Multi-head Attention and Graph Convolutional Networks with Regularized Dropout for Biomedical Relation Extraction

Mian Huang, Jian Wang(✉), Hongfei Lin, and Zhihao Yang

Dalian University of Technology, Dalian 116024, China
huangmian@mail.dlut.edu.cn, {wangjian,hflin,yangzh}@dlut.edu.cn

Abstract. Automatic extraction of biomedical relation from text becomes critical because manual relation extraction requires significant time and resources. The extracted medical relations can be used in clinical diagnosis, medical knowledge discovery, and so on. The benefits for pharmaceutical companies, health care providers, and public health are enormous. Previous studies have shown that both semantic information and dependent information in the corpus are helpful to relation extraction. In this paper, we propose a novel neural network, named RD-MAGCN, for biomedical relation extraction. We use Multi-head Attention model to extract semantic features, syntactic dependency tree, and Graph Convolution Network to extract structural features from the text, and finally R-Drop regularization method to enhance network performance. Extensive results on a medical corpus extracted from PubMed show that our model achieves better performance than existing methods.

Keywords: Regularized Dropout · Multi-head Attention · GCN · Biomedical Relation Extraction

1 Introduction

Biomedical relation extraction is an important natural language processing task, which aims to quickly and accurately detect the relations between multiple entities related to medicine from the mass medical information on the Internet, it plays an important role in clinical diagnosis [1], medical intelligence question and answer [2], and medical knowledge mapping [3]. This research can provide technical support for medical institutions and drug companies, and has great benefits for public health. At present, there are some knowledge bases of entities and relations, but more biomedical relations exist in cross-sentence documents, which brings challenges to the research of relation extraction.

With the rise of the neural network, the deep learning model has been widely used in medical relationship extraction tasks. The existing methods are mainly divided into two categories: semantic-based model and dependency-based model. Semantic-based models, such as Convolutional Neural Network (CNN), and Recurrent Neural Network (RNN), can obtain context information effectively by encoding text sequences. Ekbal

H. Xu et al. (Eds.): CHIP 2023, CCIS 1993, pp. 98–111, 2024.
https://doi.org/10.1007/978-981-99-9864-7_7

et al. [4] used the CNN model to classify relations by using the features extracted from the convolution kernel and max-pooling layer. As a feature extraction method, CNN has a good performance, but it is more suitable to capture local information features. To better capture long-distance information and reflect the importance of different information, the attention mechanism [5] attracts researchers' attention. Zhou et al. [6] proposed an attention-based Bi-directional Long Short-Term Memory (BI-LSTM) framework that automatically focuses on words that have a decisive effect on classification and captures important semantic information in sentences. At present, the attention-based Bi-LSTM model has become an important method for natural language processing tasks.

In order to fully mine the deep information in sentences, syntactic dependency structure is applied to the relation extraction task. Guo et al. [7] fused the attention mechanism based on the shortest dependency path with CNN and RNN to obtain keywords and sentence features; Zhang et al. [8] used Graph Convolutional Network to extract relations based on the Lowest Common Ancestor (LCA) rule of entities. Miwa and Bansal [9] encoded the Shortest Dependency Path (SDP) between two entities by using Tree LSTM. Peng et al. [10] divided the input graph into two directed acyclic graphs (Dags), and Song et al. [11] proposed the Graph Recurrent Network model (GRN) to obtain the semantic structure. In addition to using parsers to construct dependency graphs, researchers also begin to pay attention to and propose methods to construct dependency graphs automatically. Jin et al. [12] proposed a complete dependency forest model, to construct a weight map that adapts to terminal tasks, Guo et al. [13] proposed a "Soft pruning" strategy, the neural network of Attention-Guided graph is used to represent the graph better. Besides, Jin et al. [14] proposed a method to generate dependency forests consisting of the semantic-embedded 1-best dependency tree. Qian et al. [15] proposed an auto-learning convolution-based graph convolutional network to perform the convolution operation over dependency forests and Tang et al. [16] devised a cross-domain pruning method to equalize local and nonlocal syntactic interactions. In the general domain, Chen et al. [17] proposed to exploit the sequential form of POS tags and naturally fill the gap between the original sentence and imperfect parse tree. Zhang et al. [18] proposed a dual attention graph convolutional network with a parallel structure to establish bidirectional information flow.

Based on the above ideas, we propose a novel end-to-end model called Multi-head Attention and Graph Convolutional Networks with R-Drop (RD-MAGCN) for N-ary document-level relation extraction, which combines semantic information and syntactic dependency information. First, we interact the input representation and the relation representation with Multi-head Attention Layer to obtain the weighted context semantic representation of the text. To make full use of syntactic dependency information in cross-sentence extraction, we construct document-level syntactic dependency trees and encode them with GCN to solve the long-distance dependence problem. Then, Concatenate the two representations and feed them into the decoder. Finally, the network is enhanced by using the R-Drop mechanism and the biomedical relation is extracted.

The major contributions of this paper are summarized as follows.

- We propose a novel end-to-end model (RD-MAGCN) that effectively combines context semantic information and syntactic information.

- We introduce a regularization method for the randomness of dropout, which can enhance the performance of the network.
- We evaluate the performance of our model, the experimental results show that the performance of this model exceeds that of previous models.

2 Method

In this section, we introduce our proposed method. The input of our model is the long documents containing the relations between medical entities, and the output is a certain type of relation. There are four steps in our method: (1) preprocessing the corpus, including instance construction and other information extraction; (2) constructing document-level syntactic dependency tree; (3) building Attention and Graph convolution Networks for relation extraction; (4) utilizing R-Drop mechanism to enhance the network.

2.1 Data Preprocessing

For the texts in the corpus, we carry out a series of preprocessing processes. We first use the Stanford CoreNLP toolkit to parse each document in the corpus to obtain the syntactic parsing results and POS tags for each word. Then we construct instances for each pair of entities marked in the dataset, each instance contains the tokens of the text, the directed dependency edges of each word, the POS tags of each word, the absolute position of each entity, and the relation type used as the label.

POS tags and entity positions are used in the Input Representation Layer to enrich the text information, and the syntactic dependency information is used to build dependency trees and encodes them with GCN to capture long-range dependency information in the text.

2.2 Dependency Tree Construction

Syntactic analysis [19] is one of the important techniques in natural language processing, which is used to determine the dependencies between words in sentences. The dependency tree is a kind of syntactic analysis method, which mainly expresses the dependence relation between the words. In order to get the syntactic dependency feature of documents, we introduce a document-level dependency tree, in which the nodes represent words and the edges represent the intra-sentence and inter-sentence lexical dependency relations. As shown in Fig. 1, in this paper, we use the following three types of edges between nodes to construct the dependency trees:

1. Syntactic dependency edges: the results of parsing text by Stanford CoreNLP toolkit. They denote the dependencies between the words in a sentence.
2. Adjacent sentence edges: we connect the dependency roots of two adjacent sentences using the adjacent sentence edges. The dependency between two sentences is indicated by “next”. By using adjacent sentence edges, the entire document can form a connected graph.
3. Self-node edges: each node in the dependency tree has a self-node edge, which allows the model to learn about the node itself during training.

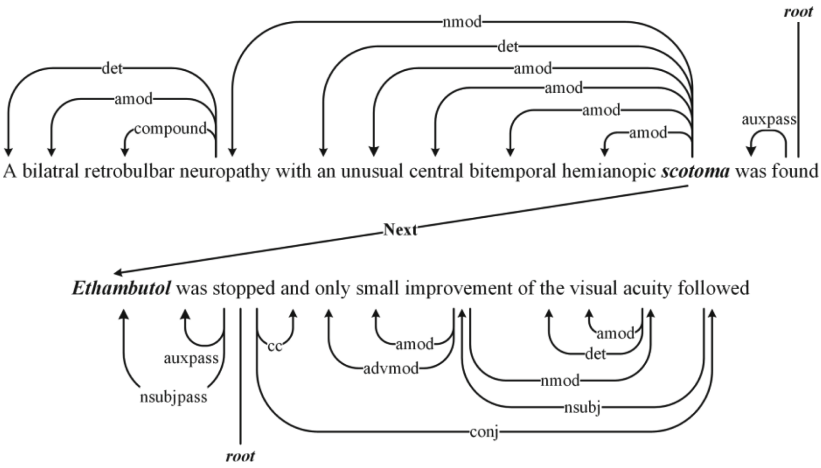

Fig. 1. An example of a constructed dependency tree consisting.

2.3 Model Structure

In this paper, we propose a novel model, Attention and Graph Convolutional Networks with R-Drop (RD-MAGCN), for N-ary document-level relation extraction. As shown in Fig. 2, the overall framework of our model consists of five parts: Input representation layer, Bi-LSTM layer, Multi-head Attention layer, GCN layer, and output layer. In addition, we utilize the R-Drop mechanism to enhance the networks and further improve the performance. The next few sections will describe the details of our model.

Input Representation Layer. For the specific domain of biomedical research, we introduce the Bio-BERT pre-trained language model [20] as the text representation encoder of Multi-head Attention Layer. However, since BERT and the improved pre-training models based on BERT use Word-Piece as the word segmentation method, and our dependency tree uses the entire words as nodes, we choose the ELMo pre-training model [21] to represent the input of GCN Layer. Furthermore, we enrich the representation of text with additional information, enabling the model to mine deeper semantics. POS tagging can strengthen the features of text, and position embedding can allow the model to locate the entities and better learn the information of the context near the entities. Therefore, our Input Representation Layer is divided into two parts, the embedding of Multi-head Attention layer is concatenated by Bio-BERT embedding, POS embedding, and position embedding:

$$w_1 = \left[w_{Bio-BERT}; w_{POS}; w_{position}\right] \tag{1}$$

The embedding of GCN Layer is concatenated by ELMo embedding, POS embedding, and position embedding:

$$w_2 = \left[w_{ELMO}; w_{POS}; w_{position}\right] \tag{2}$$

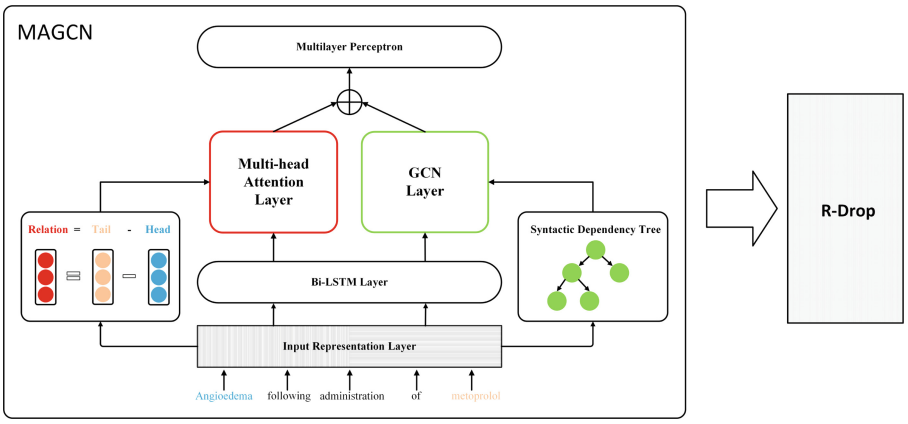

Fig. 2. Overview of our model.

Bi-LSTM Layer. RNN is very commonly used in NLP, it can capture the information of the previous text in the sentence, and LSTM utilizes the gating mechanism [22] to solve the problems of vanishing gradient, exploding gradient, and long-distance dependence that exist in RNN. Therefore, LSTM is suitable for handling document-level tasks. In this paper, we use two LSTMs, forward and backward, to encode two different input representations to obtain representations that contain both the preceding and the following information. We specify that the hidden state of the forward LSTM is h_t^f and the backward LSTM is h_t^b, the final hidden state is concatenated as:

$$h_t = \left[h_t^f; h_t^b\right] \tag{3}$$

Multi-head Attention Layer. The attention mechanism has gradually become more and more important in NLP. The attention mechanism is the focus on the input weight distribution, which can enable the model to learn more valuable information and improve the performance of relation extraction. Following Li et al. [23], we build Multi-head Attention Layer that interacts with relation representations. Based on the idea of TransE [24], we regard relation representation as to the difference between entity representations:

$$w_{relation} = w_{tail} - w_{head} \tag{4}$$

When there are only two entities, the relation representation is denoted by the tail entity minus the head entity. When there are three entities such as drug, gene, and variety, we use the third entity representation (variety) minus the first entity representation (drug) as the relation representation.

We then use normalized Scaled Dot-Product Attention to compute a weighted score for the interaction of text with relation representations:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{5}$$

where Q indicates query, from the output of Bi-LSTM Layer, represented as sequences of text. K and V indicate key and value, from relation representation. d is the dimension

of the vector and $\sqrt{d}$ is the scaling factor. The introduction of relation representation allows the model to give higher weight to text representations that are closer to the relation representation, which is helpful for relation extraction.

Eventually, concatenate the results of n heads:

$$h = [h_1; h_2; \ldots; h_n] \tag{6}$$

where n is set to 5 in our experiments. Multiple heads allow the model to learn relevant information from different representation subspaces. Finally, we perform max pooling to get the output h_{att} of Multi-head Attention Layer.

GCN Layer. GCN (Graph convolution Network) [25] is a natural extension of ConvNets on the graph structure, which can well extract the spatial structure features of images. The application of GCN to the syntactic dependency tree can extract the syntactic structure features of the text and solve the problem of long-distance separation of entities in document-level relation extraction.

In this paper, We convert the constructed document dependency tree into an adjacency matrix A, where $A_{i,j} = 1$ indicates that there is a dependency edge between word i and word j. Following Zhang et al. [8], we set the adjacency matrix as a symmetric matrix, i.e. $A_{i,j} = A_{j,i}$, and then we add self-node edges for each node, i.e. $A_{i,i} = 1$, for information about the node itself. Furthermore, we normalize the numerical values in the graph convolution to account for the large variation in node degrees in the dependency tree before adopting the activation function. At last, the graph convolution operation for node i at the l-th layer with the adjacency matrix of the dependency graph transformation can be defined as follows:

$$h_i^{(l)} = \rho\left(\frac{\sum_{j=1}^{n} A_{ij} W^{(l)} h_j^{(l-1)}}{d_i} + b^{(l)}\right) \tag{7}$$

where $h_i^{(l-1)}$ and $h_i^{(l)}$ denotes the input and the output of node i at the l-th layer. And the inputs of GCN Layer are the outputs of the Bi-LSTM Layer $h_1^{(0)}, \ldots, h_n^{(0)}$, then the outputs $h_1^{(L)}, \ldots, h_n^{(L)}$ are obtained through the graph convolution operation. W^l is the weight matrix, $b^{(l)}$ is the bias vector, $d_i = \sum_{j=1}^{n} A_{ij}$ is the degree of node i in the dependency tree for normalization, and ρ is the activation function.

Following Lee et al. [26], we also extract representations of entity nodes and concatenate them with representations of documents to highlight the role of entity nouns in the text structure and improve the performance of relation extraction. Similarly, we perform max pooling to get the output h_{GCN} of GCN Layer.

Output Layer. In this paper, our Output Layer is a two-layer perceptron. We concatenate the outputs of the two main modules to get h_{final} and then calculate as follow:

$$h_{final} = [h_{GCN}; h_{att}] \tag{8}$$

$$h_1 = ReLU\left(W_{h_1} h_{final} + b_{h_1}\right) \tag{9}$$

$$h_2 = ReLU\left(W_{h_2}h_1 + b_{h_2}\right) \tag{10}$$

In the end, we utilize the Softmax function to h_2 to determine the relation category:

$$o = Softmax(W_o h_2 + b_o) \tag{11}$$

2.4 R-Drop Mechanism

The dropout technique [27] accomplishes implicit ensemble by randomly hiding some neurons during neural network training. Liang et al. [28] introduce a simple regularization technique upon dropout, named as R-Drop. R-Drop works on the output of sub-models sampled by dropout. In each mini-batch training, each data sample undergoes two forward passes to obtain two sub-models. R-Drop forces two distributions of the same data samples outputted by two sub-models to be consistent with each other by minimizing the bidirectional Kullback-Leibler (KL) divergence between the two distributions. Finally, the two sub-models are used to jointly predict to achieve the effect of model enhancement. Results on multiple datasets show that R-Drop achieves good performance.

In this paper, we use the R-Drop mechanism to enhance our model. For the same batch of data, pass the model forward twice to get two distributions, denoted as $P_1(y_i|x_i)$ and $P_2(y_i|x_i)$. For each sub-model, we use cross-entropy as the loss function. Bidirectional KL divergence is then used to regularize the predictions of the two sub-models. Finally, the two are merged as the final loss function at the training steps:

$$L_{CE} = -logP_1(y_i|x_i) - logP_2(y_i|x_i) \tag{12}$$

$$L_{KL} = \frac{1}{2}(D_{KL}(P_1||P_2) + D_{KL}(P_2||P_1)) \tag{13}$$

$$L = L_{CE} + \alpha L_{KL} \tag{14}$$

where α is the weight coefficient, which we set to 0.5 in the experiments. In this way, R-Drop further regularizes the model space and improves the generalization ability of the model. This regularization method can be universally applied to different model structures, as long as there is randomness that can produce different outputs.

3 Experiments

3.1 Dataset

In this paper, we validate our method using the dataset introduced in Peng et al. [10], which contains 6987 drug-gene-mutation ternary relation instances and 6087 drug-mutation binary relation instances extracted from PubMed. The data we used were extracted by cross-sentence N-ary relation extraction, which extracts the triples in the

biomedical literature. Table 1 shows the statistics of the data. Most instances are documents that contain multiple sentences. There are a total of 5 relation types for labels: "resistance or nonresponse", "sensitivity", "response", "resistance" and "None". Following Peng et al., we perform relation extraction for all instances according to binary classification and multi-classification, respectively, and obtain the results using a five-fold cross-validation method. In the case of binary classification, we classify all relation types as positive examples and "None" labels as negative examples.

Table 1. The statistics of the instances in the training set.

Data	Ternary	Binary
Single	2301	2728
Cross	4956	3359
Cross-percentage	70.1%	55.2%

3.2 Parameter Settings

This section describes the details of our model experiment setup. We tune the hyperparameters based on the results on the validation set, and the final hyperparameter settings are set as follows: the dimension of Bio-BERT pre-trained language model is 768, the dimension of ELMo pre-trained language model is 1024, the dimension of POS embedding obtained by StanfordNLP and position embedding are both 100. The dimension of the Bi-LSTM hidden layer and GCN layer are both 500, the number of heads of Multi-head Attention layer is 5, and the dimension is 1000. All dropouts in the model are set to 0.5. We train the model with a batch size of 16 and the Adam optimizer [29] with a learning rate: $lr = 1e - 4$.

We evaluate our method using the same evaluation metric as the previous research, that is the average accuracy of five cross-validations.

3.3 Baselines

In order to verify the effectiveness of the model in this paper, the model in this paper is compared with the following baseline models:

1. **Feature-Based** (Quirk and Poon, 2017) [3]: a model based on the shortest dependency path between all entity pairs;
2. **DAG LSTM** (Peng et al., 2017) [10]: contains linear chains and the graph structure of the Tree LSTM;
3. **GRN** (Song et al., 2018) [11]: a model for encoding graphs using Recurrent Neural Networks;
4. **GCN** (Zhang et al., 2018) [8]: a model for encoding pruned trees using Graph Convolutional Networks;

5. **AGGCN** (Guo et al., 2019) [13]: a model that uses an attention mechanism to build dependency forests and encodes it with GCN;
6. **LF-GCN** (Guo et al., 2020) [30]: a model for automatic induction of dependency structures using a variant of the matrix tree theorem;
7. **AC-GCN** (Qian et al., 2021) [15]: a model that learns weighted graphs using a 2D convolutional network;
8. **SE-GCN** (Tang et al., 2022) [16]: a model that uses a cross-domain pruning method to equalize local and nonlocal syntactic interactions;
9. **CP-GCN** (Jin et al., 2022) [14]: a model that uses dependency forests consisting of the semantic-embedded 1-best dependency tree and adopts task-specific causal explainer to prune the dependency forests;
10. **DAGCN** (Zhang et al., 2023) [18]: a model that uses a dual attention graph convolutional network with a parallel structure to establish bidirectional information flow.

3.4 Main Results

In the experiments, we count the test accuracies of ternary relation instances and binary relation instances in binary and multi-class, respectively. In the binary-class experiment, the intra-sentence and inter-sentence situations are counted separately. The results are shown in Table 2.

As can be seen from Table 2, the performance of neural network-based methods is significantly better than that of feature-based methods. Thanks to the powerful encoding ability of GCN for graphs, GCN-based methods generally outperform RNN-based methods. Except the ternary sentence-level in Binary-class, our model RD-MAGCN achieves state-of-the-art performance.

We first focus on the multi-class relation extraction task. On the ternary relation task, RD-MAGCN achieves an average accuracy of 90.2%, surpassing the previous state-of-the-art method CP-GCN by 5.3%. On the binary relation task, RD-MAGCN achieves an average accuracy of 90.3%, surpassing AC-GCN by 9.3%. This is a huge improvement, mainly due to the greater gain effect of the R-Drop mechanism on the model in multi-classification tasks.

For the binary-class relation extraction task, although our model RD-MAGCN does not have such a large increase as the multi-classification task, it almost still exceeds CP-GCN under different tasks. The above results show that our method of combining contextual semantic features and text structure features and enhancing the model with regularization methods is effective. Next, we will introduce the ablation study we have done for each module of the method.

Table 2. Compare with related work.

Model	Binary-class				Multi-class	
	Ternary		Binary		Ternary	Binary
	Intra	Inter	Intra	Inter	Inter	Inter
Feature-Based	74.7	77.7	73.9	75.2	–	–
DAG LSTM	77.9	80.7	74.3	76.5	–	–
GRN	80.3	83.2	83.5	83.6	71.7	71.7
GCN	85.8	85.8	83.8	83.7	78.1	73.6
AGGCN	87.1	87.0	85.2	85.6	79.7	77.4
LF-GCN	88.0	88.4	86.7	87.1	81.5	79.3
AC-GCN	88.8	88.8	86.8	86.5	84.6	81.0
SE-GCN	88.7	88.4	86.8	87.7	81.9	80.4
CP-GCN	**89.5**	89.1	87.3	86.5	84.9	80.1
DAGCN	88.4	88.4	85.9	86.2	84.3	78.3
RD-MAGCN	88.7	**89.5**	**87.8**	**88.6**	**90.2**	**90.3**

3.5 Ablation Study

In this section, we have proved the effectiveness of each module in our method. First, we investigate the role of the three main modules of R-Drop, Multi-head Attention Layer and GCN Layer. We define the following variants of RD-MAGCN:

w/o R-Drop: this variant denotes using the traditional single-model cross-entropy loss function instead of two sub-models ensembles at training steps.
w/o Attention: this variant denotes removing Multi-head Attention Layer and the corresponding inputs from the model.
w/o GCN: this variant denotes removing GCN Layer and the corresponding inputs from the model.
w/o $w_{relation}$: this variant denotes that Self-Attention is applied instead of introducing relation representation in Multi-head Attention Layer, that is, Q, K, and V all from text representation.

Table 3 shows the results of the comparison of RD-MAGCN with four variants.

It can be seen from Table 3: (1) The effect of the R-Drop regularization method to enhance the model is obvious, especially in the multi-class relation extraction task. Removing the R-Drop module has a performance loss of 4.9% and 6.5% in multi-class ternary and binary tasks, respectively. We speculate the reason is that in the binary classification task, due to its low difficulty, the constraint of KL divergence makes the distribution of the output of the two sub-models roughly the same, so the ensemble effect is not obvious. In multi-classification tasks, the ensemble effect will be better. (2) The performance of removing Multi-head Attention Layer model drops in each task, indicating the usefulness of interactive contextual semantic information. (3) The performance

of removing GCN Layer model drops across tasks indicating the usefulness of syntactic structure information. Moreover, the performance of inter-sentence relation extraction drops more than that of intra-sentence relation extraction, indicating that GCN can capture long-distance structure features. (4) No introduction of relation representations in Multi-head Attention Layer degrades the results, indicating that the interaction of relation representations allows the model to pay more attention to the texts that are closer to the relation.

Table 3. The effect of the main modules of RD-MAGCN.

Model	Binary-class				Multi-class	
	Ternary		Binary		Ternary	Binary
	Intra	Inter	Intra	Inter	Inter	Inter
RD-MAGCN	**88.7**	**89.5**	**87.8**	**88.6**	**90.2**	**90.3**
w/o R-Drop	88.3	88.5	86.9	88.2	85.3	83.8
w/o Attention	86.3	86.2	84.7	84.6	86.9	86.0
w/o GCN	88.2	88.3	86.8	86.6	88.2	89.1
w/o $w_{relation}$	88.3	89.2	87.3	88.3	89.2	89.7

Next, we discuss the effects of the different input representations. We utilize the same model for the following types of inputs:

Table 4 shows the comparative performance of different input representations.

Original: The inputs to our proposed model. The input of Multi-head Attention module is the concatenation of Bio-BERT, POS and position embedding, and the input of the GCN module is the concatenation of ELMo, POS and position embedding.

Variant 1: The input of Multi-head Attention module is the concatenation of BERT, POS embedding, position embedding, and the input of the GCN module is the concatenation of ELMo, POS embedding, position embedding.

Variant 2: The input of Multi-head Attention module is the concatenation of Bio-BERT, POS embedding, and the input of the GCN module is the concatenation of ELMo, POS embedding.

Variant 3: The input of Multi-head Attention module is the concatenation of Bio-BERT, position embedding, and the input of the GCN module is the concatenation of ELMo, position embedding.

Variant 4: The input of Multi-head Attention module is Bio-BERT, and the input of the GCN module is ELMo.

From Table 4, we can see that: Bio-BERT, a domain-specific language representation model pre-trained on the large biomedical corpus, outperforms traditional BERT in the task of biomedical relation extraction. Bio-BERT enables a better understanding of complex biomedical literature. Besides, POS embedding and position embedding

Table 4. The effect of the input representation on performance.

Model	Binary-class				Multi-class	
	Ternary		Binary		Ternary	Binary
	Intra	Inter	Intra	Inter	Inter	Inter
RD-MAGCN	**88.7**	**89.5**	**87.8**	**88.6**	**90.2**	**90.3**
Variant 1	88.5	89.3	87.0	88.1	89.5	89.6
Variant 2	88.6	89.2	87.3	88.3	89.9	90.0
Variant 3	88.1	89.1	87.5	88.2	88.8	89.2
Variant 4	88.3	88.7	87.1	87.7	88.8	88.7

provide additional information for the model, which can help the model to better learn the semantics and structure of the text and locate the entities that appear in the text.

4 Conclusions

In this paper, we propose a novel end-to-end neural network named RD-MAGCN for N-ary document-level relation extraction. We extract weighted contextual features of the corpus via Multi-head Attention Layer that interacts with relation representations. We extract the syntactic structure features of the corpus through the syntactic dependency tree and GCN Layer. The combination of the two types of features can make the model more comprehensive. In addition, we ensemble the two trained sub-models through the R-Drop regularization method, and let the two sub-models jointly predict the relation type, which effectively enhances the performance of the model. Finally, we evaluate the model on multiple tasks of the medical dataset extracted from PubMed, where our RD-MAGCN achieves better results.

Our research improves the accuracy of biomedical relation extraction, which is helpful for other tasks in the medical field and the development of intelligent medicine. In future research, we will focus on applying more comprehensive techniques such as introducing medical knowledge graphs to study biomedical relation extraction more deeply.

References

1. Islamaj, R., Murray, C., Névéol, A.: Understanding PubMed user search behavior through log analysis. Database **6**(1), 1–18 (2009)
2. Yu, M., Yin, W., Hasan, S., Santos, C., Xiang, B., Zhou, B.: Improved neural relation detection for knowledge base question answering. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 571–581 (2017)
3. Quirk, C., Poon, H.: Distant supervision for relation extraction beyond the sentence boundary. In: Proceedings of the European Chapter of the Association for Computational Linguistics, pp. 1171–1182 (2017)

4. Ekbal, A., Saha, S., Bhattacharyya, P.: A deep learning architecture for protein-protein interaction article identification. In: Proceedings of the 23rd International Conference on Pattern Recognition (ICPR), pp. 3128–3133 (2016)
5. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. In: Conference on Neural Information Processing Systems (NIPS), pp. 5998–6008 (2017)
6. Zhou, P., Shi, W., Tian, J., et al.: Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 207–212 (2016)
7. Guo, X., Zhang, H., Yang, H., et al.: A single attention-based combination of CNN and RNN for relation classification. IEEE Access **7**(3), 12467–12475 (2019)
8. Zhang, Y., Qi, P., Manning, D.: Graph convolution over pruned dependency trees improves relation extraction. In: Conference on Empirical Methods in Natural Language Processing, pp. 2205–2215 (2018)
9. Miwa, M., Bansal, M.: End-to-end relation extraction using LSTMs on sequences and tree structures. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1105–1116 (2016)
10. Peng, N., Poon, H., Quirk, C., et al.: Cross-sentence *N*-ary relation extraction with graph LSTMs. Trans. Assoc. Comput. Linguist. **5**(2), 101–115 (2017)
11. Song, L., Zhang, Y., Wang, Z., et al.: N-ary relation extraction using graph-state LSTM. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 2226–2235 (2018)
12. Jin, L., Song, L., Zhang, Y., et al.: Relation extraction exploiting full dependency forests. In: Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (2020)
13. Guo, Z., Zhang, Y., Lu, W.: Attention guided graph convolutional networks for relation extraction. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 241–251 (2019)
14. Jin, Y., Li, J., Lian, Z., et al.: Supporting medical relation extraction via causality-pruned semantic dependency forest. In: Proceedings of the COLING, pp. 2450–2460 (2022)
15. Qian, M., et al.: Auto-learning convolution-based graph convolutional network for medical relation extraction. In: Lin, H., Zhang, M., Pang, L. (eds.) CCIR 2021. LNCS, vol. 13026, pp. 195–207. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88189-4_15
16. Tang, W., Wang, J., Lin, H., et al.: A syntactic information-based classification model for medical literature: algorithm development and validation study. JMIR Med. Inform. **10**(8), 378–387 (2022)
17. Chen, X., Zhang, M., Xiong, S., et al.: On the form of parsed sentences for relation extraction. Knowl.-Based Syst. **25**(1), 109–184 (2022)
18. Zhang, D., Liu, Z., Jia, W., et al.: Dual attention graph convolutional network for relation extraction. IEEE Trans. Knowl. Data Eng. **10**(11), 1–14 (2023)
19. Martin, J., Gompel, R.: Handbook of Psycholinguistics. 2nd edn. (2006)
20. Lee, J., Yoon, W., Kim, S., et al.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics **36**(4), 1234–1240 (2019)
21. Peters, M., Neumann, M., Iyyer, M., et al.: Deep contextualized word representations. In: Proceedings of the North American Chapter of the Association for Computational Linguistics, pp. 2227–2237 (2018)
22. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
23. Li, P., Mao, K., Yang, X., et al.: Improving relation extraction with knowledge-attention. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 229–239 (2019)

24. Bordes, A., Usunier, N., Weston, J., et al.: Translating embeddings for modeling multi-relational data. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 87–95 (2013)
25. Kipf, T., Welling, M.: Semi-supervised classification with graph convolutional networks. In: Proceedings of the International Conference on Learning Representations (2017)
26. Lee, K., He, L., Lewis, M., et al.: End-to-end neural coreference resolution. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (2017)
27. Hinton, G., Srivastava, N., Krizhevsky, A., et al.: Improving neural networks by preventing co-adaptation of feature detectors. Comput. Sci. **3**(4), 212–223 (2012)
28. Liang, X., Wu, L., Li, J., et al.: R-Drop: regularized dropout for neural networks. In: Advances in Neural Information Processing Systems, vol. 34, no. 1, pp. 10890–10905 (2021)
29. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference for Learning Representations, San Diego (2015)
30. Guo, Z., Nan, G., Lu, W., et al.: Learning latent forests for medical relation extraction. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, pp. 3651–3675 (2020)