SagDRE: Sequence-Aware Graph-Based Document-Level Relation Extraction with Adaptive Margin Loss

Ying Wei
Iowa State University
Ames, IA, U.S.
yingwei@iastate.edu

Qi Li
Iowa State University
Ames, IA, U.S.
qli@iastate.edu

ABSTRACT

Relation extraction (RE) is an important task for many natural language processing applications. Document-level relation extraction task aims to extract the relations within a document and poses many challenges to the RE tasks as it requires reasoning across sentences and handling multiple relations expressed in the same document. Existing state-of-the-art document-level RE models use the graph structure to better connect long-distance correlations. In this work, we propose SagDRE model, which further considers and captures the original sequential information from the text. The proposed model learns sentence-level directional edges to capture the information flow in the document and uses the token-level sequential information to encode the shortest paths from one entity to the other. In addition, we propose an adaptive margin loss to address the long-tailed multi-label problem of document-level RE tasks, where multiple relations can be expressed in a document for an entity pair and there are a few popular relations. The loss function aims to encourage separations between positive and negative classes. The experimental results on datasets from various domains demonstrate the effectiveness of the proposed methods.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; **Information extraction**.

KEYWORDS

relation extraction, document-level RE, sequence information, graph

ACM Reference Format:
Ying Wei and Qi Li. 2022. SagDRE: Sequence-Aware Graph-Based Document-Level Relation Extraction with Adaptive Margin Loss. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3534678.3539304

1 INTRODUCTION

Relation extraction (RE) aims to extract the relations among entities from text. It plays an important role in various natural language processing (NLP) tasks such as knowledge graph construction [8], question answering [42], and text summarization [12]. In the RE tasks, there are two specific scenarios: sentence-level relation extraction and document-level relation extraction [24]. Sentence-level relation extraction focuses on relationships expressed within sentences, while document-level relation extraction aims to extract relationships across sentence boundaries.

There are unique challenges for document-level RE compared to sentence-level RE. In a document, an entity can be mentioned multiple times, but only a few mentions may contribute to the targeted relation reasoning, making it harder for the RE model to focus on the most relevant parts in the document. The mentions of entities may also locate in different sentences, which requires the RE model to effectively encode long-distance information [29].

To address these challenges, some methods propose to construct a graph to represent the document and achieve the state-of-the-art performances [11, 23, 29]. However, these graph-based methods use regular graph structures with bi-directional edges for effective feature propagation, and neglect the sequence features in the original text, an important characteristic of languages. These graphs cannot encode the sequential information due to its permutation invariance property [28], which can downgrade the performance for document-level RE tasks.

Another challenge of document-level RE task is that the document may express multiple relations for the same entity pair. This leads to the multi-label problem. Existing methods convert the multi-label problem as multiple binary classification problems, and assign the corresponding label if the predicted probability is higher than a global threshold shared for all entity pairs. However, the threshold is mostly determined heuristically or tuned on validation set. The resulting threshold may not be optimal for all instances. Another commonly observed phenomenon for multi-label learning problems is the long-tail distribution of the labels. Many relations have only a few training examples, whereas only a few relations have sufficient training examples. Regular probability distribution-based loss estimations tend to over-fit the model with the popular relations but under-fit for the unpopular relations.

This work[1] proposes a Sequence-Aware Graph-based Document-level Relation Extraction model (SagDRE) to consider original text sequential information for document-level relation extraction tasks. Given a document, we first construct a sequence-aware document graph with directed edges, which can capture sentence-level sequential information in the document. In particular, forward edges from previous sentence roots to later ones are added with edge weights learned by an attention mechanism. Based on the constructed document graph, we adopt graph convolutional neural network and multi-head self attention to encode local and global features. To capture the token-level sequential information, SagDRE finds the k shortest paths from the head entity to the tail entity on the document graph and then reconstruct the paths with the original token

[1]The code is publicly available at https://github.com/IAmHedgehog/SagDRE.

KDD '22, August 14–18, 2022, Washington, DC, USA

ACM ISBN 978-1-4503-9385-0/22/08.
https://doi.org/10.1145/3534678.3539304

orders and auxiliary tokens. The paths are encoded using LSTM and a multi-head attention layer is used to aggregate paths such that relevant paths are emphasized. The resulting path encoding is concatenated with other features for prediction.

To address the long-tailed multi-label problems in document-level RE tasks, we propose a novel adaptive margin loss based on the idea of Hinge Loss. In particular, we learn a separation class for each pair of entities between positive classes and negative classes. The adaptive margin loss is invoked if an example is wrongly classified or classified within a margin to the separation class. The optimization based on this loss will encourage the separation between positive and negative classes via the separation class.

In empirical studies, we use three benchmark document-level RE datasets from both general and biomedical domains to evaluate the proposed method. The results show that the proposed SagDRE consistently outperforms state-of-the-art models. The ablation studies show that the adaptive margin loss and the sequence components are the most important contributors to the overall model performances.

The main contributions are summarized as:

- We propose SagDRE that considers and incorporates the sentence-level and token-level sequential information from the text in the graph-based document RE model.
- We propose adaptive margin loss for multi-label learning problems, which encourages the maximum separation between positive and negative classes via a separation class.
- Empirical studies on three document-level relation extraction datasets from various domains demonstrate the effectiveness of the proposed method.

2 RELATED WORK

The relation extraction task has been studied in the past decades. The applications of deep learning methods have significantly advanced the development for the task [14, 24]. Recently the research on document-level relation extraction tasks has drawn more and more attention. Comparing with sentence-level RE tasks, document-level RE tasks have a wider range of applications [40] but extracting document-level relations is more challenging since cross-sentence learning usually requires effective long-distance feature encoding and reasoning [29].

To tackle this challenge, some methods [35, 41, 45] apply BERT [7] for more informative contextual token encoding. Tang et al. [35] propose a hierarchical inference network from entity, sentence and document levels using BERT representations. Ye et al. [41] explicitly encode the coreference information to enhance the coreferential reasoning ability of BERT. Zhou et al. [45] propose an adaptive-thresholding loss, which learns an adaptive threshold to separate positive and negative classes.

Besides BERT-based methods, another line of approaches proposes to use the graph structure to shorten the distances between entities in the document [5, 11, 15, 23, 29]. Sahu et al. [29] propose the first work to adopt graph structure in document-level RE tasks. It uses linguistic tools to build various edges, such as co-reference edges, which embed inter-sentence and intra-sentence dependencies, and applies a graph convolutional neural network for feature learning. Unlike previous methods that use linguistic tools for graph construction, Guo et al. [11] and Sahu et al. [30] use attention mechanisms to construct edges in the graph. Guo et al. [11] employ an attention mechanism to automatically learn attending relevant sub-structures in the graph for relation reasoning. Instead of constructing token-level graphs, Zeng et al. [43] propose to build two graphs, including mention-level and entity-level graphs, to predict relations. Christopoulou et al. [5] construct a graph with different nodes and edges and applies edge-oriented graph neural networks for document-level relation extraction. Nan et al. [23] apply an iterative refinement strategy to aggregate multi-hop information for reasoning. Compared to previous methods that use graph neural networks to encode features, Zhou et al. [44] propose a global context-enhanced graph convolutional network to consider global context information for relation reasoning.

However, most existing works use regular graph structures, which cannot capture the sequential information in the original text. The permutation invariance property of graph structure [28] makes it hard to embed sequential information naturally, which is critical in extracting document-level relation information. This work addresses this issue by encoding sequential information in graphs and directional path information for document-level relation reasoning.

3 PRELIMINARY

In this section, we introduce graph neural networks and formulate the document-level RE task.

3.1 Graph Convolutional Networks

Given a graph $\mathcal{G} = (V, E)$, V and E represent the node set and edge set in the graph, respectively. Each node v has a feature vector $\boldsymbol{x}_v$. An adjacency matrix $\boldsymbol{A}$ is used to represent graph connections. Graph Neural Networks (GNNs) learn feature representations for nodes and the graph from the graph structure and node features. Most existing graph neural networks follow a neighborhood aggregation learning strategy, where each node iteratively aggregates features from its neighborhood and updates its features [13, 39]. Specifically to Graph Convolutional Networks (GCN), the ℓ^{th} GCN layer is defined as:

$$\boldsymbol{H}^{(\ell+1)} = \sigma\left(\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{H}^{(\ell)}\boldsymbol{W}^{(\ell)}\right), \tag{1}$$

where $\boldsymbol{A}$ is the adjacency matrix, $\boldsymbol{D}$ is the degree matrix, $\boldsymbol{H}^{(\ell)}$ is the input feature matrix at layer ℓ, $\boldsymbol{W}^{(\ell)}$ is the trainable parameter matrix, and $\sigma(\cdot)$ represents an activation function.

3.2 Relation Extraction Task Formulation

We formally formulate the task of document-level relation extraction as follows. A document $\mathcal{D}$ contains N sentences $\{s_1, s_2, \ldots, s_N\}$. s_i is the i^{th} sentence, which includes P_i tokens: $\{w_{i,1}, w_{i,2}, \ldots, w_{i,P_i}\}$. $w_{i,j}$ represents the j^{th} word in the i^{th} sentence. Each token $w_{i,j}$ is initially populated with an embedding feature vector $\boldsymbol{x}_{i,j}$. An entity e_k can have Q_k mentions $\{m_{k,1}, m_{k,2}, \cdots, m_{k,Q_k}\}$ in this document, where $m_{k,a}$ refers to the a^{th} span of tokens for the entity e_k.

Given a document $\mathcal{D}$ and a pair of entities (e_h, e_t), where e_h and e_t are head entity and tail entity, respectively, the RE task aims to predict the relations for this pair of entities based on the

document. The pre-defined relations contain labels $\{R_1, \cdots, R_C\}$, where R_i $(1 \leq i \leq C)$ represents the i^{th} pre-defined relationships. A RE model should output either an empty set or a subset of relations from $\{R_1, \cdots, R_C\}$ for each (e_h, e_t) based on the document. The relations between two entities exist if any pair of their mentions expresses the corresponding relationships. In the testing time, a relation extraction model predicts relations between all pairs of entities in a document.

4 SAGDRE

This section introduces a sequence-aware graph-based document-level relation extraction network (SagDRE), which consists of four components: sequence-aware graph construction (Section 4.1), local and global feature encoding (Section 4.2), sequence-aware path encoding (Section 4.3), and relation prediction head (Section 4.4). Figure 1 illustrates the architecture of the proposed network. In Section 4.5, we propose a novel adaptive margin loss especially designed for multi-label multi-class learning tasks such as document-level RE.

4.1 Sequence-Aware Graph Construction

Many existing document-level RE methods adopt graph structures using dependency parsers [4, 31] to construct the document graph with undirected edges. The undirected graph increases the connectivity between the head-tail entity pairs, and thus can better capture long-distance information for document-level RE tasks. However, the language sequence information cannot be explicitly reflected in this type of constructed graphs. Moreover, the permutation invariance property of a bi-directional graph makes it more challenging to capture sequential information expressed in the text [28].

It is critical to encode the original sequential information from the text as changing the order of words or the order of sentences can lead to semantic changes of relations for a pair of entities. If the sequential information in the text is neglected, it can negatively impact the performance of graph-based relation extraction models. To maintain high connectivity between the head-tail entity pairs and effectively encode original sequential information, we propose to construct a sequence-aware document graph that can capture the sentence-level sequential information.

Given a document, we first encode contextual features of each token in the document:

$$\begin{aligned} \boldsymbol{H} &= [\boldsymbol{h}_{1,1}, \cdots, \boldsymbol{h}_{N,P_N}] \\ &= \text{Encoder}([\boldsymbol{x}_{1,1}, \cdots, \boldsymbol{x}_{N,P_N}]), \end{aligned}$$

where $\boldsymbol{x}_{i,j}$ is the word embedding for the j^{th} token of the i^{th} sentence in the document and $\boldsymbol{h}_i$ is the encoded feature representation for the same token. This encoder can be a pre-trained BERT model [7] or LSTM model.

Then, we construct a document graph. This graph contains two types of nodes: token nodes and entity nodes. Each token in the document corresponds to a token node and its encoded features are used as node features. Each entity in the document corresponds to an entity node. Its node features are calculated by averaging the features of tokens in its mentions.

There are two types of edges in the graph: bi-directed edges and directed edges. The bi-directed edges are formed based on three sources: dependency syntax tree, adjacent sentence roots, and entity-token relation. Each sentence in the document is fed into a dependency parser, which generates a dependency syntax tree. Bi-directed edges are added between each pair of connected tokens in the syntax tree. Then the dependency syntax tree roots of adjacent sentences are connected by bi-directed edges since there are close context relationships between adjacent sentences. Finally, bi-directed edges between each entity and tokens of its mentions are added. In this graph, the weights for bi-directed edges are 1, which indicates strong connections among nodes.

The directed edges are added to capture the sentence-level sequential information in the document. In particular, we add forward edges from previous sentence root nodes to later ones, since information of a document tends to propagate from earlier sentences to later ones naturally. However, not all sentences are closely related to earlier sentences, we apply an attention mechanism to automatically learn the closeness between each pair of sentences for the given tasks and use the resulting similarity scores as weights for these directed edges.

Specifically, given two sentences root nodes i and j, we compute the weight $A_{i,j}$ for the directed edge from node i to node j based on their feature vectors:

$$A_{i,j} = \frac{\boldsymbol{h}_i \cdot \boldsymbol{h}_j}{||\boldsymbol{h}_i|| \cdot ||\boldsymbol{h}_j||}, \tag{2}$$

where $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ are the encodings of corresponding tokens i and j. Using these learned edges weights, our relation extraction model can automatically identify important logic flows from earlier sentences to later sentences. Note that if i and j are roots of adjacent sentences, $A_{i,j}$ and $A_{j,i}$ are always 1 as there is a bi-directed edge between them.

4.2 Local and Global Feature Encoding

Based on the constructed document graph with feature matrix $\boldsymbol{H}$, and adjacency matrix $\boldsymbol{A}$, we extract graphical features locally and globally. We employ graph convolutional network layers (GCN) [13] for feature aggregation and encoding. Since GCN layers only aggregates information from neighboring nodes, the resulting features can be considered as local feature encoding, providing information from a local context.

We also employ multi-head self attention layers [36] on contextual embeddings obtained from the GCN encode. Multi-head self attention layer can attend over all nodes in the input graph and thus can update the features from the global view, extracting features over the entire document graph. The local and global feature embedding are combined to update features of each node in the graph. We formulate this local and global feature extraction process at layer ℓ as:

$$\begin{aligned} \boldsymbol{H}'_1 &= \text{GCN}(\boldsymbol{H}^{(\ell)}, \boldsymbol{A}), \\ \boldsymbol{H}'_2 &= \text{Attn}(\boldsymbol{W}_Q \boldsymbol{H}^{(\ell)}, \boldsymbol{W}_K \boldsymbol{H}^{(\ell)}, \boldsymbol{W}_V \boldsymbol{H}^{(\ell)}), \\ \boldsymbol{H}^{(\ell+1)} &= \boldsymbol{H}'_1 + \boldsymbol{H}'_2, \end{aligned}$$

where $\boldsymbol{H}^{(\ell)}$ is the input feature matrix of layer ℓ, $\boldsymbol{W}_Q$, $\boldsymbol{W}_K$, and $\boldsymbol{W}_V$ are trainable weights. GCN and Attn represent a GCN layer and an attention layer, respectively.

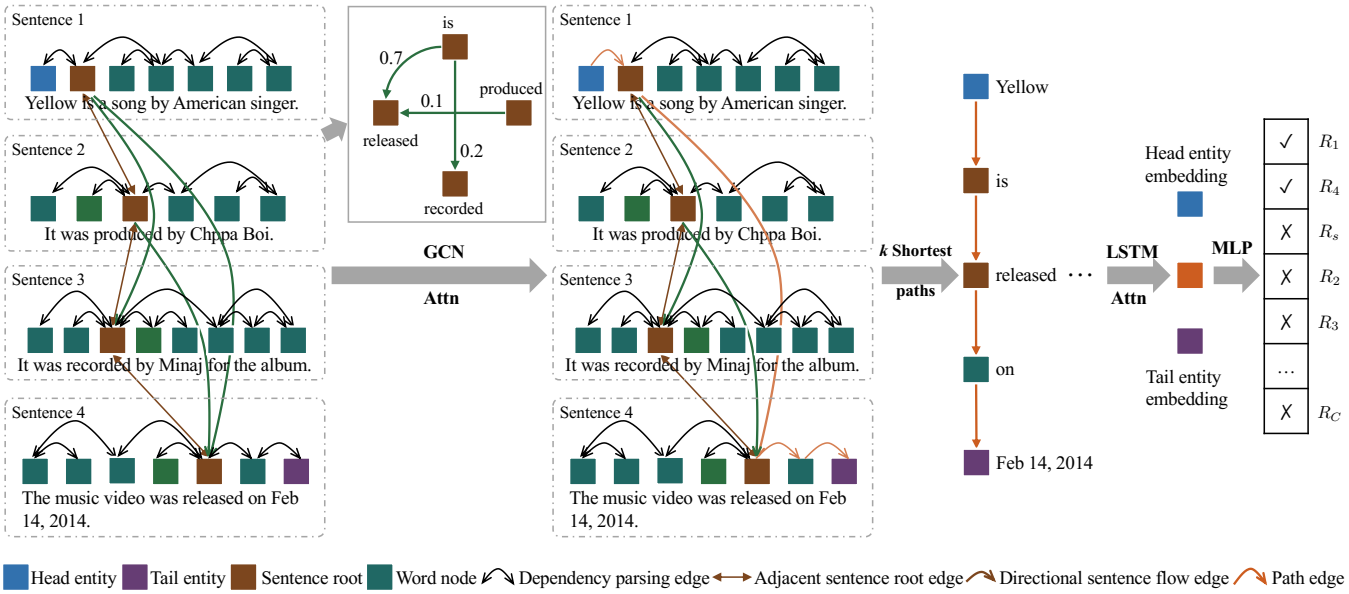


Figure 1: Illustration of the sequence-aware document-level relation extraction network. Given an input document, each token obtains its initial feature embedding from the encoder. Then a document graph is constructed. The directed cross-sentence edges are added (green edges) and their edge weights are computed using an attention mechanism. We stack several GCN layers and attention layers to learn feature representations from both local and global perspectives. Then, we extract K shortest paths from the head entity to the tail entity from the graph, encoded by LSTM, and fused by an attention layer, resulting in a path embedding. Finally, the entity and path embeddings are fed into an MLP for prediction. The adaptive margin loss is applied.

4.3 Sequence-Aware Path Encoding

The document graphs constructed can resolve the issue of long distance between entities by increasing entity connectives. However, the graph can also connect less-related information and confuse the model. To focus on the most relevant information and encode original token-level sequential information, we propose to construct a set of sequence-aware paths from the head entity to the tail entity.

Given a graph and a pair of entities (e_h, e_t), the paths from e_h to e_t in the graph usually contain the relevant reasoning information for their relationships. We select the top K shortest paths as they tend to contain the most information. We denote the k^{th} shortest path as $P^k_{h,t} = [e_h, n_{k,1}, \cdots, n_{k,d}, e_t]$, where $n_{k,j}$ represents the j^{th} node on the k^{th} shortest path. These shortest paths may neglect some important structural words for relation reasoning though such as "near" and "outside". To enrich the sequence-aware paths and include more informative nodes, we augment each extracted path with adposition words attached to this path. That is, given the k^{th} shortest path, we add the neighboring adposition word nodes of each node $n_{k,j}$ in $P^k_{h,t}$, which leads to the augmented path $Q^k_{h,t}$.

To encode the original token-level sequential information, we order the nodes in each path by their original sequential order in the text, which leads to $\bar{Q}^k_{h,t}$. We apply a directional LSTM layer to encode features and a max-pooling layer to obtain the feature representations of each path. The proposed sequence-aware path encoding for the k^{th} shortest path is formulated as:

$$\overrightarrow{\boldsymbol{u}}^k_j = \text{LSTM}(\overrightarrow{\boldsymbol{h}}^k_j) \tag{3}$$

$$\boldsymbol{p}^k_{h,t} = \max(\overrightarrow{\boldsymbol{u}_h}, \overrightarrow{\boldsymbol{u}}^k_1, \cdots, \overrightarrow{\boldsymbol{u}}^k_d, \overrightarrow{\boldsymbol{u}_t}), \tag{4}$$

where $\boldsymbol{u}^k_j$ represents the LSTM hidden representations of the j^{th} node in $\bar{Q}^k_{h,t}$.

Since not all paths contain relevant information for relation reasoning, we employ a multi-head attention layer over the K shortest paths encodings to identify the most relevant ones. We formulate this process as:

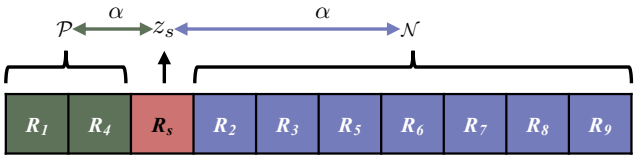
$$\boldsymbol{P} = [\boldsymbol{p}^1_{h,t}; \cdots; \boldsymbol{p}^K_{h,t}], \tag{5}$$

$$\boldsymbol{p}_{h,t} = \text{Attn}(\boldsymbol{W}'_Q(\boldsymbol{e}_h - \boldsymbol{e}_t), \boldsymbol{W}'_K\boldsymbol{P}, \boldsymbol{W}'_V\boldsymbol{P}), \tag{6}$$

where $\boldsymbol{W}'_Q$, $\boldsymbol{W}'_K$, and $\boldsymbol{W}'_V$ are trainable weights and Attn represents an attention layer. This attention layer takes the vector from head to tail entity encoding as the query, which is widely used to reflect the relations of the two entities [22]. As a result, the attention layer responds by the weighted aggregating path encoding with relevant paths emphasized.

4.4 Relation Prediction Head

We use a relation prediction head to predict relations for a pair of entities. The prediction is based on both entities' feature representations and their aggregated path encoding. Following previous methods [43], we concatenate the entity encoding of two entities $(\boldsymbol{e}_h, \boldsymbol{e}_t)$, the absolute values of subtraction of two entity encoding



$R(e_i, e_j) = R_1, R_4$

Figure 2: Illustration of our proposed adaptive margin loss. Given an entity pair (e_i, e_j) in a document, their relations are R_1 and R_4.

($|\boldsymbol{e}_h - \boldsymbol{e}_t|$), the element-wise feature multiplication ($\boldsymbol{e}_h \odot \boldsymbol{e}_t$), and the sequence-aware path encoding ($\boldsymbol{p}_{h,t}$), which leads to an overall encoding for this entity pair:

$$\boldsymbol{I}_{h,t} = [\boldsymbol{e}_h; \boldsymbol{e}_t; |\boldsymbol{e}_h - \boldsymbol{e}_t|; \boldsymbol{e}_h \odot \boldsymbol{e}_t; \boldsymbol{p}_{h,t}]. \tag{7}$$

We compute the prediction values $\boldsymbol{z} \in \mathcal{R}^C$ for all relation classes:

$$\boldsymbol{z} = \boldsymbol{W}_2\sigma_1(\boldsymbol{W}_1\boldsymbol{I}_{h,t} + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{8}$$

where $\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{b}_1, \boldsymbol{b}_2$ are trainable parameters, and σ_1 is an element-wise activation function. Additionally, we predict a separation class R_s to separate the positive classes and negative classes:

$$z_s = \boldsymbol{W}_4\sigma_2(\boldsymbol{W}_3\boldsymbol{I}_{h,t} + \boldsymbol{b}_3) + \boldsymbol{b}_4, \tag{9}$$

where $\boldsymbol{W}_3, \boldsymbol{W}_4, \boldsymbol{b}_3, \boldsymbol{b}_4$ are trainable parameters, and σ_2 is an element-wise activation function. During prediction, for each entity pair, SagDRE outputs a set of classes $[c_1, c_2, \cdots, c_d]$ where $z_{c_j} > z_s (1 \leq j \leq d)$. Note that if no class has a value bigger than z_s, SagDRE outputs an empty set, indicating no relationships between the given entity pair (e_i, e_j).

4.5 Adaptive Margin Loss

Most existing relation extraction models output $P(R_i|e_h, e_t, \mathcal{D})$ for the probability of that the i-th relation R_i exists for the pair of entities (e_h, e_t), which requires a pre-determined global threshold to convert probabilities into relation labels. Some methods [18, 23, 25] use heuristic threshold or learn a global threshold with the highest F1 score on the validation set. However, the global threshold may not be optimal for all instances and introduce errors. To address this issue, Zhou et al. [45] introduce an extra threshold class as a threshold to separate positive classes and negative classes. However, such probability distribution-based methods may suffer when the long tail problem occurs, where a majority of labels are only associated with a small number of training examples. Even when the prediction is correct (higher than the threshold), probability distribution-based losses such as cross-entropy loss may still invoke a large loss. Dominate classes with more training examples will have stronger impacts on the whole model, which leads to over-fitting to dominant classes.

Many variants of Hinge loss [9] have been proposed to overcome the long tail problem for multi-class learning tasks in various fields [3, 6, 21, 27]. Instead of modeling probabilistic distributions, Hinge loss leads to a maximum-margin classifier. However, these Hinge loss variants cannot be directly applied to multi-label learning problems where an instance can belongs to multiple classes.

To this end, we develop an adaptive margin loss function for multi-label learning tasks, which encourages more separations between positive classes and negative classes. Given a pair of entities (e_h, e_t), we first split their relation labels into positive classes $\mathcal{P}$ and negative classes $\mathcal{N}$. The positive classes $\mathcal{P}$ contains relations that exist between two entities. Note that the positive classes set $\mathcal{P}$ can be empty when there is no relation between these two entities. The negative classes set $\mathcal{N}$ contains relations that do not exist between them. An illustrative example is shown in Figure 2. We define a set of new labels $\boldsymbol{t} = [t_1, t_2, \cdots, t_C]^T$. The value of t_i is defined as:

$$t_i = \begin{cases} 1 & R_i \in \mathcal{P} \\ -1 & R_i \in \mathcal{N}. \end{cases} \tag{10}$$

The adaptive margin loss for an entity pair (e_h, e_t) includes the loss for each relation class and is formally defined as:

$$\mathcal{L} = \sum_{1 \leq i \leq C} \max(0, \alpha - t_i(z_i - z_s)), \tag{11}$$

where $\alpha \geq 0$ is a hyper-parameter for margin in the margin-based loss. Note that the proposed adaptive margin loss will reduce to Hinge loss in the binary RE tasks.

When the prediction is correct (i.e., t_i and $(z_i - z_s)$ have the same sign) and the predicted value is higher or lower than that of the separation class with the margin (i.e., $|z_i - z_s| > \alpha$), the loss will be 0. Otherwise, the loss will be linear to the distance of $|z_i - z_s|$. In this case, the model aims to make "good enough" predictions instead of "prefect" predictions. Thus, the model avoids over-fitting to any classes, especially to the dominate classes.

5 EXPERIMENTS

In this section, we evaluate the proposed SagDRE model on several document-level relation extraction benchmark datasets from various domains.

5.1 Experiments on the General Domain Dataset

Datasets and Evaluation Metrics. We conduct experiments to evaluate the proposed method on DocRED dataset [40], a general domain dataset. The DocRED dataset is a large-scale human-annotated dataset constructed from Wikipedia and Wikidata. It contains 132,275 entities, 56,354 relational facts, and 96 relation classes. More than 40.7% of the relation pairs are cross-sentence relation facts. The statistics of this dataset is summarized in Table 1. We use the evaluation metrics provided by Yao et al. [40], including Ign F1 and F1 scores, on both validation and test sets. Ign F1 scores are computed by excluding those relational facts shared by the training and dev/test sets. For both metrics, the higher the better.

Baseline Models. We compare the proposed SagDRE with the state-of-the-art models including sequence-based models and graph-based models. For sequence-based models, we compare the proposed method with two traditional neural networks: CNN-GloVe [2] and BiLSTM-GloVe [20], and BERT enhanced models including BERT [37], ATLOP-BERT [45], CorefBERT [41], and HIN-BERT [35]. The graph-based baseline models include AGGCN-GloVe [11], EoG-GloVe [5], LSR-GloVe/BERT [23], and GAIN-GloVe/-BERT [43]. We evaluate the proposed SagDRE with $K = 1$ and 3, where K is the number of shortest paths used in the sequence-aware path encoding component. Note that SagDRE uses the shortest path when $K = 1$.

Table 1: Statistics of the DocRED, CDR, and CHR datasets. On the DocRED dataset, we do not have access to the numbers of positive and negative pairs in the test dataset

	DocRED			CDR			CHR		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
#Documents	3053	1000	1000	500	500	500	7,298	1,182	3,614
#Pos pairs	38,180	12,323	-	1,038	1,012	1,066	19,643	3,185	9,578
#Neg pairs	1,198,650	396,790	-	4,198	4,069	4,119	69,843	11,466	33,339

Table 2: Results on document-level RE tasks using the DocRED dataset from the general domain. We report the Ign F1 (%) and F1 (%) scores on both the validation set and the test set. For performances on the test set, we report the official test score using the best model on the validation set. Results with † are reported from [23]. Results with * are reported from their original papers. In SagDRE, k indicates the number of the shortest paths in the sequence-aware path encoding component.

	Dev		Test	
Model	**Ign F1**	**F1**	**Ign F1**	**F1**
CNN-GloVe*	41.58	43.45	40.33	42.26
BiLSTM-GloVe*	48.87	50.94	48.78	51.06
AGGCN-GloVe†	46.29	52.47	48.89	51.45
EoG-GloVe†	45.94	52.15	49.48	51.82
LSR-GloVe*	48.82	55.17	52.15	54.18
GAIN-GloVe*	53.05	55.29	52.66	55.08
SagDRE-GloVe ($K = 1$)	53.69	56.69	53.85	56.19
SagDRE-GloVe ($K = 3$)	**53.73**	**56.71**	**53.90**	**56.23**
$\text{BERT}_{\text{BASE}}$*	-	54.16	-	53.20
$\text{LSR-BERT}_{\text{BASE}}$*	52.43	59.00	56.97	59.05
$\text{HIN-BERT}_{\text{BASE}}$*	54.29	56.31	53.70	55.60
$\text{CorefBERT}_{\text{BASE}}$*	55.32	57.51	54.54	56.96
$\text{GAIN-BERT}_{\text{BASE}}$*	59.14	61.22	59.00	61.24
$\text{ATLOP-BERT}_{\text{BASE}}$*	59.22	61.09	59.31	61.30
$\text{SagDRE-BERT}_{\text{BASE}}$ ($K = 1$)	**60.32**	**62.11**	**60.11**	**62.32**
$\text{SagDRE-BERT}_{\text{BASE}}$ ($K = 3$)	60.26	62.06	60.06	62.19

SagDRE Setups. For the proposed methods, we use Huggingface's Transformers [38] to implement BERT model [7]. A dropout [33] operation is applied in the final prediction layer with a keep rate of 0.6. We use AdamW optimizer [19] to optimize the SagDRE model with the learning rate of 1e-3. When training with the BERT encoder, a linear warmup [10] is used for the first 6% steps then decay the linear rate to 0. When using Glove embedding [26], we reduce the learning rate when the F1 value on the validation set has stopped improving. All hyper-parameters are tuned on the validation set. We train all RE models using one Tesla V100 GPU.

Main Results. We summarize the comparison results in Table 2. The results clearly show that the proposed SagDRE model consistently outperform previous state-of-the-art models. Comparing with models without using pre-trained BERT models, GAIN-GloVe achieves the best performance among the baseline methods. The proposed SagDRE-GloVe outperforms GAIN-GloVe by margins of 0.64% and 1.4% on the validation set, and by 1.19% and 1.11% on the test set, in terms of Ign F1 and F1, respectively. All methods improve significantly after applying pre-trained BERT model. Comparing with baseline models, the proposed $\text{SagDRE-BERT}_{\text{BASE}}$ achieves better performances on both validation and test sets as well. In particular, the proposed model improves the performances by 1.1% and 1.12% on the validation set, and by 0.8% and 1.02% on test set, in terms of Ign F1 and F1, respectively, compared to $\text{ATLOP-BERT}_{\text{BASE}}$. Both comparison results show that the proposed methods can bring consistent performance improvements when working with GloVe embedding or pre-trained transformer models. Note that, SagDRE with $K = 1$ achieves similar results as SagDRE with $K = 3$. This is caused as the shortest paths already contain the most information for reasoning and are also encoded with rich contextual information especially with the pre-trained Bert model.

5.2 Experiments on Biomedical Datasets

Datasets and Evaluation Metrics. In this section, we use two datasets from biomedical domains: CDR and CHR. The **CDR** or BC5CDR dataset [16] is a human-annotated relation extraction dataset with detailed annotation guidelines on corpus of PubMed. The chemicals, diseases, and their relations are annotated by four MeSH indexers with a medical training background and curation experience. The dataset includes 1,500 PubMed articles, 5,818 diseases, 4,409 chemicals, and 3,119 chemical-disease relation pairs.

Table 3: Results on document-level RE tasks using the CDR and CHR datasets from Biomedical domain. Results with ‡ are obtained using their official released code. Results with * are reported from their original papers. We report the F1 (%) scores on the test sets.

Model	CDR	CHR
CNN-BioGloVe	62.3*	84.1*
BiLSTM-BioGloVe	59.1*	86.4*
GCNN-BioGloVe	58.6*	87.5*
EoG-BioGloVe	63.6*	-
SciBERT	65.1*	88.9‡
ATLOP-SciBERT	69.4*	90.1‡
SagDRE-SciBERT(**ours**)	**71.8**	**92.9**

The task is to predict the binary relation between Chemicals and Diseases.

The **CHR** dataset is a distantly annotated document-level RE dataset [29] with chemical relations. The annotation is a two-step process. In the first step, the semantic faceted search engine Thalia [32] is used to annotate biomedical name entities on abstracts from PubMed. Then each pair of annotated chemical entities are aligned with the graph dataset Biochem4j [34]. Two chemical entities are considered to have a relation if they appear in Biochem4j. The task is to predict the binary relation between chemicals.

The statistics of these datasets are summarized in table 1. We use F1 scores to evaluate the proposed model.

Baseline Models. We compare the proposed model with sequential models including CNN-BioGloVe and BiLSTM-BioGloVe [29], and state-of-the-art models including GCNN-BioGloVe [29], EoG-BioGloVe [5], GAIN-GloVe [43], ATLOP-SciBERT [45], and SciBERT [45].

SagDRE Setups. We follow similar setups as Section 5.1 with several changes. We use SciBERT [1] as the encoder, which is a pre-trained language model trained on large-scale labeled scientific corpora. We use AdamW to optimize the SagDRE model with the learning rate of 1e-3. A linear warmup is used for the first 6% steps then decay the linear rate to 0. We evaluate the proposed SagDRE with $K = 1$ to save computational costs.

Main Results. The results are summarized in Table 3. SagDRE achieves consistently better performances than previous state-of-the-art models on both biomedical RE datasets. Compare to the previously best model ATLOP-SciBERT, the proposed SagDRE outperforms it by margins of 2.4% and 2.8% on CDR and CHR, respectively. This demonstrates the effectiveness of our proposed methods on biomedical datasets.

5.3 Ablation Study of SagDRE

We conduct ablation studies to investigate the contributions of each component to the overall model performances. Based on SagDRE model, we remove one component (GNN encoders, directed edges, path LSTM, path augmentation, and adaptive margin loss) at a time and evaluate the resulting model using the validation set of DocRED. To examine the importance of the sequence information, we also tested SagDRE model removing all sequence components including both directed edges and path LSTM. The ablation study results on SagDRE-GloVe model are shown in Table 4, while SagDRE-BERT$_{\text{BASE}}$ shows similar trends.

Table 4: Ablation study results on DocRED dataset with GloVe embedding. We report the precision (P) (%), recall (R) (%), and F1 (%) scores on the validation set.

Model	P	R	F1
SagDRE-GloVe (K=1)	57.24	56.16	56.69
(-) GCN layers	53.44	56.75	55.04
(-) Directed edges	49.88	60.59	54.72
(-) path LSTM	50.56	59.68	54.74
(-) Path augmentation	51.20	61.61	55.92
(-) Adapt margin loss			
(+) Best threshold loss	50.60	58.51	54.26
(+) Adapt threshold loss	49.98	55.02	52.38
(-) sequence components	50.48	58.76	54.30

Table 5: Analysis results of different α values on document-level relation extraction tasks using the DocRED dataset. We report the precision (P) (%), recall (R) (%), and F1 (%) scores on the validation set.

α	P	R	F1
0.0	57.86	61.85	59.79
0.5	62.50	60.00	61.39
1.0	63.33	60.97	**62.11**

Table 6: Error distribution of SagDRE on 50 wrong predictions from the CDR dataset.

	Label Noise (LN)	Hard (H)	Ambiguity (A)	Other (O)
Count	12	5	16	17
Ratio	24%	10%	32%	34%

From Table 4, we can observe that every proposed component contributes to the overall model performance. The most important contributors are the adaptive margin loss and the sequence components. When replacing the adaptive margin loss with cross entropy loss with the best threshold and the adaptive threshold loss in [45], F1 score drops by 2.43% and 4.31%, respectively. When removing sequence components, the performance drops by 2.39%, which shows that the sequential information in text is critical for document-level RE task. In particular, encoding sentence-level and word-level sequential information both contribute to the overall performances of the SagDRE model, which can be observed from the performance drops when removing the direction-aware edges and path LSTM, respectively.

5.4 Parameter Study in Adaptive Margin Loss

In the proposed adaptive margin loss, there is a hyper-parameter α that controls the margin. In this experiment, we investigate the impact of α values on model performances. We evaluate the proposed SagDRE model with different α values (0.0, 0.5, and 1.0) on

Table 7: Case study of RE results on the CDR dataset. For each case, the head entity and tail entity are colored blue and red, respectively. Important words for relation reasoning are highlighted.

Category	Document	Pred	Label
LN	*Document #24618873*: **Cerebellar and oculomotor dysfunction** induced by rapid infusion of **pethidine**. **Pethidine** is an opioid that gains its popularity for the effective pain control through acting on the opioid-receptors. . . .	1	0
H	*Document #24659727*: Tolerability of lomustine in combination with **cyclophosphamide** in dogs with lymphoma. . . . Ninety treatments were given to the 57 dogs included in the study. . . . One dog (3%) developed hematologic changes suggestive of **hepatotoxicity**. No dogs had evidence of either renal toxicity or hemorrhagic cystitis.	1	0
A	*Document #3297909*: Progressive bile duct injury after **thiabendazole** administration. A 27-yr-old man developed **jaundice** 2 wk after exposure to **thiabendazole**. . . .	1	0
A	*Document #24729111*: . . . A 62-year-old man was found to have bradycardia, hypothermia and **respiratory failure** 3 weeks after initiation of **amiodarone** therapy for atrial fibrillation. . . .	0	1
O	*Document #24897009*: Optochiasmatic and peripheral neuropathy due to **ethambutol** overtreatment. **Ethambutol** is known to cause optic neuropathy and, more rarely, axonal **polyneuropathy**. . . .	0	1

the validation set of DocRED dataset. The results are summarized in Table 5. The results show that the SagDRE model achieves the best performance when $\alpha = 1$, a popular choice for margin in margin-based losses [17]. As expected, larger margins improve the Precision, since the model requires a higher score to predict a label.

5.5 Error Analysis

To better understand the bottleneck of the SagDRE and inspire future work, we conduct a case study to investigate the errors that SagDRE makes. To this end, we choose CDR dataset since it is labeled by domain experts, and thus may have the least label noise.

We randomly selected 50 wrong predictions made by SagDRE and analyzed their reasons. In particular, we categorize in three main reasons: 1) **Label Noise (LN)**, where entity pairs in this category may obtain wrong labels by domain experts, 2) **Hard (H)**, where inferring relations between entity pairs may require extra knowledge such as statistics and advanced reasoning, 3) **Ambiguity (A)**, where the document expresses the relation vaguely, and 4) **Others (O)**, where all other wrong predictions are included. The error distribution of these wrong predictions is shown in Table 6. We also illustrate some examples under each category in Table 7.

From our analysis, the majority of the errors occurred without obvious reasons. However, we observe that the model makes more mistakes for entity pairs with certain keywords. For example, Document #24897009 states that "Ethambutol is known to cause ... rarely ... polyneuropathy". The negation word "rarely" may trigger the model to classify the relation as unrelated. Other observed keywords include negation word such as "no" and words expressing uncertainty such as "may".

Another major cause of errors is the ambiguity of the document. Almost all the errors of this type occur in reports of drug side-effect events. For example, Document #3297909 states that a man developed jaundice after exposure to thiabendazole, and Document #24729111 states that a man had respiratory failure after initiation of amiodarone therapy. It is somewhat ambiguous whether the symptoms are caused by the drugs in these two cases, and in fact, the labels provided by experts of these two cases are also different. There may be more similar cases in the training data, and thus the model cannot predict consistently.

There may also be some incorrect labels provided by domain experts. For example, Document #24618873 clearly states the causal relationship between the disease and the drug. However, the label for this pair of entities is "unrelated".

There are also some hard cases that external knowledge might be needed. For example, Document #24659727 states that 1 out of 57 dogs with the given treatments developed the symptom, so the relation is statistically insignificant. To correctly label this symptom-drug pair, the model needs to understand "3%" is too low to support the relation. Studies in common sense extraction and number extraction may be helpful for this case.

6 CONCLUSION

In this work, we propose the SagDRE model for document-level relation extraction, which encodes the sequential information in the original text. SagDRE considers both the sentence-level and the token-level sequential information in the documents. To capture sentence-level sequential information, directed edges are added in the constructed document graph and their weights are learned through an attention mechanism. These directed weighted edges can capture the logic flows of the sentences in a document. For token-level sequential information, SagDRE extracts and reconstructs augmented shortest paths from the head entity to the tail entity with the original sequential ordering, and encodes it with LSTM. To address the limitation of the regular loss function for RE model optimization, we propose the adaptive margin loss. This loss function employs a threshold class and maximizes the margins between the positive classes and the negative classes. The experimental results on document-level RE datasets from both general and biomedical domains demonstrate the effectiveness of the proposed methods. The ablation study of SagDRE shows that every proposed component contributes to the overall model performance. The most important contributions are from the adaptive margin loss and the sequence components.

ACKNOWLEDGMENTS

The work is supported in part by NIFA grant no. 2022-67015-36217 from the USDA National Institute of Food and Agriculture, and NSF IIS-2007941 from the National Science Foundation.

REFERENCES

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. 3615–3620.

[2] Phil Blunsom, Edward Grefenstette, and Nal Kalchbrenner. 2014. A Convolutional Neural Network for Modelling Sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*.

[3] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems* 32 (2019).

[4] Daniel Cer, Marie-Catherine De Marneffe, Dan Jurafsky, and Christopher D Manning. 2010. Parsing to stanford dependencies: Trade-offs between speed and accuracy. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*.

[5] Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the Dots: Document-level Neural Relation Extraction with Edge-oriented Graphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. 4925–4936.

[6] Jiankang Deng, Yuxiang Zhou, and Stefanos Zafeiriou. 2017. Marginal loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 60–68.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. 4171–4186.

[8] Bayu Distiawan, Gerhard Weikum, Jianzhong Qi, and Rui Zhang. 2019. Neural relation extraction for knowledge base enrichment. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 229–240.

[9] Claudio Gentile and Manfred KK Warmuth. 1998. Linear hinge loss and average margin. *Advances in neural information processing systems* 11 (1998), 225–231.

[10] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677* (2017).

[11] Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention Guided Graph Convolutional Networks for Relation Extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 241–251.

[12] Ben Hachey. 2009. Multi-document summarisation using generic relation extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. 420–429.

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

[14] Shantanu Kumar. 2017. A survey of deep learning methods for relation extraction. *arXiv preprint arXiv:1705.03645* (2017).

[15] Bo Li, Wei Ye, Zhonghao Sheng, Rui Xie, Xiangyu Xi, and Shikun Zhang. 2020. Graph Enhanced Dual Attention Network for Document-Level Relation Extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*. 1551–1560.

[16] 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. 2016. BioCreative V CDR task corpus: A resource for chemical disease relation extraction. *Database* 2016 (2016).

[17] Yi Lin. 2004. A note on margin-based loss functions in classification. *Statistics & probability letters* 68, 1 (2004), 73–82.

[18] Yang Liu and Mirella Lapata. 2018. Learning structured text representations. *Transactions of the Association for Computational Linguistics* 6 (2018), 63–75.

[19] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[20] Xuezhe Ma and Eduard Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. 1064–1074.

[21] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2020. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314* (2020).

[22] Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. Natural Language Inference by Tree-Based Convolution and Heuristic Matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 130–136.

[23] Guoshun Nan, Zhijiang Guo, Ivan Sekulić, and Wei Lu. 2020. Reasoning with Latent Structure Refinement for Document-Level Relation Extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 1546–1557.

[24] Sachin Pawar, Girish K Palshikar, and Pushpak Bhattacharyya. 2017. Relation extraction: A survey. *arXiv preprint arXiv:1712.05191* (2017).

[25] 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 Linguistics* 5 (2017), 101–115.

[26] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. 1532–1543.

[27] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. 2020. Balanced meta-softmax for long-tailed visual recognition. *Advances in Neural Information Processing Systems* 33 (2020), 4175–4186.

[28] Luana Ruiz, Fernando Gama, Antonio García Marques, and Alejandro Ribeiro. 2019. Invariance-preserving localized activation functions for graph neural networks. *IEEE Transactions on Signal Processing* 68 (2019), 127–141.

[29] Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Inter-sentence Relation Extraction with Document-level Graph Convolutional Neural Network. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 4309–4316.

[30] Sunil Kumar Sahu, Derek Thomas, Billy Chiu, Neha Sengupta, and Mohammady Mahdy. 2020. Relation extraction with self-determined graph convolutional network. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2205–2208.

[31] Xavier Schmitt, Sylvain Kubler, Jérémy Robert, Mike Papadakis, and Yves LeTraon. 2019. A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. In *Proceedings of the 6th International Conference on Social Networks Analysis, Management and Security*. 338–343.

[32] Axel J Soto, Piotr Przybyła, and Sophia Ananiadou. 2019. Thalia: semantic search engine for biomedical abstracts. *Bioinformatics* 35, 10 (2019), 1799–1801.

[33] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[34] Neil Swainston, Riza Batista-Navarro, Pablo Carbonell, Paul D Dobson, Mark Dunstan, Adrian J Jervis, Maria Vinaixa, Alan R Williams, Sophia Ananiadou, Jean-Loup Faulon, et al. 2017. biochem4j: Integrated and extensible biochemical knowledge through graph databases. *PloS one* 12, 7 (2017), e0179130.

[35] Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. Hin: Hierarchical inference network for document-level relation extraction. *Advances in Knowledge Discovery and Data Mining* 12084 (2020), 197.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[37] 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* (2019).

[38] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771* (2019).

[39] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*.

[40] 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 *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 764–777.

[41] Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential Reasoning Learning for Language Representation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 7170–7186.

[42] 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 *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 571–581.

[43] Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double Graph Based Reasoning for Document-level Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 1630–1640.

[44] Huiwei Zhou, Yibin Xu, Weihong Yao, Zhe Liu, Chengkun Lang, and Haibin Jiang. 2020. Global context-enhanced graph convolutional networks for document-level relation extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*. 5259–5270.

[45] Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 14612–14620.