Article

Document-Level Relation Extraction with Local Relation and Global Inference

Yiming Liu [1], Hongtao Shan [1], Feng Nie [2],*, Gaoyu Zhang [3] and George Xianzhi Yuan [4,5]

[1] School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China; m320121356@sues.edu.cn (Y.L.)
[2] School of Finance, Shanghai Lixin University of Accounting and Finance, Shanghai 201209, China
[3] School of Information Management, Shanghai Lixin University of Accounting and Finance, Shanghai 201209, China
[4] College of Science, Chongqing University of Technology, Chongqing 400054, China
[5] Business School, Chengdu University, Chengdu 610106, China
* Correspondence: 20019274@lixin.edu.cn

Abstract: The current popular approach to the extraction of document-level relations is mainly based on either a graph structure or serialization model method for the inference, but the graph structure method makes the model complicated, while the serialization model method decreases the extraction accuracy as the text length increases. To address such problems, the goal of this paper is to develop a new approach for document-level relationship extraction by applying a new idea through the consideration of so-called "Local Relationship and Global Inference" (in short, LRGI), which means that we first encode the text using the BERT pre-training model to obtain a local relationship vector first by considering a local context pooling and bilinear group algorithm and then establishing a global inference mechanism based on Floyd's algorithm to achieve multi-path multi-hop inference and obtain the global inference vector, which allow us to extract multi-classified relationships with adaptive thresholding criteria. Taking the DocRED dataset as a testing set, the numerical results show that our proposed new approach (LRGI) in this paper achieves an accuracy of 0.73, and the value of F1 is 62.11, corresponding to 28% and 2% improvements by comparing with the classical document-level relationship extraction model (ATLOP), respectively.

Keywords: document-level relation extraction; multi-hop reasoning; attention mechanism; BERT pre-trained model; Floyd algorithm

1. Introduction

In recent years, document-level relationship extraction, which extracts relationships between entity pairs in text spanning multiple sentences and multiple entities, has become a new research hotspot due to the widespread use of relational knowledge in the form of knowledge graphs by Internet companies such as search engines [1] under the framework of big data analysis and related applications in financial technology (Fintech) and related areas.

The current research on the extraction of relationship tasks mostly focuses on the sentence-level relation extraction method by identifying the relation of entity pairs by two entities and the sentence where the entity pair is located [2]. However, it seems that the sentence-level relation extraction task in general cannot meet the need of practical applications, by the fact that the same entity may appear multiple times, multiple entity pairs exist in the text, multiple relationships may be contained between the same entity pairs, and entity pairs appear in two different sentences, and so on; thus, it is expected that the approach to the document-level relationship extraction proposed in this paper is more practical.

Compared to the sentence-level relation extraction method, the document-level relation extraction approach that we discuss in this paper has several advantages in dealing with at least the following four issues:

1. Entity information needs to be fused for all mentioned words of the same entity, as the same entity can appear multiple times within one document.

2. The calculation of relationships between entity pairs requires consideration of contextual information, since a document contains multiple entity pairs.

3. The classification of these relationships extends from a single-label classification task to a multi-label classification task, as multiple relationships may exist between a pair of entities.

4. Some entity pairs require indirect logical inference to determine their relationships. This necessitates additional inference mechanisms in the construction of the document-level relationship extraction model.

So far, there have been many attempts to achieve the document-level relationship extraction task (see [3] and related references therein); here, we briefly recall some of them.

The popular approach is based on the serialization model, which encodes the text using serialization methods [4] and obtains entity information from the sequence of word vectors after encoding; the relationship between entity pairs is then obtained through neural networks, followed by the classification of relationships. This method primarily originates from sentence-level relationship extraction, and the model has a simple structure that is easy to implement. However, as we know, all document-level relation extraction models based on serialization methods face the challenge of decreased accuracy as the sentence length increases. Another approach is the use of the pre-training model method [5], which researchers have applied to the task of document-level relationship extraction. The key idea of the pre-training model is to use attention mechanisms that dynamically adjust the attention weight for each word according to context, thereby enhancing interpretability and the ability to handle long texts, improving the accuracy of document-level relation extraction.

Despite these improvements, attention mechanisms still face challenges in capturing contextual information in long texts, as more information processing leads to degraded model performance. Because of this reason, researchers also proposed an alternative research idea based on the graph-structured model method. The document is encoded using serialization methods or pre-trained models, and graph-structured data [6] are constructed to implement explicit inference based on graph neural networks with relational multi-classification. The key idea of using the graph-structured data method is to model relations and then use graph neural networks to perform explicit inference on relations in text.

We note that the constructed graph-structured data can capture long-distance information by connecting individual nodes through edges, but the model proposed by many current researchers usually contains multiple types of nodes and edges or establishes complex inference, making the extracted model structures complex and introducing redundant information. For example, the edge-driven graph neural network proposed by Christopoulou [7] contains three types of nodes: mention-level nodes, entity-level nodes, and sentence-level nodes, with a complex model structure. There is no directly connected edge between any entity pair, and the relationship vector between entity pairs needs to be represented by a path passing through several intermediate nodes. The information in the path is processed and aggregated through a multilayer graph neural network to obtain the relationship between entity pairs, introducing a lot of extra information into the relationship representation between entities.

In this paper, in order to overcome the limitations of the two methods discussed above, we first define the "local relation" as the relationship between entity pairs that can be obtained through contextual text pooling without explicit inference mechanisms, and then an extension is made to the document-level relation extraction with adaptive thresholding and localized context pooling (in short, ATLOP) model [8]; this allows us to propose a document-level relation extraction model with local relations and global inference (in short, LRGI). The key idea of our LRGI method is that the process involves using the ATLOP model to obtain local relation information, followed by constructing an inference network with bilinear functions to achieve multi-path and multi-hop inference and obtain global

inference information, and finally, the local relation information is combined with the global inference information to obtain the relationship between any two entities in the document and perform relationship classification.

Our LRGI model was tested by DocRED data [9], and the numerical results showed that the LRGI model has an accuracy of 0.73, which is a 28% improvement compared to the ATLOP model, and an F1 score of 62.11, a 2% improvement. Thus, the effectiveness of the document-level relationship extraction model that incorporates local relationships and global inference is verified.

In addition, for the convenience of interested readers to facilitate peer reproduction of this experiment, the source code and trained models have been released on GitHub, which makes this work more accessible to researchers.

Our LRGI approach to the document-level relation extraction model proposed in this paper, by incorporating the local relation and global inference, contains at least the following two major innovations or advantages:

(1) An explicit inference mechanism based on an improved Floyd algorithm is proposed. This inference mechanism first constructs graph-structured data, which contain only one type of node and edge; here, nodes represent entities in a document, while edges represent relationships between two connected entities. Based on the idea of Floyd's algorithm [10] for computing the shortest paths, the state transfer equation is modified and trainable parameters are added to calculate the relationship between two nodes passing through the intermediate node via iteration. Then, the inference mechanism has a simple structure that is easy to implement. In addition, there are directly connected edges between any two entities in the document, and the relationship vectors between entity pairs can be calculated directly without a complicated aggregation process, avoiding the impact of redundant information on the accuracy of relationship extraction. Moreover, the inference mechanism is easy to extend, and iteration can realize the transformation from single-hop multi-path inference to multi-hop multi-path inference.

(2) Relationship extraction from two different perspectives, local relationships containing contextual information and explicit inference containing indirect relationships, is proposed to construct a document-level relationship extraction model that incorporates local relationships and global inference. Local relations are obtained through context pooling in the ATLOP model: contextual information is obtained through an attention mechanism for any entity pair and appended to the vector of relations between entity pairs. Local relations depend on the entity pairs and contextual content and are affected by text length. In this way, the local relations are used as input to the global inference mechanism based on the improved Floyd algorithm, and the global inference containing indirect relations is obtained through explicit inference about the direct relations between entity pairs. Then, global inference complements the local relations to obtain more valid information and improve the accuracy of the document-level relationship extraction model that incorporates local relations and global inference.

This paper is organized into five sections, as follows.

In Section 2, the current work related to document-level relation extraction is introduced. A variety of algorithms are sorted out, and the current research status, advantages, and disadvantages of existing algorithms are analyzed. Based on the summary of previous work, the key idea of our LRGI method is proposed and discussed briefly.

In Section 3, the LRGI model is discussed from the perspective of the implementation of modeling with algorithms. An explicit inference mechanism based on the improved Floyd algorithm is proposed to address the problems that have occurred due to the complex graph structure and redundant information in existing graph-structured document-level relationship extraction models. In addition, two different perspectives of relation extraction, local relations containing contextual information and explicit inference containing indirect relations, are proposed to construct a document-level relation extraction model that incorporates local relations and global inference. This approach addresses the degradation

of model performance due to increased input text length in the serialization model based on mechanisms such as attention.

Section 4 describes the experiments and analyzes the results. The experiments were conducted on the public DocRED dataset, and the results demonstrate that the document-level relation extraction model combining local relations and global inference has high extraction accuracy. The experiments also demonstrate the effectiveness of the explicit inference mechanism based on the improved Floyd algorithm and the effectiveness of global inference as a complement to local relations to achieve the improvement on the accuracy of document-level relation extraction.

Finally, Section 5 summarizes the work of this paper and proposes future research directions and related issues.

2. Related Work

This section classifies document-level relationship extraction models into two types based on their research ideas: serialization-based models and graph-structured models. This section compares several algorithms and analyzes the current state of research on existing algorithms, including their advantages and disadvantages.

2.1. Document-Level Relationship Extraction Models Based on Serialization

Serialization-based document-level relationship extraction models do not use graph neural networks but employ serialization models such as convolutional neural networks (CNNs), long short-term memory neural networks (LSTMs), and bi-directional long short-term memory neural network (BiLSTMs). However, they often face difficulties in capturing long-range contextual information in long text analysis.

With the proposal of pre-training models such as BERT [11], using pre-training models for encoding can alleviate the problem of an insufficient ability to capture long-range contextual information due to the use of attention mechanisms.

For example, the BERT-TS$_{BASE}$ model [12] decomposes the relationship extraction task into a pipeline task, first determining whether a relationship exists between two entities and then making a relationship type judgment. HIN-BERT$_{BASE}$ [13] uses a multilayer network to implement the process of inference. The ATLOP model achieves implicit inference by using the attention mechanism again after encoding with additional contextual information.

Local context pooling improves entity-to-vector representation by introducing local context vectors. The method exploits the dependencies between word vectors in the pre-trained model directly without relearning the attention matrix, thereby saving time and computational resources.

In the pre-trained multi-head attention matrix $A \in R^{H \times l \times l}$, A_{ijk} denotes the attention coefficients in the k-th attention head from the i-th word vector to the j-th word vector. The special symbol "*" in front of the mention indicates the mentioned word, and the attention of the entity $A_i^E \in R^{H \times l}$ is represented by the mean of the attention coefficients of all mentioned words of the same entity. The attention coefficients of two entities A_S^E, A_O^E in the entity pair (e_s, e_o) are multiplied to obtain the weight of the contextual content on the entity pair, and the local context vector $c^{(s,o)}$ is calculated. As shown in Equation (1):

$$
\begin{aligned}
A^{(s,o)} &= A_s^E \bullet A_o^E \\
q^{(s,o)} &= \sum_{i=1}^{H} A_i^{(s,o)} \\
\alpha^{(s,o)} &= q^{(s,o)} / 1^T q^{(s,o)} \\
c^{(s,o)} &= H^T \alpha^{(s,o)}
\end{aligned}
\tag{1}
$$

where H is the sequence of word vectors obtained by the pre-training model. The entity representation in the entity pair is obtained by fusing the local context vector $c^{(s,o)}$ and the entity vectors h_{e_s} and h_{e_o}, as shown in Equation (2):

$$
\begin{aligned}
z_s &= tanh(W_s h_{e_s} + W_{c_1} c^{(s,o)}) \\
z_o &= tanh(W_o h_{e_o} W_{c_2} c^{(s,o)})
\end{aligned}
\tag{2}
$$

where $W_{C1}, W_{C2} \in R^{d \times d}$ is the model variable.

Local context pooling is shown in Figure 1. The word vectors are pooled by weighted averages to obtain the local context vectors $c^{(s,o)}$ of the entity pair (e_s, e_o), and context vectors $c^{(s,o)}$ are fused with the entity vectors h_{e_s} and h_{e_o}. In this process, attention coefficients of the head entity and the tail entity are multiplied to obtain the weights of the word vectors. Therefore, only the word vector with high attention coefficients for both entities will receive high weights in the context vector. The word vectors with higher weights in the context vector are marked in light yellow in Figure 1.
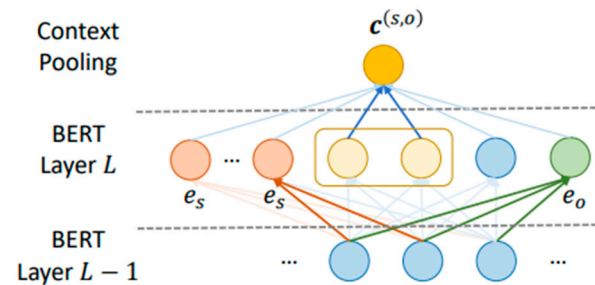

Figure 1. Illustration of local context pooling.

However, implicit inference mainly relies on the attention mechanism, which cannot solve the problem of the decreasing effectiveness of relation extraction with an increase in sentence length.

2.2. Document-Level Relationship Extraction Models Based on Graph Structures

Graph-structured document-level relationship extraction models construct graph structures based on dependency structures or structured attention, containing different types of nodes and edges such as entity nodes and sentence nodes to capture information about documents. Then, relationships between entities in the document are obtained using graph neural networks. Since scattered entities in the document are connected by the constructed graph structure data to obtain relationship information between entities, the problem of serialized encoders' (e.g., recurrent neural network encoders) inability to capture information over long distances is alleviated.

Among these models, the attention-guided graph convolutional network (AGGCN) [14] model converts the dependency tree into a fully connected weighted graph based on the attention mechanism, followed by the graph convolutional network (GCN) [15] for relationship extraction between entities. Latent structure refinement (LSR) [16] constructs entity nodes to automatically perform document graph construction without relying on dependencies such as syntax trees to construct static document graph structures.

Sahu encodes all words in the document with additional location information and constructs a graph structure based on the document, which contains two types of nodes: entity nodes and document nodes, and five types of edges. The vector of each node is obtained by GCN for each of these edges, followed by classification of the relationship for any two entities by a linear activation neural network. This model creates a document node to shorten the shortest distance between any nodes and solves the problem of long-distance dependence, as shown in Figure 2 below [17].
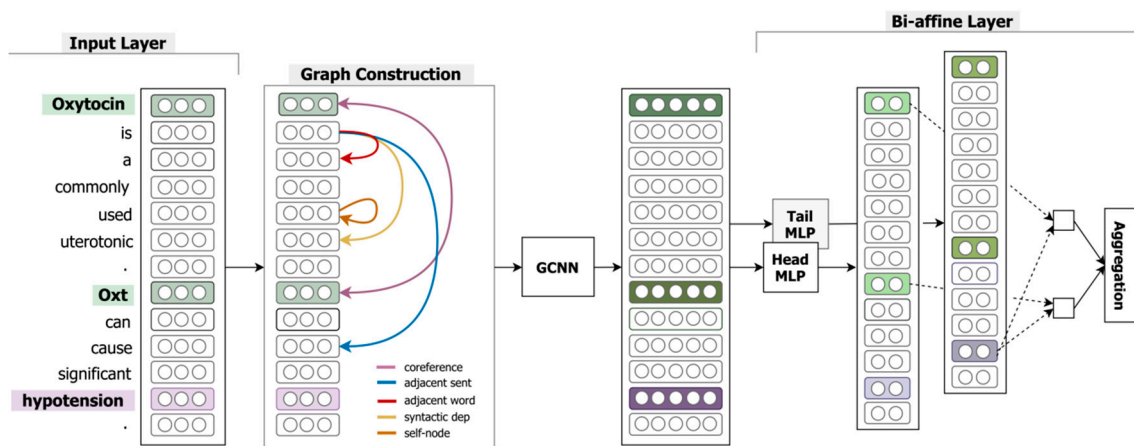
Figure 2. Framework of node-driven graph neural network model.

In addition to node-driven graph neural networks that construct the graph structure to obtain the node vector and the relationship vector between two nodes through a neural network, Christopoulou proposed the edge-driven neural network. This approach processes each sentence in the document to obtain word vectors using a BiLSTM layer, constructing a graph structure containing three types of nodes (mention nodes, entity nodes, and sentence nodes) and five types of edges. The graph is aggregated several times by graph convolution to obtain the edges between entities, which are the relationship vectors. Softmax is used to classify the relationship vectors. However, relevant information may be lost in this model's multiple GCN networks that are required to obtain edge vectors, since edges directly connecting entities do not exist.

To enhance the expressive capability, the GAIN model employs a bipartite graph comprising two layers to represent textual information. Initially, the complete document is encoded; subsequently, mention nodes and document nodes are incorporated into the first layer of the graph structure, which encompasses three types of edges. The mention nodes' vectors are acquired through the utilization of graph convolutional networks (GCNs). Afterwards, the mention vectors corresponding to the same entities are aggregated through averaging, resulting in entity vectors. These entity-to-entity paths are then represented by applying an attention-mechanism-based approach, which involves an inference process. By following the illustration in Figure 3 below, a multi-classification task is executed [18].
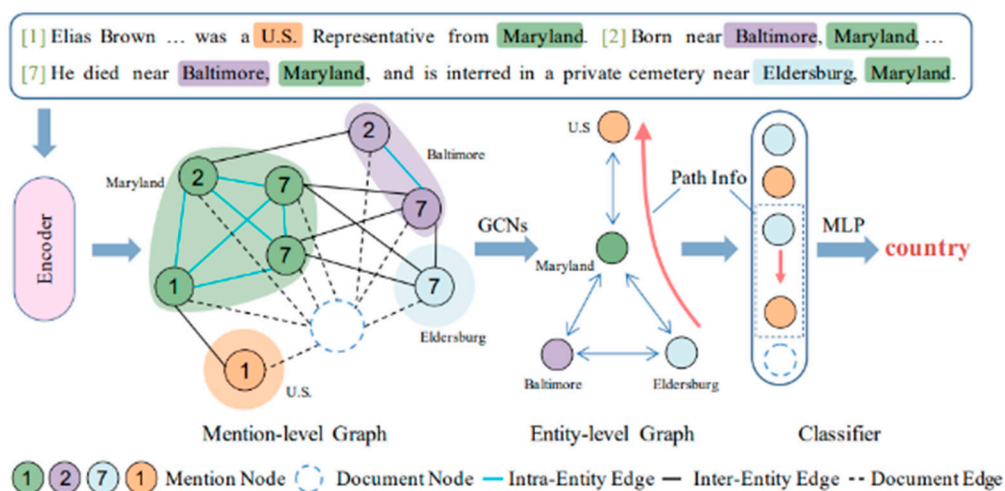

Figure 3. GAIN model framework.

In the GCN network, the information of each node is obtained by weighted summation of the information of that node in the previous layer and the information of the neighboring nodes, followed by linear and nonlinear transformations, as shown in Figure 4.
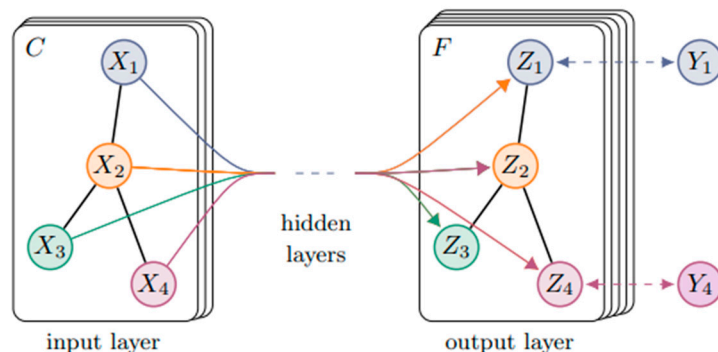

Figure 4. Illustration of GCN.

For node u at layer i in Figure 4, the equation is shown in Equation (3):

$$h_u^{l+1} = \sigma\left(\sum_{k\in\kappa}\sum_{v\in N_k(u)} W_k^l h_v^l + b_k^l\right) \tag{3}$$

where κ denotes edges of different types, and $W_k^l \in R^{d\times d}$ and $b_k^l \in R^d$ are model parameters. $N_k(u)$ denotes the nodes which node u is connected to by an edge of type k. σ is the ReLU activation function.

Document-level relationship extraction based on graph structure often requires constructing syntactic or semantic structures of documents, which is less efficient in processing. The construction process is time-consuming and computationally expensive.

3. Document-Level Relationship Extraction Model with Local Relationships and Global Inference

The proposed document-level relationship extraction model with local relationships and global inference is an extension of the ATLOP network with the addition of a global inference layer and a fusion layer of local relationships and global inference. The model framework is shown in Figure 5.

3.1. Definition of Symbols

In a document d, $d = [x_t]_{t=1}^l$, where x_t denotes the tth word, and l is the length of the document. the entity is denoted by e_i. The same entity may appear several times in a document, and the name of the entity that appears each time is called a mentioned word and denotes the jth mentioned word of the i-th entity by m_i^j. Each mentioned word can be composed of several words.

3.2. Local Relationships

The local relationships defined in this paper are the relationships between pairs of entities obtained by context pooling only. Local relationships are obtained by this model using the ATLOP network.

The position before and the position after each mentioned word in the text is filled by the pre-processing layer using the symbol "*" to mark the mentioned words.

The pre-processed text content is encoded in the BERT encoding layer using the BERT pre-training model, as shown in Equation (4):

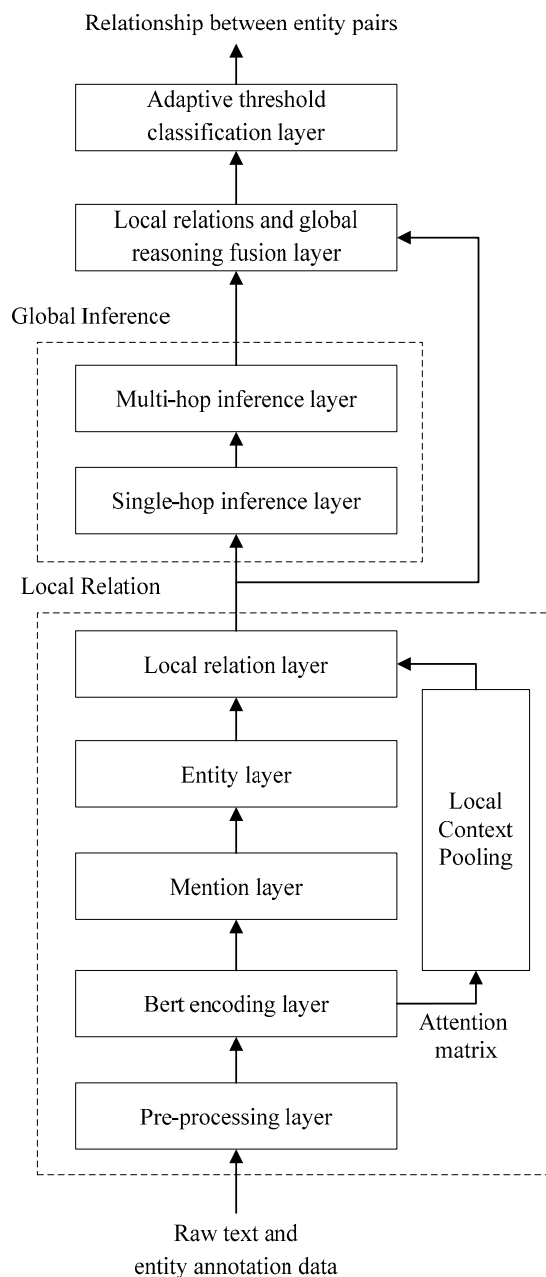$$H = [h_1, h_2, \ldots, h_l] = Bert([x_1, x_2, \ldots, x_l]) \tag{4}$$

Relationship between entity pairs

```
          ┌──────────────────────┐
          │  Adaptive threshold  │
          │ classification layer │
          └──────────────────────┘
          ┌──────────────────────┐
          │ Local relations and  │
          │ global reasoning     │◄──┐
          │ fusion layer         │   │
          └──────────────────────┘   │
Global Inference                     │
  ┌─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─┐      │
  │ ┌──────────────────────┐  │      │
  │ │ Multi-hop inference  │  │      │
  │ │       layer          │  │      │
  │ └──────────────────────┘  │      │
  │ ┌──────────────────────┐  │      │
  │ │ Single-hop inference │  │      │
  │ │       layer          │  │      │
  │ └──────────────────────┘  │      │
  └─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─┘      │
Local Relation                       │
  ┌─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─┐
  │ ┌──────────────────┐  ┌────────┐  │
  │ │ Local relation   │◄─┤        │  │
  │ │     layer        │  │ Local  │  │
  │ └──────────────────┘  │Context │  │
  │ ┌──────────────────┐  │Pooling │  │
  │ │   Entity layer   │  │        │  │
  │ └──────────────────┘  │        │  │
  │ ┌──────────────────┐  │        │  │
  │ │  Mention layer   │  │        │  │
  │ └──────────────────┘  └────────┘  │
  │ ┌──────────────────┐   Attention  │
  │ │ Bert encoding    │◄──  matrix   │
  │ │     layer        │              │
  │ └──────────────────┘              │
  │ ┌──────────────────┐              │
  │ │ Pre-processing   │              │
  │ │     layer        │              │
  │ └──────────────────┘              │
  └─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─┘
```

Raw text and
entity annotation data

Figure 5. Model framework.

The word vector encoded with the symbol "*" in front of each mentioned word in the BERT encoding layer contains the information of the mentioned word, so the mentioned word layer inputs this word vector into the entity layer as the vector $h_{m_i^j}$ of the corresponding mentioned word.

The vector h_{e_i} of an entity is obtained at the entity level by pooling a number of mentioned word vectors representing this entity by the logsumexp function, as shown in Equation (5):

$$h_{e_i} = log\sum_{j=1}^{N_{e_i}} exp(h_{m_i^j}) \tag{5}$$

The multi-headed attention matrix data in the BERT encoding layer are processed by local context pooling to obtain the attention coefficients of the entity pair in the context $c^{(i,j)}$.

Since the relationship between entity pairs is directional, for two entities e_i and e_j, the relationship between them is represented by two different relationship vectors r_{e_i,e_j} and r_{e_j,e_i}.

Taking r_{e_i,e_j} as an example, e_i is the head entity, and e_j is the tail entity. To distinguish the head entity and tail entity of the entity pair, the entity vectors h_{e_i} and h_{e_j}, and the attention coefficient $c^{(i,j)}$ of the entity pair in the context, are used by the local relationship representation layer to calculate the head entity vector z_s and the tail entity vector z_o, as shown in Equation (6):

$$z_s = tanh(W_s h_{e_i} + W_{c_1} c^{(i,j)})$$
$$z_o = tanh(W_o h_{e_j} + W_{c_2} c^{(i,j)}) \tag{6}$$

where W_s, W_o, W_{c_1}, W_{c_2} denote the training parameters of the neural network in the local relational representation layer.

A bilinear group transformation is performed on z_s and z_o to obtain the local relation r_{e_i,e_j} with e_i as the head entity and e_j as the tail entity, as shown in Equation (7):

$$[z_s^1; \ldots; z_s^k] = z_s$$
$$[z_o^1; \ldots; z_o^k] = z_o$$
$$r_{e_i,e_j} = \sigma((z_s^i)^T W_r z_o^i + b_r) \tag{7}$$

where W_r, b_r, are the training parameters of the neural network in the local relational representation layer.

3.3. Global Inference

Local relations between entity pairs are obtained using local context pooling, which relies on the attention mechanism in the BERT pre-training model. However, the attention mechanism is affected by the sentence length, and the longer the sentence length is, the more the accuracy of its relation extraction gradually decreases, which is especially obvious in document-level relation extraction.

To address this issue, the ATLOP model is extended in this paper to implement global relational inference.

On the basis of local context pooling of text contents to obtain local relations between entity pairs, Floyd's algorithm is improved to perform global inference, i.e., to derive global relations between entity pairs from local relations.

In the process of global inference, the computation of relationships between entity pairs will no longer depend on the entity positions in the document but on the local relationships.

Floyd's algorithm is improved by this model, and an explicit relation inference mechanism (explicit relation inference layer based on the improved Floyd's algorithm) is proposed to solve the global relation inf_{e_i,e_j} for any pair of entities (e_i and e_j) with e_i as the head entity and e_j as the tail entity.

3.3.1. Classical Floyd's Algorithm

The weights of the edges between any two nodes represent the distance between the nodes. In the classical Floyd algorithm, for any pair of nodes (i, j), the nodes other than node i and node j are used as intermediate nodes, and the shortest distance $D_{i,j}$ between node i and node j is continuously updated by the state transfer equation iterating over the intermediate nodes.

For any intermediate node, according to the state transfer equation

$$D_{i,j} = min(D_{i,j}, D_{i,k} + D_{k,j}) \tag{8}$$

In the case of containing at most one intermediate node k, the shortest distance $D_{i,j}$ of nodes i and j can be obtained, where $D_{i,k}$ is the distance from node i to intermediate node k, and $D_{k,j}$ is the distance from intermediate node k to node j.

In the case that the path contains multiple intermediate nodes, in order to calculate the shortest distance between node i and node j, the set of nodes other than these two nodes is defined as $(1 \ldots k)$. $D_{i,j,k}$ denotes the length of the shortest path from node i to node j using the nodes in the set $(1 \ldots k)$ as intermediate nodes.

If node k is in the shortest path:

$$D_{i,j,k} = D_{i,j,k-1} + D_{k,j,k-1} \tag{9}$$

If node k is not in the shortest path:

$$D_{i,j,k} = D_{i,j,k-1} \tag{10}$$

Therefore,

$$D_{i,j,k} = min(D_{i,j,k-1}, D_{i,k,k-1} + D_{k,j,k-1}) \tag{11}$$

If there are several different ways between two nodes, the distance between those nodes is the shortest one. The above process is iterated until the number of iterations is equal to the number of intermediate nodes or the shortest distance between all pairs of nodes no longer changes with the number of iterations, and the iterative process ends. Finally, $D_{i,j,k}$ is the shortest distance between node i and node j under the condition of the path containing more than one intermediate node.

3.3.2. Explicit Relational Inference Based on Improved Floyd's Algorithm

The explicit relational inference mechanism proposed in this paper is a modification of the state transfer equation of the classical Floyd algorithm.

(1) Single-hop relational inference mechanism

The distance D in the state transfer equation in the classical Floyd algorithm is modified to a local relation vector r. Since the inference relation inf_{e_i,e_j,e_k} containing information about the intermediate entity e_k with e_i as the head entity and e_j as the tail entity cannot be obtained by direct summation of the local relations, the computational model of the distance between node i and node j passing through intermediate node k under the constraint of a single intermediate node is modified from $D_{i,k} + D_{k,j}$ to a neural network constructed with a bilinear function whose inputs are two local relations r_{e_i,e_k} and r_{e_k,e_j}, as Equation (12) shows:

$$inf_{e_i,e_j,e_k} = tanh((r_{e_i,e_k})^T W_{inf} r_{e_k,e_j} + b_{inf}) \tag{12}$$

where W_{inf}, b_{inf} are the training parameters in the global inference layer neural network.

In order to fuse the information of all paths, the set of entities $(e_1 \ldots e_k)$ except e_i and e_j is represented by $\Re$. After traversing all intermediate nodes, the minimum value calculation method in the classical Floyd algorithm is modified to calculate the average value for all inference relation vectors containing information of a single intermediate node to obtain the global inference relation inf_{e_i,e_j} for the entity pair. The modified state transfer equation is shown in Equation (13):

$$inf_{e_i,e_j} = \frac{1}{K} \sum_{e_k \in R} tanh((r_{e_i,e_k})^T W_{inf} r_{e_k,e_j} + b_{inf}) \tag{13}$$

where K is the number of entities in the set $\Re$.

When $K = 2$, the single-hop relational inference process based on the improved Floyd algorithm is shown in Figure 6.
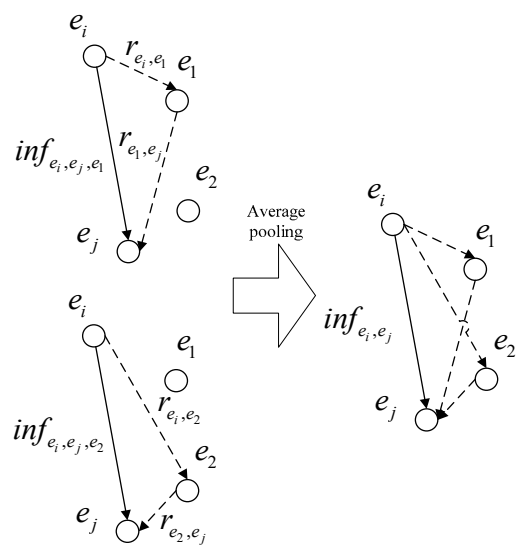
Figure 6. Single-hop relational inference mechanism based on improved Floyd's algorithm.

The nodes in Figure 6 represent entities, the edges represent local relations, and the weights of the edges are the local relation vectors, which are calculated by the method in Section 3.2. The direction of the edge is the direction of the relationship.

Single-hop relational inference takes the entities in the document other than the head entity e_i and the tail entity e_j as the intermediate entities e_k of the global inference relation, and for each intermediate entity e_k, a neural network model constructed with a bilinear function is used to obtain the inference relation inf_{e_i,e_j,e_k} between pairs of entities containing the intermediate entity e_k.

After traversing the intermediate entities, the global inference relation $\left\{ inf_{e_i,e_j,e_k} \middle| e_k \in \Re \right\}$ between pairs of entities containing a single intermediate entity is obtained by pooling the mean value between the pairs of entities containing a single intermediate entity inf_{e_i,e_j}.

In order to make full use of the rich relationship information implied between multiple entities in documents, the single-hop relationship inference algorithm is extended to propose a multi-hop relationship extraction calculation.

(2) Multi-hop relational inference mechanism

The entities in the document other than the head entity e_i and the tail entity e_j are used as intermediate entities for multiple relationship inference.

For each intermediate entity e_k, two global single-hop inference relations inf_{e_i,e_k} with e_i as the head entity and e_k as the tail entity and inf_{e_k,e_j} with e_k as the head entity and e_j as the tail entity, respectively, are obtained by inference about the single-hop relations, as shown in Equation (13).

Each global single-hop inference relation is obtained by traversing the intermediate entities other than the head and tail entities of the global single-hop inference relation in the document to complete the inference calculation.

We fed inf_{e_i,e_k} and inf_{e_k,e_j} into a neural network constructed as a bilinear function to obtain the multi-hop inference relation inf_{e_i,e_j,e_k} containing information about these two global single-hop inference relations. $\left\{ inf_{e_i,e_j,e_k} \middle| e_k \in \Re \right\}$ is the inference of all multi-hop relations computed by traversing the intermediate entity. After all multi-hop inference relations are pooled by the average value, the global multi-hop inference relation inf_{e_i,e_j} containing the information of multiple intermediate entities is obtained.

The global inference relation inf_{e_i,e_j} from entity e_i to entity e_j contains information about entities in the set $(e_1 \ldots e_k)$, as shown in Equation (14):

$$inf_{e_i,e_j,e_k} = tanh((inf_{e_i,e_k})^T W_{inf} inf_{e_k,e_j} + b_{inf})$$
$$inf_{e_i,e_j} = \frac{1}{K} \sum_{e_k \in R} inf_{e_i,e_j,e_k}$$

(14)

where inf_{e_i,e_j,e_k} is the inference relation computed using inf_{e_i,e_k} and inf_{e_k,e_j} iterations from entity e_i to entity e_j when the last traversed intermediate entity is e_k. inf_{e_i,e_j} is the global inference relation from the head entity e_i to the tail entity e_j. The same is true for inf_{e_i,e_k} and inf_{e_k,e_j}.

When $K = 2$, the multi-hop relational inference mechanism based on the improved Floyd algorithm is shown in Figure 7.
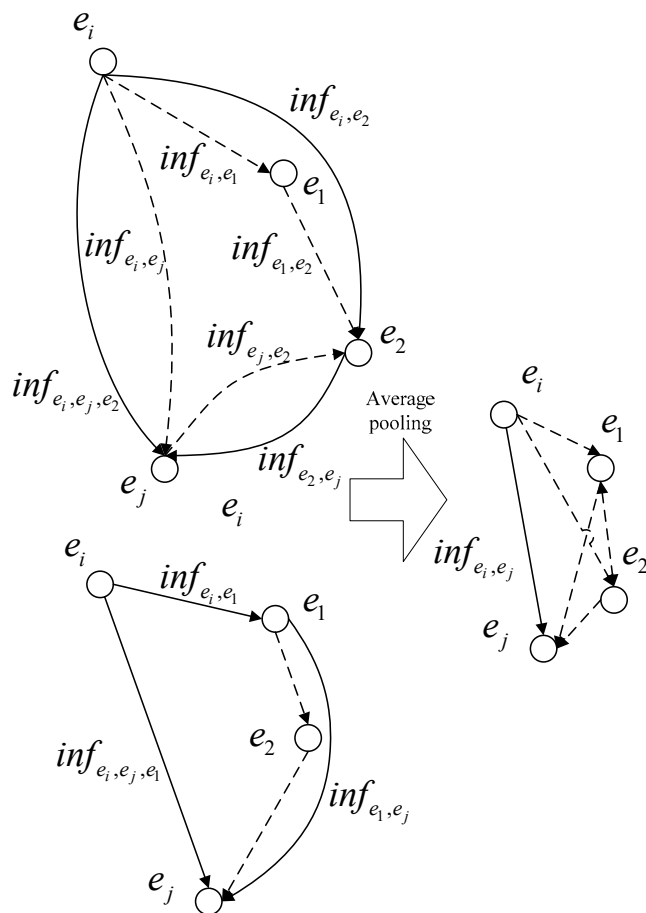

Figure 7. Multi-hop relational inference mechanism based on improved Floyd's algorithm.

In Figure 7, the single-hop inference relations inf_{e_i,e_1}, inf_{e_1,e_2}, inf_{e_i,e_j}, and inf_{e_j,e_2} are obtained according to Equation (10), and then the global inference relation inf_{e_i,e_2} is obtained after the first iteration of the multi-hop inference layer, as shown in Equation (15).

$$inf_{e_i,e_2} = \frac{1}{2}(tanh((inf_{e_i,e_1})^T W_{inf} inf_{e_1,e_2} + b_{inf}) + tanh((inf_{e_i,e_j})^T W_{inf} inf_{e_j,e_2} + b_{inf})) \tag{15}$$

The global inference relation is obtained in the same way.

The global relations inf_{e_i,e_2} and inf_{e_2,e_j} are fed into a neural network constructed with a bilinear function to obtain the inference relation inf_{e_i,e_j,e_2} with e_2 as the intermediate entity, as shown in Equation (16):

$$inf_{e_i,e_j,e_2} = tanh((inf_{e_i,e_2})^T W_{inf} inf_{e_2,e_j} + b_{inf}) \tag{16}$$

Similarly, the inference relation inf_{e_i,e_j,e_1} is obtained.

The global relation inf_{e_i,e_j} is obtained by averaging pooling for the inference relations inf_{e_i,e_j,e_1} and inf_{e_i,e_j,e_2}, as shown in Equation (17):

$$inf_{e_i,e_j} = \frac{1}{2}(inf_{e_i,e_j,e_2} + inf_{e_i,e_j,e_1}) \tag{17}$$

The second iteration process is represented by Equations (16) and (17).

Multi-path multi-hop inference can be implemented by the above procedure to obtain the global relation inf_{e_i,e_j} for the head entity e_i pointing to the tail entity e_j.

When $K > 2$, i.e., the number of intermediate entities is greater than two, the inference process is still as shown in Equation (14).

Since the intermediate entities between entity pairs can be repeatedly traversed during the iteration, the number of intermediate entities traversed by the global inference relation calculation between any entity pair is 2N−1 after N iterations of multi-path multi-hop inference.

The flowchart of the algorithm for the global inference mechanism in this model is shown in Figure 8.
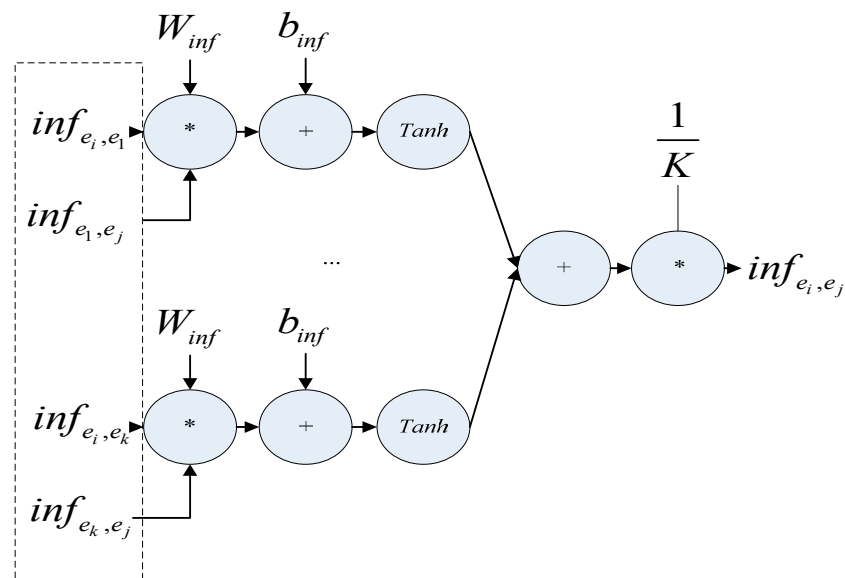

Figure 8. Algorithm flowchart of the global multi-hop relational inference mechanism.

In Figure 8, e_i, e_j are any head and tail entities in the document, and their output global multi-hop inference relations inf_{e_i,e_j}, which can be continuously updated by iteration and used as input for global multi-hop inference, are calculated to obtain global multi-hop inference relations for other head and tail entities.

3.4. Fusion of Local Relations and Global Inference

The vector r_{e_i,e_j} containing local relations of contextual information and the vector inf_{e_i,e_j} based on global inference of explicit inference are fed into the neural network for fusion to achieve document relation extraction, as shown in Equation (18):

$$\begin{aligned} rel_{e_i,e_j} &= tanh(W_{con}cat\left(r_{e_i,e_j}, inf_{e_i,e_j}\right) \\ &\quad + b_{con}) \end{aligned} \tag{18}$$

where the symbol rel_{e_i,e_j} is the relational vector fusing local relations and global inference, W_{con}, b_{con} are the training parameters of the neural network, and $cat()$ is the splicing operation. Its algorithm flowchart is shown in Figure 9.
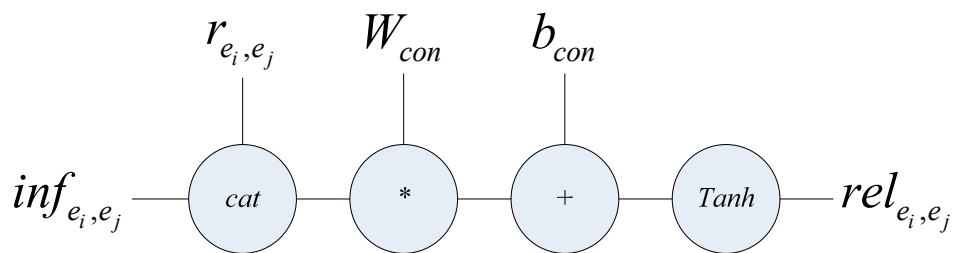
Figure 9. Flowchart of the fusion algorithm for local relations and global inference. As is shown in the figure, * means Multiplication calculation.

Adaptive threshold classification is performed on the relationship vector rel_{e_i,e_j} to obtain the type of relationship between entity pairs.

4. Experimental Design and Analysis

To ensure easy reproducibility, the proposed document-level relationship extraction model with local relation and global inference is available at https://github.com/Emir-Liu/Document-Level-Relation-Extraction-Fusing-Local-Relation-and-Global-Reasoning (accessed on 24 June 2023).

4.1. Dataset Acquisition and Processing

In this paper, based on DocRED, a large-scale dataset for document-level relationship extraction with data from English Wikipedia was manually annotated as the testing set to verify the performance of the LRGI approach, and the testing results show that more than 40.7% of the relationships in this dataset need to be judged by multiple sentences, 61.1% of the relationships need to be obtained by inference, and 7% of the entity pairs with relationships have multiple relationships, as summarized by Table 1 below.

Table 1. Statistics of DocRED dataset. As is shown in the figure, # means the number of.

Statistics	Value
Dataset	DocRED
# Train	3053
# Dev	1000
# Test	1000
# Types of relation	96
# Entities	132,275
# Relations	56,354
Avg. # sentences per Doc.	8
Avg. # entities per Doc.	19.5

4.2. Experimental Environment and Parameter Settings

4.2.1. Experimental Environment

The deep learning framework used for the experiments in this paper is PyTorch. All training, validation, and testing of the models were conducted using the PyTorch framework, and the specific experimental environment is shown in Table 2.

Table 2. Experimental environment configuration.

Software and Hardware	Configuration
Development tools and languages	Pycharm+Python3.9
Development framework	Pytorch1.9.0
Operating System	64-bit Ubuntu20.04 system
GPU	INVDA Quadro RTX 6000 24G
CPU	Intel® Xeon(R) W-2225 CPU @ 4.10 GHz $\times$ 8

4.2.2. Experimental Parameter Settings

In this research paper, a novel approach for document-level relation extraction is presented, which integrates local relations and global inference. The model utilizes the base version of the BERT pre-training model and employs the AdamW optimization algorithm [19]. The specific parameter configurations employed for this model can be found in Table 3.

Table 3. Parameters of model.

Parameter	Value
Dimension of mentioned embedding	768
Dimension of entity embedding	768
Dimension of local relations vector	97
Dimension of global inference vector	97
Learning rate	5×10^{-5}
Dropout	0.1

4.3. Evaluation Indicators

Since there may be more than one relationship between entity pairs in document-level relationship extraction tasks, they are considered a multi-classification task. To properly evaluate the performance of various relationship extraction models and considering the characteristics of the task, precision and F_1 are used as evaluation metrics, which are defined as shown in Equation (19):

$$
\begin{aligned}
Precison &= \frac{TP}{TP+FP} \\
Recall &= \frac{TP}{TP+FN} \\
F_1 &= 2 * \frac{Precision * Recall}{Precision + Recall}
\end{aligned}
\tag{19}
$$

where TP denotes the number of correctly predicted entity pair relationships; FP denotes the number of incorrectly predicted entity pair relationships; and FN denotes the number of entity pair relationships that are not predicted.

In this paper, F_1 and accuracy are used as metrics in the experiments. Please note that overfitting may occur during the training process, leading to errors when evaluating the model's performance if the same entity pairs and inter-entity pair relationships exist in the training, validation, and test sets.

To address this problem, in this paper, we also use $lgnF_1$ score as a metric, which calculates F_1 metrics for the validation set after eliminating entity pairs and inter-entity pair relationships that have already appeared in the training set.

4.4. Contrast Model

The proposed document-level relationship extraction model that incorporates local relationships and global inference was compared with serialization-based and graph-structured models, using F_1 and $lgnF_1$ as metrics. The models involved in the comparison are as follows:

1. Serialization-based models

Serialization-based models encode the entire document through neural networks or pre-trained models without using graph structures.

(1) Convolutional neural network (CNN)/long short-term memory neural network (LSTM)/BiLSTM:

Word encoding is performed on the text using GloVe [20], and after encoding, word vectors are obtained using CNN, LSTM, and BiLSTM networks, respectively. Entity vectors are obtained by mean pooling and input to a bilinear function to predict the relationship between entity pairs.

(2) Context-Aware:

Word vector encoding is performed on text using GloVe and input to the LSTM network to obtain the relationship between entity pairs. Attention mechanisms are used to achieve information fusion.

(3) BERT$_{BASE}$/BERT-TS$_{BASE}$/HIN-BERT$_{BASE}$/CorefBERT$_{BASE}$ [21]:

All these models use pre-trained models for encoding serialized models. BERT$_{BASE}$ uses BERT pre-trained models for encoding. BERT-TS$_{BASE}$ decomposes the task into a pipeline task, a dichotomous classification task that first determines whether a relationship exists between two entities, and then a multi-label classification task that makes a relationship type determination. HIN-BERT$_{BASE}$ uses a multilayer network to implement the process of inference.

(4) BERT-ATLOP$_{BASE}$:

Word encoding is performed on text using the BERT pre-training model. Word encoding is pooled to obtain entity vectors. Relationship vectors are obtained using contextual pooling, and then adaptive threshold classification is performed to obtain the type of relationship between entity pairs.

(5) E2GRE [22]:

The E2GRE model concatenates document text with head entities to help LMs concentrate on parts of the document that are more related to the head entity.

(6) DocuNet [23]:

DocuNet leverages an encoder module to capture the context information of entities and a U-shaped segmentation module over the image-style feature map to capture global interdependency among triples.

2. Graph-structure-based models

Information in the documents is used to construct graph structure data on which the computation of relationships between entity pairs is performed.

(7) BiLSTM-AGGCN/BiLSTM-LSR:

After encoding using a BiLSTM network, BiLSTM-AGGCN converts dependency trees into fully connected band-weight graphs by an attention mechanism and uses GCN for relation computation. BiLSTM-LSR constructs mentioned words, entities, and sentence nodes by an LSR model for determining relations.

(8) BERT-LSR$_{BASE}$:

After using BERT encoding, the LSR model is used for relationship extraction.

(9) GAIN:

The text is encoded using the BERT model to construct a two-layer heterogeneous graph structure referring to mentioned word-level graphs and entity-level graphs. Explicit inference is implemented using GCN networks to obtain the relationships between entity pairs.

4.5. Experimental Results and Analysis

4.5.1. Comparative Experiments

The different models were experimented on with the DocRED dataset, and the experimental results are shown in Table 4.

Both GAIN and ATLOP models use the BERT pre-training model. In order to compare the performance of each model fairly, the basic version of the BERT pre-training model, BERT Base, is used in this paper.

The above model is replicated in this paper, and the same parameters are used for the original ATLOP model and document-level relationship extraction model proposed in this paper that fuses local relationships and global inference, for better comparison.

The proposed model in this paper achieves better results, with improvements in both F_1 and $lgnF_1$ indicators, with a 2% improvement in the F_1 score compared to the ATLOP model.

Comparing the proposed model in this paper with the ATLOP model, the accuracy is improved by about 28%, as shown in Table 5.

Table 4. F1 of different RE models on DocRED.

Model	Dev		Test	
	F_1	$lgnF_1$	F_1	$lgnF_1$
Serialization-based models:				
CNN	43.45	41.58	42.26	40.33
LSTM	50.66	48.44	50.07	47.71
BiLSTM	50.95	48.87	51.06	48.78
Context-Aware	51.10	48.94	50.70	48.40
BERTBASE	54.16	-	53.20	-
BERT-TSBASE	54.42	-	53.92	-
HIN-BERTBASE	56.31	54.29	55.60	53.70
CorefBERTBASE	57.51	55.32	56.96	54.54
BERT-ATLOPBASE	61.09	59.22	61.30	59.31
E2GRE	58.72	55.22	-	-
DocuNet	61.83	59.86	61.86	59.93
Graph-structure-based models:				
BiLSTM-AGGCN	52.47	46.29	51.45	48.89
BiLSTM-LSR	55.17	48.82	54.18	52.15
BERT-LSRBASE	59.00	52.43	59.05	56.97
GAIN	61.22	59.14	61.24	59.00
Proposed model:				
LRGI	62.11	60.48	61.71	59.62

Table 5. Comparison of LRGI and ATLOP and GAIN models.

Model	LRGI	ATLOP	GAIN
F_1	62.11	60.92	59.83
$lgnF_1$	60.48	59.03	58.10
Precision	0.73	0.57	0.58
Recall	0.54	0.65	0.58

4.5.2. Analysis of the Maximum Number of Relationships Existing between Each Pair of Entity Pairs

Since more than one type of relationship may exist between entity pairs, each relationship should have a different classification threshold to determine whether a certain relationship type exists in that entity pair.

In the DocRED dataset, there are 96 relationships, and there are at most three different relationships between each pair of entities. The original ATLOP model only judges the three most probable relationships among the entity pairs. Therefore, for this experiment, the maximum number of relationships between entity pairs was set to three, which is the same as the setting of the original ATLOP model.

In the experiments, comparisons were made separately for different maximum number of relations, as shown in Table 6.

Table 6. Comparison of different maximum relationship numbers between entity pairs on dev set.

Max Num of Relations between Entity Pairs	F_1	$lgnF_1$	*Precision*	*Recall*
1	57.53	56.30	0.68	0.49
2	59.14	57.58	0.68	0.52
3	61.67	60.10	0.72	0.53

Here, the metrics F_1, $lgnF_1$, precision, and recall of the model increase when the maximum number of relationships between entity pairs is increased. The extraction precision of the model is highest when the maximum number of relationships between entity pairs is three.

4.5.3. Analysis of Model Parameters

To represent in more detail the effect of the local relation dimension and global inference relation dimension on the experimental results in the relation extraction model that fuses local relations and global inference, the model parameters were modified to analyze the effect of this mechanism. The training procedure is shown in Figures 10–12 and the configured parameters and evaluation results are shown in Table 7.
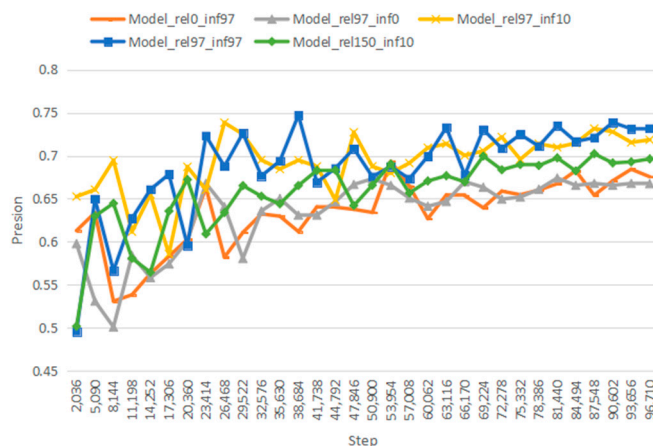

Figure 10. Precision of different models in training.
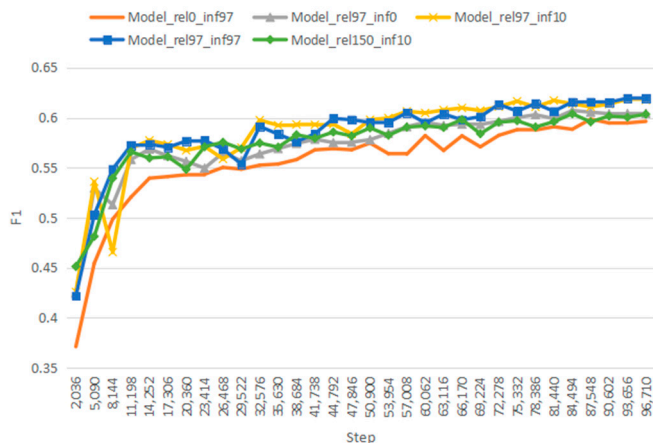
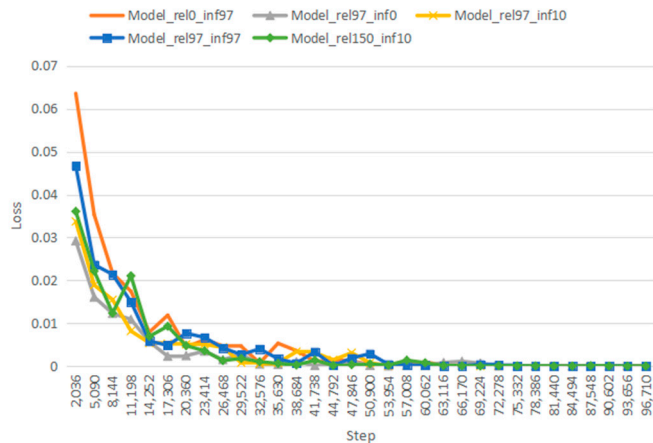**Figure 11.** F1 of different models in training.

Figure 12. Loss of different models in training.

Table 7. Parameters and results of different models. As shown in the table, # indicates dimension.

Model	# rel	# inf	F_1	$lgnF_1$	Pre	Rec
Model_rel0_inf97	0	97	59.56	57.94	0.67	0.539
Model_rel97_inf0	97	0	60.57	58.75	0.66	0.554
Model_rel97_inf10	97	10	61.67	60.10	0.72	0.539
Model_rel97_inf97	97	97	62.10	60.48	0.73	0.538
Model_rel150_inf10	150	10	60.32	58.72	0.69	0.534

Here, *Model_relx_infy* denotes a document-level relation extraction model with local relation dimension *x* and global inference dimension *y* that incorporate local relation and global inference. When *x* is zero, it represents a document-level relation extraction model that contains only global inference, and the global inference dimension is *y*. Similarly, when *y* is zero, it represents a document-level relation extraction model that contains only local relations, and the local relation dimension is *x*.

First, the *Model_rel97_inf0* model with only local relations, *Model_rel0_inf97* model with only global inference, and *Model_rel97_inf97* model with fusion of local relations and global inference are compared.

The *Model_rel97_inf0* model containing only local relations has higher recall, F_1 and $lgnF_1$, while the *Model_rel0_inf97* model containing only global inference has better precision.

The *Model_rel97_inf97* model that incorporates both local relations and global inference has higher precision compared to the two models that use only a single relation. It contains both local relations and global inference, so although the recall is slightly reduced due to the additional information added, both F_1 and $lgnF_1$ parameters are improved for the overall effect.

A change in the global inference dimension directly affects the precision and recall of the model, which in turn improves F_1 and $lgnF_1$. While keeping the local relationship dimension at 97, the precision of the model improves, and the recall rate decreases slightly with the increase in the global inference dimension. Meanwhile, both F_1 and $lgnF_1$ are improved, indicating that increasing the global inference dimension can improve the overall extraction effect of the model.

When the dimensionality of global inference remains unchanged, F_1 and $lgnF_1$ of the model greatly improve when the number of local relation changes from 0 to 97, while F_1 and $lgnF_1$ of the model become worse when the number of local relation changes from 97 to 150. This indicates that simply increasing the dimension of local relations has some limitations, and global inference can be a good supplement to improve the accuracy of extraction.

5. Conclusions

In order to address the problem of the decreasing effectiveness of the attention mechanism in document-level relation extraction tasks with increasing text length, this paper first proposes an improved Floyd algorithm, which allowed us to extend the ATLOP model, and then we were able to establish a general framework for multi-path single-hop global inference and a multi-path multi-hop global inference for the extraction of document-level relationships. Then, combining this framework on the inference with local relation computation, the LRGI (standing for "Local Relationship and Global Inference") method on document-level relation extraction with local relation and global inference is proposed.

The proposed model is compared with other document-level relation extraction models on the DocRED dataset, showing improved accuracy and F_1 values.

However, we would like to point out that when applying our LGRI model to other datasets, the dimensionality of the global inference relationship vector in the global inference layer needs to be adjusted to achieve the best relationship extraction.

The multi-hop inference layer in our LGRI model contains two iterations of the process. During each iteration, the relational inference vector between entity pairs is updated, and the number of intermediate nodes used between entity pairs is also increased, making the

global inference relationship between entity pairs contain richer information about the relationships implied between multiple entities.

Finally, we would also like to point out that using more iterative processes will lead to too many intermediate entities being used in the global inference relationship, resulting in entity pairs consisting of intermediate nodes being repeatedly inferred multiple times during inference, and this may lead to duplicate contents in the global relationship. Future research needs to explore how to choose the optimal number of iterations and avoid duplicate inference information in the inference process for the iterations.

Author Contributions: Conceptualization, Y.L.; supervision, H.S., F.N., G.Z. and G.X.Y. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yu, M.; Yin, W.; Hasan, K.S.; Santos, C.D.; Xiang, B.; Zhou, B. Improved neural relation detection for knowledge base question answering. *arXiv* **2017**, arXiv:1704.06194.
2. Shi, P.; Lin, J. Simple bert models for relation extraction and semantic role labeling. *arXiv* **2019**, arXiv:1904.05255.
3. Soares, L.B.; FitzGerald, N.; Ling, J.; Kwiatkowski, T. Matching the blanks: Distributional similarity for relation learning. *arXiv* **2019**, arXiv:1906.03158.
4. Miwa, M.; Bansal, M. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv* **2016**, arXiv:1601.00770.
5. Alt, C.; Hübner, M.; Hennig, L. Improving relation extraction by pre-trained language representations. *arXiv* **2019**, arXiv:1906.03088.
6. Tang, Y.; Huang, J.; Wang, G.; He, X.; Zhou, B. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. *arXiv* **2020**, arXiv:1911.04910.
7. Christopoulou, F.; Miwa, M.; Ananiadou, S. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. *arXiv* **2019**, arXiv:1909.00228.
8. Zhou, W.; Huang, K.; Ma, T.; Huang, J. Document-level relation extraction with adaptive thresholding and localized context pooling. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2021; pp. 14612–14620.
9. Yao, Y.; Ye, D.; Li, P.; Han, X.; Lin, Y.; Liu, Z.; Liu, Z.; Huang, L.; Zhou, J.; Sun, M. DocRED: A large-scale document-level relation extraction dataset. *arXiv* **2019**, arXiv:1906.06127.
10. Weisstein, E.W. Floyd-Warshall Algorithm. Available online: https://mathworld.wolfram.com/2008 (accessed on 21 June 2008).
11. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
12. Wang, H.; Focke, C.; Sylvester, R.; Mishra, N.; Wang, W. Fine-tune bert for docred with two-step process. *arXiv* **2019**, arXiv:1909.11898.
13. Tang, H.; Cao, Y.; Zhang, Z.; Cao, J.; Fang, F.; Wang, S.; Yin, P. Hin: Hierarchical inference network for document-level relation extraction. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, 11–14 May 2020; pp. 197–209.
14. Guo, Z.; Zhang, Y.; Lu, W. Attention guided graph convolutional networks for relation extraction. *arXiv* **2019**, arXiv:1906.07510.
15. Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; Weinberger, K. Simplifying graph convolutional networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6861–6871.
16. Nan, G.; Guo, Z.; Sekulić, I.; Lu, W. Reasoning with latent structure refinement for document-level relation extraction. *arXiv* **2020**, arXiv:2005.06312.
17. Sahu, S.K.; Christopoulou, F.; Miwa, M.; Ananiadou, S. Inter-sentence relation extraction with document-level graph convolutional neural network. *arXiv* **2019**, arXiv:1906.04684.
18. Zeng, S.; Xu, R.; Chang, B.; Li, L. Double Graph Based Reasoning for Document-level Relation Extraction. *arXiv* **2020**, arXiv:2009.13752.
19. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
20. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
21. Ye, D.; Lin, Y.; Du, J.; Liu, Z.; Li, P.; Sun, M.; Liu, Z. Coreferential Reasoning Learning for Language Representation. *arXiv* **2020**, arXiv:2004.06870.

22. Huang, K.; Qi, P.; Wang, G.; Ma, T.; Huang, J. Entity and evidence guided document-level relation extraction. In Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021), Online, 6 August 2021; pp. 307–315.

23. Zhang, N.; Chen, X.; Xie, X.; Deng, S.; Tan, C.; Chen, M.; Huang, F.; Si, L.; Chen, H. Document-level relation extraction as semantic segmentation. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, Main Track, 19–27 August 2021; pp. 3999–4006.