Understanding More Knowledge Makes the Transformer Perform Better in Document-level Relation Extraction

Haotian Chen[†] HTCHEN18@FUDAN.EDU.CN
Yijiang Chen[†] CHENYJ20@FUDAN.EDU.CN
Xiangdong Zhou XDZHOU@FUDAN.EDU.CN
School of Computer Science, Fudan University, China

Editors: Berrin Yanıkoğlu and Wray Buntine

Abstract

Relation extraction plays a vital role in knowledge graph construction. In contrast with the traditional relation extraction on a single sentence, extracting relations from multiple sentences as a whole will harvest more valuable and richer knowledge. Recently, the Transformer-based pre-trained language models (TPLMs) are widely adopted to tackle document-level relation extraction (DocRE). Graph-based methods, aiming to acquire knowledge between entities to form entity-level relation graphs, have facilitated the rapid development of DocRE by infusing their proposed models with the knowledge. However, beyond entity-level knowledge, we discover many other kinds of knowledge that can aid humans to extract relations. It remains unclear whether and in which way they can be adopted to improve the performance of the Transformer, which affects the maximum performance gain of Transformer-based methods. In this paper, we propose a novel weighted multi-channel Transformer (WMCT) to infuse unlimited kinds of knowledge into the vanilla Transformer. Based on WMCT, we also explore five kinds of knowledge to enhance both its reasoning ability and expressive power. Our extensive experimental results demonstrate that: (1) more knowledge makes the performance of the Transformer better and (2) more informative knowledge leads to more performance gain. We appeal to future Transformer-based work to consider exploring more informative knowledge to improve the performance of the Transformer.

Keywords: Document-level relation extraction; graph-based method; a weighted multi-channel Transformer

1. Introduction

Relation extraction, as the foundation of constructing the knowledge base, aims to extract the entities and relations to form relation triples from text collections. Early research on relation extraction (Zhang et al., 2019; Soares et al., 2019; Guo et al., 2019) mainly focuses on the sentence level, namely predicting the relation facts between entity pairs in a single sentence. However, at the document level, cross-sentence relation triples are common. It is reported that more than 40% of relation triples are extracted across multiple sentences (Yao et al., 2019). Therefore, document-level relation extraction has attracted increasing research interests recently (Ye et al., 2020; Nan et al., 2020; Zeng et al., 2020).

†. These authors contributed equally to this work

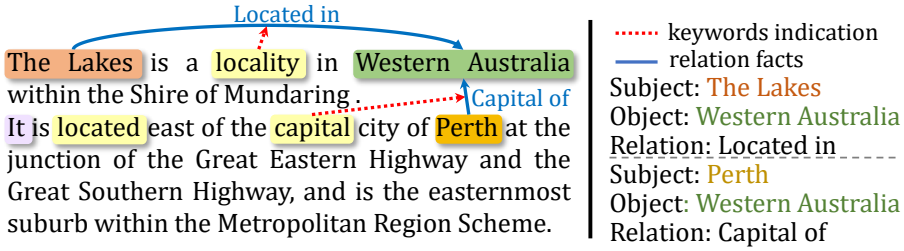

Figure 1: A multi-hop inference example. Evidence word refers to the keyword that is more relevant and informative for the inference of the corresponding relations. Mention refers to the different descriptions of an entity. In this Figure, "locality", "located", and "capital" are the evidence words and provide important clues to indicate the relations between an entity pair. "It" is a pronoun refers to "The Lakes".

In the document-level relation extraction (DocRE), the subject and object entities in a relation triple may be located in different sentences, and the same entity may appear multiple times across a document in different forms including corresponding nouns, pronouns, evidence words, and other expressions. These various mentions/entities are very informative for relation extraction in the document-level context. Figure 1 shows a multi-hop inference example. There are two mentions of the entity "The Lakes": "The Lakes" in the first sentence and "It" in the second sentence. To extract the relation "Capital of" between "Perth" and "Western Australia", we need to first recognize that "It" refers to "The Lakes". Second, we extract the fact that "The Lakes" is located in "Western Australia" and "It" is located in the capital city of "Perth". Finally, we can reason that the relation between "Perth" and "Western Australia" is "Capital of".

As Figure 1 shows, the major challenge in document-level relation extraction is the indirect and multi-hop relation extraction. To deal with the problem, the graph-based methods (Christopoulou et al., 2019; Zeng et al., 2020; Nan et al., 2020; Xiao et al., 2022) appear in recent years with the purpose of performing inter-sentence reasoning for DocRE. They employ a Transformer-based pre-trained language model (TPLM) as the encoder and then use their delicately designed rules to construct entity-level graphs, which consider entities and their mentions as nodes and their coreference relationships as edges for aiding in reasoning. However, abundant useful information (e.g., evidence words and pronouns) in the context is neglected in their constructed graphs. The information loss breaks the logic chain for reasoning and thus impedes the achievement of accurate relation predictions, which limits the performance of graph-based methods. Meanwhile, previous work (Xu et al., 2021; Yu et al., 2022) tries to facilitate the development of both graph-based methods and DocRE by using knowledge (e.g., co-occurrence and coreference entity structure) to improve their common backbone (Transformer). Therefore, using an appropriate method to incorporate comprehensive knowledge into Transformer is of the essence. This raises three crucial yet rarely discussed questions: (1) What method can infuse the Transformer with various kinds of (comprehensive) knowledge? (2) What factor of the kind can cause more significant

performance improvement? (3) Does the number of kinds affect the performance of the Transformer?

In this paper, we address the three crucial problems by proposing a novel model named weighted multi-channel Transformer (WMCT), which encodes various kinds of knowledge together with context, and then analyze the effectiveness of these kinds both individually and in combination. Specifically, we explore knowledge by capturing the mentions, pronouns, evidence words, and dependency words to construct plentiful and various kinds of knowledge. Then, to model the interactions between context and knowledge, we encode context and knowledge in the TPLM model. In order to adaptively select and incorporate key clues from various and unlimited kinds of knowledge, we propose a weighted multi-channel Transformer to aggregate them. The extensive experimental results show that: (1) our proposed WMCT achieves significant performance improvement compared with the state-of-the-art (SOTA) baseline methods that focus on improving the Transformer; (2) more informative knowledge leads to more significant performance improvement; (3) more knowledge makes the performance of the Transformer better. Our main contributions are summarized as follows:

- We explore and construct various kinds of knowledge represented by graphs. We propose some delicately designed rules and leverage the dependency tree to capture mentions, pronouns, evidence words, and dependency words to extend and enrich the knowledge for relation inference.
- We propose a weighted multi-channel Transformer (WMCT) to aggregate and adaptively embed the various kinds of knowledge. WMCT exhibits competitive performance compared with the SOTA baseline methods that focus on improving the Transformer.
- We evaluate our model and analyze our findings on three document-level relation extraction datasets including DocRED, CDR, and GDA. Our findings suggest directions for improvement on the methods that adopt TPLM.

2. Related Work

Early relation extraction approaches focus on predicting the relations between entities within a single sentence. Various methods including models based on graph (Guo et al., 2019), pre-training (Soares et al., 2019), knowledge graph (Zhang et al., 2019), and attention mechanism (Yang et al., 2019), just to name a few, are applied. Recently, researchers start to deal with the document-level relation extraction problem. Some early works (Yao et al., 2019; Christopoulou et al., 2019) indicate that the reasoning process is necessary for the document-level relation extraction because many relation facts can only be predicted based on interactions between mentions. Many previous approaches model the reasoning process by building graphs. GCNN (Sahu et al., 2019) uses co-reference links to construct the dependency graph and MULTISCALE (Jia et al., 2019) leverages the dependency graph to better capture document-specific features. Entity-GCN (Cao et al., 2019) leverages co-reference information to construct document-level entity graphs. More recently, LSR (Nan et al., 2020) captures non-local interactions of entities from the same sentence to build dependency structures. SSAN (Xu et al., 2021) explores co-occurrence and coreference entity

Name	Nodes (Knowledge Kind)
$\mathcal{G}_1$	only mentions (mention structural knowledge)
$\mathcal{G}_2$	mentions and evidence words (relation knowledge between each mention pair)
$\mathcal{G}_3$	mentions and pronouns (extended mention structural knowledge)
$\mathcal{G}_4$	mentions and their dependency nodes in 2-hop (mention dependency knowledge)
$\mathcal{G}_5$	mentions, pronouns and evidence words (mention structural and relation knowledge)

Table 1: Five kinds of knowledge represented by graphs.

structures for reasoning. RSMAN (Yu et al., 2022) improves entity-level features in previous entity-level graphs by relation-specific representations.

However, all the graphs constructed in previous work contain only a single kind of knowledge, which indicates that the power of knowledge is not sufficiently explored. Among these methods, SSAN and RSMAN improve the Transformer by either incorporating knowledge (structural guidance) or proposing a better way for incorporating knowledge. We compare our proposed WMCT with them.

3. Methodology

We present our proposed WMCT as follows. In Section 3.1, we clarify and define the task. Then, we introduce our formulated rules and methods to construct graphs by pronouns and evidence words to incorporate knowledge in Section 3.2. Finally, in Section 3.3, we detail the weighted multi-channel Transformer (WMCT).

3.1. Task Formulation

Given a document d and an entity set $\mathcal{E} = \{e_i\}_{i=1}^{n}$ in d, the target of document-level relation extraction is to predict all of the relations between entity pairs $(e_i, e_j)_{i,j=1...n;i\neq j}$ among $\mathcal{R} \cup \{\text{NA}\}$. $\mathcal{R}$ is the predefined relation set . NA stands for no relation between an entity pair. e_i and e_j denote subject and object entities. An entity may appear many times in a document, we use set $\left\{m_j^i\right\}_{j=1}^{N_{e_i}}$ to distinguish the mentions of each entity. We finally build the extracted relation triples into the form of $\{(e_i, r_{ij}, e_j) \mid e_i, e_j \in \mathcal{E}, r_{ij} \in \mathcal{R}\}$.

3.2. Graph Construction

In graph-based relation extraction, the graphs are constructed by various rule-based methods, hence involving the knowledge of relation inference. In previous work, it is often the case that the graph consists of a single type of nodes (e.g. mention/entity nodes). However, as Figure 1 shows, in the context of documents, the pronouns and some keywords are very informative and helpful for indirect or cross-sentence relation extraction. They should be involved as new types of nodes in the graphs.

Apparently, it is difficult to define and derive the graphs completely due to the diversity of knowledge and our limited scope. Consequently, based on the previous work and our own observation, we preliminarily propose five kinds of regular graphs that carry various knowledge as shown in Table 1. The graph set is denoted with $\{\mathcal{G}_i\}_{i=1}^{M}$ where $M = 5$. The motivations of the graphs in our work are given as follows: $\mathcal{G}_1$ explains a mention structure that discriminates whether two mentions reside in the same sentence or refer to the same entity (Zeng et al., 2020); $\mathcal{G}_2$ introduces our proposed evidence word nodes, which creating another path between each mention node pair to gain more attention on relevant relation; $\mathcal{G}_3$, extends the knowledge of $\mathcal{G}_1$ by the reference relations between pronouns and their referring mentions; $\mathcal{G}_4$ as a graph considering the knowledge reserved in syntactic structure, involves the words depended by each mention in a dependency tree to describe the dependencies among the mentions and involvements; $\mathcal{G}_5$ is a combination of the knowledge in $\mathcal{G}_2$ and $\mathcal{G}_3$. Figure 2 shows our proposed graphs, where each graph $\mathcal{G}_i$ has its adjacency matrix A_i.

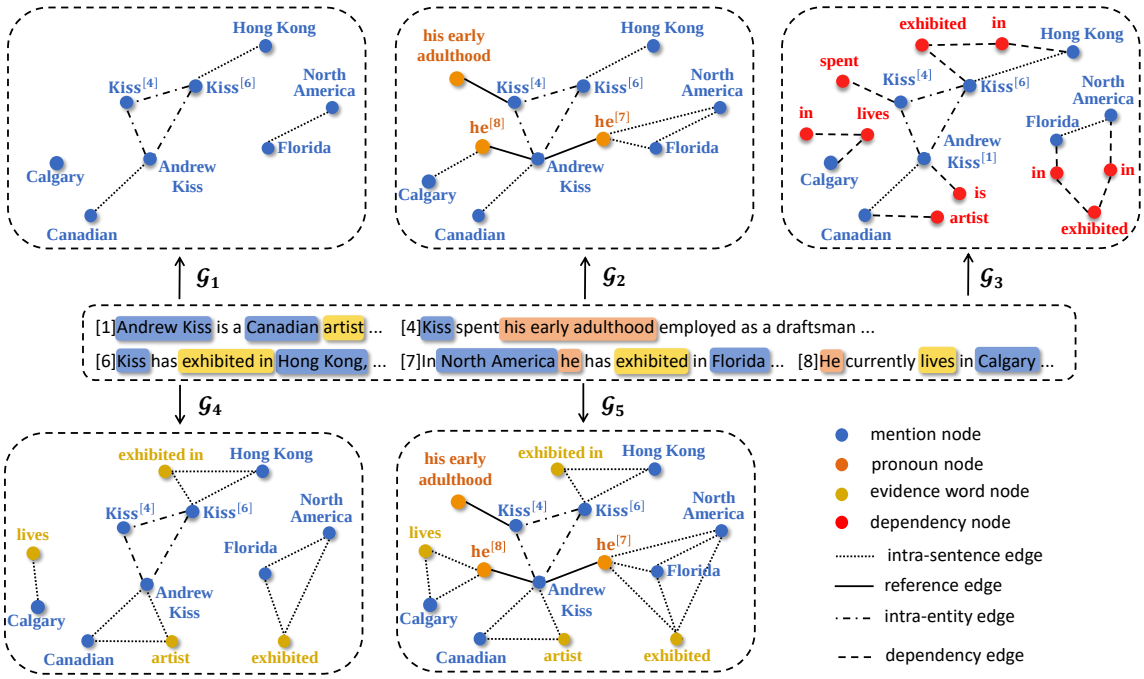

Figure 2: An example of our constructed graphs $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, $\mathcal{G}_4$, and $\mathcal{G}_5$.

To create the graph set, we first capture mentions and construct a base mention-level graph where mentions are regarded as nodes. The pronouns and evidence words are regarded as special kinds of nodes and added to the base graph to obtain the extended graphs shown in Table 1. Rules are formed to find the candidate mentions that pronouns probably refer to. The shortest dependency path (SDP) of a dependency tree is employed to capture evidence words. They are bridges of the corresponding mention pair. Our proposed rules are as follows:

- The graph consists of pronoun nodes, evidence word nodes, dependency nodes, or mention nodes, and there are bi-directional edges between each other if they come

from the same sentence (except dependency nodes where bi-directional edges are constructed by hop).

- There are two rules to confirm which entities are referred to by pronouns: 1. the positions of the referents are in front of the current pronoun in a document; 2. types of entities are in the "type set" of the current kind of pronouns.
- There are bi-directional edges between pronouns and the entities they probably refer to. If there are possessive pronouns such as "his" or "her", we replace the pronoun with the first noun that appears behind it and then add bi-directional edges between the noun and corresponding entities.
- There are bi-directional edges between mentions of the same entity.

We build the "type set" for each kind of pronoun by the concluded common regularity. Specifically, for pronouns "he", "she", "his", and "her", entity type "Person" should be in their "type set". Meanwhile, for pronouns "it", "its", "they", and "their", entity types including "Organization" and "Other" should be involved.

3.3. Weighted Multi-Channel Transformer (WMCT)

Our model partly inherits the architecture of the Transformer encoder, which consists of L ($L = 12$ in our work) stacking layers. In each layer, there are five units including a multi-channel graph encoder, graph aggregator, feed-forward network, residual connection, and layer normalization. The working flow in the layer is as follows: the graph encoder encodes the input context and various graphs through different channels by self-attention mechanism, where the input context embedding matrix is taken as initial node embeddings. Then the output embeddings of nodes and edges from different graphs are adaptively aggregated to form their new embeddings in the graph aggregator. Next, new embeddings are put into a feed-forward network followed by residual connection and layer normalization. Finally, the output embeddings are adopted as the initial embeddings and input into the next layer. The overview structure of our model is shown in Figure 3.

Graph Encoder. Our graph encoder encodes different kinds of graphs to incorporate the knowledge for improving the performance of the Transformer. Given a document $d = \{x_t\}_{t=1}^{N}$ with N words, the input context embedding matrix $\boldsymbol{X}_0^l \in \mathbb{R}^{N \times d}$ in l-th layer is first projected into query, key and value matrix respectively:

$$\boldsymbol{Q}^l = \boldsymbol{X}_0^l \boldsymbol{W}_Q^l, \quad \boldsymbol{K}^l = \boldsymbol{X}_0^l \boldsymbol{W}_K^l, \quad \boldsymbol{V}^l = \boldsymbol{X}_0^l \boldsymbol{W}_V^l \tag{1}$$

where $\boldsymbol{W}_Q^l, \boldsymbol{W}_K^l, \boldsymbol{W}_V^l \in \mathbb{R}^{d \times k}$ are trainable model parameters in l-th layer. The matrix is computed in a multi-head way. We compute five kinds of attention scores by attention mechanism in five channels and then aggregate them adaptively. Incorporated with the adjacency matrix A_i constructed in Section 3.2, the attention score $\boldsymbol{S}_i^l$ of graph $\mathcal{G}_i$ in l-th layer is calculated by:

$$\boldsymbol{S}_i^l = \boldsymbol{Q}^l \boldsymbol{W}_A^l {\boldsymbol{K}^l}^T \odot A_i, i = 1, \ldots, 5, \tag{2}$$

where $\odot$ denotes the element-wise multiplication between matrix, $\boldsymbol{W}_A^l \in \mathbb{R}^{k \times k}$ is a trainable parameter matrix in l-th layer.

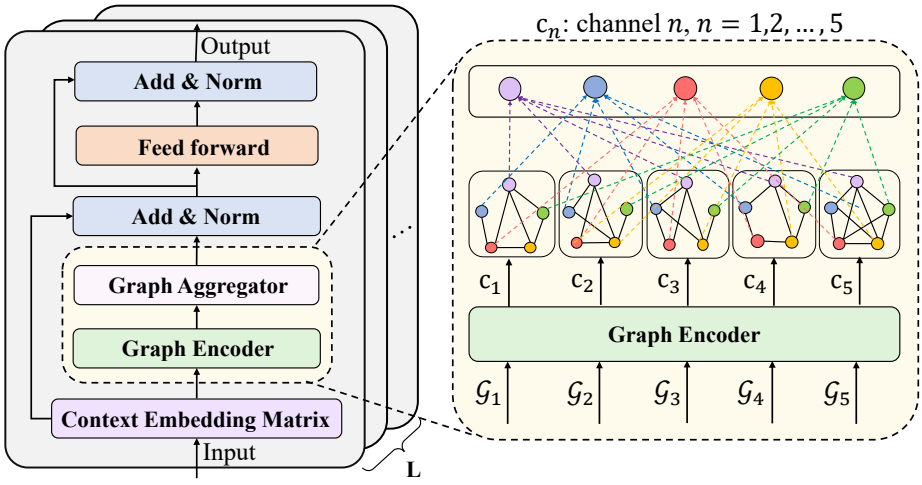

Figure 3: Overview of our model. Different colors of nodes in the right part of the figure distinguish the tokens in context. For each node, the dynamically learned weights from different graphs are represented by dash lines in the same color, which explains the different effects of graphs.

Multi-channel Graph Aggregator. A multi-channel graph aggregator is employed to allocate the learned weights to each kind of attention score. Each weight indicates the significance of the corresponding graph. All the score $\boldsymbol{S}_i^l$ are aggregated by graph aggregator. The output attention score is produced by

$$\boldsymbol{S}^l = \sum_{i=1}^{M} \alpha_i^l \boldsymbol{S}_i^l, \tag{3}$$

where $\boldsymbol{S}^l \in \mathbb{R}^{N\times N}$, α_i^l is a trainable parameter which controls the influence of $\mathcal{G}_i$ in layer l. Following the previous work (Zeng et al., 2020; Zhou et al., 2020; Nan et al., 2020; Xu et al., 2021), we adopt the attention score of full context produced by TPLM and add it to S^l to take context knowledge into consideration.

We employ the final attention score $\boldsymbol{S}^l$ to calculate the output embedding matrix of the graph aggregator by

$$\boldsymbol{X}_1^l = \text{softmax}\left(S^l\right) V^l. \tag{4}$$

Feed-Forward Network. There are three intermediate outputs in l-th layer named $\boldsymbol{X}_1^l$, $\boldsymbol{X}_2^l$ and $\boldsymbol{X}_3^l$ respectively. The obtained embedding matrix $\boldsymbol{X}_1^l$ is input into a residual net followed by layer normalization to derive the output embedding matrix $\boldsymbol{X}_2^l$ by

$$\boldsymbol{X}_2^l = \text{LayerNorm}\left(\boldsymbol{X}_1^l + \boldsymbol{X}_0^l\right). \tag{5}$$

A fully connected feed-forward network is then applied to each token embedding in matrix $\boldsymbol{X}_2^l$ by

$$\boldsymbol{X}_3^l = \text{ReLU}\left(\boldsymbol{X}_2^l \boldsymbol{W}_1^l + b_1^l\right) \boldsymbol{W}_2^l + b_2^l, \tag{6}$$

which consists of two linear transformations with a ReLU activation in between. Finally, we apply residual connection and layer normalization to the obtained matrix $\boldsymbol{X}_3^l$ again to get the output embedding matrix for the next layer:

$$\boldsymbol{X}_0^{l+1} = \text{LayerNorm}\left(\boldsymbol{X}_2^l + \boldsymbol{X}_3^l\right). \tag{7}$$

To sum up, through L layers, the context of a document is encoded into the output contextual embeddings $\boldsymbol{X}_0^L$, where we can derive the embedding h_{m_i} of mention m_i. For entity embedding matrix $\boldsymbol{v} = \left\{v_{e_p}\right\}_{p=1}^n$, we compute each entity embedding v_{e_p} for entity e_p through its mention set $\mathcal{M}_p$ by

$$v_{e_p} = \sum_{j \in \mathcal{M}_p} h_{m_j}, \tag{8}$$

where $p = 1, 2, \ldots, n$. We extract the attention score calculated in L layers for each entity to compute the corresponding entity-aware context feature embedding $\boldsymbol{A}_e^{(L)} = \{\boldsymbol{a}_e^l\}_{l=1}^L$, where each $\boldsymbol{a}_e^l$ denotes the importance of context tokens to entity e in layer l. For entity pair e_i and e_j, we obtain the entity-pair-aware context feature embedding by

$$\boldsymbol{a}_{e_i} = \frac{1}{L}\sum_{l=1}^{L} \boldsymbol{a}_{e_i}^l, \quad \boldsymbol{a}_{e_j} = \frac{1}{L}\sum_{l=1}^{L} \boldsymbol{a}_{e_j}^l, \tag{9}$$

$$\boldsymbol{a}_{ij} = \boldsymbol{a}_{e_i} \odot \boldsymbol{a}_{e_j}, \quad \boldsymbol{u}_{ij} = \boldsymbol{a}_{ij}\boldsymbol{X}_0^L, \tag{10}$$

where $\odot$ denotes the element-wise multiplication between matrix, and $\boldsymbol{u}_{ij}$ is the final context feature embedding for entity pair e_i and e_j.

Classifier. We make pairs of the entities with each other (totally $n \times n - n$ pairs) and let the classifier calculate their classification scores by using their entity embeddings selected from v. For example, given v_{e_i}, v_{e_j} where i and j denote subject and object entity in an entity pair, we first map the embeddings of both entities to subject and object entity embeddings respectively to distinguish which entity is now considered as a subject entity while estimating the relations. Second, we feed the obtained two embeddings y_i and y_j to a bilinear layer in Equation 12 to compute the final classification scores. The classification scores can be calculated as follows:

$$y_i = \sigma\left(\boldsymbol{W}_s\left[v_{e_i}; \boldsymbol{u}_{ij}\right]\right), \quad y_j = \sigma\left(\boldsymbol{W}_o\left[v_{e_j}; \boldsymbol{u}_{ij}\right]\right), \tag{11}$$

$$\mathrm{P}\left(r \mid e_i, e_j\right) = \sigma\left(y_i^\top \boldsymbol{W}_r y_j + b_r\right), \tag{12}$$

where $\boldsymbol{W}_s \in \mathbb{R}^{d\times 2d}, \boldsymbol{W}_o \in \mathbb{R}^{d\times 2d}, \boldsymbol{W}_r \in \mathbb{R}^{d\times d}, b_r \in \mathbb{R}$ are parameters that need to be trained. $[;]$ denotes the vector concatenation, which takes the context feature of each entity pair into consideration. Finally, We use binary cross entropy as the loss function in our model.

$$\begin{aligned}\mathcal{L} = -\sum_{d\in\mathcal{D}}\sum_{s\neq o}\sum_{r_i\in\mathcal{R}} &\mathbb{I}\left(r_i = 1\right)\log P\left(r_i \mid \mathbf{e}_s, \mathbf{e}_o\right) \\ &+ \mathbb{I}\left(r_i = 0\right)\log\left(1 - P\left(r_i \mid e_s, e_o\right)\right),\end{aligned} \tag{13}$$

where $\mathcal{D}$ denotes the dataset and $\mathbb{I}\left(.\right)$ is indication function.

4. Experiments

4.1. Dataset

We conduct experiments on the public document-level dataset DocRED (Yao et al., 2019) and two popular document-level relation extraction datasets in the biomedical domain including Chemical-Disease Reactions (CDR) (Li et al., 2016) and Gene-Disease Associations (GDA) (Wu et al., 2019). The DocRED dataset has two versions. The smaller named DocRED contains 3,053 labeled instances, 1,000 development instances, and 1,000 testing instances respectively. The larger is based on DocRED and named $\text{DocRED}_{\text{distant}}$ since it is amplified with 101,873 distantly supervised training instances which can be deemed automatically annotated (Yao et al., 2019). CDR contains 500 training instances, 500 development instances, and 500 testing instances. GDA consists of 29,192 training instances, 5,839 development instances, and 1,000 testing instances. Over 61.1% relations in DocRED require reasoning. Among these relations, 26.6% require logical reasoning (relation established by a bridge entity), which achieves the highest proportion among relations requiring reasoning (Yao et al., 2019).

4.2. Experimental Settings

Our model is implemented under PyTorch (Paszke et al., 2019). We apply spaCy[†] to get dependency trees employed in Section 3.2. We use AdamW (Loshchilov and Hutter, 2019) as our optimizer and apply mixed precision training (Micikevicius et al., 2017) based on the Apex library[†]. We take Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) as the TPLM.

For fair comparison, we use the parameters and published code declared in the corresponding papers, where the learning rate equals 5e-5 in DocRED and 2e-5 in CDR and GDA respectively.

4.3. Evaluation Metric

Following the previous works (Nan et al., 2020), we use Ign F1, Intra-F1 and Inter-F1. The Ign F1 denotes the F1 score of the dev/test sets that excludes the relation facts appearing in training sets. The Inter-F1 denotes the F1 score of the relational facts between entity pairs that have no mentions in the same sentence. About 45% entity pairs are involved in the "Inter" condition in DocRED. The other entity pairs are included in the calculation of Intra-F1.

4.4. Baselines

We compare our model with the following baseline models.

Sequence-based Models. These models use different neural architectures including convolutional neural network (CNN) (LeCun et al., 2015) and bidirectional long short-term memory network (Bi-LSTM) (Schuster and Paliwal, 1997) to encode a document. Then

†. https://spacy.io
†. https://github.com/NVIDIA/apex

	Dev				Test	
Model	**Intra-F1**	**Inter-F1**	**Ign F1**	**F1**	**Ign F1**	**F1**
CNN(Yao et al., 2019)	51.87	37.58	41.58	43.45	40.33	42.26
LSTM(Yao et al., 2019)	56.57	41.47	48.44	50.68	47.71	50.07
BiLSTM(Yao et al., 2019)	57.05	43.49	48.87	50.94	48.78	51.06
Context-Aware(Yao et al., 2019)	56.74	42.26	48.94	51.09	48.40	50.70
$\text{BERT}_{\text{BASE}}$(Wang et al., 2019)	61.61	47.15	-	54.16	-	53.20
$\text{SSAN}_{\text{Decomp}}$(Xu et al., 2021)	-	-	56.68	58.95	56.06	58.41
$\text{SSAN}_{\text{Biaffine}}$(Xu et al., 2021)	-	-	57.03	59.19	55.84	58.16
$\text{SSAN}_{\text{Decomp}}$ + RSMAN(Yu et al., 2022)	-	-	57.22	59.25	57.02	59.29
$\text{WMCT-BERT}_{\text{BASE}}$	**67.66**	**53.37**	**59.44**	**61.29**	**59.13**	**61.11**

Table 2: Performance comparisons on DocRED.

	Dev				Test	
Model	**Intra-F1**	**Inter-F1**	**Ign F1**	**F1**	**Ign F1**	**F1**
$\text{BERT}_{\text{BASE}}^{\dagger}$(Xu et al., 2021)	68.87	55.29	60.45	62.49	55.31	62.63
$\text{GAIN-BERT}_{\text{BASE}}^{\dagger}$(Zeng et al., 2020)	-	-	61.38	63.49	60.60	62.87
$\text{ATLOP-BERT}_{\text{BASE}}^{\dagger}$(Zhou et al., 2020)	70.63	56.04	62.23	64.19	61.79	63.90
$\text{SSAN-BERT}_{\text{BASE}}^{\dagger}$(Xu et al., 2021)	70.29	55.72	61.52	63.55	61.19	63.40
$\text{WMCT-BERT}_{\text{BASE}}^{\dagger}$	**71.39**	**56.01**	**62.57**	**64.45**	**62.04**	**64.36**

Table 3: Performance comparisons on $\text{DocRED}_{\text{distant}}$. Signal † denotes that the model is pre-trained on $\text{DocRED}_{\text{distant}}$.

they usually classify an entity pair with a bilinear function based on the encoded entity embeddings.

Graph-based methods. These models fine-tune the TPLM on document-level relation extraction under different model structures. Usually, their first step is to encode a document by TPLM. In the second step, they utilize various model structures to capture the different statistical significance of data (Wang et al., 2019). Some of these models conduct an inference graph in the second step by co-reference links (Sahu et al., 2019), dependency trees (Nan et al., 2020; Guo et al., 2019) or knowledge (Christopoulou et al., 2019) including LSR, GAIN, and CorefBERT. The others build a fully connected graph instead and use the attention mechanism to calculate every edge weight in the graph (Veličković et al., 2018). More recently, SSAN (Xu et al., 2021) and RSMAN (Yu et al., 2022) focus on facilitating graph-based methods by improving their common backbone: Transformer. They infuse the Transformer with graphs to enhance its reasoning ability.

4.5. Results on DocRED

In this section, we conduct experiments on DocRED and $\text{DocRED}_{\text{distant}}$ to evaluate the effectiveness of different kinds of methods including graph-based methods and sequence-based models. The experimental results are shown in Table 2 and Table 3.

We can observe from Table 2 that our proposed WMCT outperforms the baseline methods that aim to enhance the reasoning ability of the Transformer. Specifically, WMCT surpasses SSAN and RSMAN by at least 3.07/1.82 F1 and 2.70/2.11 Ign F1 on the test set

Dataset	Model	F1	Intra-F1	Inter-F1
CDR	CNN (Gu et al., 2017)	61.3	57.2	11.7
	CNN+CNNchar (Nguyen and Verspoor, 2018)	62.3	-	-
	BRAN (Verga et al., 2018)	62.1	-	-
	GCNN (Sahu et al., 2019)	58.6	-	-
	LSR (Nan et al., 2020)	61.2	66.2	50.3
	EoG (Christopoulou et al., 2019)	63.6	68.2	50.9
	LSR w/o MDP nodes (Nan et al., 2020)	64.8	68.9	53.1
	WMCT-BERT$_{\text{BASE}}$	**66.4**	**71.7**	**53.2**
GDA	EoG(NoInf) (Christopoulou et al., 2019)	74.6	79.1	49.3
	LSR (Nan et al., 2020)	79.6	83.1	49.6
	EoG(Full) (Christopoulou et al., 2019)	80.8	84.1	54.7
	EoG(Sent) (Christopoulou et al., 2019)	81.5	85.2	50.0
	WMCT-BERT$_{\text{BASE}}$	**82.0**	**85.6**	**59.9**

Table 4: Model performance on the test set of CDR and GDA dataset.

of DocRED, respectively. Since SSAN only adopts a single kind of graph containing entity-level information (co-occurrence and coreference links) while our proposed WMCT contains various kinds of graphs beyond entities, the results indicate that our proposed WMCT exploits the knowledge more effectively than SSAN. Furthermore, WMCT performs better than RSMAN which further improves the information aggregation of entities in SSAN. It also implies that merely exploring a single kind of knowledge will constrain the performance and potential of the Transformer. The effectiveness of various kinds of knowledge is more significant than that of improving the Transformer by using a single kind of knowledge.

By investigating the datasets, we find that DocRED just contains 3,053 labeled instances, which implies that the latent knowledge is very limited. On the contrary, the dataset of DocRED$_{\text{distant}}$ contains 101,873 distantly supervised training instances based on DocRED. It means that more plentiful knowledge can be explored for performance improvement. However, the amount of noise information also becomes larger as the dataset is automatically obtained without manually labeling. In this scenario, the guidance of the knowledge, which is exploited in an unsupervised way, is of the essence. The experimental results on DocRED$_{\text{distant}}$ shown in Table 3 demonstrate that our improved Transformer possesses stronger reasoning ability originating from the guidance of various kinds of knowledge. WMCT surprisingly surpasses the two representative methods in DocRE which adopt additional modules (e.g., graph convolutional network, adaptive threshold, and localized context pooling) to improve their performance. This also suggests directions for improvement of future methods that are based on the Transformer.

4.6. Results on CDR and GDA

We evaluate our model on CDR and GDA compared with the stat-of-the-arts on these datasets. The results are shown in Table 4, which also demonstrates the effectiveness of our model, and indicates that pronouns and evidence words are important in document-level relation extraction. According to the experimental results, our proposed model outperforms the previous work (Christopoulou et al., 2019) by 1.6 F1 on CDR and outperforms EoG by 0.5 F1 on GDA, which demonstrates the effectiveness of our formulated graphs and the joint methods.

Model	Intra-F1	Inter-F1	Ign F1	F1
Ours all-graph	**71.09**	**56.13**	**62.53**	**64.45**
w/o $\mathcal{G}_1$	70.72	56.55	62.49	64.36
w/o $\mathcal{G}_2$	70.80	55.67	62.13	64.00
w/o $\mathcal{G}_3$	70.87	56.32	62.47	64.25
w/o $\mathcal{G}_4$	70.82	55.96	62.29	64.17
w/o $\mathcal{G}_5$	70.77	55.40	62.01	63.88

Table 5: The ablation study on DocRED.

4.7. Ablation Study and Discussion

We exhibit ablation studies of our model and conduct experiments for a comprehensive discussion. As a key part of our model, the multi-channel aggregator allocates weights adaptively to different graphs. It is instructive to explore whether graphs indeed impact. For this purpose, we design two models. One model only picks one kind of graph. The other model excludes a graph and takes the rest kinds of the graphs into consideration. The experimental results are shown in Figure 4 and Table 5 respectively. In Figure 4, we evaluate the effectiveness of a single graph by adjusting its weight. Meanwhile, as Table 5 and Figure 4 show each graph impacts to the relation extraction, where $\mathcal{G}_2$, $\mathcal{G}_4$ and $\mathcal{G}_5$ effects the most. With only one kind of graph, the performance of our model drops at least by 1.20 Ign F1 and 1.43 F1 on the DocRED test set. Specifically, the exclusions of $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, $\mathcal{G}_4$ or $\mathcal{G}_5$ lead to the declines of model performance by 0.09, 0.45, 0.20, 0.28 and 0.57 in F1 respectively, while the involvements of $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, $\mathcal{G}_4$ or $\mathcal{G}_5$ improve the performance of $\text{BERT}_{\text{BASE}}$ by 0.73, 0.86, 0.78, 1.10 and 1.26 in F1 respectively. *It means that graphs extended by dependency words, evidence words, or a combination of pronouns and evidence words are more important than those that only contain mentions and pronouns.*

To further investigate the effect of each graph, we show the weights of the graphs distributed in 12 layers of our model in Figure 5. We observe that the $\mathcal{G}_2$, $\mathcal{G}_4$ and $\mathcal{G}_5$ graphs are always assigned heavier weights by the aggregator introduced in the weighted multi-channel Transformer section throughout 12 layers. The results shown in Figure 5 indicate the significance of graphs sorted in ascending order represented by $\mathcal{G}_1, \mathcal{G}_3, \mathcal{G}_2, \mathcal{G}_4, \mathcal{G}_5$ (the last layer impacts the most). The figure shows that the influence of $\mathcal{G}_1$ and $\mathcal{G}_3$ is limited when compared with the other graphs. The former is composed of mentions which are widely used in previous work, and the latter consists of mentions and pronouns which has a similar nature to the former about referring entities. Consequently, we can derive that mentions or pronouns are not enough for document-level relation extraction, and the improvement will be more significant while combining pronouns with evidence words because they can be bridges to connect the mentions for unobstructed information transmitting, and thus enrich the knowledge and extend the graphs for relation inference. In other words, a more informative graph leads to more significant improvement due to the complete logic chain.

To summarize, according to the experimental results in this section, we can derive three conclusions. First, it is that not only mentions but also dependency words, pronouns, and evidence words are also very informative for relation inference. Second, the diversity of graphs (knowledge) is helpful for document-level relation extraction. Third, we argue that graphs, especially with richer knowledge, are indeed important and can contribute to the final improvement while applying an appropriate method.

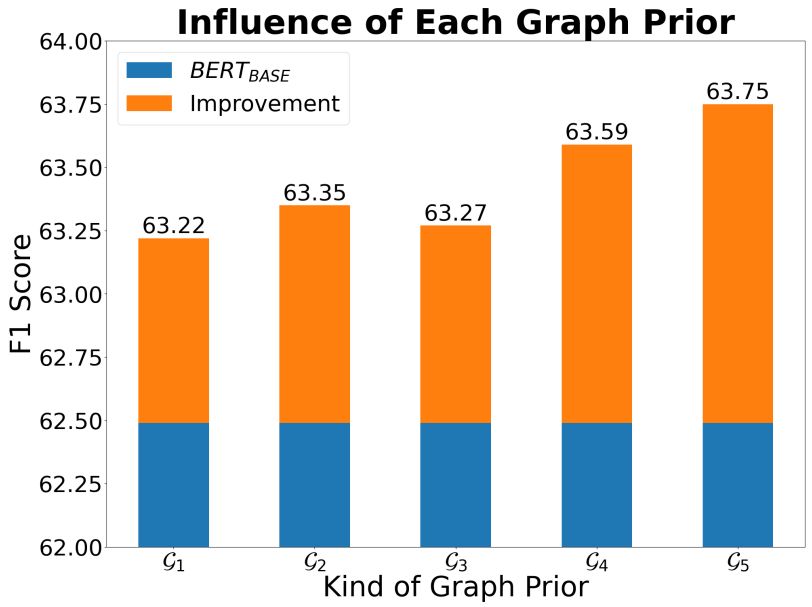

Figure 4: The influence of graphs. We evaluate the influence of five graphs one after another based on $\text{BERT}_{\text{BASE}}$ on the development set.

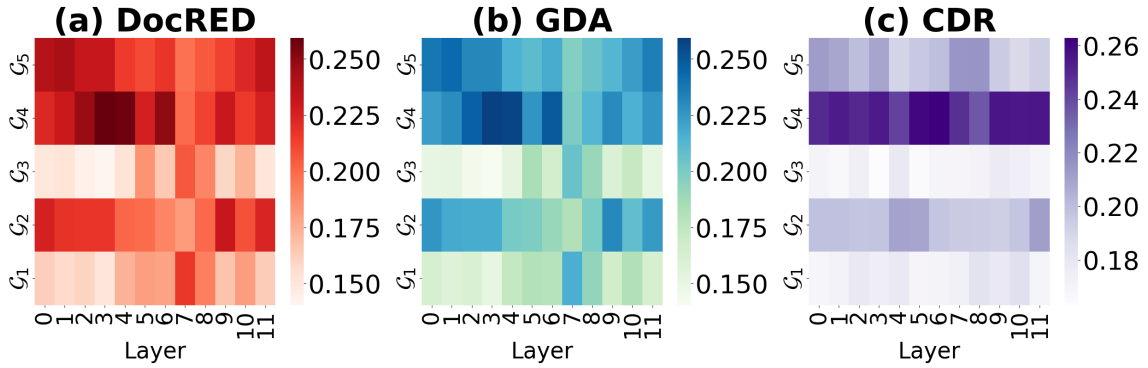

Figure 5: The weights of five graphs distributed in different layers. Our all-graph model is trained on DocRED, CDR, and GDA respectively.

5. Conclusion

In this work, we explore various kinds of knowledge represented by graphs for document-level relation extraction. We propose the weighted multi-channel Transformer to effectively aggregate such graphs in an adaptive way. Our model deals with the document-level relation extraction problems from both contextual inferring and knowledge inferring interactively and simultaneously. The experimental results demonstrate the importance of informative knowledge and the effectiveness of our proposed model. For future works, we plan to adapt

our model and newly constructed graphs to alleviate the inference challenge in multi-hop question answering and reading comprehension.

References

Nicola De Cao, Wilker Aziz, and Ivan Titov. Question Answering by Reasoning Across Documents with Graph Convolutional Networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2306–2317, 2019.

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 4924–4935, 2019. doi: 10.18653/v1/D19-1498.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2018.

Jinghang Gu, Fuqing Sun, Longhua Qian, and Guodong Zhou. Chemical-induced disease relation extraction via convolutional neural network. *Database*, 2017(1), 2017.

Zhijiang Guo, Yan Zhang, and Wei Lu. Attention Guided Graph Convolutional Networks for Relation Extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 241–251, 2019. doi: 10.18653/v1/P19-1024.

Robin Jia, Cliff Wong, and Hoifung Poon. Document-Level N-ary Relation Extraction with Multiscale Representation Learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3693–3704, 2019.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553): 436–444, 2015. doi: 10.1038/nature14539.

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, 2016.

I. Loshchilov and F. Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2019.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory F. Diamos, Erich Elsen, David García, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed Precision Training. In *International Conference on Learning Representations*, 2017.

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, 2020. doi: 10.18653/v1/2020.acl-main.141.

Dat Quoc Nguyen and Karin Verspoor. Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings. In *Proceedings of the BioNLP 2018 Workshop*, pages 129–136, 2018. doi: 10.18653/v1/W18-2314.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. Inter-sentence Relation Extraction with Document-level Graph Convolutional Neural Network. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4309–4316, 2019. doi: 10.18653/v1/P19-1423.

Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997. doi: 10.1109/78.650093.

Livio Baldini Soares, Nicholas Arthur FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, 2019. doi: 10.18653/v1/P19-1279.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. In *International Conference on Learning Representations*, 2018.

Patrick Verga, Emma Strubell, and Andrew McCallum. Simultaneously Self-Attending to All Mentions for Full-Abstract Biological Relation Extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 872–884, 2018. doi: 10.18653/v1/N18-1080.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Yang Wang. Fine-tune Bert for DocRED with Two-step Process. *arXiv preprint arXiv:1909.11898*, 2019.

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 *International Conference on Research in Computational Molecular Biology*, pages 272–284, 2019.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. SAIS: Supervising and augmenting intermediate steps for document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2395–2409, Seattle, United States, July 2022. Association for Computational Linguistics.

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. Entity Structure Within and Throughout: Modeling Mention Dependencies for Document-Level Relation Extraction. In *AAAI*, pages 14149–14157, 2021.

Baosong Yang, Jian Li, Derek F. Wong, Lidia S. Chao, Xing Wang, and Zhaopeng Tu. Context-aware self-attention networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 387–394, 2019.

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, 2019. doi: 10.18653/v1/P19-1074.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. Coreferential Reasoning Learning for Language Representation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7170–7186, 2020. doi: 10.18653/v1/2020.emnlp-main.582.

Jiaxin Yu, Deqing Yang, and Shuyu Tian. Relation-specific attentions over entity mentions for enhanced document-level relation extraction. In *North American Chapter of the Association for Computational Linguistics*, 2022. doi: 10.18653/v1/2022.naacl-main.109.

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, 2020. doi: 10.18653/v1/2020.emnlp-main.127.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, 2019. doi: 10.18653/v1/P19-1139.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. Document-Level Relation Extraction with Adaptive Thresholding and Localized Context Pooling. In *AAAI*, pages 14612–14620, 2020.