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

Anonymous ACL submission

Abstract

Relation extraction (RE) is an important task for many natural language processing applications. Document-level relation extraction 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 path from one entity to the other. In addition, we propose an adaptive margin loss to maximize the margins to separate positive and negative classes. The experimental results on datasets from various domains demonstrate the effectiveness of our proposed methods.

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 (Distiawan et al., 2019; Yu et al., 2020), question answering (Yu et al., 2017), and text summarization (Hachey, 2009). In the RE tasks, there are two specific sub-tasks: sentence-level relation extraction and document-level relation extraction (Pawar et al., 2017). 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 (Sahu et al., 2019).

To address these challenges, some methods propose to construct a graph to represent the document and achieve the state-of-the-art performances (Nan et al., 2020; Li et al., 2020; Sahu et al., 2019; Guo et al., 2019). 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 characteristics of languages. These graphs cannot encode the sequential information due to its permutation invariance property (Ruiz et al., 2019), which can downgrade the performance for document-level RE tasks.

Another challenge of document-level RE is that the document may express multiple relations for the same entity pair. This leads to the multi-label problem. Intuitively, given a document, an entity pair either has no relation expressed, or have one or more relation expressed. 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.

In this work, we propose a Sequence-Aware Graph-based Document-level Relation Extraction model (SagDRE) to consider original text sequential information for 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 previ-

ous sentence roots to later ones are added with edge weights learned by an attention mechanism. Based on the constructed document graph, we adopt GCN and multi-head self attention to encode local and global features. To capture the token-level sequential information, SagDRE finds the shortest path from the head entity to the tail entity on the document graph and then reconstruct the path with the original token orders and auxiliary tokens. The path is encoded using LSTM and concatenated with other features for prediction.

Inspired by Hinge loss, we propose an adaptive margin loss for multi-class multi-label learning tasks. In particular, we learn a threshold class for each pair of entities between positive classes and negative classes. The optimization based on this loss will encourage the maximum separation between positive and negative classes via the threshold class.

In empirical studies, we use three 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 threshold class.
- Empirical studies on three document-level relation extraction datasets from various domains demonstrate the effectiveness of the proposed method.

2 Related Work

Relation extraction task has been studied in the past decades. The applications of deep learning methods have significantly advanced the development for the task (Kumar, 2017; Pawar et al., 2017). 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 (Yao et al., 2019) but extracting document-level relations is more challenging since cross-sentence learning usually requires effective long-distance feature encoding and reasoning (Sahu et al., 2019).

To tackle this challenge, some methods (Eberts and Ulges, 2021; Zhou et al., 2021; Xie et al., 2021; Ye et al., 2020; Tang et al., 2020) apply BERT (Devlin et al., 2019) for more informative contextual token encoding. Besides BERT, some methods propose to use the graph structure to shorten the distances between entities in the document (Li et al., 2020; Sahu et al., 2019; Guo et al., 2019; Nan et al., 2020).

Sahu et al. (2019) is the first work to adopt graph structure in document-level RE tasks. It uses linguistic tools to build various edges, such as coreference 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. (2019) and Sahu et al. (2020) use attention mechanisms to construct edges in the graph. Instead of constructing token-level graphs, Zeng et al. (2020) proposes to build two graphs, including mention-level and entity-level graphs, to predict relations. Compared to previous methods that use graph neural networks to encode features, Zhou et al. (2020) proposes 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 (Ruiz et al., 2019) 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 (Kipf and Welling, 2017; Xu et al., 2018). Specifically to Graph Convolutional Networks (GCN), the ℓ^{th} GCN layer is defined as

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

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 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 $\{R_0, R_1, \cdots, R_C\}$, where R_0 represents "No relation" while R_i $(1 \leq i \leq C)$ represents the i^{th} pre-defined relationships. A RE model should output either R_0 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.

4 SagDRE

In this section, we introduce 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 that is especially designed for multi-label multi-class learning tasks such as document-level RE.

4.1 Sequence-Aware Graph Construction

Many existing methods adopt graph structures for document-level RE tasks using dependency parsers (Cer et al., 2010; Schmitt et al., 2019) 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 (Ruiz et al., 2019).

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 $\boldsymbol{x}_i$ is the word embedding for the i^{th} token 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 (Devlin et al., 2019) 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.

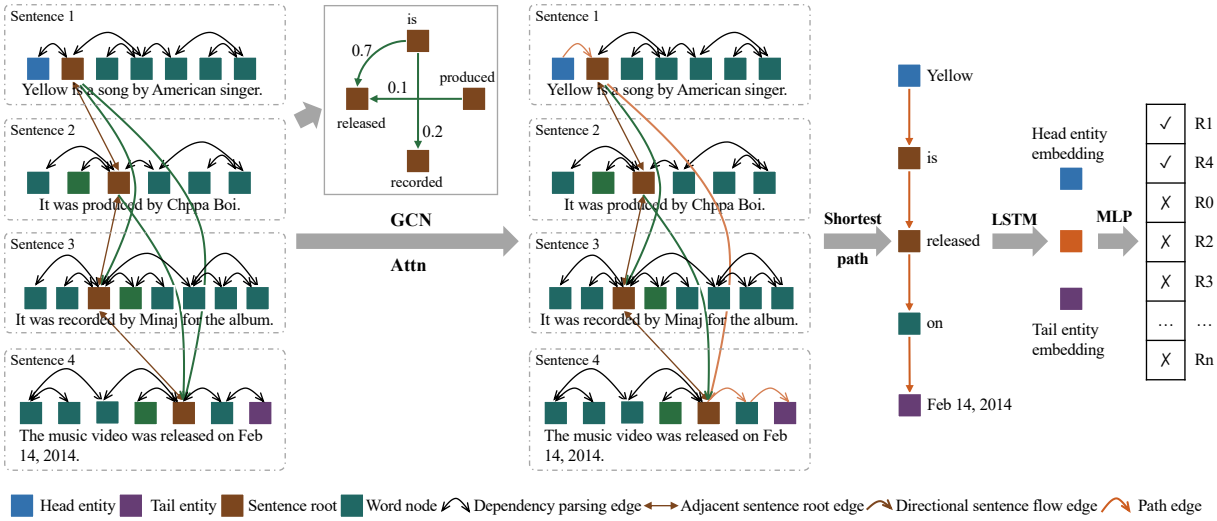

Figure 1: Illustration of the sequence-aware document-level relation extraction network. Given an input document, each token obtains its initial feature embedding from the encoder. Then a document graph is constructed. The directed cross-sentence edges are added (green edges) and their edge weights are 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 the shortest path from the head entity to the tail entity from the graph and encoded by LSTM, resulting in a path embedding. Finally, the entity embeddings and path embedding are fed into an MLP for prediction. The adaptive margin loss is applied.

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. Final 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 roots to later ones. These forward edges indicate the order information in the original text sequence and enforce the information to propagate from earlier sentences to later ones. Since not all sentences are closely related to each other, 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.

Given two sentences roots i and j, we compute the weight $A_{i,j}$ for the directed edge from i to j based on their feature vectors:

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

where $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ are the encodings of roots 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) (Kipf and Welling, 2017) 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 (Vaswani et al., 2017) 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:

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

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.

4.3 Sequence-Aware Path Encoding

The document graphs 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 sequence-aware path from the head entity to the tail entity.

Given a graph and a pair of entities (e_h, e_t), the shortest path from e_h to e_t in the graph usually contains the most relevant reasoning information for their relationships. We denote the shortest path as $P_{h,t} = [e_h, n_1, \cdots, n_k, e_t]$, where n_j represents the j^{th} node on the path. This path may neglect some important structural words for relation reasoning though such as "near" and "outside". To enrich the sequence-aware path and include more informative nodes, we augment the extracted path with adposition words attached to this path. That is, given the shortest path, we add the neighboring adposition word nodes of each node n_i in $P_{h,t}$, which leads to the augmented path $P'_{h,t}$.

To encode the original token-level sequential information, we order the nodes in the path by their original sequential order in the text, which leads to $P''_{h,t}$. We apply a directional LSTM layer to encode features of this path, and a max-pooling layer to obtain the feature representations. The proposed sequence-aware path encoding is formulated as

$$\overrightarrow{\boldsymbol{u}_j} = \text{LSTM}(\overrightarrow{\boldsymbol{h}_j}) \quad (3)$$
$$\boldsymbol{p}_{h,t} = \max(\overrightarrow{\boldsymbol{u}_h}, \overrightarrow{\boldsymbol{u}_1}, \cdots, \overrightarrow{\boldsymbol{u}_k}, \overrightarrow{\boldsymbol{u}_t}), \quad (4)$$

where $\boldsymbol{u}_j$ represents the LSTM hidden representations of the j^{th} node in $P''_{h,t}$.

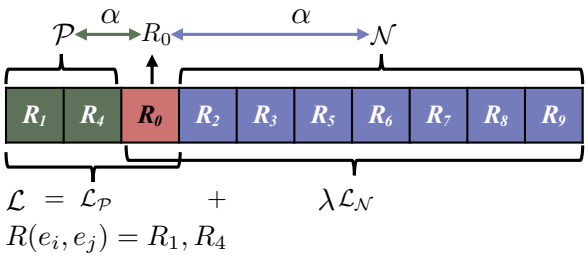

Figure 2: Illustration of our proposed adaptive margin loss. Given an entity pair (e_i, e_j) in a document, their relations are R_1 and R_4.

4.4 Relation Prediction Head

After obtain sequence-aware entity encoding and path encoding, 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 augmented shortest path encoding. Following previous methods (Zeng et al., 2020), we concatenate entity encoding of two entities, $\boldsymbol{e}_h$, $\boldsymbol{e}_t$, the absolute values of subtraction of two entity encoding, $|\boldsymbol{e}_h - \boldsymbol{e}_t|$, 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:

$$\boldsymbol{I}_{h,t} = [\boldsymbol{e}_h; \boldsymbol{e}_t; |\boldsymbol{e}_h - \boldsymbol{e}_t|; \boldsymbol{e}_h \odot \boldsymbol{e}_t; \boldsymbol{p}_{h,t}]. \quad (5)$$

We compute the prediction values $\boldsymbol{z} \in \mathcal{R}^{C+1}$ for all relation classes:

$$\boldsymbol{z} = \boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{I}_{h,t} + \boldsymbol{b}_1) + \boldsymbol{b}_2, \quad (6)$$

where $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, $\boldsymbol{b}_1$, $\boldsymbol{b}_2$ are trainable parameters, and σ is an element-wise activation function.

4.5 Adaptive Margin Loss

Most existing relation extraction models output $P(R_i|e_h, e_t, \mathcal{G})$ for the probability of that 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 (Peng et al., 2017; Liu and Lapata, 2018; Nan et al., 2020) 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. (2021) uses the probability prediction on class R_0 as a threshold between positive classes and negative classes.

Inspired by Hinge loss (Gentile and Warmuth, 1998), we further develop an adaptive margin loss function to encourage more separations between

positive classes and negative classes, thereby leading to better generalization. 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 two entities. An illustrative example is shown in Figure 2.

The adaptive margin loss for an entity pair (e_h, e_t) includes the loss between positive classes and the threshold class, and the loss between the threshold class and negative classes. The loss for the entity pair (e_h, e_t) is formally computed as

$$\mathcal{L}_{\mathcal{P}} = \sum_{i \in \mathcal{P}} \max(0, \alpha - z_i + z_0), \tag{7}$$

$$\mathcal{L}_{\mathcal{N}} = \sum_{j \in \mathcal{N}} \max(0, \alpha - z_0 + z_j), \tag{8}$$

$$\mathcal{L} = \mathcal{L}_{\mathcal{P}} + \lambda \mathcal{L}_{\mathcal{N}}, \tag{9}$$

where α is a hyper-parameter for margin in margin-based loss. The prediction on class R_0 (i.e., z_0) is used for the threshold. Since there are usually much more negative classes than positive classes in multi-class multi-label RE tasks, $\lambda \in (0, 1)$ is introduced as a hyper-parameter to balance the loss values between positive classes and negative classes.

The optimization based on this loss will pull the predictions of positive classes to be higher than that of the threshold class R_0 with margin α. Similarly, the prediction of negative classes will be pushed lower than that of the threshold class R_0 with margin α. Combining together, this loss function will try to maximize margins between positive classes and negative classes via the threshold class R_0. During the prediction, we output the classes whose prediction values are larger than that of the threshold z_0.

5 Experiments

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

5.1 Experiments on General Domain Dataset

Datasets and Evaluation Metrics. We conduct experiments to evaluate the proposed method on DocRED dataset (Yao et al., 2019), 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. There are 3,053 documents, 1,000 documents, and 1,000 documents for training, validation, and testing, respectively. The statistics of this dataset is summarized in Table 1.

We use the evaluation metrics provided by Yao et al. (2019), including Ign F1 and F1 scores, on both validation and test sets. Ign F1s exclude 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 (Blunsom et al., 2014) and BiLSTM-GloVe (Ma and Hovy, 2016), and BERT enhanced models including BERT (Wang et al., 2019), ATLOP-BERT (Zhou et al., 2021), Coref-BERT (Ye et al., 2020), and HIN-BERT (Tang et al., 2020). The graph-based baseline models include AGGCN-GloVe (Guo et al., 2019), EoG-GloVe (Christopoulou et al., 2019), LSR-GloVe/BERT (Nan et al., 2020), and GAIN-GloVe/-BERT (Zeng et al., 2020).

SagDRE Setups. For the proposed methods, we use Huggingface's Transformers (Wolf et al., 2019) to implement BERT model (Devlin et al., 2019). A dropout (Srivastava et al., 2014) operation is applied in the final prediction layer with a keep rate of 0.6. We use AdamW (Loshchilov and Hutter, 2018) to optimize the SagDRE model with the learning rate of 1e-3. When training with the BERT encoder, a linear warmup (Goyal et al., 2017) is used for the first 6% steps then decay the linear rate to 0. When using Glove embedding (Pennington et al., 2014), 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%

	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 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

	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 (**ours**)	**53.69**	**56.69**	**53.85**	**56.19**
$\text{BERT}_{\text{BASE}}$*	-	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
$\text{SagDRE-BERT}_{\text{BASE}}$ (**ours**)	**60.32**	**62.11**	**60.11**	**62.32**

Table 2: Results on document-level RE tasks using the DocRED dataset from 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 by using the best model on the validation set. Results with † are reported from (Nan et al., 2020). Results with * are reported from their original papers.

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 the baseline models, the proposed SagDRE-$\text{BERT}_{\text{BASE}}$ achieves better performances on both validation and test sets as well. In particular, the proposed SagDRE-$\text{BERT}_{\text{BASE}}$ 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 ATLOP-$\text{BERT}_{\text{BASE}}$.

5.2 Experiments on Biomedical Datasets

Datasets and Evaluation Metrics. We use two datasets from biomedical domains: CDR and CHR. The **CDR** dataset (Li et al., 2016) is a human-annotated relation extraction dataset with detailed annotation guidelines on text 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. The task is to predict the binary relation between Chemicals and Diseases.

The **CHR** dataset is a distantly annotated document-level RE dataset (Sahu et al., 2019) with chemical relations. The annotation is a two-step process. In the first step, the semantic faceted search engine Thalia (Soto et al., 2019) 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 (Swainston et al., 2017). 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.

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**

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	P	R	F1
SagDRE-GloVe	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	50.60	58.51	54.26
(-) sequence components	50.48	58.76	54.30

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.

Baseline Models. We compare the proposed model with sequential models including CNN-BioGloVe and BiLSTM-BioGloVe (Sahu et al., 2019), and state-of-the-art models including GCNN-BioGloVe (Sahu et al., 2019), EoG-BioGloVe (Christopoulou et al., 2019), GAIN-GloVe (Zeng et al., 2020), SciBERT (Zhou et al., 2021), and ATLOP-SciBERT (Zhou et al., 2021).

SagDRE Setups. We follow similar setups as Section 5.1 with several changes. We use SciBERT (Beltagy et al., 2019) 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.

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.

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 on SagDRE-GloVe model is shown in Table 4, while SagDRE-BERT$_{\text{BASE}}$ shows similar trends.

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 removing the adaptive margin loss, F1 score drops by 2.43%, which indicates the proposed loss function can help RE model achieve better generalization ability. 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.

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 an augmented shortest path 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.

References

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*, pages 3615–3620.

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*.

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*.

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*, pages 4925–4936.

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*, pages 4171–4186.

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*, pages 229–240.

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

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

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*.

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*, pages 241–251.

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

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

Shantanu Kumar. 2017. A survey of deep learning methods for relation extraction. *arXiv preprint arXiv:1705.03645*.

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*, pages 1551–1560.

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.

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

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

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*, pages 1064–1074.

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*, pages 1546–1557.

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

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*, 5:101–115.

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*, pages 1532–1543.

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*, 68:127–141.

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*, pages 4309–4316.

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*, pages 2205–2208.

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*, pages 338–343.

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

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):1929–1958.

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):e0179130.

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*, 12084:197.

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*, pages 5998–6008.

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*.

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*.

Yiqing Xie, Jiaming Shen, Sha Li, Yuning Mao, and Jiawei Han. 2021. Eider: Evidence-enhanced document-level relation extraction. *arXiv preprint arXiv:2106.08657*.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? In *International Conference on Learning Representations*.

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*, pages 764–777.

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*, pages 7170–7186.

Haoze Yu, Haisheng Li, Dianhui Mao, and Qiang Cai. 2020. A relationship extraction method for domain knowledge graph construction. *World Wide Web*, 23(2):735–753.

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*, pages 571–581.

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*, pages 1630–1640.

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*, pages 5259–5270.

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*, volume 35, pages 14612–14620.