Entity and Evidence Guided Document-Level Relation Extraction

Anonymous ACL-IJCNLP submission

Abstract

Document-level relation extraction is a challenging task, requiring reasoning over multiple sentences to predict a set of relations in a document. In this paper, we propose a novel framework *E2GRE* (Entity and Evidence Guided Relation Extraction) that jointly extracts relations and the underlying evidence sentences by using large pretrained language model (LM) as input encoder. First, we propose to guide the pretrained LM's attention mechanism to focus on relevant context by using attention probabilities as additional features for evidence prediction. Furthermore, instead of feeding the whole document into pretrained LMs to obtain entity representation, we concatenate document text with head entities to help LMs concentrate on parts of the document that are more related to the head entity. Our *E2GRE* jointly learns relation extraction and evidence prediction effectively, showing large gains on both these tasks, which we find are highly correlated. Our experimental result on DocRED, a large-scale document-level relation extraction dataset, is competitive with the top of the public leaderboard for relation extraction, and is top ranked on evidence prediction, which shows that our *E2GRE* is both effective and synergistic on relation extraction and evidence prediction.

1 Introduction

Relation Extraction (RE), the problem of predicting relations between pairs of entities from text, has received increasing research attention in recent years [Zhang *et al.*, 2017; Zhao *et al.*, 2019; Guo *et al.*, 2019]. This problem has important downstream applications to numerous tasks, such as automatic knowledge acquisition from web documents for knowledge graph construction [Trisedya *et al.*, 2019], question answering [Yu *et al.*, 2017] and dialogue systems [Young *et al.*, 2018]. While most previous work focus on relation extraction at the sentence level, in real world applications, e.g predicting relations from web articles, the majority of relations are expressed across multiple sentences. Figure 1 shows an example from the recently released DocRED dataset [Yao *et al.*, 2019], which requires reasoning over three evidence sentences to predict the relational fact that "Link" is present in the work "The Legend of Zelda". In this paper, we focus on the more challenging task of *document-level* relation extraction task and design a method to facilitate *document-level* reasoning.

Document: [0] **The Legend of Zelda** : The Minish Cap ( ) is an action - adventure game and the twelfth entry in **The Legend of Zelda** series. [1] Developed by Capcom and Flagship , with Nintendo overseeing the development process , it was released for the Game Boy Advance handheld game console in Japan and Europe in 2004 and in North America and Australia the following year . [2] In June 2014 , it was made available on the Wii U Virtual Console . [3] The Minish Cap is the third Zelda game that involves the legend of the Four Sword , expanding on the story of and . [4] A magical talking cap named Ezlo can shrink series protagonist **Link** to the size of the Minish , a bug - sized race that live in Hyrule . [5] The game retains some common elements from previous Zelda installments , such as the presence of Gorons , while introducing Kinstones and other new gameplay features . [6] The Minish Cap was generally well received among critics . [7] It was named the 20th best Game Boy Advance game in an IGN feature , and was selected as the 2005 Game Boy Advance Game of the Year by GameSpot .
Head Entity: Link
Tail Entity: The Legend of Zelda
Relation: "Present in Work"
Evidence Sentences: 0,3,4

Figure 1: An example document in the DocRED dataset, where a head and tail entity pair span across multiple sentences.

Aside from extracting entity relations from a document, it is often useful to also highlight the evidence that a system uses to predict them, so that a human or second system can verify them for consistency. What is more, evidence prediction can potentially supplement RE performance by restricting the model's focus on the correct context. In preliminary experiments, we find that current models are able to achieve around 87% RE F1 on DocRED by only keeping the gold evidence sentences when trained and evaluated only on the gold evidence sentences, which is a significant im-

Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021), pages 307–315
Bangkok, Thailand (Online), August 6, 2021.

provement on current leaderboard DocRED RE F1 numbers ($\sim$ 63% RE F1). However, evidence prediction is a challenging task, and most existing relation extraction (RE) approaches ignore the task of evidence prediction entirely.

Most recent approaches for relation extraction fine-tune large pretrained Language Models (LMs) (e.g.,BERT [Devlin *et al.*, 2019], RoBERTa [Liu *et al.*, 2019]) as input encoder. However, naively adapting pretrained LMs for document-level RE faces an issue which limits its performance. Due to the length of a given document, many more entities and relations exist in document-level RE than in intra-sentence RE. A pretrained LM has to simultaneously encode information regarding all pairs of entities for relation extraction, making the task more difficult, and limiting the pretrained LM's effectiveness.

In this paper we propose a new framework: Entity and Evidence Guided Relation Extraction (*E2GRE*), which jointly solves relation extraction and evidence prediction. For evidence prediction, we take a pretrained LM as input encoder and use its internal attention probabilities as additional features to predict evidence sentences. As a result, we use supporting evidence sentences to provide direct supervision on which tokens the LM should attend to during finetuning, which in turn helps improve relation extraction in a joint training framework. To further help LMs focus on a smaller set of relevant word context from a long document, we also introduce entity-guided input sequences as the input to these models, by appending each head entity to the document text, one at a time. This allows the LM encoder to explicitly model relations involving a specific head entity while ignoring all other entity pairs, thus simplifying the task for the LM encoder. The joint training framework helps the model locate the correct semantics that are required for each relation prediction. To the best of our knowledge[1], we are the first to present an effective joint training framework for relation extraction and evidence prediction.

Each of these ideas gives a significant boost in performance, and by combining them, we are able to achieve highly competitive results on the DocRED leaderboard. We obtain 62.5 relation extraction F1 and 50.5 evidence prediction F1 from our E2GRE trained RoBERTa$_{\text{LARGE}}$ model, which is the current state-of-the-art performance on evidence prediction. Our proposed *E2GRE* framework is a simple joint training approach that effectively incorporates information from evidence prediction to guide the pretrained LM encoder, boosting performance on both relation extraction and evidence prediction.

Our main contributions are summarized as follows:

- We propose to generate multiple new entity-guided inputs to a pretrained language model: for every document, we concatenate every entity with the document and feed it as an input sequence to a pretrained LM encoder.
- We propose to use internal attention probabilities of the pre-trained LM encoder as additional features for the evidence prediction.
- Our joint training framework of *E2GRE* which receives the guidance from entity and evidence, improves the performance on both relation extraction and evidence prediction, showing that the two tasks are mutually beneficial to each other.

2 Related Work

Early work attempted to solve RE with statistical methods with different feature engineering [Zelenko *et al.*, 2003; Bunescu and Mooney, 2005]. Later on, neural models have shown better performance at capturing semantic relationships between entities. These methods include CNN-based approaches [Zeng *et al.*; Wang *et al.*, 2016] and LSTM-based approaches [Cai *et al.*, 2016].

On top of using CNNs/LSTM encoders, previous models added additional layers to model semantic interactions. For example, Han *et al.* [2018] introduced using hierarchical attentions in order to generate relational information from coarse-to-fine semantic ideas; Zhang *et al.* [2017] applied GCNs over pruned dependency trees, and Guo *et al.* [2019] introduced Attention Guided Graph Convolutional Networks (AG-GCNs) over dependency trees. These models have shown good performance on intra-sentence relation extraction, but are not easily adapted for document-level RE.

Many approaches for document-level RE are graph-based neural network methods. Quirk and Poon [2017] first introduced a document graph being used for document-level RE; In [Jia *et al.*, 2019], an entity-centric, multi-scale representation

[1] Based on published papers on DocRED.

learning on entity/sentence/document-level LSTM model was proposed for document-level n-ary RE task. Christopoulou *et al.* [2019] recently proposed a novel edge-oriented graph model that deviates from existing graph models. Nan *et al.* [2020] proposed an induced latent graph and Li *et al.* [2020] used an explicit heterogeneous graph for DocRED. These graph models generally focus on constructing unique nodes and edges, and have the advantage of connecting and aggregating different granularities of information. Zhou *et al.* [2021] pointed out multi-entity and multi-label issues for document-level RE, and proposed two techniques: adaptive thresholding and localized context pooling, to address these problems.

Pretrained Language Models [Radford *et al.*, 2019; Devlin *et al.*, 2019; Liu *et al.*, 2019] are powerful NLP tools trained with enormous amounts of unlabelled data. In order to take advantage of the large amounts of text that these models have seen, finetuning on large pretrained LMs has been shown to be effective on relation extraction [Wadden *et al.*, 2019]. Generally, large pretrained LMs are used to encode a sequence and then generate the representation of a head/tail entity pair to learn a classification [Eberts and Ulges, 2019; Yao *et al.*, 2019]. Baldini Soares *et al.* [2019] introduced a new concept similar to BERT called "matching-the-black" and pretrained a Transformer-like model for relation learning. The models were finetuned on SemEval-2010 Task 8 and TACRED achieved state-of-the-art results. Our framework aims to improve the effectiveness of pretrained LMs for document-level relation extraction, with our entity and evidence guided approaches.

3 Method

In this section, we introduce our *E2GRE* framework. First, we describe how to generate entity-guided inputs. Then we present how to jointly train RE with evidence prediction, and finally show how to combine this with our evidence-guided attentions. We use BERT as our pretrained LM when describing our framework.

3.1 Entity-Guided Input Sequences

The goal of relation extraction is to predict *relation* label between every head/tail (*h*/*t*) pair of given entities in a given *document*. Most standard models approach this problem by feeding in an entire document and then extracting all of the head/tail pairs

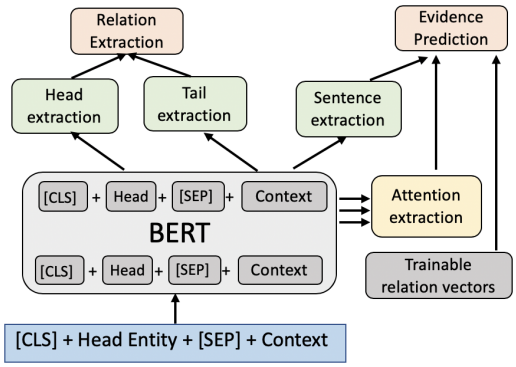

Figure 2: Diagram of our *E2GRE* framework. As shown in the diagram, we pass an input sequence consisting of an entity and document into BERT. We extract head and tails for relation extraction. We show the learned relation vectors in grey. We extract out sentence representation and BERT attention probabilities for evidence predictions.

to predict relations.

Instead, we design entity-guided inputs to give BERT more guidance towards the entities during training. Each training input is organized by concatenating the tokens of the first mention of a head entity, denoted by H, together with the document tokens D, to form: "[CLS]"+ H + "[SEP]" + D + "[SEP]", which is then fed into BERT.[2]

We generate these input sequences for each entity in the given document. Therefore, for a document with N_e entities, N_e new entity-guided input sequences are generated and fed into BERT separately.

Our framework predicts $N_e - 1$ different sets of relations for each training input, corresponding to $N_e - 1$ head/tail entity pairs.

After passing a training input through BERT, we extract the *head entity* embedding and a set of *tail entity* embeddings from the BERT output. After obtaining the head entity embedding $\boldsymbol{h} \in \mathbb{R}^d$ and all tail entity embeddings $\{\boldsymbol{t}_k | \boldsymbol{t}_k \in \mathbb{R}^d\}$ in an entity-guided sequence, where $1 \leq k \leq N_e - 1$, we feed them into a bilinear layer with the sigmoid activation function to predict the probability of i-th relation between the head entity h and the k-th tail

[2]Since the max input length for BERT is 512, for any input length longer than 512, we make use of a sliding window approach over the input and separate it into two chunks (DocRED does not have documents longer than 1024): the first chunk is the input sequence up to 512 tokens; the second chunk is the input sequence with an offset, such that offset + 512 reaches the end of the sequence. This is shown as "[CLS]"+ H + "[SEP]" + D[offset:end] + "[SEP]". We combine these two input chunks in our model by averaging the embeddings and BERT attention probabilities of the overlapping tokens in the model.

entity t_k, denoted by $\hat{y}_{ik}$, as follows

$$\hat{y}_{ik} = \delta(\boldsymbol{h}^T \boldsymbol{W}_i \boldsymbol{t}_k + b_i) \quad (1)$$

where δ is the sigmoid function, $\boldsymbol{W}_i$ and b_i are the learnable parameters corresponding to i-th relation, where $1 \leq i \leq N_r$, and N_r is the number of relations. Finally, we finetune BERT with multi-label cross-entropy loss.

During inference, we group the $N_e - 1$ predicted relations for each entity-guided input sequence from the same document, to obtain the final set of predictions for a document.

3.2 Evidence Guided Relation Extraction

3.2.1 Evidence Prediction

Evidence sentences are sentences which contain important facts for predicting the correct relationships between head and tail entities. Therefore, evidence prediction is a very important auxiliary task to relation extraction and also provides explainability for the model. We build our evidence prediction upon the baseline introduced by Yao *et al.* [2019], which we will describe next.

Let N_s be the number of sentences in the document. We first obtain the sentence embedding $\boldsymbol{s} \in \mathbb{R}^{N_S \times d}$ by averaging all the embeddings of the words in each sentence (i.e., *Sentence Extraction* in Fig. 2). These word embeddings are derived from the BERT output embeddings.

Let $\boldsymbol{r}_i \in \mathbb{R}^d$ be the relation embedding of i-th relation r_i ($1 \leq i \leq N_r$), which is learnable and initialized randomly in our model. We employ a bilinear layer with sigmoid activation function to predict the probability of the j-th sentence s_j being an evidence sentence w.r.t. the given i-th relation r_i as follows.

$$\begin{aligned} \boldsymbol{F}^i_{jk} &= \boldsymbol{s}_j \boldsymbol{W}^r_i \boldsymbol{r}_i + b^r_i \\ \hat{\boldsymbol{y}}^i_{jk} &= \delta(\boldsymbol{F}^i_{jk} \boldsymbol{W}^r_o + b^r_o) \end{aligned} \quad (2)$$

where $\boldsymbol{s}_j$ represents the embedding of j-th sentence, $\boldsymbol{W}^r_i / b^r_i$ and $\boldsymbol{W}^r_o / b^r_o$ are the learnable parameters w.r.t. i-th relation. We define the loss of evidence prediction under the given i-th relation as follows:

$$\begin{aligned} L_{Evi} = -\frac{1}{N_e - 1} \frac{1}{N_s} \sum_{k=1}^{N_e - 1} \sum_{j=1}^{N_s} (y^i_{jk} \log(\hat{\boldsymbol{y}}^i_{jk}) \\ + (1 - y^i_{jk}) \log(1 - \hat{\boldsymbol{y}}^i_{jk})) \end{aligned} \quad (3)$$

where $y^j_{ik} \in \{0, 1\}$, and $y^j_{ik} = 1$ means that sentence j is an evidence for the i-th relation. It should be noted that in the training stage, we use the embedding of true relation in Eq. 2. In testing/inference stage, we use the embedding of the relation predicted by the relation extraction model.

3.2.2 Baseline Joint Training

In [Yao *et al.*, 2019] the baseline relation extraction loss L_{RE} and the evidence prediction loss are combined as the final objective function for the joint training:

$$L_{baseline} = L_{RE} + \lambda * L_{Evi} \quad (4)$$

where $\lambda > 0$ is the weight factor to make trade-offs between two losses, which is data dependent. In order to compare to our models, we utilize a BERT-baseline to predict relation extraction loss and evidence prediction loss.

3.2.3 Guiding BERT Attention with Evidence Prediction

Pretrained language models have been shown to be able to implicitly model semantic relations internally. By looking at internal attention probabilities, Clark *et al.* [2019] has shown that BERT learns coreference and other semantic information in later BERT layers. In order to take advantage of this inherent property, our framework attempts to give more guidance to where correct semantics for RE are located. For each pair of head h and tail t_k, we introduce the idea of using internal attention probabilities extracted from the last l internal BERT layers for evidence prediction.

Let $\boldsymbol{Q} \in \mathbb{R}^{N_h \times L \times (d/N_h)}$ be the query and $\boldsymbol{K} \in \mathbb{R}^{N_h \times L \times (d/N_h)}$ be the key of the Multi-Head Self Attention layer, N_h be the number of attention heads as described in [Vaswani *et al.*, 2017], L be the length of the input sequence and d be the embedding dimension. We first extract the output of multi-headed self attention (MHSA) $\boldsymbol{A} \in \mathbb{R}^{N_h \times L \times L}$ from a given layer in BERT as follows. These extraction outputs are shown as *Attention Extractor* in Fig. 2.

$$\text{Attention} = \text{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d/N_h}}) \quad (5)$$

$$\text{Att-head}_i = \text{Attention}(\boldsymbol{Q}\boldsymbol{W}^Q_i, \boldsymbol{K}\boldsymbol{W}^K_i) \quad (6)$$

$$\boldsymbol{A} = \text{Concat}(\text{Att-head}_1, \cdots, \text{Att-head}_n) \quad (7)$$

For a given pair of head h and tail t_k, we extract the attention probabilities corresponding to head and tail tokens to help relation extraction. Specifically, we concatenate the MHSAs for the last l BERT

layers extracted by Eq. 7 to form an attention probability tensor as: $\tilde{\boldsymbol{A}}_k \in \mathbb{R}^{l \times N_h \times L \times L}$.

Then, we calculate the attention probability representation of each sentence under the given head-tail entity pair (h, t_k) as follows.

1. We first apply maximum pooling layer along the attention head dimension (i.e., second dimension) over $\tilde{\boldsymbol{A}}_k$. The max values are helpful to show where a specific attention head might be looking at. Afterwards we apply mean pooling over the last l layers. We obtain $\tilde{\boldsymbol{A}}_s = \frac{1}{l}\sum_{i=1}^{l} \text{maxpool}(\tilde{\boldsymbol{A}}_{ki})$, $\tilde{\boldsymbol{A}}_s \in \mathbb{R}^{L \times L}$ from these two steps.

2. We then extract the attention probability tensor from the head and tail entity tokens according to the start and end positions of in the document. We average the attention probabilities over all the tokens for the head and tail embeddings to obtain $\tilde{\boldsymbol{A}}_{sk} \in \mathbb{R}^{L}$.

3. Finally, we generate sentence representations from $\tilde{\boldsymbol{A}}_{sk}$ by averaging over the attentions of each token in a given sentence from the document to obtain $\boldsymbol{a}_{sk} \in \mathbb{R}^{N_s}$

Once we get the attention probabilities $\boldsymbol{a}_{sk}$, we pass the sentence embeddings $\hat{\boldsymbol{F}}_k^i$ from Eq. 2 through a transformer layer to encourage inter-sentence interactions and form the new representation $\hat{\boldsymbol{Z}}_k^i$. We combine $\boldsymbol{a}_{sk}$ with $\hat{\boldsymbol{Z}}_k^i$ and feed it into a bilinear layer with sigmoid (δ) for evidence sentence prediction as follows:

$$\hat{\boldsymbol{Z}}_k^i = \text{FFN}(\text{LayerNorm}(\text{Multi-Head}(\hat{\boldsymbol{F}}_k^i))) \quad (8)$$

$$\hat{\boldsymbol{y}}_k^{ia} = \delta(\boldsymbol{a}_{sk}\boldsymbol{W}_i^a \hat{\boldsymbol{Z}}_k^i + b_i^a) \quad (9)$$

Finally, we define the loss of evidence prediction under a given i-th relation based on attention probability representation as follows:

$$L_{Evi}^a = -\frac{1}{N_e-1}\frac{1}{N_s}\sum_{k=1}^{N_e-1}\sum_{j=1}^{N_s}(y_{jk}^{ia}\log(\hat{\boldsymbol{y}}_{jk}^{ia}) + (1-y_{jk}^{ia})\log(1-\hat{\boldsymbol{y}}_{jk}^{ia})), f \quad (10)$$

where $\hat{\boldsymbol{y}}_{jk}^{ia}$ is the j-th value of $\hat{\boldsymbol{y}}_k^{ia}$ computed by Eq. 8.

3.2.4 Joint Training with Evidence Guided Attention Probabilities

Here we combine the relation extraction loss and the attention guided evidence prediction loss as the final objective function for the joint training:

$$L_{E2GRE} = L_{RE}^e + \lambda_a * L_{Evi}^a \quad (11)$$

where $\lambda_a > 0$ is the weight factor to make trade-offs between two losses, which is data dependent.

4 Experiments

4.1 Dataset

DocRED [Yao *et al.*, 2019] is a large document-level dataset for the tasks of relation extraction and evidence prediction. It consists of 5053 documents, 132375 entities, and 56354 relations mined from Wikipedia articles. For each (head, tail) entity pair, there are 97 different relation types as candidates to predict. The first relation type is an "NA" relation between two entities, and the rest correspond to a WikiData relation name. Each of the head/tail pair that contains valid relations also includes a set of evidence sentences.

We follow the same setting in [Yao *et al.*, 2019] to split the data into Train/Development/Test for model evaluation for fair comparisons. The number of documents in Train/Development/Test is 3000/1000/1000, respectively. The dataset is evaluated with the metrics of relation extraction **RE F1**, and evidence **Evi F1**. There are also instances where relational facts may occur in both the development and train set, so we also evaluate **Ign RE F1**, which removes these relational facts.

4.2 Experimental Setup

Hyper-parameter Setting. The configuration for the $\text{BERT}_{\text{BASE}}$ model follows the setting in [Devlin *et al.*, 2019]. We set the learning rate to 1e-5, λ_a to 1e-4, the hidden dimension of the relation vectors to 108, and extract internal attention probabilities from last three BERT layers.

We conduct our experiments by fine-tuning the $\text{BERT}_{\text{BASE}}$ model. The implementation is based on the HuggingFace [Wolf *et al.*, 2020] PyTorch [Paszke *et al.*, 2017] implementation of BERT[3]. The DocRED baseline and our *E2GRE* model have 115M parameters[4]. We implement a RoBERTa-large model for the public leaderboard.

Baseline models. We compare our framework with the following published models.

1. *Context Aware BiLSTM*. [Yao *et al.*, 2019] introduced the original baseline to DocRED in their paper. They used a context-aware BiLSTM (+ additional features such as entity type, coreference and

[3]https://github.com/huggingface/pytorch-pretrained-BERT

[4]We will release the code after paper review.

Model	Dev			Test		
	Ign F_1	**RE** F_1	**Evi** F_1	**Ign** F_1	**RE** F_1	**Evi** F_1
Baseline Models						
BiLSTM [Yao *et al.*, 2019]	45.12	50.95	-	44.73	51.06	-
$\text{BERT}_{\text{BASE}}$ [Wang *et al.*, 2019]	-	54.16	-	-	53.20	-
Transformer-based Models						
$\text{BERT-TS}_{\text{BASE}}$ [Wang *et al.*, 2019]	-	54.32	-	-	53.92	-
$\text{HIN-BERT}_{\text{BASE}}$ [Tang *et al.*, 2020]	54.29	56.31	-	53.70	55.60	-
$\text{CorefBERT}_{\text{BASE}}$ [Ye *et al.*, 2020]	55.32	57.51	-	54.54	56.96	-
$\text{BERT-LSR}_{\text{BASE}}$ [Nan *et al.*, 2020]	52.43	59.00	-	56.97	59.05	-
$\text{CorefRoBERTa}_{\text{LARGE}}$ [Ye *et al.*, 2020]	57.84	59.93	-	57.68	59.91	-
$\text{RoBERTa-ATLOP}_{\text{LARGE}}$[Zhou *et al.*, 2021]	**61.32**	**63.18**	-	**61.39**	**63.40**	-
Joint Frameworks						
$\text{BERT}_{\text{BASE}}$-Joint Training	-	55.04	43.13	-	-	-
BiLSTM-Joint Training [Yao *et al.*, 2019]	-	-	-	44.60	51.10	43.8
Ours						
$\text{E2GRE-BERT}_{\text{BASE}}$	55.22	58.72	47.14	55.4	57.80	48.35
$\text{E2GRE-RoBERTa}_{\text{LARGE}}$	59.55	62.91	**51.11**	60.29	62.51	**50.51**

Table 1: **Main results (%) on the development and test set of DocRED.** We report the official test score of the best checkpoint on the development set. Our *E2GRE* framework is competitive with the top of the current DocRED leaderboard, and is the best on the public leaderboard for evidence prediction.

distance) to encode the document. Head and tail entities are then extracted for relation extraction.

2. *BERT Two-Step*. [Wang *et al.*, 2019] introduced finetuning BERT in a two-step process, where the model first does predicts the NA relation, and then predicts the rest of the relations.

3. *HIN*. [Tang *et al.*, 2020] introduced using a hierarchical inference network to help aggregate the information from entity to sentence and further to document-level in order to obtain semantic reasoning over an entire document.

4. *CorefBERT*. [Ye *et al.*, 2020] introduced a way of pretraining BERT in order to encourage the model to look more at relations between the coreferences of different noun phrases.

5. *BERT+LSR*. [Nan *et al.*, 2020] introduced an induced latent graph structure to help learn how the information should flow between entities and sentences within a document.

6. *ATLOP*. [Zhou *et al.*, 2021] introduced adaptive thresholding and localized context pooling to help alleviate multi-label and multi-entity issues in document-level RE.

4.3 Main Results

Table 1 presents the main results of our proposed *E2GRE* framework, compared with other published results. From this table, we observe that:

- Our RE result is highly competitive with the best published models using $\text{BERT}_{\text{BASE}}$ model. Our proposed framework is also the only one which solves the dual task of evidence prediction, while taking advantage of evidence sentences for relation extraction.
- By replacing $\text{BERT}_{\text{BASE}}$ with $\text{RoBERTa}_{\text{LARGE}}$, we obtain SOTA performance on the DocRED leaderboard. Our test result ranks top 3 on the public leaderboard for relation extraction, and top 1 for evidence prediction [5], which shows that our *E2GRE* is both effective and mutually beneficial for relation extraction and evidence prediction.

We see that our framework significantly boosts F1 scores on both relation extract and evidence prediction compared to previous $\text{BERT}_{\text{BASE}}$ models. Even though we do not have the state-of-the-art performance on relation extraction, we are the first paper to show that with appropriate joint training of RE and evidence prediction we can effectively improve performance for both. [6]

Table 2 compares our proposed *E2GRE* with the joint-training BERT baseline, as described in our

[5] At the time of the submission date

[6] The original DocRED paper [Yao *et al.*, 2019] did not report improvement of RE from joint training.

Models	Multi-Mention			Multi-Evidence		
	R	P	F1	R	P	F1
Relation Extraction						
$\text{BERT}_{\text{BASE}}$-Joint Training	52.42	43.88	47.77	51.20	37.55	43.33
E2GRE-$\text{BERT}_{\text{BASE}}$	55.84	47.75	**51.47**	53.04	40.78	**46.11**
Evidence Predictions						
$\text{BERT}_{\text{BASE}}$-Joint Training	42.59	31.21	36.02	40.44	34.68	37.34
E2GRE-$\text{BERT}_{\text{BASE}}$	42.04	37.78	**39.79**	38.34	40.83	**39.54**

Table 2: Analysis of how Evidence Prediction (EP) impact on Relation Extraction (RE) in the joint training framework. Results on recall, precision and F1 are shown on the dev set with BERT base model.

model section on evidence prediction. We examine the comparison under two challenging scenarios in the dev set: 1) entity pairs which consists of multiple mentions in a document; and 2) entity pairs with multiple evidence sentences for evidence prediction.

From Table 2, we observe that: *E2GRE* shows consistent improvement in terms of F1 on both settings. This is due to the evidence guided attention probabilities from the pretrained LM which helps extract relevant contexts from the document. These relevant contexts further benefit the relation extraction and thus result in significant F1 improvement comparing to the baseline. In summary, our implementation of evidence prediction enhances the performance of relation extraction, and the utilization of a pretrained LM's internal attention probability is a more effective way for joint training.

4.4 Ablation Study

To explore the contribution of different components in our *E2GRE*, we conduct an ablation study in Table 3. We start off with our full *E2GRE*, and consecutively remove the evidence-guided attention and entity-guided sequences. From this table, we observe that: both entity-guided sequences and evidence-guided attentions play a significant role in improving F1 on relation extraction and evidence prediction: entity-guided sequences improve RE by about 2 $F1$ and evidence prediction by about 3.5 $F1$. Evidence-guided attentions improve RE by about 1.7 $F1$ and evidence prediction by about 1 $F1$.

We also observe that entity-guided sequences tend to help more on precision in both tasks of RE and evidence prediction. Entity-guided sequences help by grounding the model to focus on the correct entities, allowing it to be more precise in its information extraction. In contrast, evidence-guided attentions tend to help more on recall in both tasks of

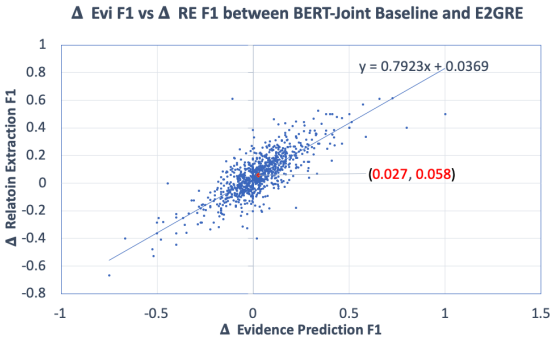

Figure 3: Plot showing the change in RE F1 and EVI F1 from $\text{BERT}_{\text{BASE}}$-Joint Training to our E2GRE-$\text{BERT}_{\text{BASE}}$ model for each document in the dev set.

RE and evidence prediction.These attentions help by giving more guidance to locate relevant contexts, therefore increasing the recall of RE and evidence prediction.

Model	Recall	Precision	F1
Relation Extraction			
E2GRE-$\text{BERT}_{\text{BASE}}$	**59.09**	56.95	**58.72**
– Evidence-guided attentions	54.07	**60.43**	57.08
– Entity-guided inputs	55.06	55.02	55.04
Evidence Prediction			
E2GRE-$\text{BERT}_{\text{BASE}}$	**44.83**	49.75	**47.14**
– Evidence-guided attentions	43.10	**49.66**	46.15
– Entity-guided inputs	47.50	38.91	43.13

Table 3: Ablation study on evidence guided attentions and entity guided input sequence components, by removing attention extraction module in Figure 2, and entity-guided input sequences consecutively on the dev set.

4.5 Analysis on number of BERT layers

Table 4 shows the impact of the number of BERT layers from which the attention probabilities are extracted on evidence prediction and relation extraction. We observe that using the last 3 layers is better than using the last 6 layers. This is because later layers in pretrained LMs tend to focus more on semantic information, whereas earlier layers focus more on syntactic information [Clark *et al.*, 2019]. We hypothesize that the last 6 layers may include noisy information related to syntax.

4.6 Analysis on Evidence/Relation Interdependence

In Fig. 3, we plot the change in RE F1 and EVI F1 between $\text{BERT}_{\text{BASE}}$-Joint Training and our E2GRE-$\text{BERT}_{\text{BASE}}$. We observe that RE F1 and EVI F1 are closely linked, with a coefficient of

Model	Recall	Precision	F1
Relation Extraction			
w/o attention	54.07	60.43	57.08
Last 3 Layers	59.09	56.95	**58.72**
Last 6 Layers	61.87	54.14	58.51
Evidence Prediction			
w/o attention	43.10	49.05	46.15
Last 3 Layers	44.83	49.75	**47.14**
Last 6 Layers	46.34	48.19	46.90

Table 4: Analysis on the number of BERT layers for relation extraction and evidence prediction. Results are shown on dev set.

Model	10%	30%	50%
Relation Extraction			
$\text{BERT}_{\text{BASE}}$-Joint Training	40.00	47.12	52.88
E2GRE-$\text{BERT}_{\text{BASE}}$	**47.37**	**53.48**	**56.55**
Evidence Prediction			
$\text{BERT}_{\text{BASE}}$-Joint Training	21.15	30.70	38.25
E2GRE-$\text{BERT}_{\text{BASE}}$	**36.27**	**41.92**	**44.82**

Table 5: Analysis on how our E2GRE model performs on 10%, 30%, and 50% data for relation extraction.

0.7923, showing that when EVI F1 improves, RE F1 also improves. We observe that the centroid of the points lies in the first quadrant $(2.7\%, 5.8\%)$, showing the overall improvement of our model.

Furthermore, we analyze the effectiveness of our E2GRE model with smaller amounts of training data. Table. 5 shows that our model achieves much larger gains on RE F1 when training with 10, 30 and 50% of the data. E2GRE-$\text{BERT}_{\text{BASE}}$ is able to achieve bigger improvements with less data, as attention probabilities used for evidence prediction provides a effective guidance for relation extraction.

5 Conclusion

In this paper we propose a simple, yet effective joint training framework *E2GRE* (Entity and Evidence Guided Relation Extraction) for relation extraction and evidence prediction on DocRED. In order to more effectively exploit pretrained LMs for document-level RE, we first generate new entity-guided sequences to feed into an LM, focusing the model on the relevant areas in the document. Then we utilize the internal attentions extracted from the last few layers to help guide the LM to focus on relevant sentences for evidence prediction. Our *E2GRE* method improves performance on both RE and evidence prediction, and achieves the state-of-the-art performance on the DocRED public leaderboard. We show that evidence prediction is an important task that helps RE models perform better.

References

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. In *ACL*, 2019.

Razvan Bunescu and Raymond Mooney. A shortest path dependency kernel for relation extraction. In *EMNLP*, Vancouver, British Columbia, Canada, October 2005.

Rui Cai, Xiaodong Zhang, and Houfeng Wang. Bidirectional recurrent convolutional neural network for relation classification. In *ACL*, Berlin, Germany, August 2016.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *EMNLP*, Hong Kong, China, November 2019.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of BERT's attention. In *ACL*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

Markus Eberts and Adrian Ulges. Span-based joint entity and relation extraction with transformer pre-training. 09 2019.

Zhijiang Guo, Yan Zhang, and Wei Lu. Attention guided graph convolutional networks for relation extraction. In *ACL*, Florence, Italy, July 2019.

Xu Han, Pengfei Yu, Zhiyuan Liu, Maosong Sun, and Peng Li. Hierarchical relation extraction with coarse-to-fine grained attention. In *EMNLP*, Brussels, Belgium, October-November 2018.

Robin Jia, Cliff Wong, and Hoifung Poon. Document-level n-ary relation extraction with multiscale representation learning. In *NAACL*, Minneapolis, Minnesota, June 2019.

Bo Li, Wei Ye, Zhonghao Sheng, Rui Xie, Xiangyu Xi, and Shikun Zhang. Graph enhanced dual attention network for document-level relation extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*, 2020.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

Guoshun Nan, Zhijiang Guo, Ivan Sekulić, and Wei Lu. Reasoning with latent structure refinement for document-level relation extraction. In *ACL*, 2020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Chris Quirk and Hoifung Poon. Distant supervision for relation extraction beyond the sentence boundary. In *ACL*, Valencia, Spain, April 2017. ACL.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. Hin: Hierarchical inference network for document-level relation extraction. In *PAKDD*, 2020.

Bayu Distiawan Trisedya, Gerhard Weikum, Jianzhong Qi, and Rui Zhang. Neural relation extraction for knowledge base enrichment. In *ACL*, Florence, Italy, July 2019. ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. Entity, relation, and event extraction with contextualized span representations. In *EMNLP*, Hong Kong, China, November 2019.

Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. Relation classification via multi-level attention CNNs. In *ACL*, Berlin, Germany, August 2016. ACL.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Wang. Fine-tune bert for docred with two-step process, 2019.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. DocRED: A large-scale document-level relation extraction dataset. In *ACL*, 2019.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Maosong Sun, and Zhiyuan Liu. Coreferential reasoning learning for language representation. *ArXiv*, abs/2004.06870, 2020.

Tom Young, Erik Cambria Cambria, Iti Chaturvedi, Minlie Huang, Hao Zhou, and Subham Biswas. Augmenting end-to-end dialog systems with commonsense knowledge. In *AAAI*, 2018.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. Improved neural relation detection for knowledge base question answering. In *ACL*, July 2017.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3:1083–1106, 08 2003.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *COLING*.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, 2017.

Yi Zhao, Huaiyu Wan, Jianwei Gao, and Youfang Lin. Improving relation classification by entity pair graph. In Wee Sun Lee and Taiji Suzuki, editors, *ACML*, volume 101, Nagoya, Japan, 2019.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. Document-level relation extraction with adaptive thresholding and localized context pooling. In *AAAI*, 2021.