Evidence-aware Document-level Relation Extraction

Tianyu Xu
School of Computer Science and Technology, Soochow University
Suzhou, China
tyxutianyu@stu.suda.edu.cn

Wen Hua
School of Information Technology and Electrical Engineering, The University of Queensland
St Lucia, Australia
w.hua@uq.edu.au

Jianfeng Qu*
School of Computer Science and Technology, Soochow University
Suzhou, China
jfqu@suda.edu.cn

Zhixu Li*
Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University
Shanghai, China
zhixuli@fudan.edu.cn

Jiajie Xu
School of Computer Science and Technology, Soochow University
Suzhou, China
xujj@suda.edu.cn

An Liu
School of Computer Science and Technology, Soochow University
Suzhou, China
anliu@suda.edu.cn

Lei Zhao
School of Computer Science and Technology, Soochow University
Suzhou, China
zhaol@suda.edu.cn

ABSTRACT

Document-level Relation Extraction (RE) is a promising task aiming at identifying relations of multiple entity pairs in a document. However, in most cases, a relational fact can be expressed enough via a small subset of sentences from the document, namely evidence sentence. Moreover, there often exist strong semantic correlations between evidence sentences that collaborate together to describe a specific relation. To address these challenges, we propose a novel evidence-aware model for document-level RE. Particularly, we formulate evidence sentence selection as a sequential decision problem through a crafted reinforcement learning mechanism. Considering the explosive search space of our agent, an efficient path searching strategy is executed on the converted document graph to heuristically obtain hopeful sentences and feed them to reinforcement learning. Finally, each entity pair owns a customized-filtered document for further inferring the relation between them. We conduct various experiments on two document-level RE benchmarks and achieve a remarkable improvement over previous competitive baselines, verifying the effectiveness of our method.

通过一个精心设计的强化学习机制，将证据句子选择表述为一个顺序决策问题

CCS CONCEPTS

• **Information systems** → **Information extraction**.

*Corresponding author

CIKM '22, October 17–21, 2022, Atlanta, GA, USA.

ACM ISBN 978-1-4503-9236-5/22/10...$15.00
https://doi.org/10.1145/3511808.3557313

KEYWORDS

Document-level relation extraction; reinforcement learning; evidence extraction

ACM Reference Format:
Tianyu Xu, Wen Hua, Jianfeng Qu, Zhixu Li, Jiajie Xu, An Liu, and Lei Zhao. 2022. Evidence-aware Document-level Relation Extraction. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557313

1 INTRODUCTION

Relation extraction (RE) is a fundamental task in information extraction, which aims at identifying semantic relations between entity pairs from plain text and has facilitated many downstream tasks including question answering and knowledge graph completion [1, 3]. Previous works mainly focus on the sentence-level RE and achieve outstanding results [11, 15, 17, 31] in discerning relations only mentioned in a single sentence. While in the real world, a large number of relational facts are expressed across multiple sentences in a document. To this end, scholars pay more attention to the general case (i.e., document-level RE).

To facilitate the research on document-level RE, [26] first proposes a large-scale dataset: DocRED. An example of DocRED is illustrated in Figure 1, where entities and their mentions are colored and dispersed in different sentences. We can infer the relation "*country*" between the entity "*Central Cemetery*" and "*Bulgaria*" by an evidence set $\{0, 3\}$ with a bridge entity "*Sofia*", while many rest sentences (e.g., 1, 2, 4...) are useless to this relational fact. In document-level RE scenarios, it is common that only a small subset of sentences from the original document is useful for reasoning the target relation. According to the statistics of DocRED, more than 95% instances require no more than three sentences as supporting evidence [5], but the average length of a document is above

超过95%的案例需要不超过三句话作为支持证据

[0] The Georgi Dimitrov Mausoleum was a ceremonial tomb on Prince Alexander of Battenberg Square in **Sofia** , **Bulgaria** , from 1949 to 1999 .
[1] The white marble mausoleum was built in 1949 to contain the embalmed body of **Bulgaria** 's first Communist leader , Georgi Dimitrov (1882 – 1949) , construction beginning immediately after the news of Dimitrov 's death .
[2] It was completed in just six days , the time it took Dimitrov 's body to be returned to **Sofia** from the USSR .
[3] Dimitrov 's body remained in the mausoleum until August 1990 , when Dimitrov 's remains were cremated and the ashes buried in the **Sofia** 's **Central Cemetery** .
[4] The mausoleum itself was destroyed by Prime Minister Ivan Kostov 's UDF government in 1999 after a heated nationwide debate .
[5] The prime minister and his party claimed that retaining the mausoleum was inappropriate following the fall of Communism in 1989 because it represented **Bulgaria** 's repressive past .
[6]~[10]…

Head entity: **Central Cemetery** **Tail entity:** **Bulgaria** **Relation**: country
Evidence set: [0, 3]

Figure 1: An example adapted from DocRED. Annotated entities are colored and a relation triplet is listed with its support evidence below.

eight sentences. Additionally, these evidence sentences are not always consecutive and may distribute anywhere in the document (e.g., {0, 3}). Thus a desired model should be able to learn long-term dependency and choose relevant context from the document.

Previous works in document-level RE focus on learning text structural information to capture the long dependency [22, 30, 32]. Generally, they abstract the document into a graph in terms of the entity or sentence nodes and specify various types of edges to characterize the interaction between these nodes. Based on the graph structure, global information is then propagated and aggregated by its neighbor nodes without the limitation of the original long sequence distance between sentences. Nevertheless, since all surrounding information will be incorporated into a node encoding, the graph-based methods are inevitably influenced by useless information brought by irrelevant nodes and edges. Although using attention mechanism [22, 29] (i.e., a soft way) might be able to alleviate this problem to some extent, it still cannot completely eliminate the side effect of the irrelevant information.

不能完全消除无关信息的副作用。

因此去除无关句子

To avoid that, we should decisively exclude irrelevant sentences by a hard decision. Evidence extraction task is then introduced by [26] and involved in a few works [4, 23]. However, these works just separately predict each sentence to be evidence or not according to the given entity pair, ignoring the strong semantic correlations with other evidence sentences. Actually, evidence sentences should be taken as a whole for expressing a specific relational fact. As shown in Figure 1, we can infer the entity "*Bulgaria*" is the country of the entity "*Central Cemetery*" only if synthesize all the following sentences :

这些工作只是根据给定的实体对分别预测每个句子是否是证据，

其忽略了与其他证据句子的强语义相关性

"The Georgi Dimitrov Mausoleum was a ceremonial tomb on Prince Alexander of Battenberg Square in Sofia, ***Bulgaria****, from 1949 to 1999." (S_0);*

"..., when Dimitrov's remains were cremated and the ashes buried in the Sofia's ***Central Cemetery****." (S_3);*

在这种情况下，单独判断一句话是否构成证据是很困难的，需要参考其他组成的证据句子。

In this case, it is difficult to judge whether a sentence constitutes the evidence alone without referring to other component evidence sentences. For example, we are unable to straightforwardly point out that S_0 can be the evidence for expressing *Country (Central Cemetery, Bulgaria)* without relying on S_3 and vice versa. Obviously, existing evidence extraction modules are too simple to fit the complexity of this problem.

In this paper, we propose a novel evidence-aware method for document-level RE, namely DRE-EA. The goal of our model is to explicitly extract the precise evidence for an entity pair. And then the extracted evidence sentences serve as a concise and clean document for judging the relation between the target entity pair. In this way, we are able to only focus on related context, solving the long dependency problem and providing an interpretable clue for each prediction. More specifically, to model the connection between sentences, we formulate evidence extraction as a sequential decision problem in which all the previously selected sentences will influence the current sentence selection. We craft reinforcement learning (RL) to implement the decision making process with Double Deep Q-learning Network (DDQN) [19]. Though RL is an effective strategy to explore evidence sentences, the search space will be extremely large if considering all sentences. According to the dataset DocRED, the average length of a document is around 8, meaning that there are $8! = 40,320$ permutations of sentence sequence for each entity pair. To reduce useless attempts among irrelevant sentences and accelerate RL, we develop a path searching strategy to narrow the search space of RL. Particularly, we organize all entities as nodes and turn the document into a graph with sentence co-occurrence edges. Sentences on the shortest path from the head entity to the tail entity are picked out as candidates. After that, evidence is examined only from these candidates by the agent of RL. In this manner, our model can get both high effectiveness and efficiency. Experiments on two document-level benchmarks: DocRED [26] and DWIE [27] show that our method outperforms previous works and yields competitive results compared to the state-of-the-art methods. We also conduct careful analysis to figure out what our evidence extraction model learns.

我们将证据提取定义为一个顺序决策问题，其中所有先前选择的句子都会影响当前句子的选择。

我们利用强化学习（RL）来实现双深度质量学习网络（DDQN）的决策过程

为了减少无关句子之间的无用尝试并加速RL，我们开发了一种路径搜索策略来缩小RL的搜索空间

从头部实体到尾部实体的最短路径上的句子作为候选。在此之后，RL的代理人只检查了这些候选者的证据。

We summarize our contributions as follows:

- For the first time, we investigate the strong semantic correlations between evidence sentences for expressing a relational fact, and formulate evidence extraction as a sequential decision problem.
- We propose a new evidence extraction framework based on reinforcement learning, which can allocate explicit actions to the candidate sentences in the document and constitute a clean evidence subset to facilitate document-level RE.
- We reconstruct the original document to a graph structure and design an efficient path searching strategy prior to evidence extraction, which significantly reduces the exploratory space and boosts both effectiveness and efficiency.
- We conduct extensive experiments on two widely used datasets and the experimental results demonstrate the superiority of our method compared to the strong baselines. Moreover, our model is proven to find accurate evidence sentences, promoting the interpretability of document-level RE.

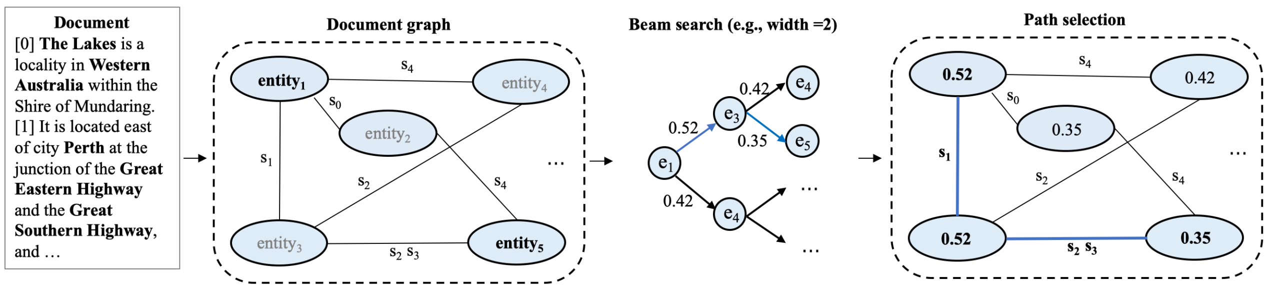

Figure 2: Entity graph with co-occurrence sentence on the edge. Value of node is measured by its centrality. We use beam search on the graph and obtain the only one optimal path, e.g., $entity_1$ to $entity_5$ is s_1, s_2, s_3.

2 METHODOLOGY

We propose a new document-level evidence extraction method through RL, in which the potential evidence for each entity pair is selected regardless of their relations. These evidence sentences act as a pure document for relation classifier then. Finally, fusing the results on both original and evidence documents, we implement a strong baseline for document-level RE.

2.1 Task formulation

We decompose our task into two sub-problems: document-level evidence extraction and relation extraction. Given a document of L sentences $D = \{s_i\}_{i=1}^{L}$, and an entity set $E = \{e_i\}_{i=1}^{n}$ where each entity of its mentions $M_e = \{m_i\}_{i=1}^{|e|}$ appears at least once in D, the document-level RE task aims at inferring a set of relations $R \cup \{NA\}$ between each entity pair (e_s, e_o). For the entity pair with non-NA relations, the evidence extraction task aims at selecting all sentences that can express some relational facts of it.

2.2 Path searching

As a document consists of L sentences with N entity pairs, there should be $L! \times N$ permutations of sentences decision sequence of each document which apparently poses huge pressure on evidence extraction module and degrades its effectiveness and efficiency. So, we design a coarse selection method with a path searching technique to heuristically exclude some irrelevant sentences.

We formulate the whole document as a graph by the following process. Each node of our graph represents a specific entity from the document. If respective mentions of two entities occur in the same sentence, we link the two entity nodes and record the co-occurrence sentence as edge information. Specifically, there might exist several sentences on an edge as mentions appear in various sentences, which corresponds to that some entity pairs co-occur in multiple sentences.

如果两个实体分别出现在同一句子中，我们将这两个实体节点连接起来，并将共同出现的句子记录作为边信息

Through the above steps, our graph can model the co-occurrence and coreference structure of a document. Thus, two entities not directly connected can be linked through multiple bridge nodes and the path between them also implies logical reasoning for RE. We assume that entities with more connections contain more information. And using these informative entities as an intermediate entity will include as much useful information as possible. To achieve this goal, we use the eigenvector centrality to measure the importance of nodes e_j or weights of the edge $e_i \rightarrow e_j$. The intuition behind this metric is that the importance of a node depends both on the number of its neighbors (degree) and on the importance of its neighbors. The eigenvector centrality λ is defined as $Mx = \lambda x$, where M denotes the adjacency matrix of the graph and x denotes the degree matrix of each node.

我们假设具有更多连接的实体包含更多的信息，使用这些信息实体作为中间实体将包含尽可能多的有用信息。

利用特征向量中心性来度量节点ej的重要性或边ei→ej的权值。

这个度量标准背后的直觉是，节点的重要性取决于它的邻居数量（度）以及邻居的重要性。

We believe that a good path should not contain too many entities (i.e., the shortest path) and should include more important entities (i.e., the largest weight path). Naturally, we adopt the beam search algorithm to obtain such a path between entity pairs. As shown in Figure 2, to get the optimal path $\{s_1, s_2, s_3\}$ between $entity_1$ and $entity_5$, the process of beam search with a width of two is described in the middle part. At last, sentences on the path are viewed as a candidate set for evidence extraction. In case there is no path between entity nodes, We also take all sentences that the entity's mentions appearing into consideration. In this way, we find up to 94.2% of the human-annotated supporting evidence in DocRED can be fully covered by our path searching strategy.

2.3 Precise evidence selection by RL

We cast evidence selection as an RL problem. The framework of our evidence extraction model is shown in Figure 3. In each episode, the model reads in an entity pair with a candidate sentence set C. The agent picks the sentence (action) within C at each time step and gets the rewards for the selected (state, action) pair. Finally, we optimize the network with the Double Deep Q-learning Network technique.

Input layer. We use a pretrained language model BERT [6] to encode the whole document. To specific the named entities and their mentions, we add two special tokens [∗] to the start and end of them. And we obtain the embedding $\mathbf{h}$ and multi-head attention $\mathbf{a}$ of each token in the document.

$$\mathbf{H}, \mathbf{A} = BERT[x_1, x_2, ..., x_{len}] \tag{1}$$

Following previous work [33], we apply logsumexp pooling to get the embedding of each entity e with mention set $M = \{m_j\}_{j=1}^{num}$ from the start special token to the end token. To get the context of the entity, attention matrix $\mathbf{H}$ from the last layer of BERT is used

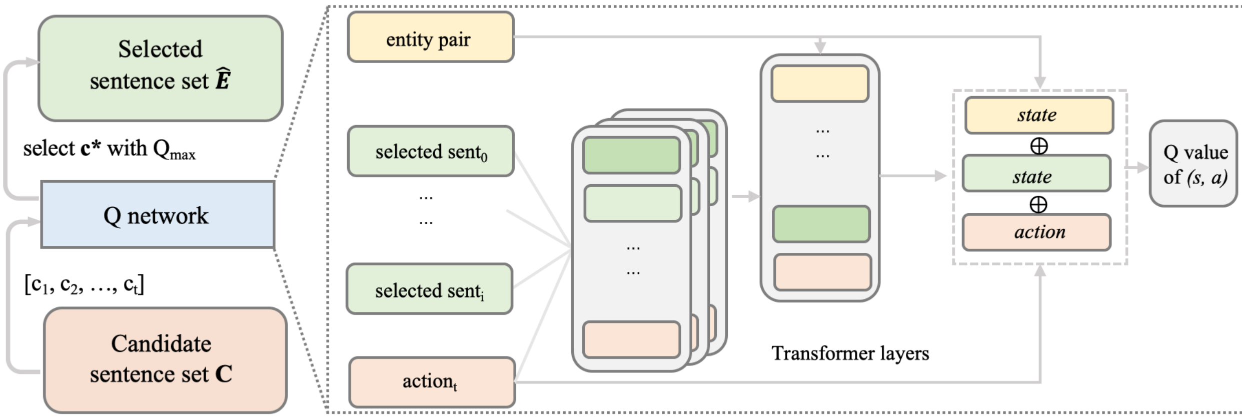

Figure 3: Framework of our evidence extraction RL model. The architecture of the Q-network is shown on the right which outputs the Q value for each (state, action) pair.

to compute a localized context embedding as shown in Equation (2-4).

$$\mathbf{h}_e = log \sum_{j=1}^{num} exp(\mathbf{h}_{m_j}) \tag{2}$$

$$\mathbf{a}^{(s,o)} = norm(\sum_{i=1}^{head} \mathbf{A}_i^{(s,o)}) \tag{3}$$

$$\mathbf{c}^{(s,o)} = \mathbf{H}^T \mathbf{a}^{(s,o)} \tag{4}$$

We also use the mean pooling of all tokens in a sentence to represent the sentence embedding $\mathbf{s}_i$ of the i_{th} sentence in the document.

$$\mathbf{s}_i = mean(\sum_{j\in|s|} \mathbf{h}_j) \tag{5}$$

我们的代理是一个句子选择器，它可以选择所有可以被视为目标实体对的证据的句子，而不管它们的潜在关系如何。

Our agent is a sentence selector to pick all sentences that can be viewed as evidence for a target entity pair regardless of their potential relations. Thus, we choose the optimal sentence from the candidate set that will feedback on more returns at each step. We introduce the action, state, and reward as follows.

Action. The action a at each step is picking one sentence from the candidate set. And no actions will be taken when getting to the terminal state. During an episode, we maintain a set of selected sentences $\hat{E}$, and a set of candidate sentences C extracted by our path searching strategy. Then we view choosing a sentence from set $C\backslash\hat{E}$ as action a_t at time t. When the total number of selected sentences reaches K, we abort the selection and reach the terminal state. We set K to three since three sentences are enough for evidence extraction in 95% cases as stated in Sec. 1. It should be noted that we pick the decision sequence with the largest Q value as the final predicted evidence, as described in Algorithm 2, which means the length of our extracted evidence is not fixed.

action at

证据句个数最多为3

State. As we assume before, the sentence chosen to be evidence will affect our next decision. To achieve this goal, the state s includes the information from previous sentences that we picked. We denote

选择作为证据的判决将影响我们的下一个决定

the selected sentence set at step t as $\hat{E}_{(t)}$, and $\hat{E}_{(t+1)} = \hat{E}_{(t)} \bigcup a_t$. Also, sentences are aggregated by N transformer layers [20] to learn the interaction between each other as defined in Equation 6.

$$H' = [h'_0, h'_1, ..., h'_{t-1}] = Transformer([\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_{t-1}]) \tag{6}$$

where $\mathbf{c}$ denotes the embedding for each sentence.

Moreover, we also add the target entity pairs to each state since a sentence performs different roles in different entities. We concatenate the embedding of the head, tail entity, and their context attention as entity pair embedding in the end.

$$\mathbf{h}_{(s,o)} = [\mathbf{h}_s \oplus \mathbf{h}_o \oplus \mathbf{c}^{(s,o)}] \tag{7}$$

In addition, we use them as a query vector to get interaction with the selected evidence.

$$\mathbf{h}_{evi} = Transformer(F(\mathbf{h}_{(s,o)}), H') \tag{8}$$

where F is a linear layer reducing dimensionality to d_0, $h \in \mathbb{R}^{d_0}$ and $H \in \mathbb{R}^{t \times d_0}$.

Finally, features of our network include the state (information of target entity pairs) and action (information of currently chosen sentence).

$$\mathbf{F}_t = [\mathbf{h}_{(s,o)} \oplus \mathbf{h}_{evi} \oplus \mathbf{h}_{a_t}] \tag{9}$$

Reward. We give an immediate reward to the action taken in step t according to the golden truth of evidence set E of each entity pair. The reward r is defined as

$$r_t = \begin{cases} 1, & a_t \in E \vee (\forall c \in \hat{E}_t : c \in E \;\; if \;\; terminal) \\ 0, & otherwise \end{cases} \tag{10}$$

where a_t denotes the action taken at step t, $\hat{E}_t$ denotes the sentences set chosen by our agent. If the action sentence belongs to the evidence set E, we give it positive feedback. When reaching the terminal of an episode, we also reward it if all true evidence sentences have been chosen.

In the end, the value for the current state and action is computed by a multi-layer perceptron as illustrated in Figure 3.

$$Q(s_t, a_t; \theta) = \sigma(\mathbf{W} * \mathbf{F}_t + \mathbf{b}) \tag{11}$$

where $\mathbf{F}_t$ is the feature vector defined in Eq.(9).

Optimization. We utilize Deep Q Networks (DQN) to learn a good policy for our agent. In this method, our primary objective is to get the Q values that neural network outputs for every pair of state and action:

$$Q(s, a) = \mathbb{E}_{s'}[r_t + \gamma \max_{a' \in A} Q(s', a')] \tag{12}$$

where s', a' denote next state and action and $\gamma \in [0, 1]$ is a discount factor that trades off the importance of immediate and later rewards.

By interacting with the environment, we can observe the tuple s, a, r, s' (state, action, reward, and the new state). These transitions are then stored and sampled uniformly from the memory bank to update the Q network. According to the Double DQN algorithm [19], we compute the temporal difference (TD) target y_j and train the model through the temporal difference error δ_j:

$$y_j = \begin{cases} r_j, & if \ s_{j+1} \in terminal \\ r_j + \gamma \max_{a \in A} Q(s_{j+1}, a; \theta') & otherwise \end{cases} \tag{13}$$

$$\delta_j = Q(s_j, a_j; \theta) - y_j \tag{14}$$

where θ and θ' denote two sets of weights and θ' can be updated symmetrically by copying θ.

Huber loss is used to minimize the TD error:

$$L_{Evi} = \begin{cases} \frac{1}{N} \sum_i^N 0.5\delta_i^2, & |\delta_i| \le 1 \\ \frac{1}{N} \sum_i^N (|\delta_i| - 0.5), & otherwise \end{cases} \tag{15}$$

where N is the batch size of transitions.

We show how to train the model in Algorithm 1. As stated in Section 2.2, sentences on the path between entity pairs are viewed as candidates and our goal is to extract evidence from the candidates. During training, we initialize the start state with an empty selected evidence set. For each step, the DQN calculates the Q value of all candidates in the rest sentence set that are not picked yet. The reward of actions in each state is computed based on Eq.(10).

训练：
对于每一步，DQN都会计算出其余句子集中尚未被选中的所有候选句的Q值。每个状态下的行为奖励都是基于

During inferring, we execute the action with the max Q value at each step and store the value throughout the whole trajectory. When all the sentences are selected, the agent reaches a termination state. The candidate set of the state with the highest value in the trajectory is used as the final prediction set, as described in Algorithm 2.

推理：
我们在每一步使用最大Q值执行动作，并在整个轨迹中存储该值。当选择所有的句子时，代理达到终止状态。将轨迹中值最高的候选状态集作为最终的预测集，如算法2所述。

2.4 Relation classifier

We use the same embedding methods to represent entities as described in Section 2.3. Briefly, head entity h_s and tail entity h_o are represented by their mentions embedding and context attention embedding:

$$\mathbf{h}_s = \sigma(\mathbf{W}_s[\mathbf{h}_s; \mathbf{c}_{(s,o)}] + \mathbf{b}_s) \tag{16}$$

$$\mathbf{h}_o = \sigma(\mathbf{W}_o[\mathbf{h}_o; \mathbf{c}_{(s,o)}] + \mathbf{b}_o) \tag{17}$$

where σ denotes the activate function *tanh*.

We consider two classifiers to obtain the final feature for RE: the bilinear layer (Eq.(18)) and the biaffine attention layer (Eq.(19)).

$$\mathbf{y}_{bilinear} = \mathbf{h}_s^T \mathbf{W} \mathbf{h}_o + \mathbf{b} \tag{18}$$

Algorithm 1 Training Procedure for evidence extraction with experience reply and ϵ-greedy exploration, where the max iteration number N, the updated interval n and the max number of selected sentences K are hyper-parameters.

Initialize an experience memory with a capacity of M
Initialize the Q-network $Q(s, a; \theta)$ randomly
Initialize the target network $\hat{Q}(s, a; \hat{\theta})$ with $\hat{\theta} = \theta$
for $epoch = 1 \rightarrow N$ **do**
  Get candidate sentence set C for target entity pairs $[e_1, e_2]$ by the strategy in Sec 3.2
  for $each([e_1, e_2], \hat{E}, C)$ **do**
    Initialize the start state $s_0 = (h_{(e_1,e_2)}, \hat{E}_0)$ with Eq.(6)-(8) where selected evidence set $\hat{E}_0 = \varnothing$
    for $t = 0 \rightarrow K$ **do**
      if $random() < \epsilon$ **then**
        Select a random action a_t where $a_t \in \{C \backslash \hat{E}_t\}$
      else
        $a_t = \underset{a \in C \backslash \hat{E}_t}{argmax}\, Q(s_t, a; \theta)$
      end if
      $s_{(t+1)} = (h_{(s,o)}, E_{(t+1)})$, where $E_{(t+1)} = E_t \cup \{a_t\}$
      Execute action a_t and compute reward r_t based on Eq.(10)
      Store transition $(s_t, a_t, r_t, s_{(t+1)})$ in M
    end for
    Sample a mini batch of transitions $(s_j, a_j, r_j, s_{(j+1)})$ from M and update $Q(s, a; \theta)$ based on Eq.(12)-(15)
    Update the target network $\hat{Q}(s, a; \hat{\theta})$ after n step by $\hat{\theta} \leftarrow \tau \cdot \theta + (1 - \tau) \cdot \hat{\theta}$
  end for
end for
return $Q(s, a; \theta)$

$$\mathbf{y}_{biaffine} = \mathbf{h}_s^T \mathbf{W}_1 \mathbf{h}_o + \mathbf{W}_2[\mathbf{h}_o; \mathbf{h}_s] + \mathbf{b} \tag{19}$$

To reduce the number of parameters in the bilinear classifier, we also use the group bilinear technique [33]. Each embedding is split into k equal-sized groups. Then bilinear is computed within these groups:

$$x = [x^1; ...; x^k] \tag{20}$$

$$x_s^T \mathbf{W} x_o = \sum_{i=1}^{k} x_s^{i\,T} \mathbf{W}_r^i x_o^i \tag{21}$$

where $\mathbf{W}_r \in \mathbb{R}^{d_0/k \times d_0/k}$.

We use the adaptive thresholding loss [33] to tackle the multi-relation problem. Specifically, Relations are divided into three classes: positive class P, negative class N and TH class. The goal of adaptive thresholding loss is to build a gap between positive and negative classes through making the probabilities of TH lower than that of P, and higher than that of N, as defined below:

$$L = -\sum_{r \in P} log(\frac{exp(prob_r)}{\sum_{r' \in P \cup TH} exp(prob_{r'})}) - log(\frac{exp(prob_{TH})}{\sum_{r' \in N \cup TH} exp(prob_{r'})}) \tag{22}$$

Algorithm 2 Inferring Procedure for evidence extraction where C_i is the candidate sentence set for i_{th} entity pair.

```
Initialize an empty list Ê to store evidence results
for each ([e_1, e_2], Ê_i, C_i) do
    Initialize the start state s_0 = (h_(s,o), E_0) where E_0 = ∅
    Initialize an empty list values to record Q-values at time t
    Initialize an empty list track to record chosen sentence set
    at time t
    for t = 0 → len(C_i) do
        a_t = argmax_{a∈C\E_t} Q(s_t, a; θ)
        q_t = Q(s_t, a_t)
        s_(t+1) = (h_(s,o), E_(t+1)), where E_(t+1) = E_t ∪ {a_t}
        add q_t to values
        add E_t to track
    end for
    ĵ = argmax(values)
    Store track[ĵ] to Ê_i
end for
return Ê
```

2.5 Final prediction

We believe that relations can be inferred with the support evidence only, and other sentences can be regarded as noisy information. Thus, only a purer document with the evidence could help RE model to make a better decision. Based on the evidence extraction results from RL model, we combine these sentences in their original order as pseudo document D'. As our evidence extraction results are not 100% correct we train the model with the original document D and infer the relation both on D and D'. Similar to [23], we also find the RE results from D and D' can be complemented with each other. And fusing these two results will lead to better performance. The fusion strategy is defined as

$$P(s,o,r) = P(s,o,r;D) + P(s,o,r;D') - \lambda \tag{23}$$

where λ is a hyperparameter tuned on the dev dataset with respect to F1 score, $P(h,t)$ denotes the relative probability of relation class r to class TH. We keep the prediction result when $P(s,o,r) \geq \lambda$.

Now, we have the final relation result for the entity pair and their support evidence. However, different relationships are often expressed by different sentences. We use a simple binary classification model to further classify the evidence that the predicted relation triples hold. The classifier is defined in Eq.(23), and optimized by the binary cross entropy loss function.

$$\mathbf{y}(s,o,r,e) = \mathbf{h}_{(s,o,r)}^{T} \mathbf{W} \mathbf{h}_{sent} + \mathbf{b} \tag{24}$$

where $\mathbf{h}_{(s,o,r)} = [\mathbf{h}_s; \mathbf{h}_o; \mathbf{c}_{(s,o)}; \mathbf{r}]$, r denotes the embedding of the relation type and h_{sent} denotes the sentence embedding.

3 RELATED WORK

Traditionally, RE aims at identifying the relation between two entities from a predefined relation set. Many efforts have been devoted to the document-level RE under the supervised learning paradigm. Meanwhile, reinforcement learning as a new paradigm of machine learning is widely used recently in Nature Language Processing. We describe these two paradigms separately as follows.

Document-level relation extraction [26] firstly proposes a large-scale document-level RE dataset in general domain. The dataset is constructed from Wikipedia and Wikidata with both distant supervision and human-annotated. If entity pair has relation in the predefined relation set, they also label the sentence set, named evidence, that can support the relation. In document-level RE, relations are inferred by multiple contexts for most time. To learn the connection between entity pairs, many works [22, 25, 29] formulate the document as a graph and aggregate information of nodes from neighbors. Others also model the structural information by pre-trained language model [24] or U-net [30]. In our work, we use the document graph structure in a different way. We don't learn about the graph structure but use the graph to narrow our choices of important contexts. Some researchers focus on reasoning or evidence extraction. [16] proposes a probabilistic model to learn latent logic reasoning rules between relations. [4, 23] design a multi-task learning framework for evidence extraction (EE). Given a sentence corresponding to an entity pair and its relation, they simply decide whether it is evidence or not, which isolates each sentence. However, we believe relational facts can be reasoned with all evidence together. So, our work takes the previous selection of evidence sentences into consideration and formulates the evidence extraction task under the reinforcement learning paradigm. In this work, we separate these two tasks (i.e., RE and EE) with a pipeline framework. What's more, our agent chooses evidence of entity pairs without knowing their relations. Our method attempts to find a sentence decision sequence that maximizes the cumulative reward thus modeling the connection between sentences.

Reinforcement Learning in NLP With the boost of deep learning, deep reinforcement learning (DRL) has risen rapidly [8, 18, 19]. DRL methods are suitable for decision making problems and have gained significant attention in NLP recent years, such as information extraction [10] and dialog [7]. As RL learns a latent policy for better decisions, [2, 12–14] apply the policy-based or value-based method to combat the noisy data in distant supervision RE. They use a deep reinforcement learning strategy to select false positives or choose high-quality instances for each relation type without supervised information. [21] proposes a DQN-based approach to retrieve supporting or refuting evidence for fact verification. In our work, we also use a value-based method to simulate the process of selecting sentences that may express some relations between two entities. Different from previous works, we do not use the metrics of the target task to design our rewards. The agent can receive immediate rewards during training and output the best evidence decision sequence during inferring.

4 EXPERIMENTS

We conduct experiments on two document-level relation extraction datasets: DocRED [26] and DWIE [27]. For DWIE, we utilize the same data splits as [16] and additionally divide the documents into sentences bounded by periods.

Implementation Details. For evidence extraction model, we pre-train the bert-base-cased encoder under supervised setting until convergence. Then we fix its parameters and only use this encoder to get the word embeddings for training the RL model. We apply the Double DQN model implemented by [21] with priority experience

Table 1: Main results(%) on the development set and the test set of DocRED. We conduct 5 runs with different random seeds and report the mean and standard deviation on the development set and choose the best results on test set. Best performance are emphasized in the table.

Model	Dev			Test		
	Ign F_1	F_1	*Evi* F_1	*Ign* F_1	F_1	*Evi* F_1
BiLSTM [26]	48.87	50.94	44.07	48.78	51.06	43.83
E2GRE-$BERT_{base}$ [4]	55.22	58.70	47.14	-	-	48.35
GLRE-$BERT_{base}$ [22]	-	-	-	55.40	57.40	-
LSR-$BERT_{base}$ [9]	52.43	59.00	-	56.97	59.05	-
GAIN-$BERT_{base}$ [29]	59.14	61.22	-	59.00	61.24	-
ATLOP-$BERT_{base}$ [33]	59.22±0.15	61.09±0.16	-	59.31	61.30	-
DRN-$BERT_{base}$ [25]	59.33	61.39	-	59.15	61.37	-
SIRE-$BERT_{base}$ [28]	59.82	61.60	-	60.18	62.05	-
EIDER-$BERT_{base}$ [23]	60.51±0.11	62.48±0.13	50.71	60.42	62.47	51.27
DRE-EA-$BERT_{base}$(Ours)	**60.76±0.17**	**62.68±0.12**	**52.49±0.15**	**60.50**	**62.58**	**52.44**

replay. We set the learning rate as $5e-5$, batch size as 512 and training epoches as 100. The capacity of replay buffer is 10,000. The discount factor γ is 0.95. The probability of ϵ-greedy policy starts at 0.9 and decays exponentially towards 0.05 with the rate of 0.05%.

For the RE model, we optimize it with AdamW. We use the feature layer (bilinear or biaffine in Sec. 2.4) with the better performance on the development set for prediction. All models are trained with 1 GTX 2080 Ti GPU by using learning rates 5e-5 with a linear warmup for the first 6% of steps. The batch size is 4 for DocRED and 1 for DWIE. The max epoch is 30 during training.

Baselines. We choose several competitive methods as our baselines for document-level RE including transformer-based and graph-based methods. Most of these methods are designed for reasoning. It is worth noting that only some of these baselines have the ability of evidence extraction (i.e., **BiLSTM**, **E2GRE** and **EIDER**). As the dataset DWIE doesn't provide annotated evidence set, we don't compare with the joint-training baseline (EIDER) on it.

Table 2: Positive evidence extraction results on DocRED. Results with † are obtained by their public code.

Method		Precision	Recall	Pos evi F_1
EIDER† [23]		77.55	78.77	78.15
DRE-EA	path searching	52.56	**94.20**	67.47
	RL on path	**90.88**	77.56	**83.69**

4.1 Results on DocRED

We compare our model, namely **DRE-EA**, with the baseline methods on both relation and evidence classification on DocRED. We report the *Ign* F_1, F_1 and *Evi* F_1 score following [26]. In particular, the *Ign* F_1 score is measured excluding those relational facts shared by the training and dev/test sets. According to [26], the *Evi* F_1 score is calculated on the entity pair with relations. As demonstrated in Table 1, our method obtains a substantial gain over the basic extractors, especially outperforms the best model by 1.78% and 1.17% in terms of *Evi* F_1 on the dev and test set. Compared to the previous best approaches that ignore the evidence extraction task, methods that combine both evidence and relation extraction achieve better performance (e.g., EIDER). This demonstrates the necessity and effectiveness of introducing evidence to document-level RE. Existing methods train these two tasks jointly, while we treat them as pipeline. In our method, all potential evidence is extracted firstly ignoring the relations which provide purer information for relation classification. During our experiments, the *Evi* F_1 metrics in this step have significantly outperformed the results of the best benchmark. With further fine-grained extraction based on relational information (Sec. 2.5), we obtained the best results. For the *Ign* F_1, we observe a 0.21% improvement on the dev set but a slight 0.08% improvement on the test set. As the test set is unavailable, the result can only be obtained by submitting to the Codalab[1]. We believe that the reason for this phenomenon is the inconsistency of the data distribution of the development set and the test set.

In fact, people care more about evidence extraction on positive relations, not NA relations. Hence, we also report a new Positive Evidence F_1 metrics on the entity with not NA relations, denoted as Pos evi F_1. As shown in Table 2, our path searching strategy can cover around 94% of the evidence and achieves the highest recall. Applying the reinforcement learning to select on these candidate sentences further achieves the highest accuracy of 90.88% while sacrificing partial recall. Overall, our methods achieve the highest Pos evi F_1 score compared to the strong baseline which shows the importance of considering the connection between sentences for evidence extraction. Moreover, we find that modeling the reasoning type explicitly (e.g., GAIN, DRN and SIRE) is more effective than modeling implicitly (e.g., LSR). Our path searching strategy also acts as an explicit signal that only contains the logic, co-occurance reasoning and excludes irrelevant information.

4.2 Results on DWIE

The DWIE dataset has a longer length of document which is about three times larger than DocRED and also meets the multi-label and multi-entity problems. Since it doesn't have the golden truth

[1] https://competitions.codalab.org/competitions/20717

Table 3: Main results(%) on the development set and the test set of DWIE.

Model	Dev		Test	
	Ign F_1	F_1	*Ign* F_1	F_1
CNN [26]	37.65	47.73	34.65	46.14
LSTM [26]	40.86	51.77	40.81	52.60
BiLSTM [26]	40.46	51.92	42.03	54.47
Context-Aware [26]	42.06	53.05	45.37	56.58
GAIN [29]	58.63	62.55	62.37	67.57
ATLOP [33]	59.03	64.82	62.09	69.94
LogiRE [16]	60.24	66.76	64.11	71.78
DRE-EA(Ours)	**63.15**	**70.76**	**67.50**	**75.63**

for evidence, we only use the result of path searching strategy as evidence and directly evaluate the RE model.

Table 3 shows the main results on the development set and the test set of DWIE. We can observe that pretrained language model has a good ability to capture long distance dependency than traditional neural networks. The baseline method LogiRE builds implicit interactions among relations through the meta path determined by their rule generator. While our method explicitly provides the related context by the path searching strategy. Our method improves the ign-F_1 by 3.39% and F_1 by 3.85% on the test set of DWIE which shows that evidence plays a key role in RE.

4.3 Ablation study

Table 4: Ablation study on the evidence extraction model.

Model	Precision	Recall	F_1
DRE-EA	90.88	77.56	83.69
2-layer	90.26	77.55	83.42
1-layer	90.22	77.45	83.34
w/o path	85.88	64.04	73.36

Table 5: Ablation study on the relation extraction.

Model	DWIE		DocRED	
	Ign F_1	F_1	*Ign* F_1	F_1
DRE-EA	63.15	70.76	60.72	62.68
- pseudo document	59.73	65.30	59.47	61.30
- original document	56.61	67.11	58.55	60.84
- original document - RL	/	/	58.43	60.73

Table 6: Training Time on DocRED.

DRE-EA	w/ path	w/o path
Training time	7.74 s/it	19.32 s/it

Table 7: Detail of relation types on dev set of DocRED. The relationship name is represented by its index.

Type	Relation set
Time-related	P570, P571, P580, P582, P585, P607
High frequency	instances $\geq$ 5%
Low frequency	instances $\leq$ 0.5%
Knowledge	P39, P150, P205, P400, P441, P1056

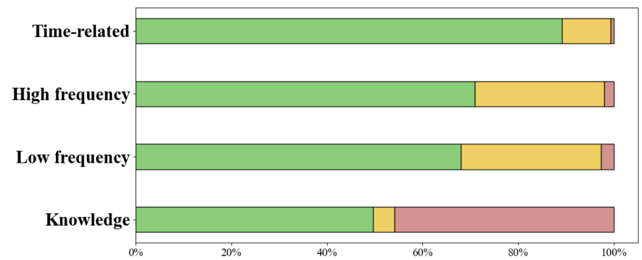

Figure 4: Error analyses on evidence extraction. Green represents our prediction contains all golden-truth evidence, yellow represents partial contain and red represents no contain at all.

We ablate each part of our evidence extraction RL model on the development set of DocRED, as shown in Table 4. Firstly, three transformer layers are enough to learn the representation of state. Secondly, we train the reinforcement learning model on all sentences without the path searching process (w/o path) and observe a sharp decrease in the performance. It is because the evidence sentences only account for a small part of the document which will cost the agent more time to explore or even be confused.

We also ablate the evidence to the relation extraction results on the development set of the two datasets. The pseudo document is consisted of the evidence extracted by the model. We can observe that only with original or pseudo document leads to a drop in the F_1 score. As our evidence extraction results cannot reach 100% correct, the results on the pseudo-document can complement the results on the original document. Finally, the last setting in Table 5 represents using the binary evidence extraction results as the pseudo document. A slight drop is observed compared to our reinforcement learning methods which model the connections between evidence. It is according to the property of the dataset DocRED that about 95% of evidence is shorter than three sentences.

We compare the training time on original document with that on the candidate extracted by our path searching strategy in Table 6, where s/it denotes second per iterator. This shows our path searching strategy can save 40% on training time which further verifies its effectiveness in reducing the search space.

In summary, our method significantly improves the performance of evidence extraction and further enhances relation extraction.

4.4 Case study and error analyses

We first discuss if our evidence extraction model learns a potential relation type without considering it. We gather the extracted

Table 8: Case study of evidence extraction on DocRED. We use different color to represent target entities and their mentions respectively. The index of a sentence is according to its position in the original document.

[0] TY.O is the third studio album by British record producer and recording artist Taio Cruz. **[1]** Taking a more electronic music sound than his previous releases, TY.O was released in December 2011 by Universal Island Records but for reasons unknown to Cruz, its British and American release were held off. **[2]** Instead, a year after its original release, ... **[3]** TY.O features a range of top-twenty and top-thirty singles including " Hangover "(featuring Flo Rida), "Troublemaker", "There She Goes" (sometimes featuring Pitbull), the limited release "World in Our Hands" and "Fast Car" which features on the Special Edition and Fast Hits versions of the album. **[4]** ...
Entity pair: (Troublemaker, December 2011) **Evidence**: [1, 3] **EIDER**: [3] **PATH**: [0, 1, 3] **DRE-EA**: [1, 3]
Entity pair: (Hangover, Universal Island Records) **Evidence**: [1, 3] **EIDER**: [0, 1, 3] **PATH**: [0, 1, 3] **DRE-EA**: [1, 3]
[0] This is a list of Presidents of Ethiopia and also a list of heads of state after the fall of the Ethiopian Empire in 1974. **[1]** Until 1974, the heads of state of the Ethiopian Empire were either Emperors or regents. **[2]** From the coup d'état of the Derg leading to the fall of the Empire in September 1974 until March 1975, the Derg considered the Crown Prince Asfaw Wossen as the nominal head of state which the Crown Prince refused to accept. **[3]** During this time, the Chairmen of the Derg, the leaders of the Derg, were to be considered as acting heads of state. **[4]** After 21 March 1975, the Derg military junta fully took over. **[5]** Until the establishment of the People's Democratic Republic of Ethiopia in 1987, still dominated by Derg figures, Chairmen of the Derg have to be considered heads of state–but not presidents. **[6]** After the fall of the Derg and the establishment of the Transitional Government of Ethiopia in 1991, the first immediate President (Meles Zenawi) has to be considered...
Entity pair: (Ethiopian Empire, Meles Zenawi) **Evidence**: [0, 6] **EIDER**: [0, 2, 5, 6, 9] **PATH**: [0,1,2,3,4,5,6] **DRE-EA**: [0, 6]

evidence with golden relation labels and list some typical relation types in Figure 4. For example, *"Time related"* concludes relation types such as "start time" and "date of birth". More details are shown in Table 7. On the x-axis, green represents that all the evidence can be identified, yellow represents that partial evidence can be identified, and red represents the extracted evidence is completely wrong.

From Figure 4, we can observe that our model is good at finding temporal relationships with a 89.13% total correct rate and 0.65% error rate. We believe information of such relation type is usually very clear thus making the decision easier. Moreover, relations of high or low frequency do not show much difference which shows that our model is not sensitive to the size of samples as it has no guidance of relations. Finally, when some relations require background knowledge (e.g., religion), the model appears polarized, either completely correct or completely wrong, and we believe the reason behind is that these relations require common sense or additional knowledge to assist the identification.

We also present two cases to understand the effect of our evidence extraction model as shown in Table 8. The key point in the first case is to figure out the bridge entity "TY.O" in S_1 and S_3. We can observe that EIDER presents either redundant or missing evidence results. In the second case that the mentions of target entities are scattered in different sentences, EIDER selects multiple sentences including many irrelevant sentences (S_2, S_5, S_9) as it can not identify the evidence from such confusing information. While our model cares about the connection between sentences and picks the precise evidence (S_0, S_6).

5 CONCLUSION

In this paper, we propose an evidence-aware pipeline method for evidence and relation extraction on the document-level. We formulate evidence extraction as a sequential decision problem by modeling the connection between sentences. Our method applies the Double DQN framework to select the possible evidence for each entity pair. We also propose a novel path searching strategy on the document graph to reduce the search space. The extracted evidence constitutes a pseudo document that provides supplementary information for relation extraction. Experiment results on two public datasets (DocRED and DWIE) demonstrate the validity of the method. In the future, we will explore using external knowledge to pick related contexts in document-level tasks.

6 ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 62102276; No. 62072323), the Natural Science Foundation of Jiangsu Province (Grant No. BK20210705, Grant No. BK20211307), the Natural Science Foundation of Educational Commission of Jiangsu Province, China (Grant No. 21KJD520005), Shanghai Municipal Science and Technology Major Project (No. 2021SHZDZX0103), the National Key Research and Development Project (No. 2020AAA0109302), Shanghai Science and Technology Innovation Action Plan (No. 19511120400), NH33714722 Youth Team on Interdisciplinary Research Soochoow University - Research on Subjectivity and Reasoning Theory in Artificial Intelligence, the Major Program of the Natural Science Foundation of Jiangsu Higher Education Institutions of China (Grant No. 19KJA610002), the major project of natural science research in universities of Jiangsu province (Grant No. 20KJA52000) and supported by Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

REFERENCES

[1] Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jijnasa Nayak, and Lun-Wei Ku. 2019. UHop: An Unrestricted-Hop Relation Extraction Framework for Knowledge-Based Question Answering. In *Proceedings of NAACL-HLT*. 345–356.

[2] Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[3] Xu Han, Zhiyuan Liu, and Maosong Sun. 2018. Neural Knowledge Acquisition via Mutual Attention Between Knowledge Graph and Text. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[4] Kevin Huang, Guangtao Wang, Tengyu Ma, and Jing Huang. 2020. Entity and evidence guided relation extraction for docred. *arXiv preprint arXiv:2008.12283* (2020).

[5] Quzhe Huang, Shengqi Zhu, Yansong Feng, Yuan Ye, Yuxuan Lai, and Dongyan Zhao. 2021. Three Sentences Are All You Need: Local Path Enhanced Document Relation Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. 998–1004.

[6] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*. 4171–4186.

[7] Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep Reinforcement Learning for Dialogue Generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 1192–1202.

[8] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

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

[10] Karthik Narasimhan, Adam Yala, and Regina Barzilay. 2016. Improving Information Extraction by Acquiring External Evidence with Reinforcement Learning. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2355–2365.

[11] Hao Peng, Tianyu Gao, Xu Han, Yankai Lin, Peng Li, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2020. Learning from Context or Names? An Empirical Study on Neural Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 3661–3672.

[12] Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust Distant Supervision Relation Extraction via Deep Reinforcement Learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2137–2147.

[13] Jianfeng Qu, Wen Hua, Dantong Ouyang, and Xiaofang Zhou. 2021. A Noise-aware Method with Type Constraint Pattern for Neural Relation Extraction. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[14] Jianfeng Qu, Wen Hua, Dantong Ouyang, Xiaofang Zhou, and Ximing Li. 2019. A fine-grained and noise-aware method for neural relation extraction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 659–668.

[15] Jianfeng Qu, Dantong Ouyang, Wen Hua, Yuxin Ye, and Xiaofang Zhou. 2019. Discovering correlations between sparse features in distant supervision for relation extraction. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 726–734.

[16] Dongyu Ru, Changzhi Sun, Jiangtao Feng, Lin Qiu, Hao Zhou, Weinan Zhang, Yong Yu, and Lei Li. 2021. Learning Logic Rules for Document-Level Relation Extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 1239–1250.

[17] Livio Baldini Soares, Nicholas Fitzgerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2895–2905.

[18] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[19] Hado Van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 30.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[21] Hai Wan, Haicheng Chen, Jianfeng Du, Weilin Luo, and Rongzhen Ye. 2021. A DQN-based Approach to Finding Precise Evidences for Fact Verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 1030–1039.

[22] Difeng Wang, Wei Hu, Ermei Cao, and Weijian Sun. 2020. Global-to-Local Neural Networks for Document-Level Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 3711–3721.

[23] Yiqing Xie, Jiaming Shen, Sha Li, Yuning Mao, and Jiawei Han. 2022. EIDER: Empowering Document-level Relation Extraction with Efficient Evidence Extraction and Inference-stage Fusion. In *Findings of the 60th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

[24] Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021. Entity Structure Within and Throughout: Modeling Mention Dependencies for Document-Level Relation Extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 14149–14157.

[25] Wang Xu, Kehai Chen, and Tiejun Zhao. 2021. Document-level relation extraction with reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[26] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A Large-Scale Document-Level Relation Extraction Dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 764–777.

[27] Klim Zaporojets, Johannes Deleu, Chris Develder, and Thomas Demeester. 2021. DWIE: An entity-centric dataset for multi-task document-level information extraction. *Information Processing Management* 58, 4 (2021), 102563. https://doi.org/10.1016/j.ipm.2021.102563

[28] Shuang Zeng, Yuting Wu, and Baobao Chang. 2021. SIRE: Separate Intra-and Inter-sentential Reasoning for Document-level Relation Extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 524–534.

[29] 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 (EMNLP)*. 1630–1640.

[30] Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021. Document-level Relation Extraction as Semantic Segmentation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, Zhi-Hua Zhou (Ed.). International Joint Conferences on Artificial Intelligence Organization, 3999–4006. Main Track.

[31] Zexuan Zhong and Danqi Chen. 2021. A Frustratingly Easy Approach for Entity and Relation Extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 50–61.

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

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