MHGEE: Event Extraction via Multi-granularity Heterogeneous Graph

Mingyu Zhang[1,2], Fang Fang[1,2](✉), Hao Li[1,2], Qingyun Liu[1,2], Yangchun Li[3], and Hailong Wang[3]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{zhangmingyu,fangfang0703,lihao1998,liuqingyun}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Chinese Academy of Cyberspace Studies, Beijing, China

Abstract. Event extraction is a key task of information extraction. Existing methods are not effective due to two challenges of this task: 1) Most of previous methods only consider a single granularity information and they are often insufficient to distinguish ambiguity of triggers for some types of events. 2) The correlation among intra-sentence and inter-sentence event is non-trivial to model. Previous methods are weak in modeling interdependency among the correlated events and they have never modeled this problem for the whole event extraction task. In this paper, we propose a novel Multi-granularity Heterogeneous Graph-based event extraction model **(MHGEE)** to solve the two problems simultaneously. For the first challenge, MHGEE constructs multi-granularity nodes, including word, entity and context and captures interactions among nodes by R-GCN. It can strengthen semantic and distinguish ambiguity of triggers. For the second, MHGEE uses heterogeneous graph neural network to aggregating the information of relevant events and hence capture the interdependency among the events. The experiment results on ACE 2005 dataset demonstrate that our proposed MHGEE model achieves competitive results compared with state-of-the-art methods in event extraction. Then we demonstrate the effectiveness of our model in ambiguity of triggers and event interdependency.

Keywords: Event extraction · Heterogeneous graph · R-GCN

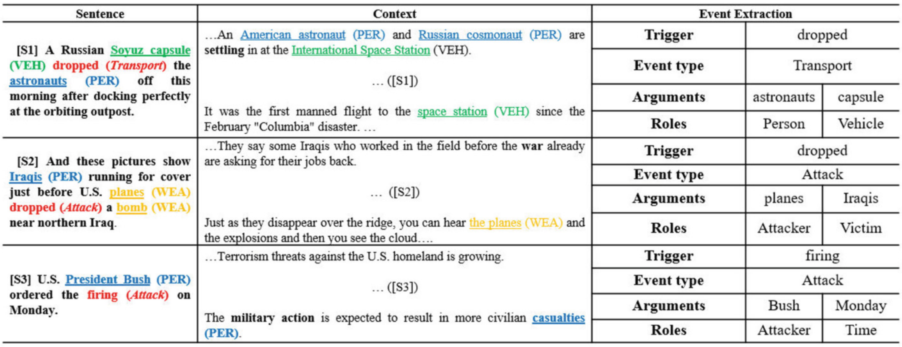
1 Introduction

Generally, event extraction(EE) tasks consist of two subtasks. **Event detection** aims to identify and classify event triggers, and **argument extraction** aims to

Supported by the National Key R&D Program of China (2021YFB3101300) and the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDC02030000).

D. Groen et al. (Eds.): ICCS 2022, LNCS 13350, pp. 473–487, 2022.
https://doi.org/10.1007/978-3-031-08751-6_34

identify arguments of event and label their roles. In Fig. 1, event detection task aims to identify the trigger "dropped" and event type "Conflict: Attack", then event argument extraction task aims to identify the arguments "U.S. planes" and "Iraqis", and their roles "Attacker" and "Victim".



Sentence	Context	Event Extraction		
[S1] A Russian Soyuz capsule (VEH) dropped (*Transport*) the astronauts (PER) off this morning after docking perfectly at the orbiting outpost.	…An American astronaut (PER) and Russian cosmonaut (PER) are **settling** in at the International Space Station (VEH). … ([S1]) It was the first manned flight to the space station (VEH) since the February "Columbia" disaster. …	**Trigger**	dropped	
		Event type	Transport	
		Arguments	astronauts	capsule
		Roles	Person	Vehicle
[S2] And these pictures show Iraqis (PER) running for cover just before U.S. planes (WEA) dropped (*Attack*) a bomb (WEA) near northern Iraq.	…They say some Iraqis who worked in the field before the **war** already are asking for their jobs back. … ([S2]) Just as they disappear over the ridge, you can hear the planes (WEA) and the explosions and then you see the cloud….	**Trigger**	dropped	
		Event type	Attack	
		Arguments	planes	Iraqis
		Roles	Attacker	Victim
[S3] U.S. President Bush (PER) ordered the firing (*Attack*) on Monday.	…Terrorism threats against the U.S. homeland is growing. … ([S3]) The **military action** is expected to result in more civilian **casualties (PER)**.	**Trigger**	firing	
		Event type	Attack	
		Arguments	Bush	Monday
		Roles	Attacker	Time

Fig. 1. Example documents in ACE 2005 corpus. Triggers and event types are marked in red. Arguments and roles are marked in other colors. The event extraction results of the three sentences are on the far right of the figure. (Color figure online)

There are a lot of research works about sentence-level event extraction, and they still face two critical challenges. 1) **Ambiguity of Triggers**: A word will express different meanings in different sentence so as it will trigger different events. In Fig. 1, both S1 and S2 contain the word "dropped". It would be quite challenging to detect the word "dropped" trigger an "Transport" event in S1 and trigger an "Attack" event in S2 without considering more information with different granularities, such as argument role (Weapon, Place , etc.) and context information. Only the event is detected correctly, can the argument in the event be extracted better. Therefore, how to distinguish different event semantics by a comprehensive understanding of information with different granularities is crucial for improving the accuracy of event extraction. 2) **Event Interdependency**: A sentence may express several correlated events simultaneously. For example, in the event mention "Three people plus the bomber were killed, and at least 30 others were hurt.", a "Die" event is triggered by "killed", and an "Injure" event is triggered by "hurt". This kind of event co-occurrence is also called multiple-event in one sentence. And it is common in ACE 2005 corpus. According to the previous statistics by [9], nearly 27% sentences have more than one events. And these events are often associated with each other, having similar event types with the same roles of arguments. Only modeling the interdependency among them, which is the fundamental to successful extraction, can we extract all events in a sentence correctly, so as to prompt the effect of event extraction. Therefore, how to effective model on such interdependency among the correlated events is one of the key challenges in event extraction.

For **challenge 1)**, most of the existing methods only consider a single granularity information [1,9,11,17,18], especially the inter-sentence information, using entity type or dependency tree. These methods try to make the best of the sentence granularity information to distinguish semantics. Unfortunately, ambiguity can't be solved only by inter-sentence information in many cases. For example, in Fig. 1, it is impossible to distinguish the event type "Attack" or "End_Position" by using entity information in S3. It requires context information "military action" and "casualties". Therefore, we should comprehensively handle the information with different granularities to distinguish different semantic. [2,7] use document granularity information, but bringing a lot of redundant information in the document. Meanwhile, they also neglect sufficient understanding of sentence granularity information. Because of that, there is no obvious improvement in the effect and no method to consider the influence of argument in event, especially roles. For **challenge 2)**, some methods use recurrent neural network to remember previous correlated events [11,13], but they still have the long-distance dependence problem. Another kind of method is graph neural network (GNN), which can effectively model the interdependency among nodes [3,9,12,17]. In order to model the interdependency of events, they construct a graph of word nodes in sentence by dependency tree. But they only consider the words of sentence and are used in the event detection subtask [3,12,17]. That is to say, they not only neglect the key multi-granularity information mentioned above, but also neglect another argument extraction subtask in event extraction and the interaction between the two subtasks.

In order to solve the above problems, inspired by [14], we propose a **M**ulti-granularity **H**eterogeneous **G**raph model for sentence-level **E**vent **E**xtraction **(MGHEE)** to complete the task of event detection and argument extraction simultaneously. Unlike previous works which only take words as nodes, MHGEE contains another two types of nodes with different granularities: entity and context. Besides, we construct six types of edges. We design three type of nodes by considering the nearest context of the sentence and inter-sentence information to learn multi-granularity semantic information. Then we use R-GCN to enable rich interactions among nodes, so as to distinguish the semantic information of the same trigger word in different events. At the same time, our model also constructs heterogeneous graph to model intra-sentence and inter-sentence event interdependency by aggregating the information of relevant events in the same sentence or context, so as to solve the challenge of multiple events in one sentence. The contributions of this paper can be summarized as follows:

- We propose a novel event extraction model based on Multi-granularity Heterogeneous Graph (MHGEE). Our MGHEE designs multi-granularity nodes and enables rich interactions among nodes by R-GCN, which strengthens the semantic and helps distinguish ambiguity of triggers.
- We are the first to construct heterogeneous graph for whole event extraction. Our MHGEE can model intra-sentence and inter-sentence event interdependency and capture multiple events in one sentence effectively

– Experiments on the ACE 2005 dataset show that our model outperforms the previous SOTA models by nearly 5% F1 on the trigger identification and 2% F1 on the argument identification.

2 Related Work

From the perspective of text, event extraction can be divided into sentence-level event extraction and document-level event extraction. And sentence-level event extraction can be divided into extraction and generation methods. Our work focuses on sentence-level event extraction. We will classify the relevant works based on deep learning from the perspective of the methods used.

Event extraction models based on basic neural network have been widely used to extract features automatically, such as convolutional neural networks (CNN) [1], recurrent neural networks (RNN) [11,13].

Some works operated BERT as the pretrained language model [4,10,16] in recent several years, since BERT has been proven its validity to improve the performance of downstream natural language processing tasks including event extraction.

And with the application of GNN in various fields of natural language processing, some researchers propose to transform the syntactic dependency tree, which contains syntactic information and plays an important role in event extraction, into a graph and employ GCN [5] to conduct event detection through information propagation over the graph [3,12,17]. These works only consider event detection task and ignore argument information.

However, the above existing methods, all focus on the single granularity information in sentence-level, neglecting the multiple granularity information aggregation across sentences. Considering that adjacent sentences from context also store some relevant event information to solve the above challenges, these methods will not integrate multiple granularity information, which would enhance the event signals of the sentence that triggers belong to.

3 Approach of MHGEE Model

Our MHGEE model consists of the following four modules: 1) **Input Layer**: we aim to get initialization vector representations of words, entities, and contexts; 2) **Graph Construction**: we build multi-granularity heterogeneous graph, including three types of nodes and six types of edges; 3) **Information Aggregation over MHG**: we use R-GCN algorithm with gating mechanism to propagate information among multi-granularity information sources, so as to enhance information flow from context and entity nodes for event extraction; 4) **Classification Layer**: we obtain the final embedding representation of words and entities, and get trigger candidates of certain types from trigger labels in BIO form annotation schema, then predict the roles that each entity plays in such events after aggregating word embedding representations to trigger candidate vector t_i and entity vector e_i. Figure 1 gives the architecture of MHGEE model.

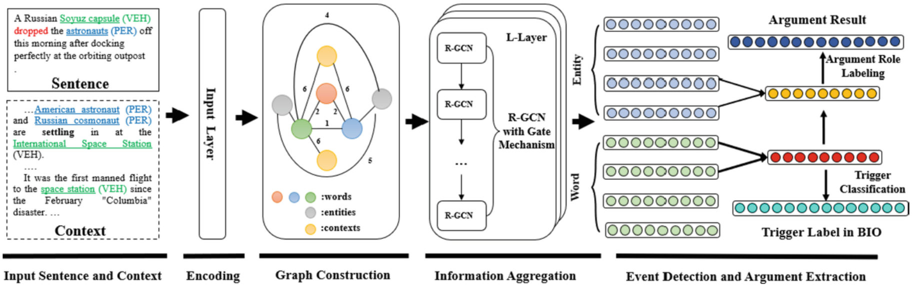

A Russian Soyuz capsule (VEH) dropped the astronauts (PER) off this morning after docking perfectly at the orbiting outpost .
Sentence
…American astronaut (PER) and Russian cosmonaut (PER) are settling in at the International Space Station (VEH).
….
It was the first manned flight to the space station (VEH) since the February "Columbia" disaster. …
Context
Input Layer
4
6
6
2
2
1
6
5
:words
:entities
:contexts
R-GCN
R-GCN
…
R-GCN
L-Layer
R-GCN with Gate Mechanism
Entity
Word
Argument Result
Argument Role Labeling
Trigger Classification
Trigger Label in BIO
Input Sentence and Context
Encoding
Graph Construction
Information Aggregation
Event Detection and Argument Extraction

Fig. 2. The architecture of our MHGEE model. Different types of nodes are represented by circles with different colors, and similarity, different types of edges are represented by lines with different numbers in Graph Construction module. Due to space limitations, not all nodes and edges are represented in the graph.

3.1 Input Layer

We need to get the initial embedding representation vector of words, entities and contexts respectively. Let $W = w_1, w_2, \ldots, w_n$ be a sentence of length n where w_i is the i-th word. Similarly, let $E = m_1, m_2, \ldots, m_k$ be the entity in the sentence where m_k is the k-th entity.

The word embedding vector $\mathbf{x}_\mathrm{i}$. In order to get the word embedding, each token w_i in the sentence is transformed to a real-valued vector $\mathbf{x}_\mathrm{i}$ by looking up in embedding matrices and concatenating the following vectors: 1) **The word embedding vector of $\mathbf{w}_\mathrm{i}$:** The word embedding vector is obtained by looking up a pre-trained word embedding matrix GloVe; 2) **The POS-tagging label embedding vector $\mathbf{pos}_\mathrm{i}$:** The POS-tagging label embedding is generated by looking up the randomly initialized POS-tagging label embedding table; 3) **The positional embedding vector $\mathbf{p}_\mathrm{i}$:** If w_c is the current word in a sentence, then we encode the relative distance $i - c$ from w_i to w_c as a real-valued vector by looking up the randomly initialized position embedding table [11,12]; 4) **The entity type label embedding vector $\mathbf{n}_\mathrm{i}$:** Similarly to the POS-tagging label embedding vector of w_i, we annotate the entities in a sentence using BIO annotation schema and transform the entity type labels to real-valued vectors by looking up the embedding table. Thus, the input embedding of w_i can be defined as:

$$\mathbf{x}_\mathrm{i} = [\mathbf{w}_\mathrm{i}; \mathbf{pos}_\mathrm{i}; \mathbf{p}_\mathrm{i}; \mathbf{n}_\mathrm{i}] \in \mathbb{R}^{d_w + 2\times d_p + d_{pos} + d_n} \tag{1}$$

where d_w, d_p, d_{pos} and d_n denote the dimension of word embedding, positional embedding, POS-tagging label embedding and entity type embedding respectively.

The entity embedding vector $\mathbf{e}_\mathrm{i}$. We calculate the entity embedding vector $\mathbf{e}_\mathrm{i}$ with the mean-pooling operation of the vectors of all words, from w_1 to w_n, that make up the entity m_i. Thus, the input embedding of $\mathbf{e}_\mathrm{i}$ can be defined as:

$$\mathbf{e}_\mathrm{i} = \text{mean-pooling}\left(\sum_{w_i \in m_i}^{n} \mathbf{w}_\mathrm{i}\right) \in \mathbb{R}^{d_e} \tag{2}$$

where d_e denotes the dimension of entity.

The context embedding vector $\mathbf{c}_\mathrm{i}$. We take the vectors generated by Word2Vec of two sentences above and below the current sentence. These four sentences are made up of words, from w_1 to w_i. Therefore, each sentence vector $\mathbf{W}_\mathrm{j}$ is concatenated from all word vectors, from $\mathbf{w}_1$ to $\mathbf{w}_\mathrm{m}$. Then we concatenate these four sentence vectors $\mathbf{W}_1$ to $\mathbf{W}_4$ simultaneously, and then average each bit of the vectors obtained after splicing:

$$\mathbf{W}_\mathrm{j} = [\mathbf{w}_1; \mathbf{w}_2; ...; \mathbf{w}_\mathrm{m}] \in \mathbb{R}^{d_w} \tag{3}$$

$$\mathbf{c}_\mathrm{i} = [\mathbf{W}_1; \mathbf{W}_2; \mathbf{W}_3; \mathbf{W}_4] \in \mathbb{R}^{d_c} \tag{4}$$

where d_c denotes the dimension of context.

3.2 Graph Construction

We construct the graph in a multi-granularity way, motivated by the fact that each sentence constituting context contains multiple entities. Then we design three types of nodes to learn multi-granularity semantic information by considering the nearest context, the inter and intra sentence information of sentences: word, entity and context. Here we consider that entities and words are not one-to-one correspondence. Then the MHGEE model is expected to aggregate information from different granularities, as well as model interactions among these nodes for event extraction.

We also define the following six types of edges to reflect the various structural information and intra-sentence and inter-sentence event interdependency in MHGEE. 1) **Word-Word Edge:** via syntactic dependency tree. Then we make these edges exist based on the following assumptions: 2) **Word-Word Edge:** if a word may be the trigger since it has been a trigger ever; 3) **Word-Entity Edge:** if a word belongs to an entity; 4) **Word-Entity Edge:** if an entity and a word ever appeared in a centain event; 5) **Entity-Entity Edge:** if the types of these two entities have been arguments involved in the same event before; 6) **Context-Entity Edge:** if an entity appears in context. These edges enable nodes with different granularities to connect with each other simultaneously with a short path, and enables the MHGEE model to learn node representations specific to different edge types. Different edges are used to learn information of different granularity, which is related to the type of nodes they connect.

3.3 Information Aggregation over MHG

Since previous GNN only considers the node-wise connectivity, ignoring edge types, we employ R-GCN to perform information dissemination over our model. R-GCN performs well at handling high-relational data and distinguishing six different edge types when updating nodes, and the information dissemination over graph nodes can be achieved by aggregation and combination. The update process of the i-th node at the l-th layer can be formally formulated as:

$$\mathbf{n}_i^{(l)} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \sum_{j \in \mathcal{R}_{ij}} f_r \left(\mathbf{h}_j^{(l)} \right) \tag{5}$$

$$\mathbf{u}_i^{(l)} = f_s\left(\mathbf{h}_i^{(l)}\right) + \mathbf{n}_i^{(l)} \tag{6}$$

where $\mathcal{N}_i$ is the set of neighbors of node i, R_{ij} is the set of edge types between i and j, $\mathbf{h}_j^{(l)}$ is the representation of node j in layer l. f_r is a parametrized function specific to an edge type $r \in R$, and both f_r and f_s are implemented with a MLP, $\mathbf{u}_i^{(l)}$ represents the updated representation of node i.

We apply a gate mechanism to provide the way to prevent completely overwriting past information in this module, since it has been shown that GNNs suffer from the smoothing problem when the number of layers is large [5]. Formally:

$$\mathbf{g}_i^{(l)} = \sigma\left(f_g\left(\left[\mathbf{u}_i^{(l)}; \mathbf{h}_i^{(l)}\right]\right)\right) \tag{7}$$

where σ is the sigmoid function and f_g is implemented with a MLP. Gating vector $\mathbf{g}_i^{(l)}$ is then applied to control the amount information from neighbor nodes or the original node:

$$\mathbf{h}_i^{(l+1)} = \phi\left(\mathbf{u}_i^{(l)}\right) \odot \mathbf{g}_i^{(l)} + \mathbf{h}_i^{(l)} \odot \left(\mathbf{1} - \mathbf{g}_i^{(l)}\right) \tag{8}$$

where ϕ is the tanh function and $\odot$ denotes element-wise multiplication. After L times of information dissemination, the information of each node will be propagated to L-node distance away, generating L-hop-reasoning relation-aware node representations.

3.4 Classification Layer

We formulate event extraction as a sequence labeling task following previous works [1,8,11,17]. Thus, each word in sentence is assigned a label that contributes to event annotation. We apply the BIO annotation schema to assign trigger label t_i to each token w_i, as there are triggers that consist of multiple tokens, and tag "O" represents the "Other" tag, which means that the corresponding word is irrelevant of the target events. In addition, another two tags "B-type" and "I-type" consist of two parts: the word position in the trigger and any event type.

After aggregating word and entity node embedding representations from R-GCN, we feed the word representation into a fully-connected network, which is followed by a softmax function to compute distribution over all event types:

$$y_{t_i} = \text{softmax}\left(\mathbf{W}_t \mathbf{h} + b_t\right) \tag{9}$$

where $\mathbf{W}_t$ maps the word node representation $\mathbf{h}$ to the feature score for each event type, and b_t is a bias term. We choose event label with the largest probability as the classification result according to the value of y_{t_i}.

After we get trigger candidates of certain types from trigger labels, we then need to predict the roles that each entity e_j plays in such events. We aggregate word embedding representations to trigger candidate vector t_i and entity vector e_j by average pooling along the sequence length dimension. The trigger candidate vector t_i consists of words that combine to form the trigger. Then we concatenate them together and feed into a new fully-connected network to predict the argument role as:

$$y_{a_{ij}} = \text{softmax}\left(\mathbf{W}_a\left[t_i, e_j\right] + b_a\right) \tag{10}$$

where $y_{a_{ij}}$ represents the final output of which role the j-th entity plays in the event triggered by the i-th trigger candidate, and b_a is also a bias term.

3.5 Biased Loss Function

We minimize the joint negative log-likelihood loss function with a bias item as follow:

$$J(\theta) = -\sum_{k=1}^{N}\left(\sum_{i=1}^{n_k} I(O)\log\left(p\left(y_{t_i} \mid \theta\right)\right) + \beta\sum_{i=1}^{t_k}\sum_{j=1}^{e_k}\log\left(p\left(y_{a_{i,j}} \mid \theta\right)\right)\right) \tag{11}$$

where N is the number of sentences in training dataset; n_p, t_p and e_p are the number of words, extracted trigger candidates and entities of the k-th sentence; $I(O)$ represents a switching function to distinguish the loss of tag "O" and event type tags, it outputs number 1 if the tag is "O", otherwise 0; β is a bias weight.

4 Experiments and Results

4.1 Experiment Settings

Dataset and Evaluation Metrics. We conduct our whole experiments on the standard supervised ACE 2005 dataset, which consists of 599 documents annotated with 33 event subtypes, and 34 role classes. Then we add the NONE class and BIO annotation schema to role classes. Therefore, the total number of labels for event detection is 67, and the total number of labels for argument extraction is 37. Tag "O" in both subtasks represents the "Other" tag, which means that the corresponding word is irrelevant of any types. We use the same data split method [1,11,16,17] to compare with the previous works. The data split includes 40 articles with 881 sentences for the test set, 30 other documents with 1087 sentences for the development set and 529 remaining documents with 21,090 sentences for the training set. We follow the traditional evaluation metrics for evaluation: 1) Trigger Identification (TI); 2) Trigger Classification (TC); 3) Argument Identification (AI); 4) Argument Classification (AC). We use the official scorer Precision, Recall and F1-score at the evaluation stage.

Hyper-parameter Setting. The learning rate and batch size we set in our experiments is 2 and 32 respectively. For all experiments below, we use 300 dimensions for word embeddings and 50 dimensions for POS-tagging embedding, positional embedding and entity type embedding. In the R-GCN module, we use a two-layer GCN. The bias parameter in biased loss function β is set to 5.

4.2 Baselines

We compare our proposed MHGEE model with a range of state-of-the-art models in order to comprehensively evaluate performance boost results: 1) **DMCNN** [1], builds a dynamic multi-pooling convolutional model to learn sentence feature; 2) **Cross-Event** [7], uses document level information to improve the performance; 3) **GAIL** [18], bases on an inverse reinforcement learning; 4) **JointBeam** [6], extracts events based on structure prediction by manually designed features; 5) **Joint3EE** [15], bases on the shared hidden representations; 6) **JRNN** [11], employs bidirectional RNN and manually designed features to event extraction jointly; 7) **Embedding+T**: uses word embedding vectors and the traditional sentence-level features; 8) **PSL** [8], uses a probabilistic reasoning model to classify events; 9) **HBTNGMA** [2], models sentence event inter-dependency via a hierarchical and bias tagging model. Some baseline methods operate BERT as the pre-trained language model. 10) **BERT_QA** [4], is a QA-based model which uses machine reading comprehension model for both two subtasks; 11) **TEXT2EVENT** [10], presents a generation-based paradigm; 12) **DMBERT** [16], mainly focuses on the training data augmentation, with external unlabeled data through adversarial mechanism. And some models build a GNN over the dependency tree of a sentence to exploit syntactical information. 13) **GCN-ED** [12], is the first attempt to explore how to effectively use GCN in event detection; 14) **JMEE** [9], enhances GCN with self-attention and highway network to improve the performance of GCN for event detection; 15) **MOGANED** [17], improves GCN with aggregated attention to combine multi-order word representation from different GCN layers.

4.3 Overall Performance and Ablation Analysis

Table 1 shows the overall performance. Our MHGEE model achieves the best F1 scores for event extraction among all the compared methods. There is a significant gain with the trigger identification, which is nearly 5% higher over the best-reported models. There is also a significant gain with the argument identification, which is over 2% higher over the best-reported models. In addition, our MHGEE model still outperforms BERT-based models without using BERT as a pre-trained language model, although the encoder of BERT has been proven its validity to improve the performance of event extraction, which is one of the downstream natural language processing tasks. It demonstrates the effectiveness of aggregating information with different granularities for event extraction tasks. Compared with the previous GNN-based models, our MHGEE model complete the subtask of argument extraction with the consideration of argument information, and the interaction between two subtasks. This information interaction between arguments and trigger has a good effect on improving the performance of event extraction.

Table 1. Overall performance comparing to the SOTA methods

Models	TI			TC			AI			AC		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
DMCNN	80.4	67.7	73.5	75.6	63.6	69.1	68.8	51.9	59.1	62.2	46.9	53.5
Cross-Event		N/A		68.7	68.9	68.8	50.9	49.7	50.3	45.1	44.1	44.6
GAIL	78.9	66.5	72.2	75.3	63.4	68.9	**69.8**	52.7	60.0	61.6	45.7	52.4
JointBeam	76.9	65.0	70.4	73.7	62.3	67.5	69.8	47.9	56.8	**64.7**	44.4	52.7
Joint3EE	70.5	74.5	72.5	68.0	71.8	69.8	59.9	59.8	59.9	52.1	52.1	52.1
JRNN	68.5	75.7	71.9	66.0	73.0	69.3		N/A			N/A	
Embedding+T	76.9	65.0	70.4	73.7	62.3	67.5	69.8	47.9	56.8	64.7	44.4	52.7
PSL		N/A		75.3	64.4	69.4		N/A			N/A	
HBTNGMA		N/A		77.9	69.1	73.3		N/A			N/A	
BERT-based												
BERT-QA	74.3	77.4	75.8	71.1	73.7	72.4	58.9	52.1	55.3	56.8	50.2	53.3
TEXT2Event		N/A		69.6	74.4	71.9		N/A		52.5	55.2	53.8
DMBERT		N/A		71.6	72.3	70.9		N/A		53.1	54.2	52.8
GNN-based												
JMEE	80.2	72.1	75.9	76.3	71.3	73.7		N/A			N/A	
MOGANED		N/A		**79.5**	72.3	75.7		N/A			N/A	
GCN-ED		N/A		77.9	68.8	73.1		N/A			N/A	
MHGEE	79.8	**81.3**	**80.5**	75.1	**76.0**	**75.7**	63.8	**61.5**	**62.6**	56.6	54.5	**55.5**

Table 2 shows the ablation analysis of our study. We assume that if one type of nodes has been removed, it means the corresponding edges also do not exist in the heterogeneous graph. The F1 score drops more than 2 points regardless of the different edge types, context nodes or entity nodes we remove. If we remove entity nodes, we observe a more significant decline on F1 score than we remove context nodes. It indicates that all kinds of nodes and edges in our MHGEE model play important roles, but entity nodes are more essential. This is because when we alleviate the challenges, we more dependent on the entity information, which means entity nodes can be used as key nodes. If there is no entity information in context to help us determine the triggers, then context is not so necessary.

Table 2. Results of ablation studies on ACE 2005 dataset

Setting	TI	TC	AI	AC
Full (using identified trigger)	80.4	75.6	48.9	44.6
Full (using golden trigger)	**80.5**	**75.7**	**62.6**	**55.5**
- Different Edge Types	77.3	71.5	60.1	52.2
- Context Nodes	77.9	72.3	59.8	54.2
- Entity Nodes	77.1	72.2	57.6	53.4

Additionally, under the condition of using identified trigger, the F1 score of event detection task will not drop significantly. However, the F1 score drops by more than 10% in event argument extraction task. This result shows that we utilize golden trigger other than identified trigger to complete the event argument detection task, since identified trigger can cause the error propagation problem.

Overall, information with different granularities, and all edges with different types, can promote the interaction among nodes by R-GCN, in order to capture the information from the multi-granularity heterogeneous graph to complete event extraction, and finally benefits the performance.

4.4 Effect on Event Interdependency

Following some previous works [1,9,11], we split the test data into two parts: 1/1 and 1/N to evaluate the effect of our model for alleviating the multiple-events phenomena. 1/1 means that one sentence only has a single trigger, and 1/N is for all remaining cases. We perform evaluations separately.

Table 3. Performance on single event sentences and multiple event sentences

Setting	Model	1/1	1/N	All
Trigger classification	Embedding+T	68.1	25.5	59.8
	CNN	72.5	43.1	66.3
	DMCNN	74.3	50.9	69.7
	JRNN	75.6	64.8	69.3
	JMEE	75.2	72.7	73.7
	HBTNGMA	**78.4**	59.5	73.3
	MHGEE	74.8	**75.8**	**75.7**
Argument classification	Embedding+T	37.4	15.5	32.6
	CNN	51.6	36.6	48.9
	DMCNN	**54.6**	48.7	53.5
	JRNN	50.0	**55.2**	55.4
	MHGEE	43.7	52.7	**55.5**

We use F1 scores to illustrate the performance of Embedding+T [6], CNN [1], JRNN [11], DMCNN [1], and our model for event extraction in Table 3. CNN is similar to DMCNN, except that it applies the standard max-pooling mechanism. Our MHGEE model significantly outperforms all the other mentioned methods in trigger classification subtask. In the **1/N** data split of triggers, our model is 3.1% better than the JMEE. It demonstrates that our model, with the utilization of multi-granularity heterogeneous graph and the model of intra-sentence and inter-sentence event interdependency, can capture multiple events in one sentence effectively.

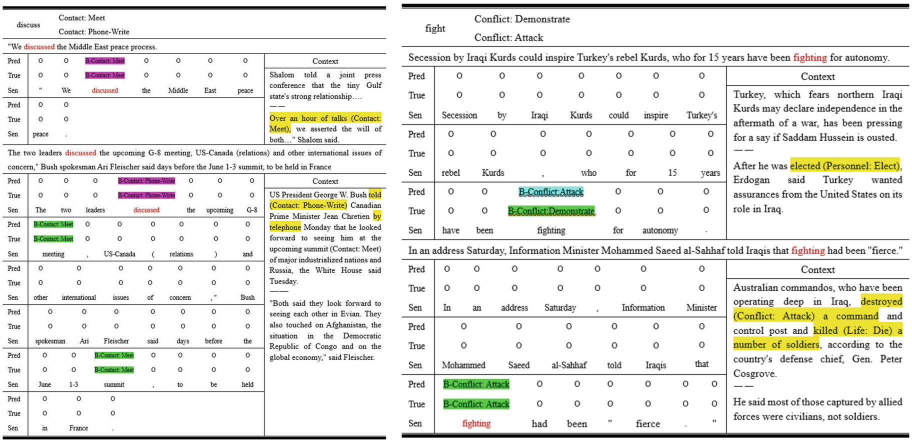
Table 4. Performance on single event sentences and multiple event sentences.

Setting	TI	TC	AI	AC
All	80.5	75.7	62.6	55.5
1/1	79.5	74.8	51.8	43.7
1/N	**81.4**	**75.8**	**63.6**	**58.0**

Table 4 shows the event extraction effect of our MHGEE model on both 1/1 and 1/N. Our model performs better on 1/N. It indicates that multiple granularity information has greater gain on distinguishing the semantics between different triggers in one sentence, that is, event interdependency, which sourced from the fact that multiple events are often associated with each other, having similar event types. We model this intra-sentence and inter-sentence event interdependency phenomenon through a heterogeneous graph to not only capture multiple events in one sentence, but also mitigate their similarity.

4.5 Case Study and Effect on Ambiguity of Triggers

In Fig. 3, we show two examples of case study on ambiguity of triggers. In both (a) and (b), both words "discuss" and "fight" trigger two different events respectively. Ambiguity occurs in both cases. However, our MHGEE model can solve this problem in (a), but fails in (b). According to the idea of our model, we need



discuss
Contact: Meet
Contact: Phone-Write

"We discussed the Middle East peace process.

Pred	O	O	B-Contact: Meet	O	O	O	O
True	O	O	B-Contact: Meet	O	O	O	O
Sen	"	We	discussed	the	Middle	East	peace
Pred	O	O					
True	O	O					
Sen	peace	.					

Context

Shalom told a joint press conference that the tiny Gulf state's strong relationship….

——

Over an hour of talks (Contact: Meet), we asserted the will of both…" Shalom said.

The two leaders discussed the upcoming G-8 meeting, US-Canada (relations) and other international issues of concern," Bush spokesman Ari Fleischer said days before the June 1-3 summit, to be held in France

Pred	O	O	O	B-Contact: Phone-Write	O	O	O
True	O	O	O	B-Contact: Phone-Write	O	O	O
Sen	The	two	leaders	discussed	the	upcoming	G-8
Pred	B-Contact: Meet	O	O	O	O	O	O
True	B-Contact: Meet	O	O	O	O	O	O
Sen	meeting	,	US-Canada	(	relations	)	and
Pred	O	O	O	O	O	O	O
True	O	O	O	O	O	O	O
Sen	other	international	issues	of	concern	,"	Bush
Pred	O	O	O	O	O	O	O
True	O	O	O	O	O	O	O
Sen	spokesman	Ari	Fleischer	said	days	before	the
Pred	O	O	B-Contact: Meet	O	O	O	O
True	O	O	B-Contact: Meet	O	O	O	O
Sen	June	1-3	summit	,	to	be	held
Pred	O	O	O				
True	O	O	O				
Sen	in	France	.				

Context

US President George W. Bush told (Contact: Phone-Write) Canadian Prime Minister Jean Chretien by telephone Monday that he looked forward to seeing him at the upcoming summit (Contact: Meet) of major industrialized nations and Russia, the White House said Tuesday.

——

"Both said they look forward to seeing each other in Evian. They also touched on Afghanistan, the situation in the Democratic Republic of Congo and on the global economy," said Fleischer.

(a)

fight
Conflict: Demonstrate
Conflict: Attack

Secession by Iraqi Kurds could inspire Turkey's rebel Kurds, who for 15 years have been fighting for autonomy.

Pred	O	O	O	O	O	O	O
True	O	O	O	O	O	O	O
Sen	Secession	by	Iraqi	Kurds	could	inspire	Turkey's
Pred	O	O	O	O	O	O	O
True	O	O	O	O	O	O	O
Sen	rebel	Kurds	,	who	for	15	years
Pred	O	O	B-Conflict:Attack	O	O	O	
True	O	O	B-Conflict:Demonstrate	O	O	O	
Sen	have	been	fighting	for	autonomy	.	

Context

Turkey, which fears northern Iraqi Kurds may declare independence in the aftermath of a war, has been pressing for a say if Saddam Hussein is ousted.

——

After he was elected (Personnel: Elect), Erdogan said Turkey wanted assurances from the United States on its role in Iraq.

In an address Saturday, Information Minister Mohammed Saeed al-Sahhaf told Iraqis that fighting had been "fierce."

Pred	O	O	O	O	O	O	O
True	O	O	O	O	O	O	O
Sen	In	an	address	Saturday	,	Information	Minister
Pred	O	O	O	O	O	O	
True	O	O	O	O	O	O	
Sen	Mohammed	Saeed	al-Sahhaf	told	Iraqis	that	
Pred	B-Conflict: Attack	O	O	O	O	O	O
True	B-Conflict: Attack	O	O	O	O	O	O
Sen	fighting	had	been	"	fierce	.	"

Context

Australian commandos, who have been operating deep in Iraq, destroyed (Conflict: Attack) a command and control post and killed (Life: Die) a number of soldiers, according to the country's defense chief, Gen. Peter Cosgrove.

——

He said most of those captured by allied forces were civilians, not soldiers.

(b)

Fig. 3. The example of case study. Yellow highlighted content indicates entity information that can be used to solve ambiguity. The context of the sentence where the trigger is located, as well as the true BIO annotation and predicted BIO annotation results of the sentence through other colors are shown. If the colors are the same, our model predicts correctly, otherwise it is different. (Color figure online)

to learn rich different granularity information, including entities and contexts, to solve the ambiguity problem. In (a), there is enough information to help us solve the ambiguity problem, but in (b), there is not enough semantic information.

The two event types triggered by “fight” have certain similarities, and “elected” in context serves as a trigger word for event type “Personnel: elected”, which is irrelevant to the two events triggered by “fight”, having no evidence to solve the ambiguity problem. The case study shows that multi-granularity information does help to alleviate the problem of ambiguity of triggers. There is indeed rich multi-granularity information around some event mentions, and our MHGEE model can solve the ambiguity problem by aggregating this information.

5 Conclusions and Future Works

In this paper, we propose a novel model MHGEE for event extraction. In order to disambiguate triggers, our MHGEE model aggregate nodes and edges simultaneously into a heterogeneous graph to enable rich information interactions among nodes with different granularities by R-GCN. In addition, we consider the multiple-event phenomenon with modeling intra-sentence and inter-sentence event interdependency. The whole experimental results demonstrate that our MHGEE model can achieve new state-of-the-art performance on the ACE 2005 dataset. In the future, we would like to apply MHGEE to other information extraction tasks, such as aspect extraction and named entity recognition.

References

1. Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J.: Event extraction via dynamic multi-pooling convolutional neural networks. In: ACL-IJCNLP 2015–53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference, vol. 1, pp. 167–176 (2015). https://doi.org/10.3115/v1/p15-1017
2. Chen, Y., Yang, H., Liu, K., Zhao, J., Jia, Y.: Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, pp. 1267–1276 (2020). https://doi.org/10.18653/v1/d18-1158
3. Cui, S., Yu, B., Liu, T., Zhang, Z., Wang, X., Shi, J.: Edge-enhanced graph convolution networks for event detection with syntactic relation. In: Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020, pp. 2329–2339 (2020). https://doi.org/10.18653/v1/2020.findings-emnlp.211
4. Du, X., Cardie, C.: Event extraction by answering (almost) natural questions. In: EMNLP 2020–2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, pp. 671–683 (2020). https://doi.org/10.18653/v1/2020.emnlp-main.49
5. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings (2017)

6. Li, Q., Ji, H., Huang, L.: Joint event extraction via structured prediction with global features. In: ACL 2013–51st Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, vol. 1, pp. 73–82 (2013)
7. Liao, S., Grishman, R.: Using document level cross-event inference to improve event extraction. In: ACL 2010–48th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, pp. 789–797 (2010)
8. Liu, S., Liu, K., He, S., Zhao, J.: A probabilistic soft logic based approach to exploiting latent and global information in event classification. In: 30th AAAI Conference on Artificial Intelligence, AAAI 2016, pp. 2993–2999 (2016)
9. Liu, X., Luo, Z., Huang, H.: Jointly multiple events extraction via attention-based graph information aggregation. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, pp. 1247–1256 (2020). https://doi.org/10.18653/v1/d18-1156
10. Lu, Y., et al.: TEXT2EVENT: controllable sequence-to-structure generation for end-to-end event extraction. In: ACL-IJCNLP 2021–59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference, pp. 2795–2806 (2021). https://doi.org/10.18653/v1/2021.acl-long.217
11. Nguyen, T.H., Cho, K., Grishman, R.: Joint event extraction via recurrent neural networks. In: 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference, pp. 300–309 (2016). https://doi.org/10.18653/v1/n16-1034
12. Nguyen, T.H., Grishman, R.: Graph convolutional networks with argument-aware pooling for event detection. In: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, pp. 5900–5907 (2018)
13. Sha, L., Qian, F., Chang, B., Sui, Z.: Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction. In: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, pp. 5916–5923 (2018)
14. Tang, H., Cao, Y., Zhang, Z., Jia, R., Fang, F., Wang, S.: Multi-granularity heterogeneous graph for document-level relation extraction. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, vol. 2021-June, pp. 7683–7687 (2021). https://doi.org/10.1109/ICASSP39728.2021.9414755
15. Wadden, D., Wennberg, U., Luan, Y., Hajishirzi, H.: Entity, relation, and event extraction with contextualized span representations. In: EMNLP-IJCNLP 2019–2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, pp. 5784–5789 (2020). https://doi.org/10.18653/v1/d19-1585
16. Wang, X., Han, X., Liu, Z., Sun, M., Li, P.: Adversarial training for weakly supervised event detection. In: NAACL HLT 2019–2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, vol. 1, pp. 998–1008 (2019). https://doi.org/10.18653/v1/n19-1105

17. Yan, H., Jin, X., Meng, X., Guo, J., Cheng, X.: Event detection with multi-order graph convolution and aggregated attention. In: EMNLP-IJCNLP 2019–2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, pp. 5766–5770 (2020). https://doi.org/10.18653/v1/d19-1582
18. Zhang, T., Ji, H., Sil, A.: Joint entity and event extraction with generative adversarial imitation learning. Data Intell. **1**(2), 99–120 (2019). https://doi.org/10.1162/dint_a_00014