Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 14, No. 2, 2025

Document-level Relation Extraction based on Graph Convolutional Neural Networks

Pengfei Song, Haoyue Lu

Sias University, Zhengzhou, Henan, China

Abstract: The objective of extracting relations between document components lies in identifying the Within a single document, the focus often lies on understanding the connections between different entities. This type of analysis goes beyond individual sentences, as it demands an understanding of how information from multiple sentences interacts to form these connections. Over recent years, the importance of exploring relationships involving several entities simultaneously has grown significantly. To advance the field of studying such connections across entire documents, a novel collection of data points, known as DocRED, has been introduced. Currently, the standard approach for this task involves using BiLSTM networks to process the entire document as a whole. However, this method struggles to effectively capture the intricate relationships that exist among various entities. To overcome this limitation, a new model designed for document-level relationship identification has been developed, which leverages Graph Convolutional Networks (GCN). GCNs are particularly useful here because they can gather information from surrounding entities, allowing for a more detailed modeling of their interactions. The proposed approach starts by identifying coreferential links to gather features that represent the relationships between pairs of entities. These features are then analyzed using GCN to construct a graph structure that represents the entire document, ultimately revealing the complex interactions between different entities. Testing this model on the large-scale DocRED dataset from Tsinghua University has demonstrated its strong performance in this challenging task.

Keywords: Document-level Relation Extraction; Graph Convolutional Network; DocRED.

1. Introduction

At its core, Relation Extraction (RE) focuses on discerning meaningful connections between entities within unstructured text [1] and presenting these connections in a structured format. By converting fragmented information into organized knowledge that is readily interpretable, RE serves as a critical component in various natural language processing (NLP) applications, including knowledge graph construction [2], data retrieval, question-answering systems [3], and conversational interfaces [4]

Nowadays, the task of extracting relationships between entities is primarily carried out at the sentence level, meaning identifying connections within a single sentence. However, such sentence-based models have an inherent drawback: they fail to grasp the relationships existing between entities across multiple sentences. As a result, to fully comprehend the information contained in a text, extracting relations from an entire document becomes essential. In recent times, Yao et al. [5] introduced a large-scale dataset that is manually annotated, which expands the scope of sentence-level relationships to the document context. This dataset includes numerous relational facts and demands that models predict the connections between every pair of entities present in the document. This new setup presents greater difficulties due to the fact that many relational facts are spread across several sentences and complex interactions among entities need to be considered and modeled.

But document-level relation extraction also faces two challenges. First, the relationship between two entities may involve multiple different sentences, and the relationship between entities cannot be obtained based on just one sentence. Apparently, sentence-level relation extraction methods do not apply to document-level relation extraction. Secondly, the same entity may have different names in a sentence, that is, the same entity may be mentioned in multiple different sentences, so document-level relation extraction has a deeper level of feature extraction to aggregate the context information of the entity.

Based on these two challenges, this paper proposes a document-level relation extraction method based on graph convolutional neural networks. The model uses coreference relations at the input layer to address the issue of the same entity being mentioned multiple times in a sentence, further extracting the deep feature information of the entity. The graph convolution model overcomes the problem that traditional deep learning models cannot aggregate entity context information, effectively controlling the impact of redundant data on the experimental results. Finally, the model was evaluated on the DocRED dataset, and the results showed that compared with existing methods, the model made some significant progress in relation extraction.

2. Related Work

Relation extraction tasks are often regarded as multi-classification problems, and the most representative of traditional methods is the eigenvector-based approach [6], which performs relation extraction by modeling the eigenvectors; Kernel function-based methods utilize the structural information of the corpus itself to achieve relation extraction by calculating similarity [7]; Deep learning methods based on neural network models [8], such as convolutional neural networks (CNNS)[9] and recurrent neural networks (RNNS)[10], can automatically learn sentence features, thus avoiding problems such as error propagation brought by NLP tools [11], and also improving the effect of relation extraction. However, these methods all require a large amount of manually labeled corpora, which limits the development of relation extraction.

To address the severe shortage of manually annotated corpora, Mintz et al. [12] proposed a remote supervision method that does not require manual annotation, that is, using the FreeBase knowledge base and Wikipedia text for alignment to obtain large-scale relation triples. However, the method is prone to noise annotation problems, so filtering noise annotations has become the focus of remote supervision methods. To address the challenges posed by imprecise annotations and noisy data in relation extraction tasks, researchers have explored various approaches. Earlier works, such as those by Hoffmann et al. [13] and Surdeanu et al. [14], introduced techniques like multi-instance learning and multi-instance multi-label methods to mitigate the impact of incorrect labeling. Additionally, Benjamin et al. [15] combined topic modeling (LDA) with decision learning to reduce noise in remote supervision results. In recent years, deep learning has emerged as a powerful tool in this field, with studies like Zeng et al. [16] utilizing convolutional networks with segmented max pooling to automatically extract sentence features, followed by integrating multi-instance learning for remote supervised relation extraction; Lei et al.[17] proposed a neural relation extraction framework with bidirectional knowledge distillation in order to effectively simulate the relationship patterns between text corpora and knowledge graph information, which can use different information sources in coordination to reduce the noise label problem in remote supervision.

But these methods did not take into account relationships among multiple sentences. Peng et al. [18] studied a graph-based LSTM framework capable of extracting cross-sentence n-element relationships. DocRED, proposed by Yao et al. [5], is based on large document-level relation extraction datasets constructed from Wikipedia and Wikidata and tests several state-of-the-art neural network models on them. Wang et al. [19] used Transformer [20] text encoding into context representations and fine-tuned with BERT to adopt a two-step strategy on DocRED to improve the model's performance. GCN was first applied by Kipf and Welling[21] on citation networks and knowledge graph datasets. Subsequently, it was used in areas such as semantic role tagging [22], multi-document summarization [23], and temporal relationship externalization [24]. Zhang et al. [25] used GCN on dependency trees and proposed a novel path-centric pruning technique that helps models remove irrelevant information to the greatest extent possible without disrupting the key content to improve relation extraction.

The document-level graph convolutional network structure proposed in this paper enables deep training of the model by aggregating and modeling the context information of entities, thereby capturing rich local and non-local dependency features in sentences. This paper experiments the proposed model on the DocRED dataset and achieves better results than existing models.

3. Methods

This section commences with an overview of the GCN framework employed for extracting relationships at the document level. The framework takes documents composed of n terms, converts each into a series of hidden state vectors, and then computes the entity representations. These entity features are combined to form vector embeddings, which are subsequently used to construct a graph structure using GCN. This graph-based representation is then analyzed to determine the final relational predictions. The model architecture diagram of this paper is shown in Figure 1.

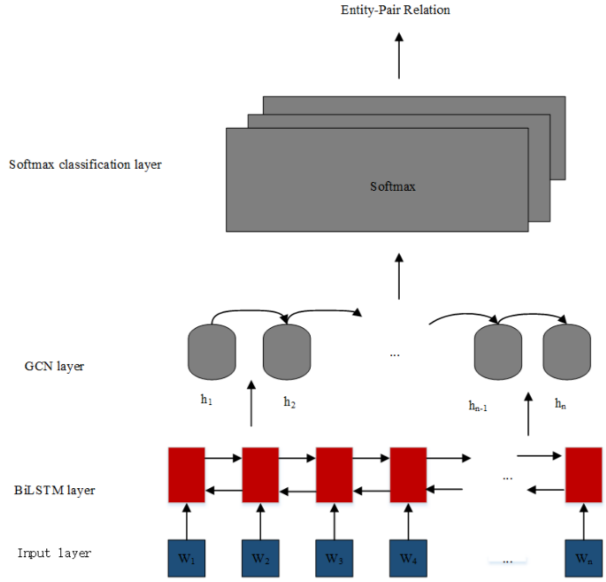

Figure 1. Model Architecture diagram

3.1. Input Layer

The input layer is an essential part of relationship extraction tasks, which can represent sentences in text in a vectorized form for further processing. The specific representation of the input layer is shown in Figure 2.

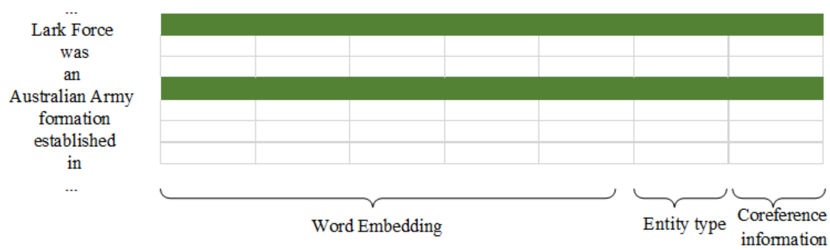

Figure 2. Schematic diagram of input layer

In Figure 2, each word in the sentence is composed of word embeddings, entity types, and co referential information. White represents the vector of words in the sentence, while dark green represents the head and tail entities in the sentence.

3.1.1. Word Embeddings

Word embedding is the collective term for language models and representation learning techniques in natural language processing (NLP). Specifically, it refers to embedding a high-dimensional space of all words into a low-dimensional continuous vector space, where each word or phrase is mapped to a vector over the real number field.

This paper mainly uses GloVe word embeddings [26], which overcomes the shortcomings of global matrix factorization and local context Windows by learning word vectors through the statistical information of global lexical co-occurrence, thereby combining the statistical information with the advantages of the local context window method to better utilize the context.

3.1.2. Entity Type Embeddings

Entity type embeddings are achieved by mapping the entity types assigned to words (such as PER, LOC, ORG, etc.) to vectors using the embedding matrix. Entity types are assigned manually to annotated data, and through the BERT model to remotely supervised data.

3.1.3. Coreference Relations

Co-reference information refers to the use of the same entity id (determined based on the order in which the entity first appears in the document) to map co-reference relationships to vectors when several different entities refer to the same entity. For example, England and UK both refer to the United Kingdom.

This paper utilizes word vectors, coreference information, and entity types as basic features, representing them in vector form as W^w, W^c, and W^t, respectively. These vectors are concatenated to form the feature x_i for the initial word i, i.e., $x_i=[W_i^w;W_i^c;W_i^t]$.

3.2. BiLSTM Layer

This section inputs the feature information obtained from the input layer into the BiLSTM layer to obtain the feature representation of each entity, while the BiLSTM layer outputs a vectorized form of word sequence information. Specifically, it involves encoding and analyzing the word features x_i obtained from the input layer, and concatenating the hidden layer vector representations of the forward LSTM and backward LSTM to obtain the sequence $h_1, h_2, \ldots, h_n$ that integrates contextual information. The calculation and update formulas for the hidden layer node h_t at a certain moment t are shown in equations (1) to (6).

$$i_t=\sigma(W_{xi}x_t+W_{hi}h_{t-1}+W_{ci}c_{t-1}+b_i) \quad (1)$$

$$f_t=\sigma\left(W_{xf}x_t+W_{hf}h_{t-1}+W_{cf}c_{t-1}+b_f\right) \quad (2)$$

$$g_t=tanh\left(W_{xg}x_t+W_{hg}h_{t-1}+W_{cg}c_{t-1}+b_g\right) \quad (3)$$

$$c_t=f_t\otimes c_{t-1}\text{-}i_t\otimes g_t \quad (4)$$

$$o_t=\sigma(W_{xo}x_t+W_{ho}h_{t-1}+W_{co}c_t+b_o) \quad (5)$$

$$h_t=o_t\otimes tanh(c_t) \quad (6)$$

Among them, σ represents the sigmoid activation function, ⊗ represents the multiplication of vector elements, and the forget gate, input gate, and output gate are respectively represented by f_t, i_t, and o_t. g_t is the new value tensor of the unit, x_t is the input vector at time t, c_t is the tensor of the forget gate, input gate, and output gate at time t, and h_t is the output of the hidden layer, $W_{xi}, W_{xf}, W_{xg}, W_{xo}, W_{ci}, W_{cf}, W_{cg}$ Representing the weight matrices of x_t running on different gate mechanisms, $W_{hi}, W_{hf}, W_{hg}, W_{ho}, W_{co}$ represent the weight matrices of h_t on different gate mechanisms, and b is the bias term.

At time t, the forward output of BiLSTM is $\overrightarrow{\boldsymbol{h}_t}$, while the reverse output is $\overleftarrow{\boldsymbol{h}_t}$. Then, concatenate the outputs from both directions to obtain the final output h_t at time t, as shown in equation (7).

$$h_t=[\overrightarrow{\boldsymbol{h}_t},\overleftarrow{\boldsymbol{h}_t}] \quad (7)$$

3.3. Figure Convolutional Layers

Graph convolution networks involve applying convolutional neural network techniques to graph data structures. Consider a graph consisting of n nodes; its structural information can be captured by an n×n adjacency matrix A, where an entry $A_{ij}=1$ (or a corresponding weight) if there exists a connection between node i and node j.In a GCN model with L layers, the input feature vector of a node i at layer l-1 is typically denoted as h_i^{l-1}. After applying the graph convolution operation, the corresponding output feature vector at layer l becomes h_i^l, and this process can be formally expressed as:

$$h_i^l=\sigma\left(\sum_{j=1}^{n} A_{ij}\, W^l h_j^{l-1}+b^l\right) \quad (8)$$

where W^l is a linear transformation, b^l is a bias term, σ is a nonlinear function (e.g.Relu). In the convolution process of each graph, each node collects and aggregates information from adjacent nodes in the graph.

If formula (8) is directly used, there will be a significant magnitude difference between connections of different nodes, resulting in the feature representation of the sentence not containing information about the nodes themselves, but simply favoring higher-order nodes. Therefore, in practical applications, normalization is required for the adjacency matrix A_{ij}; in addition, nodes in formula (8) are not connected to themselves, meaning that no information is transmitted between h_i^{l-1} and sh_i^l. Based on this, this paper adds a self-loop mechanism to the graph structure, and then sets the diagonal elements of A_{ij} to 1, that is, A_{ij}=1, thus obtaining the optimized adjacency matrix $\tilde{A}_{ij}$. Finally, it is fed back to GCN using a nonlinear function. This improvement makes the main features in the graph still the nodes themselves, which is more in line with the principle of feature extraction. The revised calculation formula is shown in formula (9).

$$h_i^l=\sigma\left(\sum_{j=1}^{n} \tilde{A}_{ij}\, W^l h_j^{l-1}+b^l\right) \quad (9)$$

Where $\tilde{A}_{ij}$ is the improved adjacency matrix.

The GCN model described above uses the same parameters for all edges in the graph. Use different transition matrices W for top-down, bottom-up, and self-cyclic edges; Add specific parameters for controlling edge connectivity, similar to the approach proposed by Marcheggiani and Titov [27] in 2017. It was found through experiments that adding directed edges to the model did not improve the model's performance, and that adding control over edge connectivity reduced the model's accuracy. It is hypothesized that this is because the proposed GCN model is usually capable of capturing the information of the edges needed for classifying relations, and adding features for the direction and connectivity of the edges again does not provide a stronger classification ability for the model, but instead causes overfitting. For example, the relations contained in "A's son, B" and "B's son, A" can be easily distinguished by the's on different entities, even without considering the directionality of the edges.

3.4. Categorical Layers

Softmax is a multi classification model with a wide range of applications. For each named entity mentioned (which appears in different sentences) m_k, from the s-th word to the t-th word, the calculation is shown in equation (10).

$$m_k=\frac{1}{t\text{-}s+1}\sum_{j=s}^{t}h_j \tag{10}$$

The entity e_i with K mentions represents the average of these mentions, as shown in equation (11).

$$e_i=\frac{1}{K}\sum_{k}m_k \tag{11}$$

In this section, the obtained entity representationse_i and e_j will be concatenated with the distance features d_{ij} and d_{ji} of the two entity pairs and input into the classification layer. Finally, the probability of each category is calculated using the Sigmoid function, and the entity with the highest probability is selected as the relationship between entities. The calculation formula is shown in equation (14).

$$\hat{e}_i=[e_i;d_{ij}] \tag{12}$$

$$\hat{e}_j=[e_j;d_{ji}] \tag{13}$$

$$P(r|\hat{e}_i,\hat{e}_j)=sigmoid(\hat{e}_iW_r\hat{e}_j+b_r) \tag{14}$$

Among them, [;] represents concatenation, r is the relationship type, W_r, $b_r$are the weight parameters and bias terms of the relationship type.

4. Experiment

4.1. Dataset

To construct the DocRED dataset, researchers employed remote supervision techniques involving Wikipedia documents and Wikidata. This process began with identifying named entities within each document, after which these entities were mapped to corresponding entries in Wikidata. Entities sharing the same knowledge base identifier were then combined into a single entity. Additionally, relationships between entity pairs were determined by querying Wikidata, with supplementary steps such as named entity and co-parameter retrieval, entity merging, and collection of relational evidence all carried out under the framework of remote supervision [12]

Table 1. Dataset Statistics

Settings	Number of documents	Number of relationships	Number of relationship instances	Number of relationship facts
Train	3053	96	38269	34715
Dev	1000	96	12332	11790
Test	1000	96	12842	12101

The DocRED dataset encompasses a diverse spectrum of subjects. Its entity categories feature individuals, locations, groups, periods, quantities, and proper nouns, while relationship classifications include scientific, artistic, temporal, and interpersonal domains, among others. Achieving optimal performance on this dataset demands an array of reasoning capabilities, such as pattern identification, logical deduction, anaphora resolution, and everyday knowledge inference. Additionally, the dataset offers both annotated training examples (drawn from a combination of remote supervision and manual labeling efforts) and automatically generated remote supervision data. During our experimental process, only the manually annotated instances were utilized. Detailed statistics regarding the dataset are presented in Table 1.

4.2. Experimental Parameter Settings

Table 2. Model Parameter Settings

Model Parameters	value
Learning rate	0.0001
Number of network iterations	200
Batch number of samples	20
Number of word vector dimensions	100
Coreference vector dimensions	20

All experiments in this paper are based on the Pytorch deep learning framework, using the sigmoid activation function as the activation function within the model, training the model with the Adam optimization algorithm, and using the method of minimizing cross-entropy to select the optimal parameters of the model. For the convenience of model comparison, the common parameters of different models in subsequent experiments were compared using the optimal parameters, and the optimal parameter Settings of the models are shown in Table 2.

4.3. Experimental Results

This paper compares the current mainstream relation extraction model with the model of this paper on the DocRED dataset, and the corresponding model is as follows.

1)CNN[28]/LSTM[29]/BiLSTM[30]: CNN/LSTM/BiLSTM is used as an encoder to encode the document into a hidden sequence of state vectors, and then a bilinear function is input to predict the relationship of each entity pair.

2) Context-aware [31]: By using an LSTM-based encoder, all the representations of the relationships in the Context are jointly learned, and then other context relationships are combined with the target relationship to make the final classification.

3)BERT [32]: Use BERT to encode the document, represent entities with average word embeddings, and use bilinear layers to predict the relationships between entity pairs.

Table 3. Performance of different models on the DocRED dataset

Model	Dev		Test	
	Ign F1	F1	Ign F1	F1
CNN	41.58	43.45	40.33	42.26
LSTM	48.44	50.68	47.71	50.07
BiLSTM	48.87	50.94	48.78	51.06
Context-Aware	48.94	51.09	48.40	50.70
BiLSTM-GCN	**49.02**	**51.45**	**49.46**	**51.52**

4.4. Experimental Analysis

As shown in Table 3, the F1 value of the model has increased by more than 8% compared to CNN. This is because the "BiLSTM" model introduced in the BiLSTM-GCN model can learn the feature information of the entities in the dataset, which greatly overcomes the long-distance dependency problem of "CNN". Compared with the "LSTM" model, the "F1" value of "BiLSTM-GCN" has increased by "0.77%", because the model can obtain rich semantic context information. Compared with the "BiLSTM" model, the "F1" value of this model has increased by 0.51%, because the BiLSTM-GCN model, based on the BiLSTM model, uses graph convolution of context information to better aggregate the semantic feature information of the target entity pairs, effectively improving the aggregation ability of the model compared to the traditional GCN method. Compared with the Context-Aware model, the F1 value of BiLSTM-GCN has increased by 0.36%. This is because BiLSTM-GCN retains the context feature information of the entities to a greater extent, while also enhancing the learning ability of the model and improving its accuracy.

5. Conclusion

This paper proposes a document-level relation extraction method based on graph convolutional neural networks, which learns the context information of entities through word embeddings, aggregates entity feature information using graph convolutional neural networks, and extracts long-distance dependency features using deep graph convolutional networks. This can effectively combine local and non-local dependency features in a sentence to obtain a more accurate sentence representation. Through a series of experimental comparisons, it is proved that the method proposed in this paper can effectively improve the effect of entity relation extraction. The model in this paper is based on an English dataset. In subsequent work, the model will be further extended to a Chinese corpus dataset.

References

[1] Chinchor N, Marsh E, MUC-7 information extraction task definition[C]// Proceedings of the 7th Message Under standing Conference(MUC-7). Stroudsburg, PA, USA: Association for Computational Linguistics, 1998: 359- 367.

[2] Bayu Distiawan Trisedya, Gerhard Weikum, Jianzhong Qi, and Rui Zhang. 2019. Neural relation extraction for knowledge base enrichment. In ACL, pages 229-240, Florence, Italy. ACL.

[3] Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved neural relation detection for knowledge base question answering. In ACL, pages 571-581, Van-couver, Canada. ACL.

[4] Tom Young, Erik Cambria Cambria, Iti Chaturvedi, Minlie Huang, Hao Zhou, and Subham Biswas. 2018. Augmenting end-to-end dialog systems with commonsense knowledge. In AAAI.

[5] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In Pro- ceedings ofACL 2019.

[6] Gan L X, Wan C X,Liu D X, et al. Chinese Named Entity Relation Extraction Based on Syntatic and Semantic Features [J]. Journal of Computer Research and Development, 2016, 53(2):284-302.

[7] Choi S P ,Lee S ,Jung H et al.An intensive case study on kernel-based relation extraction[J]. Multimedia Tools & Applications ,2014 ,71(2): 741-767.

[8] Zeng D ,Liu K ,Lai S et al.Relation Classification via Convolutional Deep Neural Network[C]//Proceedings of the 25th Inter national Conference on Computational Linguistics. 2014:2335-2344.

[9] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Proceedings of Conference on Empirical Methods in Natural Language Processing, pages 1746-1751. Association for Computational Linguistics.

[10] Sunil Kumar Sahu and Ashish Anand. 2018. Drug- drug interaction extraction from biomedical texts using long short-term memory network. Journal of Biomedical Informatics, 86:15 -24.

[11] Li F L,Ke J .Research Progress of Entity Relation Ex traction Base on Deep Learning Framework [J].Information Science, 2018, v36(3):169-176.

[12] Mintz M ,Steven B,Rion S,et al .Distant super vision for relation extraction without labeled data[C]//Proceedings of Joint Conference of the Meeting of the ACL .Stroudsburg:Association for Computational Linguistics,2009:1003-1011.

[13] Hoffmann R,Zhang C,Ling X,et al. Knowledge-based weak supervision for information extraction of overlapping relations [C]//Proceedings of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: Association for Computational Linguistics, 2011:541-550.

[14] Surdeanu M,T Ibshirani J,Nallapat I R,et al.Multi-instance multi-label learning for relation extraction[C]//Proceedings of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natura Language Learning. 2012: 455-465.

[15] Benjamin R ,Dietrich K .Combining Generative and Discriminative Model Scores for Distant Supervision[C]// Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing.2013:24-29.

[16] Zeng D ,Liu K ,Chen Y ,et al .Distant supervision for relation extraction via piecewise convolutional neural networks[C]// Proceedings of Conference on Empirical Methods in Natural Language Processing.2015:1753-1762.

[17] Lei K ,Chen D ,Li Y ,et al.Cooperative Denoising for Distantly Supervised Relation Ex traction[C]//Proceedings of the 27th International Conference on Computational Linguistics.2018: 426-436.

[18] Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. Transactions of the Association for Computational Linguis- tics, 5:101-115.

[19] H.Wang, C. Focke, R. Sylvester, N. Mishra, andW. Wang, Fine-tune bert for DocRED with two-step process,2019, arXiv:1909.11898. [Online]. Available: http:// arxiv. org/ abs/ 1909. 11898.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, Attention is all you need,in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

[21] Thomas N Kipf and Max Welling. 2017. Semi- supervised classification with graph convolutional networks. In International Conference on Learning Representations.

[22] Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for se- mantic role labeling. In Proceedings of Conference on Empirical Methods in Natural Language Processing, pages 1506–1515. Association for Computational Linguistics.[. 2017. Graph-based neural multi-document summarization. In Proceedings of the 21st Conference on Computational Linguistics.

[23] Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In Proceedings of the 21st Conference on Computational Natural Language Learning, pages 452–462. Association for Computational Linguistics.

[24] Shikhar Vashishth, Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018. Dating documents using graph convolution networks. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, pages 1605–1615. Association for Computational Linguistics.

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

[26] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In Proceedings of EMNLP, pages 1532-1543.

[27] Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017).

[28] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In Proceedings of COLING, pages 2335–2344.

[29] Sepp Hochreiter and Jurgen Schmidhuber.1997.Long short-term memory. Neural Computation, 9:1735–1780.

[30] Rui Cai, Xiaodong Zhang, and Houfeng Wang. 2016. Bidirectional recurrent convolutional neural network for relation classification. In Proceedings of ACL, pages 756-765.

[31] Daniil Sorokin and Iryna Gurevych. 2017. Contextaware representations for knowledge base relation extraction. In Proceedings of EMNLP, pages 1784-1789.

[32] Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Wang. 2019. Fine-tune bert for docred with two-step process. arXiv preprint arXiv:1909.11898.