ET-MIER: Entity Type-guided Key Mention Identification and Evidence Retrieval for Document-level Relation Extraction

Xin Li[†], **Huangming Xu**[†], **Fu Zhang**[*], **Jingwei Cheng**
School of Computer Science and Engineering, Northeastern University, China
Key Laboratory of Intelligent Computing in Medical Image of Ministry of Education
Northeastern University, China
{lxin690,xuhuangming}@foxmail.com, {zhangfu,chengjingwei}@neu.edu.cn

Abstract

Document-level relation extraction (DocRE) task aims to identify relations between entities in a document. In DocRE, an entity may appear in multiple sentences of a document in the form of mentions. In addition, relation inference requires the use of evidence sentences that can provide key clues to entity pairs. These make DocRE more challenging than sentence-level relation extraction. Existing work does not fully distinguish the contribution of different mentions to entity representation and the importance of mentions in evidence sentences. To address these issues, we observe that entity types can provide consistent semantic constraints for entities of the same type and implicitly preclude impossible relations between entities, which may help the model better understand both intra- and inter-entity mentions. Therefore, we propose a novel model ET-MIER, which for the first time leverages **E**ntity **T**ypes to guide key **M**ention **I**dentification and **E**vidence **R**etrieval. In this way, entity types not only help learn better entity representation but also enhance evidence retrieval, both of which are crucial for DocRE. We conduct experiments on widely-adopted datasets and show that our model achieves state-of-the-art performance.[1]

1 Introduction

Relation extraction (RE) is a fundamental task in natural language processing (NLP) and knowledge graph construction, typically categorized into two levels: sentence-level and document-level (Delaunay et al., 2023). Document-level relation extraction (DocRE) focuses on identifying relations between entities within a document. In DocRE, an entity can have multiple mentions, and multiple relations for an entity pair can exist and may be expressed across different evidence sentences. Due to this complexity, DocRE better aligns with real-world scenarios and is, thus, more challenging.

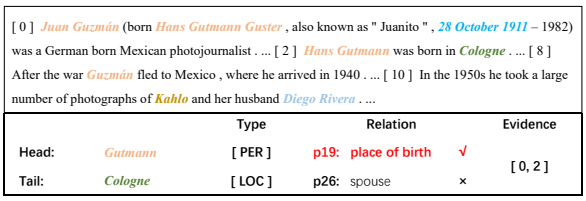


Figure 1: An example of DocRE. Multiple mentions of an entity are marked with the same color.

In DocRE, an important issue is *how to learn accurate semantic representation of entities*. Given that the semantic information of an entity is mainly conveyed through its mentions (including type, context, and connections to other entities), many studies focus on enhancing entity representation based on these mentions. They usually use pooling functions (the most commonly used of which is log-sumexp pooling (Zhou et al., 2021)) to aggregate the semantic information in mentions into entity representation. However, since the pooling is essentially an aggregation operation, it may not be able to effectively distinguish the unique contributions of individual mentions, especially when dealing with scenarios involving complex interactions and multiple relations.

As illustrated in Figure 1, the contributions of different mentions to entity representation may vary, where the mentions in the sentences [0] and [2] offer richer semantics for "Gutmann" than the mention in the sentence [8]. These mentions may also help extract evidence sentences and play a more important role in extracting the relations between ("Gutmann", "Cologne"). Existing methods may overlook such subtle differences. While some studies generate relation-specific entity representation (Yu et al., 2022; Dai et al., 2023), this can lead to excessive parameters due to the diversity of relations in DocRE. Therefore, effectively leveraging

[1]Code: https://github.com/NEU-IDKE/ET-MIER
[†] Equal contribution. [*] Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 17701–17714
November 4-9, 2025

mentions to capture fine-grained entity semantic information remains a key challenge.

To better capture entity semantics, it is crucial to apply fine-grained attention based on the contribution of each mention, enabling the model to focus on key mentions. As shown in Figure 1, entity types provide consistent semantic cues for similar entities and implicitly constrain possible relations (e.g., no "spouse" relation between PER and LOC), helping the model better understand both intra- and inter-entity mentions. Based on the above observations, we propose for the first time leveraging entity types to guide key mention identification and evidence retrieval for DocRE.

Specifically, we propose three innovative strategies: (1) **Optimization of entity type representation**. The existing method mainly uses entity type information to directly filter out impossible relations (Xiao et al., 2022), without considering the semantic information contained within the entity types themselves and the differences between entity types. Therefore, we propose a method to optimize entity type representation, which will further facilitate the identification of key mentions. (2) **Identification of key mentions guided by entity types**. Our goal is to generate type-specific entity representation with high discrimination of mentions. To achieve this, we carefully design a key mention identification task, which distinguishes the importance of mentions for entity representation by identifying the semantic relevance between mentions and entity types, thereby generating high-quality entity representation. (3) **Enhancement of evidence retrieval incorporating entity types**. We introduce evidence retrieval following previous work (Xie et al., 2022; Dai et al., 2023; Lu et al., 2023), as an auxiliary task to help the model filter out irrelevant sentences. Different from these works that capture evidence sentences by integrating all mention features, we innovatively introduce the type-specific entity representation into this task, thereby focusing on local key mentions at the same time. This enables the task to extract evidences from both global and local perspectives.

Based on the above three strategies, we propose a novel **E**ntity **T**ype-guided key **M**ention **I**dentification and **E**vidence **R**etrieval (ET-MIER) for DocRE. Our contributions include:

- We propose a contrastive learning-based method to optimize entity type representations, generating more accurate representations while capturing differences between entity types.
- We design an entity type-guided key mention identification task, which distinguishes the contribution of mentions to entity representation, thereby learning better type-specific entity representation to achieve DocRE. This is the first work to consider the impact of entity types on mentions.
- We innovatively incorporate the entity types to guide evidence retrieval, enabling the model to extract evidence sentences more comprehensively.
- We conduct experiments on two widely-adopted DocRE benchmarks, DocRED (Yao et al., 2019) and Re-DocRED (Tan et al., 2022b). Experiments show that our model achieves state-of-the-art (SOTA) results while maintaining good efficiency.

2 Related Work

Existing DocRE methods may be roughly divided into two categories: graph-based and transformer-based methods.

Graph-based DocRE constructs a heterogeneous graph by treating mentions, entities, or sentences as nodes to model their interactions, thereby enabling relation reasoning. Representative methods include GAIN (Zeng et al., 2020), SIRE (Zeng et al., 2021), DRN (Xu et al., 2021b), and FCDS (Zhu et al., 2024), among others.

Transformer-based DocRE leverages Transformers (Vaswani et al., 2017) to capture long-range dependencies among entities for relation prediction. ATLOP (Zhou et al., 2021) first introduces an adaptive threshold for multi-relation prediction. EIDER (Xie et al., 2022) proposes an evidence-enhanced DocRE framework. Other works have also proposed various frameworks to further enhance performance, including RSMAN (Yu et al., 2022), DREEAM (Ma et al., 2023), AA (Lu et al., 2023), SRF (Zhang et al., 2024), TTM-RE (Gao et al., 2024), AMTL (Xu et al., 2025) and VaeDiff-DocRE (Tran et al., 2025), among others. Additionally, several large-model-based DocRE methods have also emerged gradually (Xue et al., 2024; Zhang et al., 2025a,b).

With respect to **entity types in the DocRE task**, SAIS (Xiao et al., 2022) employs entity type classification to filter out impossible relations but does

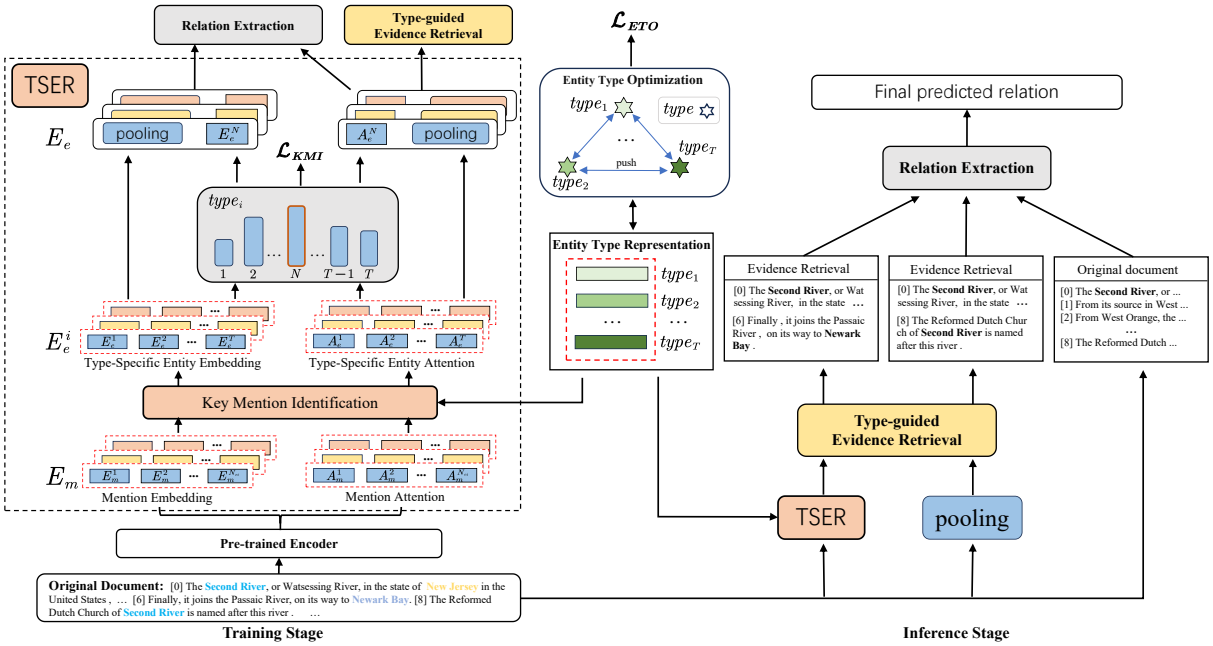


Figure 2: The overall architecture of our model ET-MIER.

not incorporate type information into entity representations. **In contrast**, our approach integrates types into entity representations and leverages them to guide key mention identification and evidence retrieval, leading to improved performance. Moreover, **entity types** are also used **in other NLP tasks**, but for different purposes. For instance, (Raiman, 2022) uses types to eliminate invalid candidates in entity linking, (Ayoola et al., 2022) uses them to initialize entity scores for entity disambiguation, and (Bhargav et al., 2022) treats hierarchical types as an auxiliary task to enhance entity linking. **This is fundamentally different from** our method and purpose of using entity types: we dynamically use entity types in DocRE to guide key mention selection and evidence retrieval.

3 Problem Formulation

Given a document D containing sets of tokens $\mathcal{X}_D = \{x_l\}_{l=1}^{|\mathcal{X}_D|}$, sentences $\mathcal{S}_D = \{s_n\}_{n=1}^{|\mathcal{S}_D|}$, and entities $\mathcal{E}_D = \{e_i\}_{i=1}^{|\mathcal{E}_D|}$, each entity e_i corresponds to an entity type $\tau_i \in \mathcal{T}$, where $\mathcal{T} = \{\tau_i\}_{i=1}^{T}$ is a pre-annotated set containing T different entity types. And, there is a set of mentions $\mathcal{M}_{e_i} = \left\{m_j^i\right\}_{j=1}^{N_{e_i}}$ for an entity e_i, where N_{e_i} is the number of mentions of e_i. The DocRE task is a multi-label classification task, aiming to predict a subset of relations $\mathcal{R} = R \cup \{\text{NA}\}$ for each entity pair (e_h, e_t), where R is a predefined set of relations, and NA denotes no relation between entities.

4 Methodology

Our ET-MIER model in Figure 2 consists of three main parts: a **type-specific entity representation** (TSER) module, which aims to obtain entity representation through joint entity type optimization and key mention identification tasks; a **type-guided evidence retrieval** module, which extracts more relevant evidence sentences by introducing type-specific entity pair features when calculating the importance distribution of sentences; and a **relation extraction** module.

4.1 Document Encoding

Given a document D, a pair of tokens "*" are inserted at the beginning and end of each mention to indicate the position of the entity mention (Zhang et al., 2017). Then, we feed D into a pre-trained language model (PLM) to obtain d-dim token embeddings $\boldsymbol{H}$ and the cross-token attention $\boldsymbol{A}$:

$$\boldsymbol{H}, \boldsymbol{A} = PLM(\left[x_1, x_2, \ldots, x_{|\mathcal{X}_D|}\right]) \quad (1)$$

where $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{X}_D| \times d}$, d is the PLM dimension of the encoder, and $\boldsymbol{A} \in \mathbb{R}^{|\mathcal{X}_D| \times |\mathcal{X}_D|}$ is the average of the attention heads in the last transformer layer. For the j-th mention $m_j^i \in \mathcal{M}_{e_i}$ of an entity e_i, we use the feature of the special beginning token "*" to get its embedding $\boldsymbol{m}_j^i$ and attention $\boldsymbol{a}_j^i$. To obtain the embedding $\boldsymbol{h}_{e_i} \in \mathbb{R}^d$ for entity e_i, most previous

works use pooling functions (e.g., the *logsumexp* (Zhou et al., 2021)), to fuse mention embeddings:

$$\boldsymbol{h}_{e_i} = \log \sum_{j=1}^{N_{e_i}} \exp(\boldsymbol{m}_j^i) \tag{2}$$

4.2 Type-Specific Entity Representation

In the DocRE task, the performance of the model usually depends on the quality of entity representation. The semantic information of an entity is mainly expressed by the mentions corresponding to the entity. Existing methods ignore the contribution of mentions to the semantic representation of entities when using Eq. (2) to obtain entity representation. To this end, we propose the type-specific entity representation (TSER) module to obtain entity representation by jointly optimizing entity types and identifying key mentions.

4.2.1 Entity Type Optimization

To better capture the semantic information contained within the entity types themselves and highlight the differences between entity types, we propose an entity type optimization method. The optimized entity type representation will also further facilitate the identification of key mentions.

We first use the Xavier initialization (Glorot and Bengio, 2010) to encode each entity type into a prototype representation $\boldsymbol{\mathcal{P}}_\tau \in \mathbb{R}^d, \tau \in \mathcal{T}$. Then, we introduce contrastive learning to adjust the prototype representations of different entity types to make them significantly distinguishable, which helps the model learn the semantic differences between entity types. Based on this idea, we design the following entity type contrastive learning loss:

$$\mathcal{L}_{ETO} = \sum_{\tau \in \mathcal{T}} \log \sum_{t \in \mathcal{T}, t \neq \tau} exp(\boldsymbol{\mathcal{P}}_\tau \cdot \boldsymbol{\mathcal{P}}_t / \varsigma) \tag{3}$$

where $\varsigma \in \mathbb{R}^+$ is a temperature parameter. During the training process, the entity types will be used to guide the model to select key mentions for subsequent relation extraction. The interaction between mentions and entity types will be updated to make the type semantics expressed by entities of the same type more consistent and the semantics expressed by different entity types more distinguishable.

4.2.2 Type-guided Key Mention Identification

To distinguish the contribution of mentions to entity representation, we first propose an entity type-guided key mention identification method, which identifies the semantic relevance of an entity's mentions to its entity type to help the model distinguish which mentions are key ones, thereby learning better type-specific entity representation.

Specifically, we calculate the semantic correlation between each optimized $\boldsymbol{\mathcal{P}}_\tau$ and $\boldsymbol{m}_j^i$, and obtain attention weight α_{ij}^τ:

$$\alpha_{ij}^\tau = \frac{\exp\left(f(\boldsymbol{m}_j^i, \boldsymbol{\mathcal{P}}_\tau)\right)}{\sum_{k=1}^{N_{e_i}} \exp\left(f(\boldsymbol{m}_k^i, \boldsymbol{\mathcal{P}}_\tau)\right)} \tag{4}$$

where f is a dot product, and $\alpha_{ij}^\tau \in \mathbb{R}^+$ represents the degree of attention paid by entity e_i to the mention m_j^i under a specific entity type τ. We fuse the key mentions to obtain type-specific entity features:

$$\boldsymbol{e}_i^\tau = \sum_{j=1}^{N_{e_i}} \alpha_{ij}^\tau \boldsymbol{m}_j^i; \qquad \boldsymbol{A}_i^\tau = \sum_{j=1}^{N_{e_i}} \alpha_{ij}^\tau \boldsymbol{a}_j^i \tag{5}$$

where $\boldsymbol{e}_i^\tau \in \mathbb{R}^d$ represents the embedding of entity e_i belonging to the specific type τ, $\boldsymbol{a}_j^i \in \mathbb{R}^{|\mathcal{X}_D|}$ represents the attention score of the j-th mention of entity e_i to all tokens, and $\boldsymbol{A}_i^\tau \in \mathbb{R}^{|\mathcal{X}_D|}$ represents the attention score of entity e_i belonging to the specific type τ. This process leverages type information to identify key mentions that better capture entity semantics, yielding fine-grained type-specific entity features and enhancing entity representation.

On this basis, for a given entity pair (e_h, e_t), assuming that the specific entity types are τ_h and τ_t, the localized context embedding is calculated:

$$\boldsymbol{p}^{\tau_h \tau_t} = \frac{\boldsymbol{A}_h^{\tau_h} \circ \boldsymbol{A}_t^{\tau_t}}{\mathbf{1}^\top (\boldsymbol{A}_h^{\tau_h} \circ \boldsymbol{A}_t^{\tau_t})}; \boldsymbol{c}_{(h,t)}^{\tau_h \tau_t} = \boldsymbol{H}^\top \boldsymbol{p}^{\tau_h \tau_t} \tag{6}$$

where $\circ$ is the Hadamard product, $\boldsymbol{p}^{\tau_h \tau_t} \in \mathbb{R}^{|\mathcal{X}_D|}$ represents the importance of each token for the pair (τ_h, τ_t), $\boldsymbol{c}_{(h,t)}^{\tau_h \tau_t} \in \mathbb{R}^d$ represents the localized context embedding calculated for (e_h, e_t) based on the entity types τ_h and τ_t.

Further, as shown in Eq. (2), the previous work fuses all mention features through pooling to form a coarse-grained entity representation. In contrast, the type-specific entity embedding in Eq. (5) focuses on key mentions to form a fine-grained representation. To make the final entity representation more comprehensive, we fuse these two representations and use localized context embedding to form type-specific context-aware entity embeddings:

$$\boldsymbol{z}_{e_h}^\tau = \tanh(\boldsymbol{W}_h[\boldsymbol{h}_{e_h}; \boldsymbol{e}_h^\tau] + \boldsymbol{W}_{c_h} \boldsymbol{c}_{(h,t)}^{\tau \tau_t^*}) \tag{7}$$

$$\boldsymbol{z}_{e_t}^\tau = \tanh(\boldsymbol{W}_t[\boldsymbol{h}_{e_t}; \boldsymbol{e}_t^\tau] + \boldsymbol{W}_{c_t} \boldsymbol{c}_{(h,t)}^{\tau_h^* \tau}) \tag{8}$$

where $[\cdot;\cdot]$ represents concatenation, $\boldsymbol{W}_h, \boldsymbol{W}_t \in \mathbb{R}^{d\times 2d}$, $\boldsymbol{W}_{c_h}, \boldsymbol{W}_{c_t} \in \mathbb{R}^{d\times d}$ are learnable parameters. When calculating the tail entity, the head entity uses its ground-truth type τ_h^*.

Now, for an entity e, it can appear at the head or tail of the entity pair (i.e., e_ν for $\nu \in \{h,t\}$), we obtain its embedding $\boldsymbol{z}_{e_\nu}^{\tau}$ corresponding to each entity type $\tau \in \mathcal{T}$, thus forming an embedding set $\Phi_{e_\nu} = \{\boldsymbol{z}_{e_\nu}^{\tau} | \tau \in \mathcal{T}\}$. We further design an embedding selection way based on entity type recognition, which aims to select the embedding from the set Φ_{e_ν} that best matches the entity's type. We use a bilinear classifier to calculate the probability that the entity e_ν belongs to each entity type and employ cross-entropy as the objective:

$$\mathbb{P}(\tau|e_\nu) = \sigma(\boldsymbol{z}_{e_\nu}^{\tau}\boldsymbol{W}_\tau\boldsymbol{\mathcal{P}}_\tau + b_\tau) \tag{9}$$

$$\mathcal{L}_{KMI} = -\sum_{e\in\mathcal{E}_D}\sum_{\tau\in\mathcal{T}}\sum_{\nu\in\{h,t\}} y_{e,\tau}^{ET}(\log \mathbb{P}(\tau|e_\nu)) \tag{10}$$

where $\boldsymbol{W}_\tau \in \mathbb{R}^{d\times d}$, $b_\tau \in \mathbb{R}$ are parameters, $\boldsymbol{\mathcal{P}}_\tau \in \mathbb{R}^d$ is the optimized prototype representation of entity type τ, $y_{e,\tau}^{ET} \in \{0,1\}$, and $y_{e,\tau}^{ET} = 1$ denotes the type of the entity e is τ.

This selected embedding $\boldsymbol{z}'_{e_\nu} \in \Phi_{e_\nu}$ serves as final *type-specific representation of the entity* e_ν, used for subsequent relation extraction and evidence retrieval. During inference, to ensure alignment with datasets, we directly consider the embedding of pre-annotated entity type label in datasets as the final type-specific entity representation.

4.3 Type-guided Evidence Retrieval

To reduce the noise introduced by sentences irrelevant to entity pairs, we propose an enhanced evidence retrieval method guided by entity types.

Previous evidence retrieval works (Xie et al., 2022; Ma et al., 2023; Lu et al., 2023) calculate the importance distribution of sentences by integrating all mention features corresponding to an entity pair. The importance of the sentences in which the mentions appeared is not fully distinguished. Through our TSER module, we effectively distinguish the key mentions when obtaining the feature of the entity pair. Therefore, we propose to further introduce the type-specific entity pair feature that incorporate key mentions when calculating the importance distribution of sentences, which can help to determine the importance of sentences, thereby extracting more relevant evidence sentences.

For each entity pair (e_h, e_t) with a valid relation, we first calculate the sentence-level importance distribution $\boldsymbol{q}^{(h,t)} \in \mathbb{R}^{|\mathcal{S}_D|}$:

$$\boldsymbol{p}^{(h,t)} = \frac{\boldsymbol{a}_h \circ \boldsymbol{a}_t}{\mathbf{1}^\top(\boldsymbol{a}_h \circ \boldsymbol{a}_t)}; \qquad q_n^{(h,t)} = \sum\nolimits_{j\in s_n} p_j^{(h,t)} \tag{11}$$

where $\boldsymbol{a}_h, \boldsymbol{a}_t \in \mathbb{R}^{|\mathcal{X}_D|}$ denote entity-level attention vectors, obtained by pooling the mention-level attentions $\boldsymbol{a}_j^i$ (Eq. (5)) for each entity, and $\boldsymbol{p}^{(h,t)} \in \mathbb{R}^{|\mathcal{X}_D|}$ is a distribution representing the importance of each token for an entity pair (e_h, e_t).

Further, we calculate the sentence-level importance distribution $\boldsymbol{q}^{\tau_h'\tau_t'} \in \mathbb{R}^{|\mathcal{S}_D|}$ that incorporates type-specific entity features:

$$q_n^{\tau_h'\tau_t'} = \sum\nolimits_{j\in s_n} p_j^{\tau_h'\tau_t'} \tag{12}$$

where $p_j^{\tau_h'\tau_t'} \in \boldsymbol{p}^{\tau_h'\tau_t'}$ is the token importance distribution (Eq. (6)) corresponding to the type-specific entity representations $\boldsymbol{z}'_{e_h}$ and $\boldsymbol{z}'_{e_t}$.

Then, we fuse two distributions $\boldsymbol{q}^{\tau_h'\tau_t'}$ and $\boldsymbol{q}^{(h,t)}$:

$$\hat{\boldsymbol{q}}^{(h,t)} = \varepsilon\boldsymbol{q}^{\tau_h'\tau_t'} + (1-\varepsilon)\boldsymbol{q}^{(h,t)} \tag{13}$$

where $\varepsilon \in \mathbb{R}^+$ is a balance parameter, $\hat{\boldsymbol{q}}^{(h,t)} \in \mathbb{R}^{|\mathcal{S}_D|}$ represents the fused sentence-level importance distribution. Final, we optimize the evidence retrieval by minimizing the Kullback-Leibler (KL) divergence between $\hat{\boldsymbol{q}}^{(h,t)}$ and the evidence distribution $\boldsymbol{v}^{(h,t)}$ derived from gold evidence labels:

$$\mathcal{L}_{Evi} = -D_{KL}(\boldsymbol{v}^{(h,t)} || \hat{\boldsymbol{q}}^{(h,t)}) \tag{14}$$

4.4 Relation Extraction

This module extracts the relation between an entity pair (e_h, e_t) by using their type-specific representations $\boldsymbol{z}'_{e_h}$ and $\boldsymbol{z}'_{e_t}$. A grouped bilinear classifier (Tang et al., 2020) computes the relation score r:

$$\mathbb{P}_r^{(h,t)} = \sigma(\sum_{k=1}^{K} \boldsymbol{z}'^{k\top}_{e_h}\boldsymbol{W}_r^k\boldsymbol{z}'^{k}_{e_t} + b_r) \tag{15}$$

where $\boldsymbol{W}_r^k \in \mathbb{R}^{d/K\times d/K}$ for $k = 1...K$ and $b_r \in \mathbb{R}$ are learnable parameters, σ is the sigmoid function, and $\mathbb{P}^{(h,t)} \in \mathbb{R}^{|\mathcal{R}|}$ represents the probability score of the relation between the entity pair (e_h, e_t). To dynamically adjust the prediction probability of each relation, we use the Adaptive Thresholding Loss (ATL) from ATLOP (Zhou et al., 2021). During training, a virtual category TH is

Model	PLM	Dev		Test	
		Ign-F1	F1	Ign-F1	F1
DocGNRE (Li et al., 2023) ◇	LLMs	11.10	11.18	11.04	11.12
LMRC (Li et al., 2024) ◇	LLMs	52.29	52.56	52.15	52.45
D-F (Xue et al., 2024) ◇	LLMs	53.48	54.22	52.50	53.33
D-R-F (Xue et al., 2024) ◇	LLMs	56.10	56.58	54.35	54.84
AutoRE (Xue et al., 2024) ◇	LLMs	59.25	60.17	58.33	59.29
EP-RSR (Zhang et al., 2025b) ◇	LLMs	63.93	65.14	63.03	64.24
ATLOP (Zhou et al., 2021) *	BERT_base	73.35	74.22	73.22	74.02
DocuNet (Zhang et al., 2021) *	BERT_base	73.68	74.65	73.60	74.49
KD-DocRE (Tan et al., 2022a) *	BERT_base	73.76	74.69	73.67	74.55
KMGRE (Jiang et al., 2022) *	BERT_base	73.33	74.44	73.39	74.46
DocRE-BSI (Zhang et al., 2023) *	BERT_base	75.03	75.85	74.85	75.77
FCDS (Zhu et al., 2024) ‡	BERT_base	-	73.26	-	72.79
SRF (Zhang et al., 2024) †	BERT_base	73.76	74.66	73.16	74.06
VaeDiff-DocRE (Tran et al., 2025) †	BERT_base	74.96	75.89	74.13	75.07
ET-MIER	BERT_base	**77.54±0.21**	**78.26±0.18**	**77.13**	**77.84**
ATLOP (Zhou et al., 2021)	RoBERTa_large	76.88	77.63	76.94	77.73
DocuNet (Zhang et al., 2021)	RoBERTa_large	77.53	78.16	77.27	77.92
KD-DocRE (Tan et al., 2022a)	RoBERTa_large	77.92	78.65	77.63	78.35
DREEAM (Ma et al., 2023)	RoBERTa_large	-	-	79.66	80.73
PEMSCL (Guo et al., 2023)	RoBERTa_large	79.02	79.89	79.01	79.86
AA (Lu et al., 2023)	RoBERTa_large	80.04	81.15	80.12	81.20
TTM-RE (Gao et al., 2024) †	RoBERTa_large	78.22	78.25	78.54	80.08
VaeDiff-DocRE (Tran et al., 2025) †	RoBERTa_large	78.35	79.19	78.22	79.03
ET-MIER	RoBERTa_large	**80.72±0.07**	**81.36±0.11**	**80.83**	**81.41**

Table 1: Results on Re-DocRED dataset. Results with * are from Zhang et al. (2023). ‡ from our reproduction utilizing its public code. † from the original paper, and ◇ from Zhang et al. (2025b). Others are reported in Lu et al. (2023). Best results are in bold.

learned for each entity pair (e_h, e_t) as a threshold, ensuring that valid relation scores $\mathcal{P}_{h,t} \subset R$ exceed TH, while invalid relation scores $\mathcal{N}_{h,t} \subseteq R$ remain below it. The relation extraction loss is defined as:

$$\mathcal{L}_{RE} = -\sum_{h \neq t} \sum_{r \in \mathcal{P}_{h,t}} \log\left(\frac{\exp(\mathbb{P}_r^{(h,t)})}{\sum_{r' \in \mathcal{P}_{h,t} \cup \{\mathrm{TH}\}} \exp(\mathbb{P}_{r'}^{(h,t)})}\right) - \log\left(\frac{\exp(\mathbb{P}_{\mathrm{TH}}^{(h,t)})}{\sum_{r' \in \mathcal{N}_{h,t} \cup \{\mathrm{TH}\}} \exp(\mathbb{P}_{r'}^{(h,t)})}\right) \quad (16)$$

Finally, we jointly optimize the model and use λ_1, λ_2, and λ_3 to balance the impact of the losses:

$$\mathcal{L}_{All} = \mathcal{L}_{RE} + \lambda_1 \mathcal{L}_{Evi} + \lambda_2 \mathcal{L}_{ETO} + \lambda_3 \mathcal{L}_{KMI} \quad (17)$$

5 Experiments and Analysis

5.1 Experiment Setup

Datasets and Parameters. We evaluate our model on two widely-adopted document-level relation extraction datasets containing entity types, DocRED (Yao et al., 2019) and Re-DocRED (Tan et al., 2022b). We use BERT_base (Devlin et al., 2019) and RoBERTa_large (Liu et al., 2019) as document encoders. We run 5 times independently and give the average results. Comprehensive dataset statistics, parameter configurations, and parameter experimental analysis are provided in ***Appendix A***.

Baselines. We conduct a comprehensive comparison with *graph-based* methods (including GAIN, SIRE, etc.) and *transformer-based* methods (including ATLOP, EIDER, AA, etc.) as detailed in Section 2 of Related Work. Additionally, a comparison is also made with methods based on large models. We also use F1 and Ign-F1 as evaluation metrics. Ign-F1 is measured by ignoring relation triples present in train set.

5.2 Main Results

We perform experiments on two datasets. Results on Re-DocRED in Table 1 indicate that our method is **consistently better than the competitive baselines**. On the test set, our model achieves improvements of 2.07 in F1 and 2.28 in Ign-F1 over the previous best baseline under the BERT_base

Model	PLM	Dev		Test	
		Ign-F1	F1	Ign-F1	F1
DocGNRE (Li et al., 2023) *	LLMs	13.65	13.84	13.67	13.93
LMRC (Li et al., 2024) *	LLMs	38.62	39.25	38.09	38.66
D-F (Xue et al., 2024) *	LLMs	44.77	46.38	45.30	47.08
D-R-F (Xue et al., 2024) *	LLMs	44.32	45.77	45.98	47.50
AutoRE (Xue et al., 2024) *	LLMs	45.58	47.17	45.45	47.15
EP-RSR (Zhang et al., 2025b) *	LLMs	51.25	53.77	51.77	54.57
LSR (Nan et al., 2020)	BERT_base	52.43	59.00	56.97	59.05
RSMAN (Yu et al., 2022)	BERT_base	57.22	59.25	57.02	59.29
GAIN (Zeng et al., 2020)	BERT_base	59.14	61.22	59.00	61.24
ATLOP (Zhou et al., 2021)	BERT_base	59.22	61.09	59.31	61.30
DocuNet (Zhang et al., 2021)	BERT_base	59.86	61.83	59.93	61.86
KD-DocRE (Tan et al., 2022a)	BERT_base	60.08	62.03	60.04	62.08
SAIS (Xiao et al., 2022)	BERT_base	59.98	62.96	60.96	62.77
EIDER (Xie et al., 2022)	BERT_base	60.51	62.48	60.42	62.47
DREEAM (Ma et al., 2023)	BERT_base	60.51	62.55	60.03	62.49
RSEEA (Dai et al., 2023)	BERT_base	60.87	62.91	60.79	62.84
AA (Lu et al., 2023)	BERT_base	61.31	63.38	60.84	63.10
SRF (Zhang et al., 2024)	BERT_base	60.46	62.50	59.84	62.11
ET-MIER	BERT_base	**61.55±0.13**	**63.51±0.14**	**61.08**	**63.24**
ATLOP (Zhou et al., 2021)	RoBERTa_large	61.32	63.18	61.39	63.40
DocuNet (Zhang et al., 2021)	RoBERTa_large	62.23	64.12	62.39	64.55
KD-DocRE (Tan et al., 2022a)	RoBERTa_large	62.16	64.19	62.57	64.28
EIDER (Xie et al., 2022)	RoBERTa_large	62.34	64.27	62.85	64.79
DREEAM (Ma et al., 2023)	RoBERTa_large	62.29	64.20	62.12	64.27
AA (Lu et al., 2023)	RoBERTa_large	**63.15**	**65.19**	62.88	64.98
ET-MIER	RoBERTa_large	63.07±0.15	64.93±0.19	**62.91**	**65.02**

Table 2: Performance on DocRED dataset. Results of RSMAN, RSEEA and SRF are from their original papers. Results of the other models are referred from Lu et al. (2023). * from (Zhang et al., 2025b). Best results are in bold.

setting. When using RoBERTa_large, our model also surpasses the previous SOTA model AA, and demonstrates significant gains over the representative model ATLOP, with improvements of 3.68 in F1 and 3.89 in Ign-F1. Moreover, we directly compare our model (ET-MIER) with recent LLMs approaches on Re-DocRED. ET-MIER achieves state-of-the-art performance on both the dev and test sets, reaching an Ign-F1 of 80.83 and an F1 of 81.41 on the test set, surpassing the EP-RSR model by +17.17 F1.

Results on DocRED are shown in Table 2. Our model outperforms most of baselines and achieves improvements of 1.77 and 1.94 in F1 and Ign-F1 over the representative ATLOP-BERT_base model on test set. Compared with the DocRED, our model has a more obvious improvement on the Re-DocRED, which may be attributed to the clearer data annotation in Re-DocRED, providing a more reliable basis for comparison. The improvements demonstrate the effectiveness of our idea of leveraging entity types as guidance for DocRE. Moreover, we compare ET-MIER directly with several recent competitive LLMs approaches. Using a BERT_base encoder, it surpasses the strongest baseline (EP-RSR) by around 8.7 F1 points on the test set, showing that its lightweight design does not compromise its generalization ability or its effectiveness in handling long-context documents.

5.3 Ablation Study

To verify the effectiveness of different modules, we conduct a series of ablation experiments as shown in Table 3:

w/o TSER or w TSER (frozen). We use the original pooling function instead of the type-specific entity representation we proposed (w/o TSER). We also freeze the prototype representation of the entity type (w TSER frozen), leaving the semantic information contained in the type representation unchanged, resulting in a decrease in performance. This emphasizes that updating type

Model	Dev		Test	
	Ign-F1	F1	Ign-F1	F1
DocRED				
Ours-BERT_base	**61.55**	**63.51**	**61.08**	**63.24**
w TSER (frozen)	61.16	63.31	60.69	63.02
w/o TSER	60.90	63.04	60.33	62.65
w/o KMI	60.92	62.91	60.40	62.62
w/o ETO	61.04	63.08	60.86	63.12
w/o Type-guided EviR ($\varepsilon = 0$)	61.16	63.24	60.92	63.22
w/o Type-guided EviR ($\varepsilon = 1$)	60.82	63.07	60.48	62.97
Re-DocRED				
Ours-RoBERTa_large	**80.72**	**81.36**	**80.83**	**81.41**
w TSER (frozen)	80.33	80.98	80.38	80.98
w/o TSER	80.17	80.88	80.26	80.90
w/o KMI	79.84	80.53	80.05	80.68
w/o ETO	80.43	81.12	80.56	81.19
w/o Type-guided EviR ($\varepsilon = 0$)	80.34	81.02	80.65	81.27
w/o Type-guided EviR ($\varepsilon = 1$)	80.17	80.88	80.26	80.90

Table 3: Ablation study on DocRED and Re-DocRED.

Model	Memory (GiB)	Trainable Params (M)
(a) *without* Evidence Retrieval		
ATLOP (Zhou et al., 2021)	10.8	115.4
SSAN (Xu et al., 2021a)	6.9	113.5
RSMAN (Yu et al., 2022)	13.5	117.8
KD-DocRE (Tan et al., 2022a)	15.2	200.1
(b) *with* Evidence Retrieval		
EIDER (Xie et al., 2022)	43.1	120.2
SAIS (Xiao et al., 2022)	46.2	118.0
DREEAM (Ma et al., 2023)	11.8	115.4
AA (Lu et al., 2023)	21.9	130.8
Ours (*with* Evidence Retrieval)	20.4	115.4

Table 4: Complexity comparison in terms of memory and parameters on DocRED. BERT_base is used as the document encoder.

representation can better capture semantic information, thereby improving performance.

w/o KMI or w/o ETO. We eliminate $\mathcal{L}_{KMI}$, $\mathcal{L}_{ETO}$ during training and find that all indicators decreased to varying degrees. This shows the effectiveness of our entity type optimization and type-guided key mention identification, and also reflects that the model's ability to understand entity types directly affects the overall performance.

w/o Type-guided EviR. In evidence extraction module, when we only use the entity feature calculated by pooling ($\varepsilon = 0$) or only use the entity feature of a specific entity type ($\varepsilon = 1$), we find that the performance on both datasets decreases. This may be because the fused features contain more semantic information from multiple angles compared to a single feature, thereby enhancing the model's expressiveness.

5.4 Complexity Analysis

To evaluate the complexity of our model, we show in Table 4 the comparison results. We find that our model is close to ATLOP in the number of trainable parameters, but less than RSMAN. This indicates that the number of parameters introduced by the type-specific entity representation we proposed is similar to the number of parameters introduced by the pooling function, but less than the number of parameters introduced by the relation-specific entity representation. Our model achieves a good balance between performance and complexity without introducing too many additional parameters. **Compared with the models with evidence retrieval**, our model shows better efficiency in both memory usage and the number of trainable parameters, demonstrating that **our approach has relatively lower complexity while maintaining higher performance**.

5.5 Comparison of Our TSER with Other Entity Representation Methods

We compare the proposed type-specific entity representation (TSER) method with other mainstream methods for entity representation. To ensure fairness, we uniformly compare the performance without evidence retrieval. There are two main types of existing methods for computing entity representation: using pooling (Zhou et al., 2021) and using relation-specific entity representation (Yu et al., 2022). The results in Table 5 show that our TSER method achieves better performance than other methods. This demonstrates that, compared to other methods, the **type-specific entity representation more effectively** captures key mention information, thereby improving the quality of entity representation.

Model	Dev		Test	
	Ign-F1	F1	Ign-F1	F1
ATLOP (LogSumExp)	59.22	61.09	59.31	61.30
RSMAN (Relation-Specific)	58.25	60.22	57.40	59.68
Ours (Type-Specific)	**59.90**	**61.73**	**59.86**	**61.69**

Table 5: Comparison of different methods for calculating entity representation on DocRED.

5.6 Generalization Analysis of TSER

To evaluate the generalization performance of the TSER module, we integrate it as a plug-in to several different backbone models. The results on DocRED, presented in Table 6, demonstrate that

integrating TSER enhances the extraction performance of the baselines. On ATLOP, the Ign-F1 and F1 scores on the test set are increased by 0.55 and 0.39, indicating that the TSER module plays a positive role in improving entity representation and relation extraction. Similarly, the DREEAM and AA models also show comparable improvements after incorporating the TSER module, further verifying the effectiveness and generalization of our idea of leveraging entity types as guidance for DocRE.

Model	Dev		Test	
	Ign-F1	F1	Ign-F1	F1
ATLOP	59.22	61.09	59.31	61.30
+TSER	**59.90**	**61.73**	**59.86**	**61.69**
AA	61.31	63.38	60.84	63.10
+TSER	**61.35**	**63.41**	**61.01**	**63.23**
DREEAM	60.51	62.55	60.03	62.49
+TSER	**61.16**	**63.31**	**60.69**	**63.02**

Table 6: Generalization analysis of the TSER module.

5.7 Weakly Supervised Generalization Ability

To further evaluate the generalization ability of ET-MIER, we conduct experiments by training the model on DocRED and testing it directly on Re-DocRED. As shown in Table 7, ET-MIER continues to outperform large language model (LLM)-based methods under this challenging setting, achieving an Ign-F1 of 58.95 and an F1 of 59.94 on the Re-DocRED test set, with a +32.80 F1 improvement over the GPT-4o + ICL (3-shot) model. These results clearly validate the robustness and generalization capability of ET-MIER, highlighting its superiority not only in fully supervised settings but also under weakly supervised generalization scenarios.

Model	F1	Ign-F1
GPT-3.5 *	4.68	-
GPT-3.5 + NLI *	9.74	-
LLaMA2 *	9.32	8.04
LLaMA2 + DP *	10.56	9.03
GPT-4o *	21.41	21.17
GPT-4o + ICL (1 shot) *	27.75	27.36
GPT-4o + ICL (3 shot) *	27.14	26.53
ET-MIER (ours) *	**59.94**	**58.95**

Table 7: Performance of LLM-based models trained on DocRED and tested on the Re-DocRED. Results marked with * are reported from Fan et al. (2024). In our model, BERT_base serves as the document encoder.

5.8 Case Studies

We conduct some case studies in Appendix B; the cases also demonstrate the positive role of the TSER, entity type optimization, and type-guided evidence retrieval modules in improving model performance.

6 Conclusions

In this paper, we propose an entity type-guided key mention identification and evidence retrieval method for DocRE. We first propose the idea of optimizing and leveraging entity types to guide the model to distinguish the importance of different mentions and obtain better type-specific entity representation based on these key mentions. We also incorporate the entity types to guide evidence retrieval, enabling the model to extract evidence sentences more comprehensively. Experimental results show that our method outperforms the existing methods while maintaining good efficiency and generalization.

Limitations

For the dependency on entity type annotations, while ET-MIER leverages entity type information to guide key mention identification and evidence retrieval, thereby improving performance on document-level relation extraction, this design introduces a reliance on the quality of entity type annotations. In cases where type labels are missing, inaccurate, or inconsistent in the training or testing data, the model's performance may be adversely affected. Although the model demonstrates a certain ability to predict types after training, such predictions are based on previously annotated data. As a result, noise in type information may propagate into the relation inference process and impact final predictions. **For the generalization in cross-domain scenarios**, although ET-MIER demonstrates strong performance on DocRE tasks, it may still pose challenges in cross-domain scenarios, where type definitions can be ambiguous or differ significantly from those seen during training. In such cases, strategies based on type-guided mention selection and evidence filtering may not generalize well, potentially limiting the model's ability to accurately extract relations when entity types differ significantly in cross-domain scenarios. We leave the challenge of improving type-robustness under domain shift as a promising direction for future work.

Acknowledgments. The authors sincerely thank the reviewers for their valuable comments, which improved the paper. The work is supported by the National Natural Science Foundation of China (62276057).

References

Tom Ayoola, Joseph Fisher, and Andrea Pierleoni. 2022. Improving entity disambiguation by reasoning over a knowledge base. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2899–2912.

GP Shrivatsa Bhargav, Dinesh Khandelwal, Saswati Dana, Dinesh Garg, Pavan Kapanipathi, Salim Roukos, Alexander Gray, and L Venkata Subramaniam. 2022. Zero-shot entity linking with less data. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1681–1697.

Qizhu Dai, Jiang Zhong, Wei Zhu, Chen Wang, Hong Yin, Qin Lei, Xue Li, and Rongzhen Li. 2023. Enhancing document-level relation extraction with relation-specific entity representation and evidence sentence augmentation. In *26th European Conference on Artificial Intelligence (ECAI)*, pages 526–532.

Julien Delaunay, Thi Hong Hanh Tran, Carlos-Emiliano González-Gallardo, Georgeta Bordea, Nicolas Sidere, and Antoine Doucet. 2023. A comprehensive survey of document-level relation extraction (2016-2022). *arXiv preprint arXiv:2309.16396*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (NAACL-HLT)*, pages 4171–4186.

Shengda Fan, Yanting Wang, Shasha Mo, and Jianwei Niu. 2024. Logicst: A logical self-training framework for document-level relation extraction with incomplete annotations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5496–5510.

Chufan Gao, Xuan Wang, and Jimeng Sun. 2024. TTM-RE: Memory-augmented document-level relation extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 443–458, Bangkok, Thailand. Association for Computational Linguistics.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 249–256.

Jia Guo, Stanley Kok, and Lidong Bing. 2023. Towards integration of discriminability and robustness for document-level relation extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 2606–2617.

Quzhe Huang, Shibo Hao, Yuan Ye, Shengqi Zhu, Yansong Feng, and Dongyan Zhao. 2022. Does recommend-revise produce reliable annotations? an analysis on missing instances in docred. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*, pages 6241–6252.

Feng Jiang, Jianwei Niu, Shasha Mo, and Shengda Fan. 2022. Key mention pairs guided document-level relation extraction. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING)*, pages 1904–1914.

Junpeng Li, Zixia Jia, and Zilong Zheng. 2023. Semi-automatic data enhancement for document-level relation extraction with distant supervision from large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5495–5505.

Xingzuo Li, Kehai Chen, Yunfei Long, and Min Zhang. 2024. Llm with relation classifier for document-level relation extraction. *arXiv preprint arXiv:2408.13889*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Chonggang Lu, Richong Zhang, Kai Sun, Jaein Kim, Cunwang Zhang, and Yongyi Mao. 2023. Anaphor assisted document-level relation extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 15453–15464.

Youmi Ma, An Wang, and Naoaki Okazaki. 2023. Dreeam: Guiding attention with evidence for improving document-level relation extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 1971–1983.

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 (ACL)*, pages 1546–1557.

Jonathan Raiman. 2022. Deeptype 2: Superhuman entity linking, all you need is type interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 36, pages 8028–8035.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022a. Document-level relation extraction with adaptive focal loss and knowledge distillation. In *Findings of the Association for Computational Linguistics (ACL)*, pages 1672–1681.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022b. Revisiting docred–addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8472–8487.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2020. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2713–2722.

Khai Phan Tran, Wen Hua, and Xue Li. 2025. Vaediff-docre: End-to-end data augmentation framework for document-level relation extraction. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING)*, pages 7307–7320.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. 2022. Sais: Supervising and augmenting intermediate steps for document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2395–2409.

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 *60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 257–268.

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021a. Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pages 14149–14157.

Huangming Xu, Fu Zhang, and Jingwei Cheng. 2025. An adaptive multi-threshold loss and a general framework for collaborating losses in document-level relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20996–21007.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021b. Discriminative reasoning for document-level relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pages 1653–1663.

Lilong Xue, Dan Zhang, Yuxiao Dong, and Jie Tang. 2024. Autore: Document-level relation extraction with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 211–220.

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 (ACL)*, pages 764–777.

Jiaxin Yu, Deqing Yang, and Shuyu Tian. 2022. Relation-specific attentions over entity mentions for enhanced document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1523–1529.

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*, pages 524–534.

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)*, pages 1630–1640.

Fu Zhang, Xinlong Jin, Jingwei Cheng, Hongsen Yu, and Huangming Xu. 2025a. Rethinking the role of llms for document-level relation extraction: a refiner with task distribution and probability fusion. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6293–6312.

Fu Zhang, Qi Miao, Jingwei Cheng, Hongsen Yu, Yi Yan, Xin Li, and Yongxue Wu. 2024. Srf: Enhancing document-level relation extraction with a novel secondary reasoning framework. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 15426–15439.

Fu Zhang, Hongsen Yu, Jingwei Cheng, and Huangming Xu. 2025b. Entity pair-guided relation summarization and retrieval in llms for document-level relation extraction. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4022–4037.

Liang Zhang, Zijun Min, Jinsong Su, Pei Yu, Ante Wang, and Yidong Chen. 2023. Exploring effective inter-encoder semantic interaction for document-level relation extraction. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5278–5286.

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)*.

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

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 (AAAI)*, volume 35, pages 14612–14620.

Xudong Zhu, Zhao Kang, and Bei Hui. 2024. Fcds: Fusing constituency and dependency syntax into document-level relation extraction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7141–7152.

A Datasets and Parameter Analysis

Datasets. We evaluate our model on two widely-adopted document-level relation extraction datasets containing entity types. DocRED (Yao et al., 2019) is the most commonly used manually annotated dataset for DocRE, built from Wikipedia and Wikidata. It contains 96 predefined relations and 6 entity types, with an average of 19.5 entities per document. However, Huang et al. (2022) finds that many entity-pair relations in DocRED are incorrectly labeled, which will introduce noise during model training and reduce model performance. To address this problem, Tan et al. (2022b) re-label the DocRED dataset and propose the Re-DocRED dataset. Compared to DocRED, this dataset provides a cleaner dev and test set. Many recent works suggest that the results on Re-DocRED should be considered as a fair comparison.

The statistics of the DocRED and Re-DocRED datasets are shown in Table 8, including the number of documents and the number of entity types in the datasets. Moreover, Table 9 gives the entity type information in the DocRED and Re-DocRED datasets.

Datasets	**DocRED**	**Re-DocRED**
# Train	3053	3053
# Dev	1000	500
# Test	1000	500
# Entity Type (T)	6	6
# Relations	97	97
Avg. # Entities	19.5	19.4

Table 8: Statistics of DocRED and Re-DocRED.

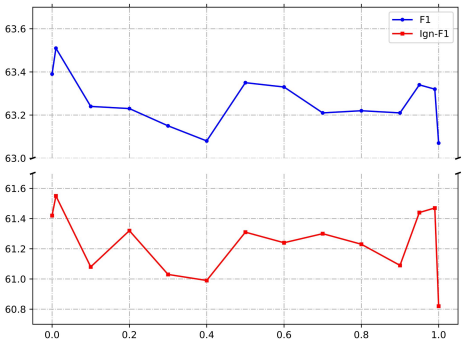
Types	**Description**
PER (Person)	People, including fictional.
ORG (Organization)	Companies, universities, institutions, political or religious groups, etc.
LOC (Location)	Geographically defined locations, including mountains, waters, etc. Politically defined locations, including countries, cities, states, streets, etc. Facilities, including buildings, museums, stadiums, hospitals, factories, airports, etc.
TIME (Time)	Absolute or relative dates or periods.
NUM (Number)	Percents, money, quantities. Products, including vehicles, weapons, etc.
MISC (Other Types)	Events, including elections, battles, sporting events, etc. Laws, cases, languages, etc.

Table 9: Entity type information in the DocRED and Re-DocRED datasets.

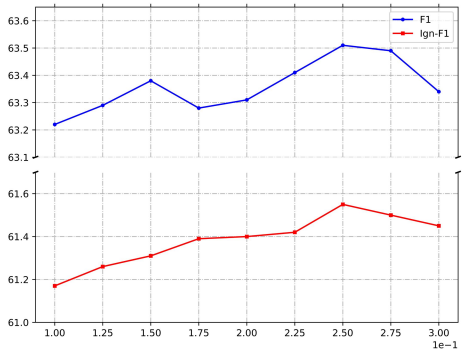
Parameters. We implement our model based on HuggingFace's Transformers (Wolf et al., 2019) and conduct experiments on a single NVIDIA V100 32GB GPU. We use BERT_base (Devlin et al., 2019) and RoBERTa_large (Liu et al., 2019) as document encoders. The embedding dimension d is 768 for BERT and 1,024 for RoBERTa. The model is trained with 30 epochs using a batch size of 4, a warmup ratio is 6e-2. We conduct grid search for the temperature parameter $\varsigma \in \{0.1, 0.2, 0.5, 1.0, 2.0\}$, $\varepsilon \in \{0, 0.01, 0.1, 0.2, ..., 1.0\}$, loss coefficients $\lambda_i \in \{0.1, 0.25, 0.3, 0.325, 0.5, 1.0, 2.0\}$, $i \in \{1, 2, 3\}$. The type optimization parameter ς in Eq. (3) is 2. The number of groups K in Eq. (15) is 64. During inference, similar to DREEAM, we set $\hat{q}_n^{(h,t)} > 0.2$ in Eq. (13) as evidence sentences. We tune all hyperparameters based on the performance of the model on the dev set, and list the key hyperparameters in Table 10.

Dataset	**DocRED**		**Re-DocRED**	
	BERT	**RoBERTa**	**BERT**	**RoBERTa**
epoch	30	30	30	30
lr_encoder	5e-5	3e-5	5e-5	3e-5
lr_classifier	1e-4	1e-4	1e-4	1e-4
batch size	4	4	4	4
warmup	6e-2	6e-2	6e-2	6e-2
ς/ε	2/0.01	2/0.1	2/0.9	2/0.9
$\lambda_1/\lambda_2/\lambda_3$	0.1/2/0.25	0.1/2/0.3	0.1/2/0.25	0.1/2/0.325

Table 10: Training parameters for DocRED and Re-DocRED datasets.



(a) ε



(b) λ_3

Figure 3: Analysis of key hyperparameters.

Analysis of Key Hyperparameters. We analyze the impact of two key hyperparameters, ε and λ_3, on model performance.

First, in the evidence retrieval module, ε controls the fusion ratio between global and local information. Figure 3(a) shows the performance of the BERT_base model on the DocRED dataset. The results indicate that the model achieves the best performance when $\varepsilon = 0.01$. This suggests that combining global and local information yields better results than using either alone (i.e., $\varepsilon = 0$ or $\varepsilon = 1$). This is because the granularity of information captured by pooling functions and by type-aware attention differs, and their combination provides complementary perspectives that enhance the model's capability.

Moreover, another important parameter λ_3 balances the weight of the loss term $\mathcal{L}_{KMI}$, which governs the model's ability to recognize key mentions under the guidance of entity types. As shown in Figure 3(b), increasing λ_3 initially improves overall performance, but performance begins to decline after reaching a peak. This is because when λ_3 is too small, the model struggles to identify entity types accurately, impairing relation extraction. Conversely, when λ_3 is too large, the model overemphasizes type information and neglects other important features, also leading to performance degradation. Therefore, tuning λ_3 appropriately is crucial for overall performance. In other words, while type-guided relation extraction benefits from entity type information, excessive reliance on it may hinder the model's ability to leverage other features, ultimately affecting its performance.

B Case Study

Figure 4 provides detailed case studies. In the cases, we conduct a detailed analysis of the model's key modules, including type-specific entity representation (TSER), entity type optimization, and type-guided evidence retrieval.

In (a), we evaluated the impact of TSER and entity type optimization modules. The analysis showed that when both modules were removed (w/o T+O), the model failed to effectively utilize document information to predict the relation between the entity pair ("Thorvald Hansen" and "Norway"), incorrectly classifying it as "NA". This result indicates that the TSER and entity type optimization modules enable the model to fully leverage entity type information, thereby identifying new potential relations (such as "P27") and enhancing its relation prediction capability. This demonstrates that the two modules strengthen the model's ability to understand the complex interactions between entities and their types.

In (b), we evaluated the role of the type-guided evidence retrieval module. When this module was removed, the model failed to retrieve all relevant evidence sentences for the entity pair ("Gifu" and "Japan"). Although the model identified some relevant information, the absence of complete evidence led to an incorrect prediction of the relation ("P131"). In contrast, when this module was included, the model could retrieve the full set of relevant evidence sentences, enabling it to correctly predict the relation ("P17"). This analysis highlights that the module not only helps the model comprehend document but also accurately extracts the complete set of evidence sentences related to an entity pair.

The case studies also demonstrate the positive role of the TSER, entity type optimization, and type-guided evidence retrieval modules in improving model performance.

T: TSER **O**: Entity Type Optimization **E**: Type-guided Evidence Retrieval

... [4]When the union of ***Norway*** and Sweden was ended in 1905, ***Norway*** sent the consul-general, ***Thorvald Hansen***, to Shanghai. ...

(a)	Head	Tail	Relation	Evidence
w/o T+O	***Thorvald Hansen*** [LOC]	***Norway*** [LOC]	NA ×	[4]
w T+O			P27 √	[4]

[0] The is a subregion of the Chūbu region and Kansai region in ***Japan*** that ... [2] ... Aichi, ***Gifu*** and Mie prefectures. [4] ... ***Gifu***, and Mie; this area is sometimes referred to as the ...

(b)	Head	Tail	Relation	Evidence
w T+O	***Gifu*** [LOC]	***Japan*** [LOC]	P131 ×	[0,2] ×
w T+O+E			P17 √	[0,2,4] √

Figure 4: Several case studies.