OneEE: A One-Stage Framework for Fast Overlapping and Nested Event Extraction

Hu Cao[1]*, Jingye Li[1]*, Fangfang Su[1], Fei Li[1], Hao Fei[2], Shengqiong Wu[2], Bobo Li[1], Liang Zhao[3], Donghong Ji[1]†

[1] Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, China

[2] School of Computing, National University of Singapore, Singapore

[3] Department of Computing and Mathematics, University of São Paulo, Brazil

{whucaohu, theodorelee, lifei_csnlp, dhji}@whu.edu.cn

Abstract

Event extraction (EE) is an essential task of information extraction, which aims to extract structured event information from unstructured text. Most prior work focuses on extracting flat events while neglecting overlapped or nested ones. A few models for overlapped and nested EE includes several successive stages to extract event triggers and arguments, which suffer from error propagation. Therefore, we design a simple yet effective tagging scheme and model to formulate EE as word-word relation recognition, called OneEE. The relations between trigger or argument words are simultaneously recognized in one stage with parallel grid tagging, thus yielding a very fast event extraction speed. The model is equipped with an adaptive event fusion module to generate event-aware representations and a distance-aware predictor to integrate relative distance information for word-word relation recognition, which are empirically demonstrated to be effective mechanisms. Experiments on 3 overlapped and nested EE benchmarks, namely FewFC, Genia11, and Genia13, show that OneEE achieves the state-of-the-art (SoTA) results. Moreover, the inference speed of OneEE is faster than those of baselines in the same condition, and can be further substantially improved since it supports parallel inference.[1]

1 Introduction

Event Extraction (EE) is a fundamental yet challenging task in information extraction research (Miwa and Bansal, 2016; Katiyar and Cardie, 2016; Fei et al., 2020b; Li et al., 2021b; Fei et al., 2022a). EE facilitates the development of practical applications such as knowledge graph construction (Wei et al., 2019b; Bosselut et al., 2021), biological process analysis (Miwa et al., 2013), and financial market surveillance (Nuij et al., 2013). The goal of EE is to recognize event triggers as well as the associated arguments from texts. As an example, Figure 1(a) illustrates a `Share Reduction` event including a trigger "reduced" and a subject argument "Wang Yawei".

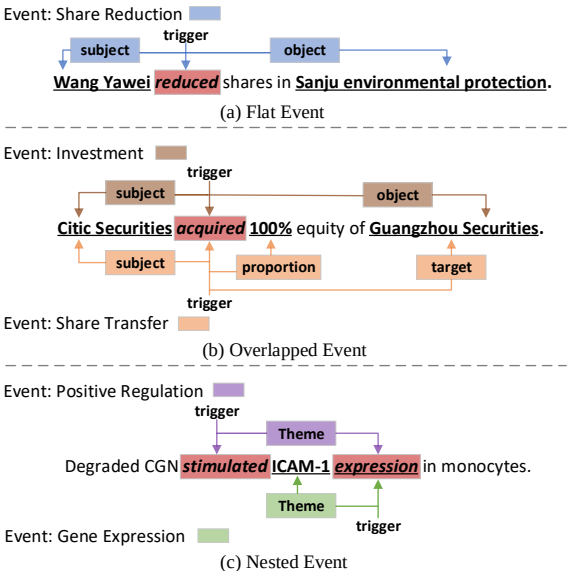


Figure 1: Examples of three kinds of events, including a flat event (a), overlapped events (b), and nested events (c). Different event mentions are denoted in distinct colors. Triggers are marked with red boxes while arguments are underlined.

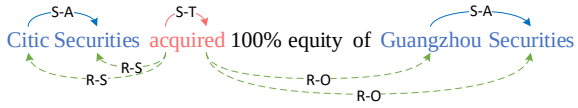
Traditional methods for EE (Li et al., 2013; Chen et al., 2015; Nguyen et al., 2016; Liu et al., 2018; Nguyen and Nguyen, 2019) regard event extraction as a sequence labeling task, assuming that event mentions do not overlap with each other. However, they neglect complicated irregular EE scenarios (i.e., overlapped and nested EE) (Fei et al., 2020a, 2021a). As exemplified in Figure 1(b), there are two overlapped events, `Investment`, and `Share Transfer`, which share the same trigger word "acquired" and the argument words "Guangzhou Securities". Figure 1(c) illustrates an example of nested events where the event `Gene`

*Equal contribution

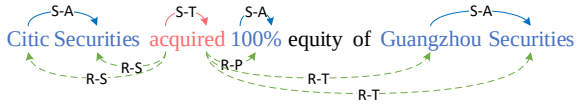
†Corresponding author

[1]The codes at https://github.com/Cao-Hu/OneEE

Proceedings of the 29th International Conference on Computational Linguistics, pages 1953–1964
October 12–17, 2022.



(a) Event: Investment



(b) Event: Transfer Share

Figure 2: Two examples to illustrate our tagging scheme. We formalize the overlapping and nested EE as word-word relation recognition, where S-T and S-A denote the relations between the head and tail boundary words of a trigger or argument, and R-S, R-O, R-T, and R-P denote the relations between the trigger word and the argument words with the roles "subject", "object", "target" and "proportion".

`Expression` is the `Theme` argument of another event `Positive Regulation`.

Prior studies for overlapped and nested EE (Yang et al., 2019; Li et al., 2020) employ pipeline-based methods that extract event triggers and arguments in several successive stages. Recently, the state-of-the-art model Sheng et al. (2021) also uses such a method that consecutively performs event type detection, trigger extraction, and argument extraction. The main problem with such a method is that the latter stage relies on the former stage, which inherently brings the error propagation problem.

To address the above issue, we present a novel tagging scheme that transforms overlapping and nested EE into word-word relation recognition. As shown in Figure 2, we design two types of relations, including the span relation (S-$\star$) and role relation (R-$\star$). S-$\star$ handles trigger and argument identification, denoting whether two words are the head-tail boundary of a trigger (T) or argument (A). R-$\star$ addresses argument role classification, indicating whether the argument plays the "$\star$" role in the event.

Based on this scheme, we further propose a one-stage event extraction model, OneEE, which mainly includes three parts. First, it adopts BERT (Devlin et al., 2019) as the encoder to get contextualized word representations. Afterward, an adaptive event fusion layer composed of an attention module and two gate fusion modules are used to obtain event-aware contextual representations for each event type. In the prediction layer, we parallelly predict the span and role relations between each pair of words by calculating distance-aware scores. Finally, event triggers, arguments, and their roles can be decoded out using these relation labels in one stage without error propagation.

We evaluate OneEE on 3 overlapped and nested EE datasets (FewFC (Zhou et al., 2021), Genia11 (Kim et al., 2011), and Genia13 (Kim et al., 2013)), and conduct extensive experiments and analyses. Our contributions can be summarized as follows:

- We design a new tagging scheme that casts event extraction as a word-word relation recognition task, providing a novel and simple solution for overlapped and nested EE.
- We propose OneEE, a one-stage model that effectively extracts word-word relations in parallel for overlapped and nested EE.
- We further present an adaptive event fusion layer to obtain event-aware contextual representations and effectively integrate event information.
- OneEE outperforms the SoTA model with regard to both the performance and inference speed.

2 Related Work

2.1 Event Extraction

Information extraction is one of the key research track in natural language processing (Miwa and Bansal, 2016; Fei et al., 2021c), among which the event extraction is the most complicated task (Chen et al., 2015; Fei et al., 2022c). Traditional EE (i.e., flat or regular EE) (Li et al., 2013; Nguyen et al., 2016; Liu et al., 2018; Sha et al., 2018; Nguyen and Nguyen, 2019) formulates EE into a sequence labeling task, assigning each token with a label (e.g., *BIO* tagging scheme). For example, Nguyen et al. (2016) uses two bidirectional RNNs to get richer representation which is then utilized to predict event triggers and argument roles jointly. Liu et al. (2018) jointly extracts multiple event triggers and arguments by introducing attention-based GCN to model the dependency graph information (Fei et al., 2021b; Li et al., 2021a; Fei et al., 2022b). However, their underlying assumption that event mentions do not overlap with each other is not always valid. Irregular EE (i.e., overlapped and nested EE) has not received much attention, which is more challenging and realistic.

Existing methods for overlapped and nested EE (Yang et al., 2019; Li et al., 2020) perform event extraction in a pipeline manner with several steps. To solve the argument overlap, Yang et al. (2019) adopts multiple sets of binary classifiers where

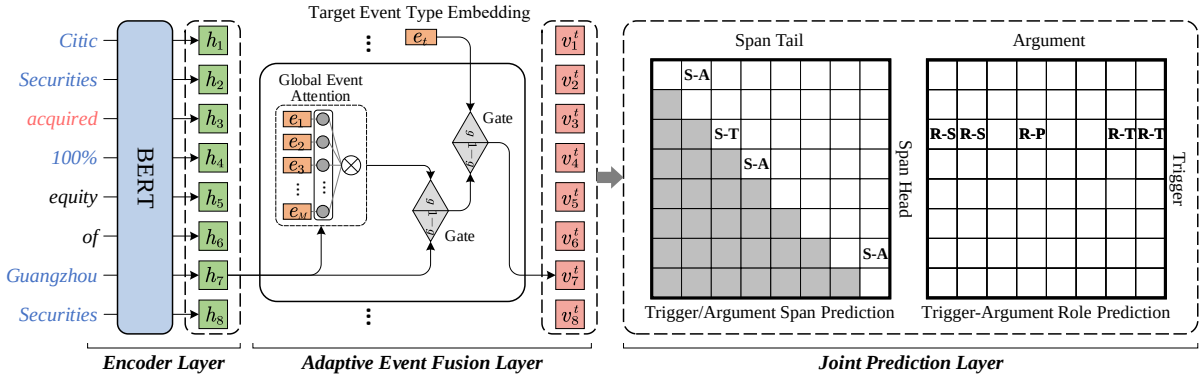


Figure 3: The architecture of our framework. Given a target event type embedding $\boldsymbol{e}_t$ of type t (e.g., transfer share), the goal of our framework is to identify its triggers, arguments, and corresponding roles in the input sentence.

each severs for a role to detect the role-specific argument spans but fails in solving trigger overlap. Except for pipeline methods, the latest attempt dealing with overlapped EE is Sheng et al. (2021) in a joint framework with cascade decoding. They are the first to simultaneously tackle all the overlapping patterns. Sheng et al. (2021) sequentially performs type detection, trigger extraction, and argument extraction, where the overlapped targets are extracted separately conditioned on the specific former prediction. Nevertheless, most of the multi-stage methods suffer from error propagation.

2.2 Tagging-based Information Extraction

Tagging scheme in the field of information extraction has been extensively investigated. Traditional sequence labeling approaches tagging each token once (e.g., *BIO*) is hard to tackle irregular information extraction (e.g., overlapped NER). Several researchers (Zheng et al., 2017) extend the BIO label scheme to adapt to more complex scenarios. However, they suffer from the label ambiguity problem due to limited flexibility. Recently, the grid tagging scheme is used in a lot of information extraction tasks, such as opinion mining (Wu et al., 2020), relation extraction (Wang et al., 2020), and named entity recognition (Wang et al., 2021), due to its characteristic of presenting relations between word pairs. For example, TPLinker (Wang et al., 2020) realizes one-stage joint relation extraction without a gap between training and inference by tagging token pairs with link labels. Inspired by these works, we design our tagging scheme to address overlapping and nested EE, which predicts relations between trigger or argument words parallelly in one stage.

Also it is noteworthy explicitly that this work inherits the recent success of the idea of word-word relation detection, as in Li et al. (2022). Li et al. (2022) propose to unify all the NER (including the flat, nested and discontinuous mentions) with a word-word modeling based on the grid tagging scheme. This work however differs from Li et al. (2022) in two folds. First, we extend the idea of the word-word tagging from NER to EE successfully, where we re-design two relation types for the nested and overlapped events. Second, from the modeling perspective, we devise an adaptive event fusion layer to fully support the one-stage (end-to-end) complex event detection, which greatly helps avoid error propagation.

3 Problem Formulation

The goal of event extraction includes extracting event triggers and their arguments. We can formalize overlapping and nested EE as follows: given an input sentence consisting of N tokens or words $X = \{x_1, x_2, \ldots, x_N\}$ and event type $e \in \mathcal{E}$, the task aims to extract the span relations $\mathcal{S}$ and the role relations $\mathcal{R}$ between each token pair (x_i, x_j), where $\mathcal{E}$ denotes the event type collection, $\mathcal{S}$ and $\mathcal{R}$ are pre-defined tags. These relations can be explained below, and we also give an example as demonstrated in Figure 2 for better understanding.

- $\mathcal{S}$: the span relation indicates that x_i and x_j are the starting and ending token of the extracted trigger span `S-T` or argument span `S-A`, where $1 \leq i \leq j \leq N$.
- $\mathcal{R}$: the role relation indicates that the argument with x_j acts the certain role `R-`$\star$ of the event with the trigger containing x_i, where $1 \leq i, j \leq N$. $\star$ indicates the role type.
- `NONE`, indicating that the word pair does not have any relation defined in this paper.

4 Framework

The architecture of our model is illustrated in Figure 3, which mainly consists of three components. First, the widely-used pre-trained language model, BERT (Devlin et al., 2019), is used as the encoder to yield contextualized word representations from the input sentences. Then, an adaptive event fusion layer consisting of an attention module and two gate modules is used to integrate the target event type embedding into contextual representations. Afterward, a prediction layer is employed to jointly extract the span relations and the role relations between word pairs.

4.1 Encoder Layer

We leverage BERT as the encoder for our model since it has been demonstrated to be one of the SoTA models for representation learning in EE. Given an input sentence $X = \{x_1, x_2, \ldots, x_N\}$, we convert each token x_i into word pieces and then feed them into a pre-trained BERT module. After the BERT calculation, each sentential word may involve vectorial representations of several pieces. Here we employ max pooling to produce word representations $\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_N\} \in \mathbb{R}^{N \times d_h}$ based on the word piece representations.

4.2 Adaptive Event Fusion Layer

Since the goal of our framework is to predict the relations between word pairs for the target event type e_t, it is important to generate event-aware representations. Therefore, to fuse the event information and contextual information provided by the encoder, we design an adaptive fusion layer. As shown in Figure 3, it consists of an attention module, modeling the interaction among events and obtaining the global event information, and two gate fusion modules for integrating the global and target event information with contextualized word representations.

Attention Mechanism Motivated by the self-attention in Transformer (Vaswani et al., 2017; Wei et al., 2019a), we first introduce an attention mechanism, of which input consists of queries, keys, and values. The output is computed as a weighted sum of the values, where the weight assigned to each value is the dot product of the query with the corresponding key. The attention mechanism can be formulated as:

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_h}})\boldsymbol{V} \,, \tag{1}$$

where $\sqrt{d_h}$ is a scaling factor, $\boldsymbol{Q}$, $\boldsymbol{K}$ and $\boldsymbol{V}$ are query, key and value tensors, represented by Eq. 4.

Gate Fusion Mechanism We design a gate fusion mechanism to integrate two kinds of features and filter the unnecessary information. The gate vector $\boldsymbol{g}$ is produced by a fully-connection layer with the sigmoid function, which can adaptively control the flow of the input:

$$\text{Gate}(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{g} \odot \boldsymbol{p} + (1 - \boldsymbol{g}) \odot \boldsymbol{q} \,, \tag{2}$$

$$\boldsymbol{g} = \sigma(\boldsymbol{W}_g[\boldsymbol{p}; \boldsymbol{q}] + \boldsymbol{b}_g) \,, \tag{3}$$

where $\boldsymbol{p}$ and $\boldsymbol{q}$ are input vectors, represented by Eq. 5 and Eq. 6. $\sigma(\cdot)$ is a sigmoid activation function, $\odot$ and $[;]$ denote element-wise product and concatenation operations, respectively. $\boldsymbol{W}_g$ and $\boldsymbol{b}_g$ are trainable parameters.

We leverage the attention mechanism to obtain the global event embeddings for each contextualized word representation. Given a set of randomly initialized event type embeddings $\boldsymbol{E} = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_M\} \in \mathbb{R}^{M \times d_h}$, where M is the number of event types, the calculation can be formulated as:

$$\boldsymbol{E}^g = \text{Attention}(\boldsymbol{W}_q\boldsymbol{H}, \boldsymbol{W}_k\boldsymbol{E}, \boldsymbol{W}_v\boldsymbol{E}) \,, \tag{4}$$

where $\boldsymbol{E}^g$ is the output of the attention mechanism, $\boldsymbol{W}_q$, $\boldsymbol{W}_k$ and $\boldsymbol{W}_v$ are learnable parameters.

To encode global event information into word representations, we adopt a gate module to fuse the contextual word representations and global event representations. After that, we employ another gate mechanism to integrate the target event type embedding and the output of the last gate module. the overall process can be formulated as:

$$\boldsymbol{H}^g = \text{Gate}(\boldsymbol{H}, \boldsymbol{E}^g) \,, \tag{5}$$

$$\boldsymbol{V}^t = \text{Gate}(\boldsymbol{H}^g, \boldsymbol{e}_t) \,, \tag{6}$$

where $\boldsymbol{e}_t \in \boldsymbol{E}$ denotes the target event type embedding, $\boldsymbol{V}^t = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_N\} \in \mathbb{R}^{N \times d_h}$ is the final event-aware word representations.

4.3 Joint Prediction Layer

After the adaptive event fusion layer, we obtain the event-aware word representations $\boldsymbol{V}^t$, which are used to jointly predict the span and role relations between each pair of words. For each word pair (w_i, w_j), we calculate a score to measure the possibility of them for the relation $s \in \mathcal{S}$ and $r \in \mathcal{R}$.

Distance-aware Score To integrate relative distance information and word pair representations, we introduce a distance-aware score function. For two vectors $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ from a sequence of representations, we combine them with corresponding position embeddings from Su et al. (2021), and then calculate the score by the dot product of them:

$$\begin{aligned} \text{Score}(\boldsymbol{p}_i, \boldsymbol{p}_j) &= (\boldsymbol{R}_i \boldsymbol{p}_i)^\top (\boldsymbol{R}_j \boldsymbol{p}_j) \\ &= \boldsymbol{p}_i^\top \boldsymbol{R}_{j-i} \boldsymbol{p}_j \ , \end{aligned} \tag{7}$$

where $\boldsymbol{R}_i$ and $\boldsymbol{R}_j$ are position embeddings of $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$, $\boldsymbol{R}_{j-i} = \boldsymbol{R}_i^\top \boldsymbol{R}_j$. Thus, we can obtain the span score c_{ij}^s and the role score c_{ij}^r of the word pair (w_i, w_j) for target event type t:

$$c_{ij}^s = \text{Score}(\boldsymbol{W}_{s1}\boldsymbol{v}_i^t, \boldsymbol{W}_{s2}\boldsymbol{v}_j^t) \ , \tag{8}$$

$$c_{ij}^r = \text{Score}(\boldsymbol{W}_{r1}\boldsymbol{v}_i^t, \boldsymbol{W}_{r2}\boldsymbol{v}_j^t) \ , \tag{9}$$

where $\boldsymbol{W}_{s1}$, $\boldsymbol{W}_{s2}$, $\boldsymbol{W}_{r1}$ and $\boldsymbol{W}_{r2}$ denote parameters. $\boldsymbol{v}_i^t$ and $\boldsymbol{v}_j^t$ are from Eq. 6.

4.4 Training Details

For the score $c_{ij}^\star$, where $\star$ denotes the relation s or r, our training target is to minimize a variant of circle loss (Sun et al., 2020) which extends softmax cross-entropy loss to figure out multi-label classification problem. In addition, we introduce a threshold score δ, noting that the scores of the pairs with relation are larger than δ, and the other pairs are less than it. The loss function can be formulated as:

$$\mathcal{L}^\star = \log(e^\delta + \sum_{(i,j)\in\Omega^\star} e^{-c_{ij}^\star}) + \log(e^\delta + \sum_{(i,j)\notin\Omega^\star} e^{c_{ij}^\star}) \ , \tag{10}$$

where $\Omega^\star$ denotes the pair set of relation $\star$, δ is set to zero.

Finally, we enumerate all event types in the selected event type set $\mathcal{E}'$ and get the total loss:

$$\mathcal{L} = \sum_{t\in\mathcal{E}'} (\sum_{s\in\mathcal{S}} \mathcal{L}^s + \sum_{r\in\mathcal{R}} \mathcal{L}^r) \ , \tag{11}$$

where $\mathcal{S}'$ is a subset sampled from $\mathcal{S}$, we detail the sampling strategy in the appendix.

4.5 Inference

During the inference period, our model is able to extract all events by parallelly injecting their event type embeddings to the adaptive event fusion layer. As shown in Figure 4, once all the tags of a certain event type are predicted by our model in one stage, the overall decoding process can be summarized as four steps: First, we get starting and ending indices of the trigger or argument. Second, we obtain the trigger and argument spans.[2] Third, we match the trigger and arguments according to the R-$\star$ relations. Finally, the event type is assigned to this event structure. Specially, we repeat the above four steps for each event type.

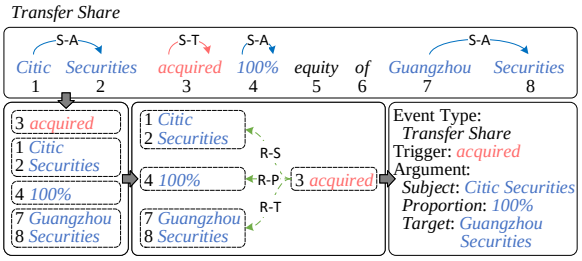


Figure 4: A decoding case of our system with four steps.

		#Ovlp.	#Nest.	#Sent.	#Events
FewFC	train	1,560	-	7,185	10,227
	dev	205	-	899	1,281
	test	210	-	898	1,332
Genia11	train	954	1,628	8,730	6,401
	dev	121	199	1,091	824
	test	125	197	1,092	775
Genia13	train	347	784	4,000	2,743
	dev	44	100	500	352
	test	42	88	500	320

Table 1: Statistics of the datasets. "Ovlp." and "Nest." denote the sentences with overlapped and nested events, respectively.

5 Experiments Settings

5.1 Datasets

As shown in Table 1, we follow previous work (Sheng et al., 2021), adopting FewFC (Zhou et al., 2021), a Chinese financial event extraction benchmark for overlapped EE. FewFC annotates 10 event types and 18 argument role classes with about 22% sentences containing overlapped events. We also experiment on two biomedical datasets for nested EE, namely Genia11 (Kim et al., 2011) and Genia13 (Kim et al., 2013), with around 18% sentences containing nested events. Genia11 annotates 9 event types and 10 argument role classes while the figures for Genia13 are 13 and 7. We split the train/dev/test as 8.0:1.0:1.0 for both of them.

5.2 Implementation Details

We employ the Chinese Bert-base model for FewFC and BioBERT (Lee et al., 2020) for Ge-

[2]Note that if two pairs with the same span relation clash in the boundaries, the pair with higher score will be selected.

		TI(%)			TC(%)			AI(%)			AC(%)		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Flat EE	BERT-softmax	89.8	79.0	84.0	80.2	61.8	69.8	74.6	62.8	68.2	72.5	60.2	65.8
	BERT-CRF	**90.8**	80.8	85.5	**81.7**	63.6	71.5	75.1	64.3	69.3	72.9	61.8	66.9
	BERT-CRF-joint	89.5	79.8	84.4	80.7	63.0	70.8	**76.1**	63.5	69.2	**74.2**	61.2	67.1
Ovlp. & Nest. EE	PLMEE	83.7	85.8	84.7	75.6	74.5	75.1	74.3	67.3	70.6	72.5	65.5	68.8
	MQAEE	89.1	85.5	87.4	79.7	76.1	77.8	70.3	68.3	69.3	68.2	66.5	67.3
	CasEE	89.4	87.7	88.6	77.9	78.5	78.2	72.8	73.1	72.9	71.3	71.5	71.4
Ours	OneEE	88.7	**88.7**	**88.7**	79.1	**80.3**	**79.7**	75.4	**77.0**	**76.2**	74.0	**72.9**	**73.4**

Table 2: Results for extracting all kinds of events on FewFC, where TI, TC, AI, AC denote trigger identification, trigger classification, argument identification, and argument classification, respectively. We run our model for 5 times with different random seeds and report the median values.

nia11 and Genia13. We adopt AdamW (Loshchilov and Hutter, 2019) optimizer with the learning rate of $2e-5$ for BERT and $1e-3$ for the other modules. The batch size is 8 and the hidden size d_h is 768. We train our model with 20 epochs on FewFC and Genia11 and 30 epochs on Genia13. All the hyper-parameters are tuned on the development set. All the event type embeddings are trained from scratch.

5.3 Evaluation Metrics

For evaluation, we follow the traditional criteria of previous work (Chen et al., 2015; Du and Cardie, 2020; Sheng et al., 2021). 1) Trigger Identification (TI): A trigger is correctly identified if the predicted trigger span matches with a golden label; 2) Trigger Classification (TC): A trigger is correctly classified if it is correctly identified and assigned to the right type; 3) Argument Identification (AI): An argument is correctly identified if its event type is correctly recognized and the predicted argument span matches with a golden label; 4) Argument Classification (AC): An argument is correctly classified if it is correctly identified and the predicted role matches any of the golden labels. We report Precision (P), Recall (R), and F measure (F1) for each of the four metrics.

5.4 Baselines

Sequence Labeling Methods for Flat EE These methods cast the EE task into a sequence labeling task by assigning each token a label. **BERT-softmax** uses BERT to get feature representations for classifying triggers and arguments. **BERT-CRF** adds the CRF layer on BERT to capture label dependencies. **BERT-CRF-joint** extends the BIO tagging scheme to joint labels of type and role as `B/I/O-type-role`, inspired by joint extraction of entity and relation (Zheng et al., 2017). All these methods are incapable to solve the overlapping problem due to label conflicts.

Multi-stage Methods for Overlapped and Nested EE These methods perform EE in several stages. **PLMEE** (Yang et al., 2019) solves the argument overlap problem by extracting role-specific argument according to the trigger predicted by the trigger extractor in a pipeline manner. **CasEE** (Sheng et al., 2021) sequentially performs type&trigger&argument extractions, where the overlapped targets are separately extracted conditioned on former predictions and all subtasks are jointly learned.

6 Experimental Results

6.1 Results of All EE

Table 2 reports the result of all methods on the overlapped EE dataset, FewFC, while Table 3 reports the results of the nested EE datasets, Genia11 and Genia13. We can observe that:

1) Our method significantly outperforms all other methods and achieves the state-of-the-art F1 score on all three datasets.

2) In comparison with sequence labeling methods, our model achieves better recall and F1 scores. Specifically, our model outperforms BERT-CRF-joint by 11.7% and 6.3% in recall and the F1 score of AC on the FewFC dataset and achieves a substantial improvement of 4.4% in F1 score of AC on two Genia datasets averagely. It shows the effectiveness of our model on overlapped and nested EE since the sequence labeling methods can only solve flat EE.

3) In comparison with multi-stage methods, our model also improves the performance on the F1 score considerably. Our model outperforms the

	TI(%)	TC(%)	AI(%)	AC(%)
• **Genia11**				
BERT-softmax	67.8	64.4	57.4	56.0
BERT-CRF	68.3	64.8	58.3	56.9
BERT-CRF-joint	67.0	64.1	60.2	58.1
PLMEE	67.3	65.5	60.7	59.4
CasEE	70.0	67.0	62.0	60.4
OneEE	**71.5**	**69.5**	**65.9**	**62.5**
• **Genia13**				
BERT-softmax	77.4	75.9	69.9	67.7
BERT-CRF	78.8	77.4	70.1	68.2
BERT-CRF-joint	77.6	75.7	71.9	68.2
PLMEE	79.3	78.3	72.1	70.7
CasEE	80.5	78.5	73.7	71.9
OneEE	**81.9**	**80.8**	**76.8**	**72.7**

Table 3: F1 scores for extracting all events on Genia11 and Genia13.

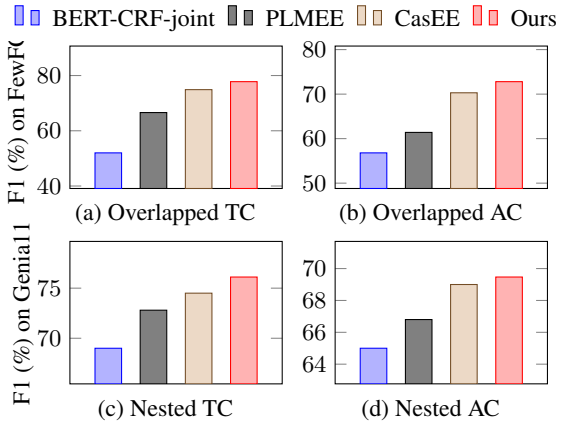


Figure 5: Results for overlapped trigger (a) and argument (b) extraction on FewFC, and nested trigger (c), and argument (d) extraction on Genia11. Note that only the sentences that contain at least one such event are used.

state-of-the-art model, CasEE, by 2.1% in the F1 score of TC on three datasets averagely. We consider this is because that the event feature is well learned by our adaptive event fusion module. Especially, our model improves 3.4% on AI and 1.6% on AC over CasEE on an average of three datasets. The results reveal the superiority of our one-stage framework which elegantly realizes overlapped and nested event extraction without error propagation.

6.2 Results of Overlapped and Nested EE

To evaluate the effectiveness of our proposed model in recognizing overlapping and nested event mentions, we further report the results on sentences containing at least one overlapping event in FewFC and sentences containing at least one nested event in Genia11, respectively.

	TI(%)	TC(%)	AI(%)	AC(%)
OneEE	88.7	79.7	76.2	73.4
w/o Attention	88.3	79.5	75.9	72.8
w/o Gate	88.4	79.3	75.3	72.6
w/o Fusion Layer	88.0	78.7	75.2	72.2
w/o Position Emb.	88.1	78.7	74.1	71.8

Table 4: Ablation studies using FewFC.

Figure 5 illustrates the results of TC and AC on overlapping and nested sentences in testing. It shows that our method outperforms other methods on overlapping and nested sentences. The reasons are mainly two-fold: 1) We solve all the overlapping patterns while BERT-CRF-joint could not handle overlapped and nested EE and PLMEE only solve the argument overlap. 2) Our one-stage model outperforms CasEE because we effectively learn event-aware representations and extract word-word relations in parallel, while CasEE performs in three sequential steps with error propagation.

6.3 Effects of the Modules in the Fusion Layer

To verify the effectiveness of each component, we conduct ablation studies on the FewFC dataset, as shown in Table 4. First, without the attention mechanism, we observe slight performance drops. By replacing the gate mechanism with an addition operation, the performance also decreases to a small degree. Furthermore, a significant drop can be found when the adaptive event fusion layer is substituted by addition, which indicates the usefulness of event representation and context. Finally, removing the position embeddings results in a remarkable drop on all F1 scores, especially 1.6% in the F1 score of AC, which suggests that the information of positions is essential to recognize word-word relations.

6.4 Effect of the Distance-aware Tag Prediction

In this section, we investigate the effect of position embeddings for the prediction layer of OneEE. We divide the arguments in the test set of FewFC into 6 groups according to their distance from corresponding triggers and report the recall scores of the model with and without position embeddings. As shown in Figure 6, the AC recall declines as the distance between trigger and argument in an

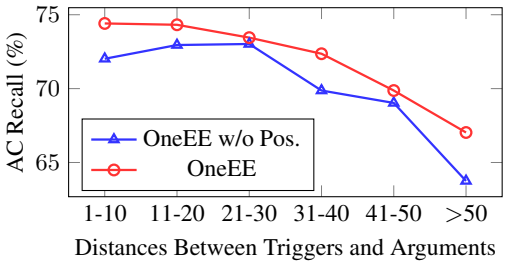


Figure 6: FewFC results of extracting triggers and arguments with different distances. The red line denotes that position embeddings are used while the blue line not.

Model	Stage	#Param.	Speed (sent/s)	Ratio
PLMEE	Two	204.6M	19.8	×1.0
CasEE	Three	120.7M	62.3	×3.1
OneEE	One	114.2M	79.4	×4.0
OneEE†	One	114.2M	186.5	×9.4

Table 5: Parameter number and inference speed comparisons on FewFC. All models are tested with batch size 1, † denotes that the model is tested with batch size 8. The ratio denotes the multiple of the speed increase with regard to PLMEE.

event go up. This indicates that it is more difficult for the model to detect roles correctly if the distance is longer in an event. Furthermore, the model with position embeddings outperforms another one without position embeddings, revealing that the relative distance information is beneficial for event extraction.

6.5 Parameter Number & Efficiency Comparisons

Table 5 lists the stage numbers, parameter numbers, and inference speeds of two baselines and our model. For a fair comparison, all of these models are implemented using PyTorch and tested using the NVIDIA RTX 3090 GPU, where the batch size is set as 1. As seen, PLMEE has 2 times as many parameters as the other two models, due to the utilization of two BERT-based modules for each stage. Moreover, the inference speed of our model is about 3 times faster than that of PLMEE (Yang et al., 2019) and 0.3 times faster than that of CasEE (Sheng et al., 2021), which verifies the efficiency of our model. Last but not least, when the batch size is set as 8, the inference speed of our model is 9.4 times as fast as that of PLMEE, which also demonstrates the advantage of our model, that is, it supports parallel inference. In one word, our model leverages fewer parameters but achieves better performance and faster inference speed.

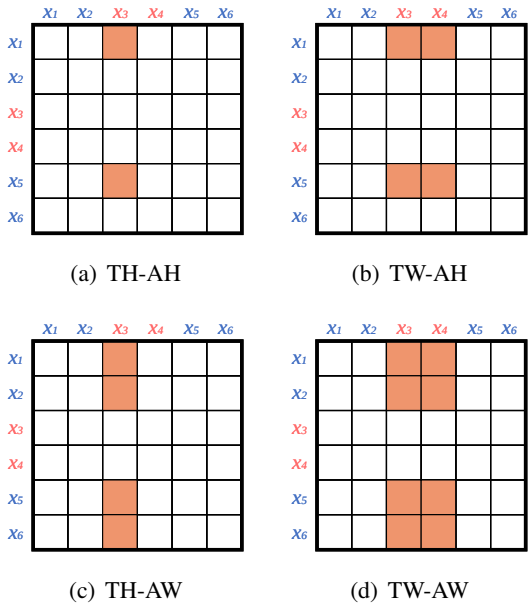


Figure 7: Four kinds of role label strategies. The goal is to predict the relation between trigger head and argument head (a), trigger word and argument head (b), trigger head and argument word (c), and trigger word and argument word (d).

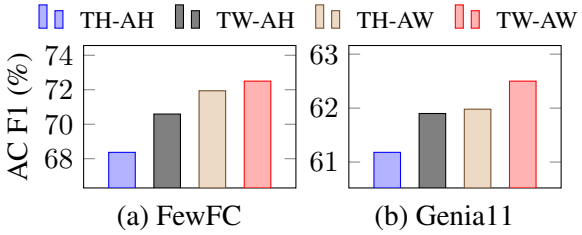


Figure 8: Results of AC with different role label strategies on FewFC (a) and Genia11 (b) datasets.

6.6 Analysis of 4 Role Label Strategies

In this section, we investigate the effect of the role strategies for AC performance. As shown in Figure 7, we introduce 4 different strategies to predict the role relation between trigger and argument: the role labels only exist in 1) trigger and argument head pairs (TH-AH), 2) trigger word and argument head pairs (TW-AH), 3) trigger head and argument word pairs (TH-AW), and 4) trigger and argument word pairs (TW-AW). The results of our model with 4 strategies are demonstrated in Figure 8. We can learn that TW-AW achieves the best results against all other strategies on both FewFC and Genia11 datasets. It is largely due to that its labels are denser than other strategies.

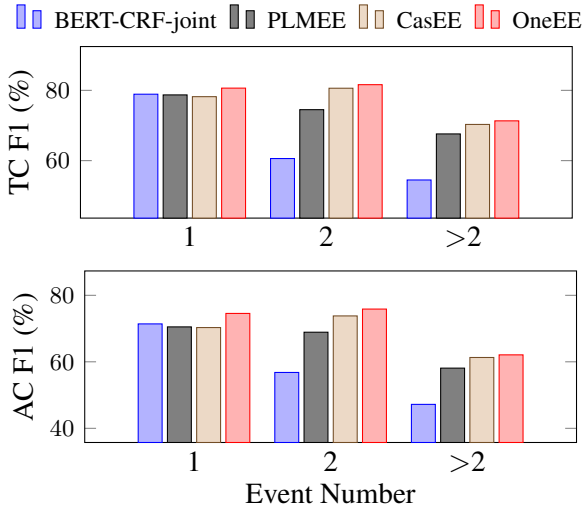


Figure 9: Results of different event numbers on FewFC.

6.7 Analysis of the Event Number

We further investigate the effect of the event number for EE, and the results are shown in Figure 9. We can observe that BERT-CRF-joint, PLMEE, and CasEE achieve similar performances on single-event sentences, while CasEE outperforms PLMEE and BERT-CRF-joint on the sentences with multiple events. Most importantly, our system achieves the best results against all other baselines for different event numbers, indicating the advances of our proposed method.

7 Conclusion

In this paper, we propose a novel one-stage framework based on word-word relation recognition to address overlapped and nested EE concurrently. The relations between word pairs are pre-defined as the word-word relations within a trigger or argument and cross a trigger-argument pair. Moreover, we propose an efficient model that consists of an adaptive event fusion layer for integrating the target event representation, and a distance-aware prediction layer for identifying all kinds of relations jointly. Experimental results show that our proposed model achieves new SoTA results on three datasets and faster speed than the SoTA model. Through ablation studies, we find that the adaptive event fusion layer and distance-aware prediction layer are effective in improving the model performance. In future work, we will extend our method to other structured prediction tasks, such as structured sentiment analysis and overlapped entity relation extraction.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62176187), the National Key Research and Development Program of China (No. 2017YFC1200500), the Research Foundation of Ministry of Education of China (No. 18JZD015), the Youth Fund for Humanities and Social Science Research of Ministry of Education of China (No. 22YJCZH064), the General Project of Natural Science Foundation of Hubei Province (No.2021CFB385). L Zhao would like to thank the support from Center for Artificial Intelligence (C4AI-USP), the Sao Paulo Research Foundation (FAPESP grant #2019/07665-4), the IBM Corporation, and China Branch of BRICS Institute of Future Networks.

References

Antoine Bosselut, Ronan Le Bras, and Yejin Choi. 2021. Dynamic neuro-symbolic knowledge graph construction for zero-shot commonsense question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4923–4931.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 167–176.

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 North American Chapter of the Association for Computational Linguistics*, pages 4171–4186.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 671–683.

Hao Fei, Donghong Ji, Bobo Li, Yijiang Liu, Yafeng Ren, and Fei Li. 2021a. Rethinking boundaries: End-to-end recognition of discontinuous mentions with pointer networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12785–12793.

Hao Fei, Fei Li, Bobo Li, and Donghong Ji. 2021b. Encoder-decoder based unified semantic role labeling with label-aware syntax. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12794–12802.

Hao Fei, Jingye Li, Shengqiong Wu, Chenliang Li, Donghong Ji, and Fei Li. 2022a. Global inference with explicit syntactic and discourse structures for dialogue-level relation extraction. In *Proceedings*

of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI, pages 4082–4088.

Hao Fei, Yafeng Ren, and Donghong Ji. 2020a. Boundaries and edges rethinking: An end-to-end neural model for overlapping entity relation extraction. *Information Processing & Management*, 57(6):102311.

Hao Fei, Shengqiong Wu, Yafeng Ren, and Meishan Zhang. 2022b. Matching structure for dual learning. In *Proceedings of the International Conference on Machine Learning, ICML*, pages 6373–6391.

Hao Fei, Shengqiong Wu, Meishan Zhang, Yafeng Ren, and Donghong Ji. 2022c. Conversational semantic role labeling with predicate-oriented latent graph. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI*, pages 4089–4095.

Hao Fei, Meishan Zhang, and Donghong Ji. 2020b. Cross-lingual semantic role labeling with high-quality translated training corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7014–7026.

Hao Fei, Meishan Zhang, Bobo Li, and Donghong Ji. 2021c. End-to-end semantic role labeling with neural transition-based model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12803–12811.

Arzoo Katiyar and Claire Cardie. 2016. Investigating LSTMs for joint extraction of opinion entities and relations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 919–929.

Jin-Dong Kim, Yue Wang, Toshihisa Takagi, and Akinori Yonezawa. 2011. Overview of genia event task in bionlp shared task 2011. In *Proceedings of BioNLP shared task 2011 workshop*, pages 7–15.

Jin-Dong Kim, Yue Wang, and Yamamoto Yasunori. 2013. The genia event extraction shared task, 2013 edition-overview. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 8–15.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Fayuan Li, Weihua Peng, Yuguang Chen, Quan Wang, Lu Pan, Yajuan Lyu, and Yong Zhu. 2020. Event extraction as multi-turn question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (Fingdings)*, pages 829–838.

Fei Li, ZhiChao Lin, Meishan Zhang, and Donghong Ji. 2021a. A span-based model for joint overlapped and discontinuous named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 4814–4828.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10965–10973.

Jingye Li, Kang Xu, Fei Li, Hao Fei, Yafeng Ren, and Donghong Ji. 2021b. MRN: A locally and globally mention-based reasoning network for document-level relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1359–1370.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 73–82.

Xiao Liu, Zhunchen Luo, and He-Yan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations*.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using LSTMs on sequences and tree structures. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1105–1116.

Makoto Miwa, Tomoko Ohta, Rafal Rak, Andrew Rowley, Douglas B Kell, Sampo Pyysalo, and Sophia Ananiadou. 2013. A method for integrating and ranking the evidence for biochemical pathways by mining reactions from text. *Bioinformatics*, 29(13):44–52.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*, pages 300–309.

Trung Minh Nguyen and Thien Huu Nguyen. 2019. One for all: Neural joint modeling of entities and events. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6851–6858.

Wijnand Nuij, Viorel Milea, Frederik Hogenboom, Flavius Frasincar, and Uzay Kaymak. 2013. An automated framework for incorporating news into stock trading strategies. *IEEE TKDE*, 26(4):823–835.

Lei Sha, Feng Qian, Baobao Chang, and Zhifang Sui. 2018. Jointly extracting event triggers and arguments by dependency-bridge rnn and tensor-based argument interaction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5916–5923.

Jiawei Sheng, Shu Guo, Bowen Yu, Qian Li, Yiming Hei, Lihong Wang, Tingwen Liu, and Hongbo Xu. 2021. Casee: A joint learning framework with cascade decoding for overlapping event extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (Findings)*, pages 164–174.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced transformer with rotary position embedding. *CoRR*, abs/2104.09864.

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 Conference on Computer Vision and Pattern Recognition*, pages 6397–6406.

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 *Proc. of NeurIPS*, pages 5998–6008.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of the International Conference on Computational Linguistics*, pages 1572–1582.

Yucheng Wang, Bowen Yu, Hongsong Zhu, Tingwen Liu, Nan Yu, and Limin Sun. 2021. Discontinuous named entity recognition as maximal clique discovery. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 764–774.

Yinwei Wei, Xiang Wang, Weili Guan, Liqiang Nie, Zhouchen Lin, and Baoquan Chen. 2019a. Neural multimodal cooperative learning toward micro-video understanding. *IEEE Transactions on Image Processing*, 29:1–14.

Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019b. Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1437–1445.

Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for end-to-end fine-grained opinion extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (Findings)*, pages 2576–2585.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 5284–5294.

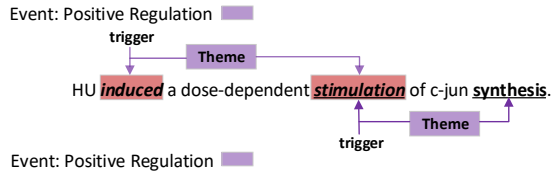
Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1227–1236.

Yang Zhou, Yubo Chen, Jun Zhao, Yin Wu, Jiexin Xu, and Jinlong Li. 2021. What the role is vs. what plays the role: Semi-supervised event argument extraction via dual question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14638–14646.

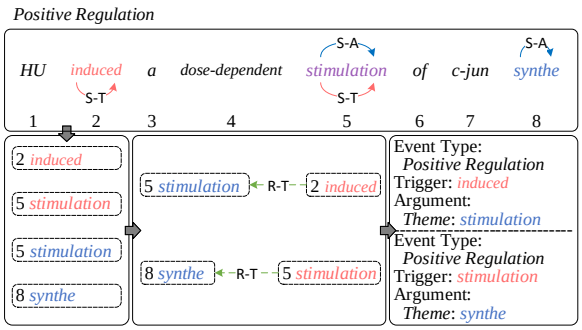
A Parallel Training with Sampling

We parallelly inject multiple target event type embeddings at the adaptive event fusion layer during training period, which results in huge computation resources. To this end, we use a subset $\mathcal{E}'$ to replace $\mathcal{E}$ for each sample, where the number of $\mathcal{E}'$ is K. It consists of one positive event type (the event type annotated in the sample) and $K-1$ negative event types selected randomly from the event types that does not appear in the sample. In other words, we inject K different event type embeddings into the gate module of Eq. 6 simultaneously. If there is no positive event type in the sample, we will select K negative event types.

B Decoding for Nested EE



(a) Nested Event Example



(b) Nested EE Decoding

Figure 10: Example of nested event (a) and its decoding process (b).

In the manuscript, we have already shown the decoding process of our model for overlapped EE in Section 4.5. Due to page limitation, we show an example of nested in Figure 10(a). We also demonstrate its decoding process in Figure 10(b), which is the same as the overlapped EE decoding.

C Analysis of the Event Sampling Number

To further analyze the effect of sampling number K and the sampling strategy, we also evaluate our model with positive and negative sampling and random sampling and compare them with different sampling numbers. Figure 11 shows the TC F1 change trend as the number of sampling increases. As seen, both two models with 6 event type samplings achieve the best performance, compared with the other sampling numbers. Specifically, our model with one positive sampling and $K-1$ negative samplings outperforms the model with K randomly selected samplings when K is less than 7, which demonstrates that our sampling strategy is helpful for the model training.

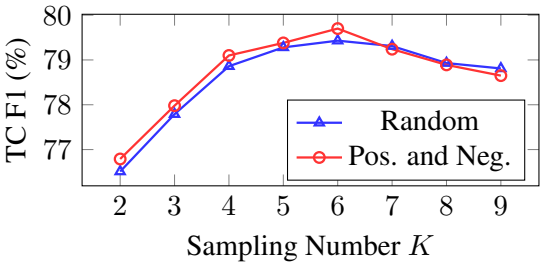


Figure 11: Results on different sampling numbers of random sampling, and positive and negative sampling.