Document-Level Relation Extraction via Pair-Aware and Entity-Enhanced Representation Learning

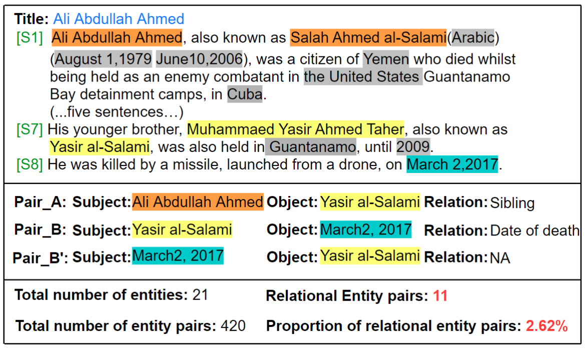
Xiusheng Huang[1,2], **Hang Yang**[1,2], **Yubo Chen**[1,2], **Jun Zhao**[1,2], **Kang Liu**[1,2,4], **Weijian Sun**[3] and **Zuyu Zhao**[3]
[1]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[2]National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[3]Huawei Technologies Co., Ltd, Shenzhen, China
[4]Beijing Academy of Artificial Intelligence, Beijing, China
huangxiusheng2020@ia.ac.cn,
{hang.yang,yubo.chen,jzhao,kliu}@nlpr.ia.ac.cn,
{sunweijian,zhaozuyu1}@huawei.com

Abstract

Document-level relation extraction aims to recognize relations among multiple entity pairs from a whole piece of article. Recent methods achieve considerable performance but still suffer from two challenges: a) the relational entity pairs are sparse, b) the representation of entity pairs is insufficient. In this paper, we propose **P**air-**A**ware and **E**ntity-**E**nhanced(PAEE) model to solve the aforementioned two challenges. For the first challenge, we design a Pair-Aware Representation module to predict potential relational entity pairs, which constrains the relation extraction to the predicted entity pairs subset rather than all pairs; For the second, we introduce a Entity-Enhanced Representation module to assemble directional entity pairs and obtain a holistic understanding of the entire document. Experimental results show that our approach can obtain state-of-the-art performance on four benchmark datasets DocRED, DWIE, CDR and GDA.

1 Introduction

Relation extraction (RE) is a primary task in the field of information extraction, which aims to identify the relationships between two entities in a document. Previous works mainly focus on sentence-level relation extraction, i.e, recognizing the relationships between entities in a sentence. However, large amounts of relationships are expressed over multiple sentences in real-world applications. According to DocRED (Yao et al., 2019), above 40.7% of the relational facts can only be extracted from multiple sentences. Therefore, it requires the model to capture complex interactions among entities in the whole document. Previous work commonly referred to this problem as document-level relation extraction which has attracted much attention recently (Nan et al., 2020; Zhou et al., 2021; Zhang et al., 2021). Although the considerable performance of these methods, there are still two critical challenges in document-level RE to be addressed.



Title: Ali Abdullah Ahmed
[S1] Ali Abdullah Ahmed, also known as Salah Ahmed al-Salami(Arabic) (August 1,1979 June10,2006), was a citizen of Yemen who died whilst being held as an enemy combatant in the United States Guantanamo Bay detainment camps, in Cuba.
(...five sentences…)
[S7] His younger brother, Muhammaed Yasir Ahmed Taher, also known as Yasir al-Salami, was also held in Guantanamo, until 2009.
[S8] He was killed by a missile, launched from a drone, on March 2,2017.

Pair_A: Subject: Ali Abdullah Ahmed **Object:** Yasir al-Salami **Relation:** Sibling
Pair_B: Subject: Yasir al-Salami **Object:** March2, 2017 **Relation:** Date of death
Pair_B': Subject: March2, 2017 **Object:** Yasir al-Salami **Relation:** NA

Total number of entities: 21 **Relational Entity pairs:** 11
Total number of entity pairs: 420 **Proportion of relational entity pairs:** 2.62%

Fig. 1: An example with entity pairs and relations from DocRED. Entity mentions only involved in these relation instances are colored, other entities in the document are highlighted in grey.

The first challenge is how to identify relational entity pairs that are **sparse** in a document. Specifically, given a document with n entities, there will be $n(n-1)$ combinations of entities to classify. However, only a few entity pairs have predefined relationships. For example, as shown in Figure 1, this document contains 21 entities with 420 potential entity pairs. However, the number of relational entity pairs is only 11, accounting for 2.62 % of the total entity pairs. According to statistics, for DocRED (Yao et al., 2019) dataset, the proportion are 3.18% and 3.11% in the train set and dev set, respectively. To further explore the impact of sparsity on performance bottlenecks, we conduct a diagnostic experiment on DocRED dataset. Utilizing previous SOTA model ATLOP (Zhou et al., 2021), we divulge the information of whether existing a predefined relationship between the entities to the model. Specifically, we just concatenate a 0-1 variable on the original representation of entity pairs, where "1" represents the entity pair exists a predefined

Proceedings of the 29th International Conference on Computational Linguistics, pages 2418–2428
October 12–17, 2022.

relationship. Experimental results show that the F1 score reaches 93.50% in dev set which is 32.20% higher than normal setting. This demonstrates that the importance of identifying the relational entity pairs when facing the sparsity problem (Wang et al., 2019a).

The second challenge is how to effectively model the representations of entity pairs. There are commonly two characteristics for entity pairs. Firstly, the **entities-scattering**, which means the entities of an entity pair may scatter across multiple sentences. Figure 1 illustrates an example from the DocRED dataset. For Pair_A, the subject $Ali\ Abdullah\ Ahmed$ and object $Yasir\ al\text{-}Salami$ are distributed in different sentences ([S1] and [S7]), which requires model to capture the long-distance dependency among entities across sentences. Secondly, the **directivity** of entity pairs, which means that the relationships of entity pairs are directional. For example, the Pair_B and Pair_B$'$ in the Figure 1, their subject and object are opposite, and the relations of them are different. Therefore, this challenge requires the model to assemble directional entity pairs and obtain a holistic understanding of the cross-sentence context. To model the representation of entity pairs, most current approaches include graph-based methods and transformer-based methods. Specifically, some methods (Christopoulou et al., 2019; Nan et al., 2020; Wang et al., 2020) construct a document graph with structured attention, dependency structures or heuristics. Meanwhile, considering the transformer can capture long-distance information, some studies (Wang et al., 2019a; Tang et al., 2020; Zhou et al., 2021) directly apply pre-trained language models without introducing graph structures. However, they directly concatenate two entities together to obtain the representation of entity pair, without considering the directivity of entity pairs and modeling the representations of entity pairs adequately.

In this paper, we propose a **P**air-**A**ware and **E**ntity-**E**nhanced (PAEE) model for document-level RE. To deal with the sparsity of entity pairs, we propose the $Pair\text{-}Aware\ Representation$(PAR) module to identify potential relational entity pairs, which constrains the relation extraction to the predicted pairs subset rather than all pairs. Furthermore, to capture the global features of triples, PAR utilizes TNet (Papadopoulos et al., 2021) to model the relation between entity pairs, unlike previous methods, PAR designs a Sliding Window Filling Strategy for filling relation matrix, which enhances the interaction between entity pairs. To effectively model the representation of entity pairs, we focus on the global interactions among sentences and entities. Specifically, we propose a $Entity\text{-}Enhanced\ Representation$(EER) module. The EER first introduces a Representation-Enhanced Encoder to facilitate the interaction between all sentences and entities. In this way, EER obtains a holistic understanding of the entire document. Then, considering that the characteristics of entities as subjects and objects are different, especially in different relationship categories, EER utilizes a Cross Matching method to assemble directional entity pairs.

Experiments on four document-level relation extraction datasets, DocRED (Yao et al., 2019), DWIE (Zaporojets et al., 2021), CDR (Li et al., 2016) and GDA (Wu et al., 2019), demonstrate that our PAEE model significantly outperforms the state-of-the-art methods. To our best knowledge, we are first to consider the sparsity and the directivity of relational entity pairs for the task.

We summarize our contributions as follows:

- To alleviate the negative impact of sparsity, we propose Pair-Aware Representation(PAR) module, which promotes the interaction between entity pairs and accurately identifies potential relational entity pairs.
- To model the representation of entity pairs better, we propose Entity-Enhanced Representation(EER) module, which is based on a Representation-Enhanced Encoder to capture the global context for the scattered entities and a Cross Matching method to assemble directional entity pairs.
- We conduct experiments on four public document-level relation extraction datasets. Experimental results demonstrate that our PAEE model can achieves state-of-the-art performance compared with baselines.

2 Methodology

Before introducing our proposed approach for PAEE in this section, we first introduce the problem definition. Given a document d and a set of entities $\{e_i\}_{i=1}^{n}$, and there are $n(n-1)$ entity

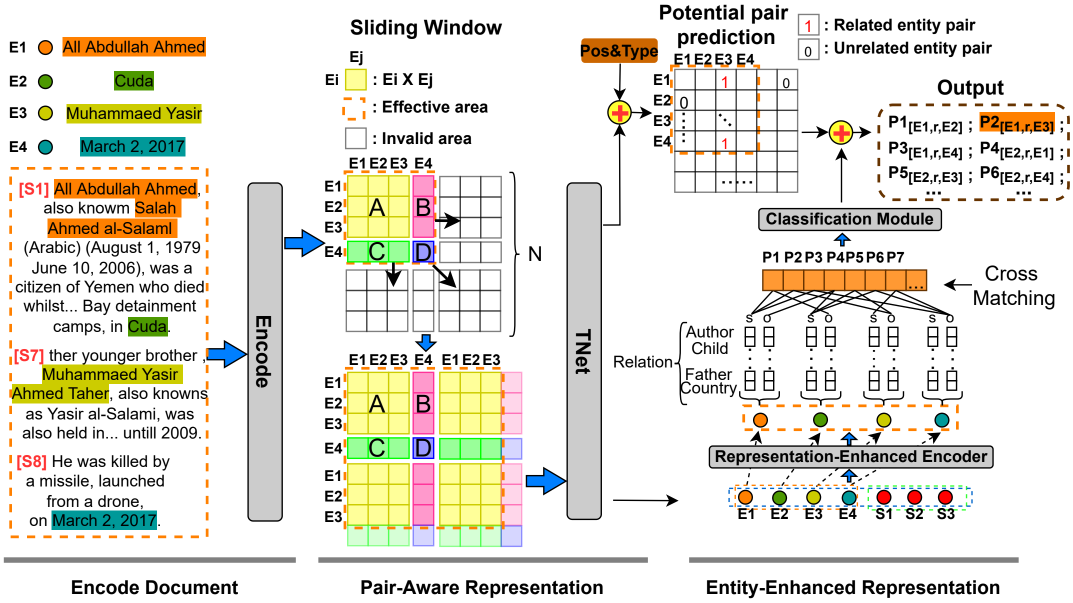

E1 All Abdullah Ahmed
E2 Cuda
E3 Muhammaed Yasir
E4 March 2, 2017
[S1] All Abdullah Ahmed, also knowm Salah Ahmed al-Salaml (Arabic) (August 1, 1979 June 10, 2006), was a citizen of Yemen who died whilst... Bay detainment camps, in Cuda.
[S7] ther younger brother , Muhammaed Yasir Ahmed Taher, also knowns as Yasir al-Salami, was also held in... untill 2009.
[S8] He was killed by a missile, launched from a drone, on March 2, 2017.
Encode
Sliding Window
Ej
Ei : Ei X Ej
: Effective area
: Invalid area
E1 E2 E3 E4
A
B
C
D
N
TNet
Pos&Type
Potential pair prediction
1 : Related entity pair
0 : Unrelated entity pair
Output
P1[E1,r,E2] ; P2[E1,r,E3] ;
P3[E1,r,E4] ; P4[E2,r,E1] ;
P5[E2,r,E3] ; P6[E2,r,E4] ;
Classification Module
P1 P2 P3 P4 P5 P6 P7
Cross Matching
Relation
Author
Child
Father
Country
s o
Representation-Enhanced Encoder
E1 E2 E3 E4 S1 S2 S3
Encode Document
Pair-Aware Representation
Entity-Enhanced Representation

Fig. 2: The overall architecture of our PAEE. Given a document, PAEE will obtain the representation of entities with the Encoder module; Then, the Pair-Aware Representation module and Entity-Enhanced Representation module are designed to get the representation of entity pairs. Finally, PAEE will obtain the relations between entities with the Classification Module.

pairs in this document. The task of document-level relation extraction is to predict a subset of relations from $R \cup \{\text{NA}\}$ between the entity pairs $(e_s, e_o)_{s,o=1...n; s\neq o}$, where R is a pre-defined set of relationships, e_s, e_o are identified as subject and object entities, respectively. The entity pairs that do not express any relation are labeled NA. In addition, the model needs to predict the label of all entity pairs $(e_s, e_o)_{s,o=1...n; s\neq o}$ at the test time. To model relation extraction between e_s and e_o, we define a $N \times N$ matrix V, where entry $V_{s,o}$ indicates the relation type between e_s and e_o. Entities in V are arranged according to their first appearance in the document. Unlike Zhang et al. (2021), we utilize the sliding window filling strategy to fill matrix V, which can enhance the interaction between entity pairs and is beneficial to relation extraction.

2.1 Encoder

Given a document $D = [x_t]_{t=1}^{l}$ with l tokens, we insert special symbols " $< e >$ " and " $< /e >$ " to mark the entity positions at the start and end of mentions. It is adapted from the entity marker technique (Zhang et al., 2017; Shi and Lin, 2019; Soares et al., 2019). We leverage the pre-trained language model as an encoder to obtain the embedding as follows:

$$H = [h_1, h_2, ..., h_l] = \text{Encoder}([x_1, x_2, ..., x_l]) \tag{1}$$

where h_i is the embedding of the token x_i. Note that some documents are longer than 512, we thus leverage a *dynamic window* to encoder whole documents (Zhou et al., 2021). We take the embedding of " $< e >$ " at the start of mentions as the mention embeddings. Then, for an entity e_i with mentions $\{m_j^i\}_{j=1}^{N_{e_i}}$, we leverage a logsumexp pooling to obtain the entity embedding $\mathbf{e}_i$:

$$\mathbf{e}_i = \log \sum_{j=1}^{N_{e_i}} \exp(\mathbf{m}_j) \tag{2}$$

This pooling accumulates signals from mentions in the document. Compared with the mean pooling, the logsumexp pooling shows better performance in the experiment. We calculate the entity-level relation matrix based on entity-to-entity relevance. Specifically, we constructed a D-dimensional feature vector $\mathbf{V}(e_s, e_o)$ to capture the relevance between entities. Note that we add the position and type information of entities to enrich the vector $\mathbf{V}(e_s, e_o)$. For intra-sentential and inter-sentential entity pairs, their position captured by a 0-1 variable *pos*.

$$a_{(s,o)} = softmax(\sum_{i=1}^{K} A_i^s \cdot A_i^o),$$
$$\mathbf{V}(e_s, e_o) = W_1 \cdot H \cdot a_{(s,o)} \tag{3}$$

where W_1 is the learnable weight matrix, $a_{(s,o)}$ is the attention weight of last layer for entity-aware attention and A_i^s refers to the tokens' importance to the i-th entity, H is the contextual embedding in Eq.1. The K is the number of head in the transformer.

2.2 Pair-Aware Representation

In this section, we propose Pair-Aware Representation(PAR) module to enhance the interaction between entity pairs and identify potential entity pairs. We build the module base on existing BERT baselines (Zhou et al., 2021; Zhang et al., 2021) and integrate other techniques to further improve the performance.

Sliding Window Filling Strategy. To capture the relevance of entity pairs, we utilize TNet (Papadopoulos et al., 2021) to expand receptive field and learn more global and local information. The TNet is a novel multi-scale hard-attention architecture that constantly adjusts the number of elements to help us focus on the related entity pairs. We take the matrix $\mathbf{V} \in R^{N \times N \times D}$ as a D-channel variable and feed it into TNet, where N is the largest number of entities, counted from all the dataset samples. However, the number of entities annotated in each document is usually different and often less than N, thus, we propose a sliding window filling strategy to fill matrix before feeding matrix $\mathbf{V}$ into TNet.

$$\mathbf{V}^{'} = \text{Sliding}(\mathbf{V}),$$
$$\mathbf{Y} = \text{TNet}(W_2 \mathbf{V}^{'}) \tag{4}$$

where $\mathbf{Y} \in R^{N \times N \times D^{'}}$ denotes the entity-level relation matrix. W_2 is the learnable weight matrix and $D^{'}$ is much smaller than D. As it shows in the Figure 2, the diagonal dots are far apart in the matrix $\mathbf{V}$, which makes their interaction poor (Ronneberger et al., 2015). Instead of previous zero filling (Ronneberger et al., 2015), we utilize the sliding window filling strategy to shorten the distance between entity pairs. Specifically, for the orange dashed window in matrix $\mathbf{V}$, we slide the window in three directions: transverse, longitudinal and oblique, then we will obtain a filled matrix $\mathbf{V}^{'}$. Furthermore, in the whole matrix $\mathbf{V}^{'}$, the spacing between dots that were originally far away was significantly shortened, which facilitates the interaction between them.

Potential Pair Prediction. This component is shown as a 0-1 distribution box in Figure 2, where "1" means potential relational entity pairs. Given a document which contains multiple entity pairs, different from previous works (Zhou et al., 2021; Zhang et al., 2021) which redundantly perform relationship classification to every entity pair, we utilize this module to predict potential relational entity pairs. Specifically, we utilize the average pooling operation (Lin et al., 2013) to obtain the representation $\mathbf{P}_{pair}$ of each entity pair, and then feed it into the binary classifier to get the potential entity pairs.

$$\mathbf{P}^{'}_{pair} = \kappa(\mathbf{P}_{pair}; \lambda; pos; sub_{emb}; obj_{emb}) \tag{5}$$

where κ and λ denote the binary classifier and threshold, sub_{emb} is the type embedding of subject in entity pairs, obj_{emb} is the type embedding of object in entity pairs. We model it as a binary classification task, and the corresponding entity pairs will be assigned with tag "1" if the probability exceeds a certain threshold λ or with tag "0" otherwise (as shown in Figure 2). By concatenating the classification results $\mathbf{P}^{'}_{pair}$ with matrix $\mathbf{Y}$ in Eq.4, we will obtain a entity-level relation matrix $\mathbf{Y}_{pair} \in R^{D^{'}+1}$ incorporating the information of candidate pairs.

2.3 Entity-Enhanced Representation

In this section, we propose a Entity-Enhanced Representation module to model the representation of entity pairs. Specifically, we introduce Representation-Enhanced Encoder to facilitate the interaction between all sentences and entities. Then, considering that the characteristics of entity as subject and object are different, especially in different relational categories, we propose Cross Matching method to assemble directional entity pairs.

Representation-Enhanced Encoder. To enable the awareness of document-level contexts for sentences and entities, we employ a Representation-Enhanced Encoder to facilitate the interaction between all sentences and entities. Formally, we can obtain the entity embedding $\mathbf{e}_i$ from Eq.2 and the embedding $[h_1, h_2, ..., h_l]$ of every token in sentence S_i^l from Eq.1, where l is the sentence length. Hence the sentence embedding S_i can be obtained by a max-pooling operation over the token sequence representation. Then we employ the Transformer (Vaswani et al., 2017) module, Representation-Enhanced Encoder, as the encoder to obtain the document-aware embedding for sentences and entities. Note that we add the sentence representation with sentence position embeddings to inform the sentence order before feeding them into the Representation-Enhanced Encoder.

$$[\mathbf{H}^e; \mathbf{H}^s] = \text{RE-Encoder}(\mathbf{e}_1 ... \mathbf{e}_{N_e}; S_1 ... S_{N_s}) \quad (6)$$

where S_i is the local representation for i-th sentence and $\mathbf{e}_i$ is the representation for i-th entity. Utilizing the Representation-Enhanced Encoder, we can obtain the document-aware entities representation $\mathbf{H}^e \in R^{N_e \times D}$, N_e is the number of entities in a document.

Cross Matching. To extract the different features of entity as subject and object respectively, we utilize the Sub-Obj layer (a Linear Layer($N_e \times D$, $2 \times N_e \times N_c$)) for feature separation. Meanwhile, we map these features to each relationship category, which enhances the interaction between entities in each relationship. For the Sub-Obj layer, we set a corresponding loss (Appendix A.1) to learn that a single entity may have several relationships . The features of entity as subject and object in each relationship can be calculated as:

$$[F_{sub}; F_{obj}] = \text{Sub-Obj}(\mathbf{H}^e) \quad (7)$$

where $F_{sub}, F_{obj} \in R^{N_e \times N_c}$ denotes the features of entities as subjects and objects respectively, N_c is the number of relationship categories and N_e is the number of entities. Meanwhile, we concatenate these features with the representation $\mathbf{H}^e$ of entities, then we will obtain $\mathbf{e}_{sub}$, $\mathbf{e}_{obj} \in R^{N_e \times (D+N_c)}$, which are the representations of entities as subjects and objects respectively.

Classification Module. Given the entity embedding $\mathbf{e}_{sub}$ and $\mathbf{e}_{obj}$ with entity-level relation matrix $\mathbf{Y}_{pair}$ in section 2.2, we map them to hidden representations z with a feedforward neural network. Then we calculate the probability of relation r by bilinear function and sigmoid activation. Formally, we obtain:

Statistics/Datasets	DocRED	DWIE	CDR	GDA
# Train	3,053	602	500	23,353
# Dev	1,000	98	500	5,839
# Test	1,000	99	500	1,000
# Relation	97	65	2	2
Avg. # entity per Doc.	19.5	14	7.6	5.4
Avg. # Ment. per Ent.	1.4	1.6	2.7	3.3

Table 1: Statistics of the experimental datasets.

$$\begin{aligned} z_s &= \tanh(\mathrm{W_s} \cdot \mathrm{e_{sub}} + \mathbf{Y}_{\mathrm{s,o}}), \\ z_o &= \tanh(\mathrm{W_o} \cdot \mathrm{e_{obj}} + \mathbf{Y}_{\mathrm{s,o}}), \\ P(r \mid \mathbf{e}_{sub}, \mathbf{e}_{obj}) &= \sigma(z_s W_r z_o + b_r) \end{aligned} \quad (8)$$

where $\mathbf{Y}_{s,o}$ is the entity-pair representation of (s, o) in matrix $\mathbf{Y}_{pair}$, σ denotes the sigmoid function, $W_s \in R^{d \times d}$, $W_o \in R^{d \times d}$, $b \in R$, and $W_r \in R^{d \times d}$ are learnable parameters.

3 Experiments

3.1 Experimental Setup

Datasets. We evaluated our method on four document-level RE datasets. The statistical results of the datasets are shown in Table 2.

- **DocRED** (Yao et al., 2019) is a large-scale document-level relation extraction dataset. It is constructed from Wikipedia articles. DocRED contains 96 relationships and 3,053/1,000/1,000 instances for training, validating and test, respectively.

- **DWIE** (Zaporojets et al., 2021) is a document-level RE dataset after processing. This dataset has 700 documents for train and 99 documents for test. The training set is then randomly split into two parts: 602 documents for train and 98 for development.

- **CDR** (Li et al., 2016) is a relation extraction dataset in the biomedical domain, which is human-annotated and aims to predict the binary interactions between Chemical and Disease concepts.

- **GDA** (Wu et al., 2019) is a large-scale dataset in the biomedical domain, which aims to predict the binary interactions between Gene and Disease concepts.

Pretrained Transformers. We initialize PAEE with three different pretrained language models including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and SciBERT (Beltagy et al., 2019).

Model	Dev			Test	
	Ign F1 (%)	F1 (%)	Rela (%)	Ign (%)	F1 (%)
BERT$_{\text{base}}$ (Wang et al., 2019b)	-	54.16	58.41	-	53.20
BERT-TS$_{\text{base}}$ (Wang et al., 2019a)	-	54.42	-	-	53.92
HIN-BERT$_{\text{base}}$ (Tang et al., 2020)	54.29	56.31	-	53.70	55.60
CorefBERT$_{\text{base}}$ (Ye et al., 2020)	55.32	57.51	-	54.54	56.96
SSAN-BERT$_{\text{base}}$ (Xu et al., 2021a)	57.03	59.19	68.37	56.06	58.41
ATLOP-BERT$_{\text{base}}$ (Zhou et al., 2021)	59.22	61.09	70.42	59.31	61.30
DocuNet-BERT$_{\text{base}}$ (Zhang et al., 2021)	59.86	61.28	70.55	59.45	61.42
PAEE-BERT$_{\text{base}}$ (Ours)	**60.38** (↑0.52)	**62.62** (↑1.34)	**74.61** (↑4.06)	**60.42** (↑0.97)	**62.98** (↑1.56)
BERT$_{\text{large}}$ (Ye et al., 2020)	56.67	58.83	67.42	56.47	58.69
CorefBERT$_{\text{large}}$ (Ye et al., 2020)	56.82	59.01	68.78	56.4	58.83
RoBERTa$_{\text{large}}$ (Ye et al., 2020)	57.14	59.22	69.23	57.51	59.62
CorefRoBERTa$_{\text{large}}$ (Ye et al., 2020)	57.35	59.43	69.77	57.9	60.25
SSAN-RoBERTa$_{\text{large}}$ (Xu et al., 2021a)	60.25	62.08	73.21	59.47	61.42
ATLOP-RoBERTa$_{\text{large}}$ (Zhou et al., 2021)	61.32	63.18	74.39	61.39	63.4
DocuNet-RoBERTa$_{\text{large}}$ (Zhang et al., 2021)	61.43	63.40	74.56	61.52	63.52
PAEE-RoBERTa$_{\text{large}}$ (Ours)	**62.44** (↑1.01)	**64.82** (↑1.42)	**79.02** (↑4.46)	**63.06** (↑1.54)	**65.09** (↑1.57)

Table 2: Main results on the development and test set of DocRED. We report the official test score on the CodaLab scoreboard with the best checkpoint on the development set. The performance of our method is followed by the improvements (↑) over the previous state-of-the-art method DocuNet.

- **BERT** employs a Transformer encoder to learn from large unlabeled text corpora and sub-word units to represent textual tokens, which contains 12 and 24 self-attention layers.
- **RoBERTa** is an improved version of BERT, which removes the Next Sentence Prediction task and adopts way larger text corpora as well as more training steps.
- **SciBERT** adopts the same model architecture as BERT, but is trained on scientific text instead. In this paper, we provide SciBERT-initialized PAEE on the two biomedical domain datasets **CDR** and **GDA**.

Implementation Detail. We used cased BERT-base, or RoBERTa-large as the encoder on DocRED and SciBERT-base on CDR and GDA. We use mixed-precision training (Micikevicius et al., 2018) based on the Apex library. Our model is optimized with AdamW (Loshchilov and Hutter, 2018) using learning rates $\in [2e{-}5, 3e{-}5, 5e{-}5, 1e{-}4]$, with a linear warmup (Goyal et al., 2018) for the first 6% steps followed by a linear decay to 0. We set the matrix size $N{=}42$ in the Figure 2 and $\lambda = 0.3$. We preprocess CDR and GDA dataset following Christopoulou et al. (2019). For GDA, we split 20% of the training set for development. For CDR, we merge the training set and dev set to train the final model after the best hyper-parameter is set. The calculation of loss will be provided in the appendix A.1. We report the mean and standard deviation of F1 on the development set by conducting 5 runs of training using different random seeds.

Evaluation. Our primary evaluation metric are **F1**, **Ign F1** (Yao et al., 2019) and **Rela**. **Ign F1** is computed by excluding relational facts that already appeared in the training set. It avoids information leakage from the training set. We propose **Rela** for evaluating the accuracy of identifying relational entity pairs. The prediction results of entity pairs are processed into two classification tasks. The relationship between entity pairs is divided into NA and non NA.

3.2 Experiment Results

We conduct experiments on four DocRE datasets to verify the effectiveness of our method PAEE.

Results on the DocRED Dataset. In the DocRED dataset, we compare PAEE with transformer-based models, including BERT$_{\text{base}}$ (Wang et al., 2019b), BERT-TS$_{\text{base}}$ (Wang et al., 2019a), CorefBERT$_{\text{base}}$ (Ye et al., 2020), HIN-BERT$_{\text{base}}$ (Tang et al., 2020), SSAN (Xu et al., 2021a) and ATLOP$_{\text{base}}$ on the DocRED dataset; and graph-based models, including GEDA (Li et al., 2020), LSR (Nan et al., 2020), GLRE (Wang et al., 2020), GAIN (Zeng et al., 2020), HeterGSAN (Xu et al., 2021b) and DocuNet (Zhang et al., 2021). Results in Table 2 shows that PAEE performs better than these methods. Our best model, PAEE built upon RoBERTa$_{\text{large}}$, is **+1.42 / +1.57 F1** better on dev/test set than DocuNet-RoBERTa$_{\text{base}}$

Model	Dev			Test		
	Ign F1 (%)	F1 (%)	Rela (%)	Ign (%)	F1 (%)	Rela (%)
CNN	37.65	47.73	56.43	34.65	46.14	55.83
LSTM	40.86	51.77	59.31	40.81	52.60	61.42
BiLSTM	40.46	51.92	59.49	42.03	54.47	64.78
GAIN	58.63	62.55	74.75	62.37	67.57	78.89
ATLOP	59.03	64.82	77.43	62.09	69.94	82.12
PAEE (Ours)	**62.05**(↑3.02)	**67.52**(↑2.70)	**82.01**(↑4.58)	**66.45**(↑4.36)	**73.10**(↑3.16)	**86.45**(↑4.33)

Table 3: Main results on the development and test set of DWIE. The performance of our method is followed by the improvements (↑) over the previous state-of-the-art method ATLOP.

Model	CDR	GDA
BRAN (Verga et al., 2018)	62.1	-
LSR (Nan et al., 2020)	64.8	82.2
DHG (Zhang et al., 2020)	65.9	83.1
GLRE (Wang et al., 2020)	68.5	-
SciBERT (Beltagy et al., 2019)	65.1	82.5
SSAN-SciBERT (Xu et al., 2021a)	68.7	83.7
ATLOP-SciBERT (Zhou et al., 2021)	69.4	83.9
DocuNet-SciBERT (Zhang et al., 2021)	76.3	85.3
PAEE-SciBERT	**78.2** (↑1.9)	**87.7** (↑2.4)

Table 4: Test F1 score (%) on CDR and GDA dataset. Our PAEE model with the SciBERT encoder outperforms the current state-of-the-art results. The performance of our method is followed by the improvements (↑) over the previous state-of-the-art method DocuNet.

Model	Ign F1	F1
PAEE-BERT$_{\text{base}}$	60.38	62.62
w/o PAR	57.67 (↓ 2.71)	59.61 (↓ 3.01)
w/o EER	59.57 (↓ 0.81)	61.53 (↓ 1.09)
w/o SW	59.72 (↓ 0.66)	61.72 (↓ 0.90)
w/o PPP	59.63 (↓ 0.75)	61.52 (↓ 1.10)

Table 5: Ablation study of PAEE on DocRED. We turn off different components of the model one at a time.

(Zhang et al., 2021), and obtains a new state-of-the-art(SOTA) result. Meanwhile, our method achieves **4.46**% improvements of Rela score on the DocRED dataset. The significant performance gain of our method over the baselines demonstrates that the proposed PAEE is very effective for this task.

Results on the DWIE Dataset. As show in Table 5, Our method improves upon the basic ATLOP model (Zhou et al., 2021) by **2.70**% and **3.16**% in term of F1 score on the Dev and Test sets of DWIE dataset, respectively. Meanwhile, our PAEE achieves **4.33**% improvements of Rela score. We attribute the improvements to that our method PAEE takes advantage of Pair-Aware Representation and Entity-Enhanced Representation, thus achieving superior performance than the previous model ATLOP.

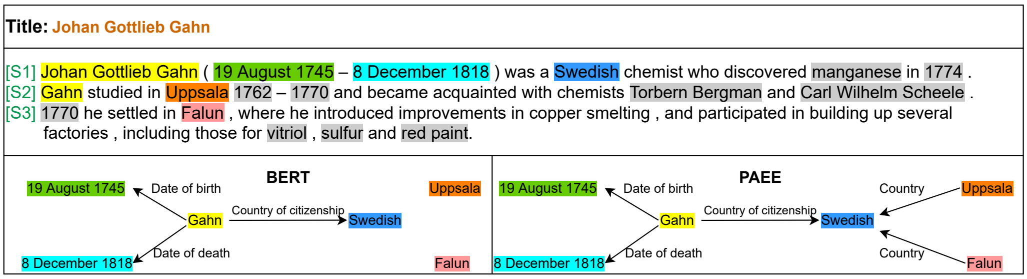
Results on the Biomedical Datasets. In the biomedical datasets, we compare PAEE with baselines including: BRAN (Verga et al., 2018), LSR (Nan et al., 2020), DHG (Zhang et al., 2020), GLRE (Wang et al., 2020), SciBERT (Beltagy et al., 2019), SSAN (Xu et al., 2021a), ATLOP (Zhou et al., 2021) and DocuNet (Zhang et al., 2021). Following ATLOP (Zhou et al., 2021), we replace the encoder with SciBERT (Beltagy et al., 2019), which is pre-trained on the scientific publication corpora. Results in Table 4 shows that PAEE has achieved new state-of-the-art with the F1 score reached to **78.2**% and **87.7**% on CDR and GDA datasets.

3.3 Ablation Study

To show the efficacy of our proposed techniques, we conduct an ablation study experiment by turning off one component at a time. 1) w/o PAR, which removes the Pair-Aware Representation module; 2) w/o EER, which removes the Entity-Enhanced Representation module, we directly splice two entities as the representation of entity pairs; 3) w/o SW, which removes the Sliding Window strategy, the previous zero filling method is used to fill the whole relationship matrix; 4) w/o PPP, which removes the Potential Pair Prediction module. We present the results of ablation study in Table 5. From the results, we can observe that:

(1) **Effectiveness of Pair-Aware Representation.** When we remove the Pair-Aware Representation module from the PAEE, the F1 score drops by 3.01% on DocRED dataset. It proves the Pair-Aware Representation module is very effective for the task.

(2) **Effectiveness of Entity-Enhanced Representation.** Compared with the model removed Entity-Enhanced Representation module, our method PAEE achieves 1.09% improvements



Title: Johan Gottlieb Gahn

[S1] Johan Gottlieb Gahn (19 August 1745 – 8 December 1818) was a Swedish chemist who discovered manganese in 1774 .
[S2] Gahn studied in Uppsala 1762 – 1770 and became acquainted with chemists Torbern Bergman and Carl Wilhelm Scheele .
[S3] 1770 he settled in Falun , where he introduced improvements in copper smelting , and participated in building up several factories , including those for vitriol , sulfur and red paint.

Fig. 3: Case study on our proposed PAEE and baseline model. Entity mentions only involved in these relation instances are colored, other entities in the document are high-lighted in grey. We utilize arrows to connect relational entity pairs.

Model	ACC	F1
BERT$_{\text{base}}$	48.83	54.16
CorefBERT	59.37	57.51
ATLOP	65.42	61.09
PAEE-BERT$_{\text{base}}$ (Our)	70.30 (↑4.88)	64.82 (↑3.73)

Table 6: The ACC means the accuracy of identifying relational entity pairs.

of F1 score on the DocRED dataset. It demonstrates that the EER module is able to effectively model the directivity of entity pairs.

(3) **Effectiveness of Sliding Window strategy.** Removing the SW, the performance drops significantly. Specifically, the F1 score drops from 62.62% to 61.72% on the DocRED dataset.

(4) **Effectiveness of Potential Pair Prediction.** When we remove the PPP module, the F1 score drops from 62.62% to 61.52%. It indicates that the performance of the model can be effectively improved by predicting potential relational entity pairs.

3.4 Discussion and Analysis

In order to explore whether the performance bottleneck of the model is effectively solved, we utilize experiments to analyze it.

The effect of PAEE on sparsity. To assess the effectiveness of PAEE on identifying relational entity pairs, we analyze it from contrast experiments, the experiments are based on the pre-training model BERT$_{\text{base}}$ and DocRED dataset. As show in Table 6, the ACC score if 70.3% which is 4.88% more than previous SOTA model ATLOP. This shows that PAEE model can effectively identify potential relational entity pairs.

The effect of Entity-Enhanced Representation(EER). To show that our EER can model the representation of entity pairs better, we divide the documents in dev set of DocRED into different groups by the proportion of relational entity pairs, and evaluate models trained with or without the EER. Experiment results are shown in Figure 4. We observe that for both models, their performance gets better when the proportion of relational entity pairs becomes larger, and the model w/ EER consistently outperforms the model w/o EER. This demonstrates that EER can model the representation of entity pairs better.

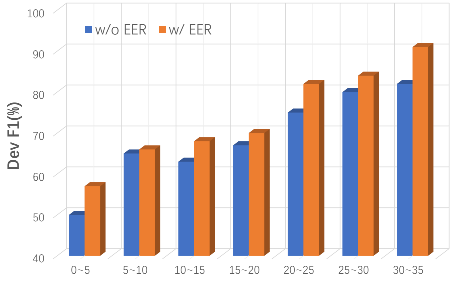


Fig. 4: Dev $F1$ score on DocRED. The x-axis refers to the proportion of relational entity pairs per document (Unit: %), the y-axis refers to the dev F1.

3.5 Case Study

We select a sample from the dev set of the DocRED dataset and conduct a case study to further illustrate the effectiveness of our model PAEE compared with the baseline. As shown in Figure 3, we notice that both BERT$_{\text{base}}$ and PAEE-BERT$_{\text{base}}$ can successfully extract the "$Country\ of\ citizenship$" relation between "$Gahn$" and "$Swedish$". However, only our PAEE-BERT$_{\text{base}}$ can deduce that the "$Country$" of "$Uppsala$" and "$Falun$" are same, namely "$Swedish$".

4 Conclusion

In this paper, we propose the **P**air-**A**ware and **E**ntity-**E**nhanced (PAEE) model. Specifically, PAEE introduces Pair-Aware Representation(PAR) module to alleviate the negative impact of sparsity, which constrains the following relation extraction to the predicted entity pairs subset rather than all pairs. In addition, PAEE also designs Entity-Enhanced Representation(EER) module to assemble directional entity pairs and obtain holistic understanding of document. Experiments on four benchmark datasets DocRED, DWIE, CDA and GDA, show that PAEE outperforms the previous methods and obtains new state-of-the-art results.

5 Acknowledgements

This work is supported by the National Key Research and Development Program of China (No.2020AAA0106400), the National Natural Science Foundation of China (No.61922085, 61976211, 62176257). This work is also supported by the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No.XDA27020200), the Youth Innovation Promotion Association CAS, and Yunnan Provincial Major Science and Technology Special Plan Projects (No.202202AD080004).

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. *arXiv preprint arXiv:1909.00228*.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2018. Accurate, large minibatch sgd: Training imagenet in 1 hour.

Bo Li, Wei Ye, Zhonghao Sheng, Rui Xie, Xiangyu Xi, and Shikun Zhang. 2020. Graph enhanced dual attention network for document-level relation extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1551–1560.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Min Lin, Qiang Chen, and Shuicheng Yan. 2013. Network in network. *arXiv preprint arXiv:1312.4400*.

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*.

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

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2018. Mixed precision training. In *International Conference on Learning Representations*.

Guoshun Nan, Zhijiang Guo, Ivan Sekulić, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. *arXiv preprint arXiv:2005.06312*.

Athanasios Papadopoulos, Pawel Korus, and Nasir Memon. 2021. Hard-attention for scalable image classification. *Advances in Neural Information Processing Systems*, 34.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

Livio Baldini Soares, Nicholas Fitzgerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905.

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6398–6407.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. Hin: Hierarchical inference network for document-level relation extraction. *Advances in Knowledge Discovery and Data Mining*, 12084:197.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 872–884.

Difeng Wang, Wei Hu, Ermei Cao, and Weijian Sun. 2020. Global-to-local neural networks for document-level relation extraction. *arXiv preprint arXiv:2009.10359*.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Wang. 2019a. Fine-tune bert for docred with two-step process. *arXiv preprint arXiv:1909.11898*.

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

Ye Wu, Ruibang Luo, Henry CM Leung, Hing-Fung Ting, and Tak-Wah Lam. 2019. Renet: A deep learning approach for extracting gene-disease associations from literature. In *International Conference on Research in Computational Molecular Biology*, pages 272–284. Springer.

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*, volume 35, pages 14149–14157.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021b. Document-level relation extraction with reconstruction. In *The 35th AAAI Conference on Artificial Intelligence (AAAI-21)*.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. Docred: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7170–7186.

Klim Zaporojets, Johannes Deleu, Chris Develder, and Thomas Demeester. 2021. Dwie: an entity-centric dataset for multi-task document-level information extraction.

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.

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. *arXiv preprint arXiv:2106.03618*.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45.

Zhenyu Zhang, Bowen Yu, Xiaobo Shu, Tingwen Liu, Hengzhu Tang, Wang Yubin, and Li Guo. 2020. Document-level relation extraction with dual-tier heterogeneous graph. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1630–1641.

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

A Appendix

A.1 Training Strategy

In the **Relationship Classification** stage, previous work (Wang et al., 2019b) observed that there is an imbalance relation distribution for RE (the relational entity pairs are sparse). To alleviate the negative impact of sparsity, Zhang et al. (2021) introduces a balanced softmax method inspired by the circle loss (Sun et al., 2020). Based on this, we design **Adaptive Softmax** loss, which introduces a addition threshold class TH, which is automatically learned in the same way as other classes. The class TH aims to separate positive classes and negative classes, hoping that the scores of the target category are all greater than s_{TH} and the scores of the non-target categories are all less than s_{TH}. Formally,

$$L_{rel} = \log(e^{s_{TH}} + \sum_{i \in \omega_{neg}} e^{s_i}) + \log(e^{-s_{TH}} + \sum_{i \in \omega_{pos}} e^{-s_i}) \tag{9}$$

In the **Potential Pair Prediction** stage, in order to match binary classification task, we design the loss as:

$$L_{pot} = -\frac{1}{n_p} \sum_{i=1}^{n_p} (y_i \log \mathbf{P}_{pair} + (1-y_i) \log(1-\mathbf{P}_{pair})) \tag{10}$$

where n_p is the size of full entity pairs set. In the **Cross Matching** stage, to capture the features of entities as subject and object respectively, we design the loss as:

$$L_{sub} = -\frac{1}{n_c n_e} \sum_{j=1}^{n_e} \sum_{i=1}^{n_c} (y_i \log F_{sub}^{j} + (1-y_i) \log(1-F_{sub}^{j})),$$

$$L_{obj} = -\frac{1}{n_c n_e} \sum_{j=1}^{n_e} \sum_{i=1}^{n_c} (y_i \log F_{obj}^{j} + (1-y_i) \log(1-F_{obj}^{j})) \tag{11}$$

where n_c is the size of full relation set, n_e is size of full entities set. The total loss is the sum of the above losses:

$$L_{total} = \alpha L_{rel} + \beta L_{pot} + \gamma \frac{L_{sub} + L_{obj}}{2} \tag{12}$$

Performance might be better by carefully tuning the weight of each sub-loss, but we just assign equal weights for simplicity (ie., $\alpha{=}\beta{=}\gamma{=}1$).