



Research article

Advancing document-level event extraction: Integration across texts and reciprocal feedback

Min Zuo^{1,2}, Jiaqi Li^{1,2}, Di Wu³, Yingjun Wang^{1,2}, Wei Dong^{1,2,*}, Jianlei Kong^{1,4} and Kang Hu^{5,*}

¹ National Engineering Research Centre for Agri-Product Quality Traceability, Beijing Technology and Business University, Beijing 100048, China

² China Food Flavor and Nutrition Health Innovation Center, Beijing Technology and Business University, Beijing 100048, China

³ Beijing Academy of TCM Beauty Supplements Co., Ltd., Beijing 102401, China

⁴ Artificial Intelligence College, Beijing Technology and Business University, Beijing 100048, China

⁵ National Institutes for Food and Drug Control, Beijing 100050, China

* **Correspondence:** Email: dongwei2019@btbu.edu.cn, hukang@nifdc.org.cn.

Abstract: The primary objective of document-level event extraction is to extract relevant event information from lengthy texts. However, many existing methods for document-level event extraction fail to fully incorporate the contextual information that spans across sentences. To overcome this limitation, the present study proposes a document-level event extraction model called Integration Across Texts and Reciprocal Feedback (IATRF). The proposed model constructs a heterogeneous graph and employs a graph convolutional network to enhance the connection between document and entity information. This approach facilitates the acquisition of semantic information enriched with document-level context. Additionally, a Transformer classifier is introduced to transform multiple event types into a multi-label classification task. To tackle the challenge of event argument recognition, this paper introduces the Reciprocal Feedback Argument Extraction strategy. Experimental results conducted on both our COSM dataset and the publicly available ChFinAnn dataset demonstrate that the proposed model outperforms previous methods in terms of F1 value, thus confirming its effectiveness. The IATRF model effectively solves the problems of long-distance document context-aware representation and cross-sentence argument dispersion.

Keywords: document-level event extraction; heterogeneous graphs; entity extraction; Graph Convolutional Networks; transformer

1. Introduction

Amid the swift advancement of social media, event extraction (EE) has emerged as a crucial component in monitoring and evaluating public opinion from social media information. The primary objective of EE is to derive structured data, such as time, place, and person, from textual data. This endeavor occupies a prominent niche within the realm of Natural Language Processing (NLP), and the extracted structured information can be utilized for various applications. These applications span knowledge graph construction [1,2], recommender systems [3,4], intelligent question answering [5,6], as well as other tasks [7–11] for more in-depth and precise analysis.

Event extraction enables us to swiftly capture the essential elements and crucial information regarding public opinion events in social media data [12], facilitating a comprehensive understanding and analysis of these events. The extracted event information empowers decision-makers to promptly comprehend the focal point of public opinion and stay abreast of the development and impact of events in real-time. However, practical applications often encounter challenges, such as scattered argument entities across multiple sentences within a document, as well as limitations imposed by most existing EE models when it comes to text length. To broaden the scope of EE's applicability [13,14], an increasing number of researchers are turning their attention to document-level event extraction (DEE). DEE proves particularly advantageous for monitoring public opinion on social media platforms. Recent statistics indicate a growing proportion of public opinion focusing on the food and cosmetic industries [15–17], highlighting the critical role of DEE in analyzing harmful public opinion specifically related to food and cosmetic products. Moreover, the abundance of unstructured and ambiguous textual data available on internet platforms poses challenges for processing of extensive news articles. The key lies in transforming these text-based data into structured information that can be efficiently queried [18]. Given this context, the significance of DEE becomes even more prominent.

Figure 1 illustrates an example of document-level event extraction. In fact, a document can encompass multiple event records, and an entity can serve multiple parameter roles. It's also possible for event records to be missing certain event arguments. The first challenge is the effective capture of event information that spans multiple sentences [19]. Considering entities are distributed throughout multiple sentences within a document, it's imperative for the model to comprehensively grasp the contextual information to extract them accurately. The second challenge is that previous models use a predefined order to extract event parameters while neglecting to dynamically adjust the order based on the importance of the parameter roles. Such a rigid extraction approach might falter when faced with the diverse order and relevance of various arguments in individual events. Furthermore, the relationships and interactions between multiple arguments are instrumental in determining argument roles. Regrettably, many current methods tend to neglect this crucial detail. To tackle these issues, we propose the Reciprocal Feedback Argument Extraction strategy, which seeks to enhance the performance of previous methods.

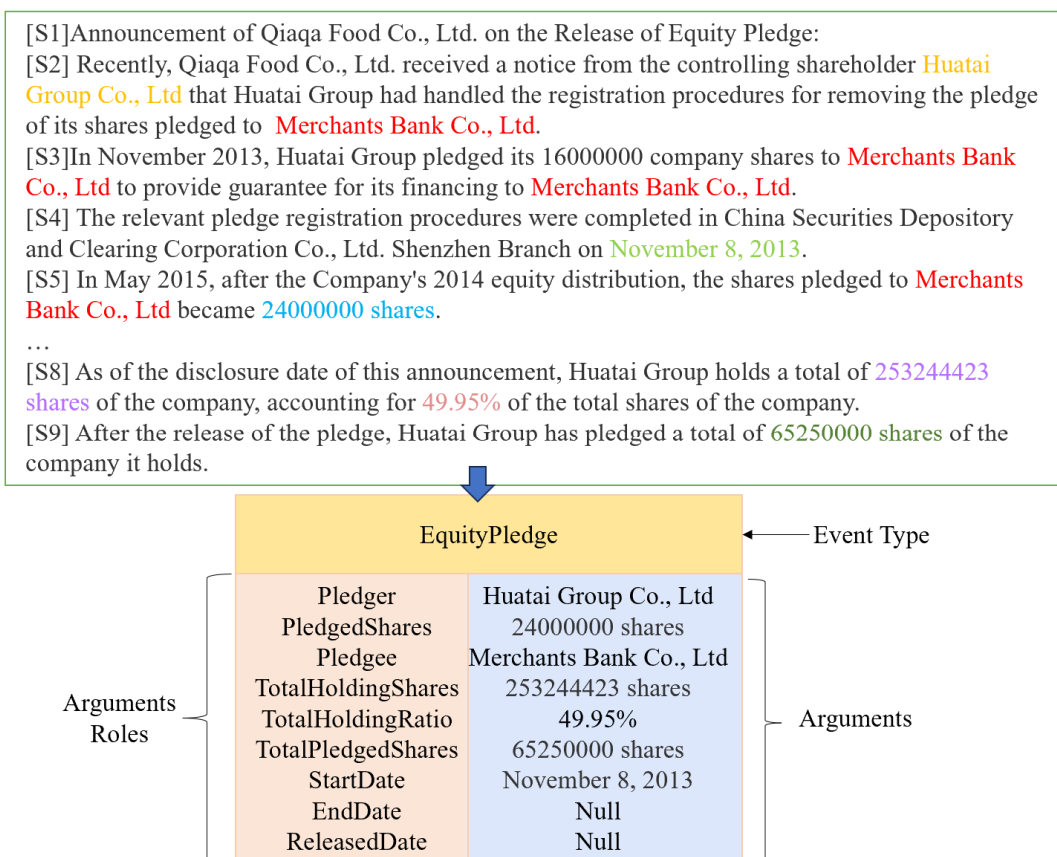


Figure 1. Examples of event extraction.

To address the aforementioned challenges, this paper presents an improved model based on the GIT model. The key to enhancing the accuracy of event extraction in DEE lies in incorporating dispersed contextual information. Therefore, we propose a novel document-level opinion event extraction model called IATRF, which integrates contextual semantics. This model constructs a Heterogeneous Graph that incorporates entities and sentences at multiple granularities. By leveraging the Graph Convolutional Network (GCN) [20], we obtain entity and sentence representations with contextual awareness at the document level. This approach effectively addresses the issue of long-distance dependencies between entity mentions in documents. For handling multiple event types, we employ a transformer classifier for multi-label classification to enhance the predictive ability of the classifier and enable event detection for multiple event types. Instead of extracting arguments in a predefined role order, the argument recognition process utilizes the knowledge of already extracted arguments to determine the roles of difficult-to-decide arguments individually. By leveraging the newly acquired information, it improves the decisions of previously extracted arguments. This interactive feedback process allows for the efficient utilization of argument relations, enabling a better understanding of sentences and improving the extraction of event arguments. As a result, it leads to a correctly extracted structured representation of events and provides a more effective solution to the problem of event thesis element fragmentation.

The contributions of this paper are as follows:

- We propose the Reciprocal Feedback Argument Extraction strategy, which utilizes knowledge of extracted argument roles in a descending order and improves the accuracy of argument extraction

through a feedback process.

- For handling multiple event types, we introduce a transformer classifier for multi-label classification, effectively achieving event detection and enhancing the predictive ability of the classifier.
- The model IATRF in this paper has been extensively experimented on the DEE benchmark, and a series of experiments on our own dataset COSM and the publicly available dataset ChFinAnn confirms the validity of our proposed methodology, which outperforms the existing Sota on F1.

2. Related works

In earlier research, scholars endeavored to represent the semantic information of text using low-dimensional dense vectors, which encode words, sentences, or entire documents. The composition of a document's semantics determines its overall semantic representation. Therefore, we will analyze typical models and methods for event extraction that obtain semantic representations at three different levels: words, sentences, and documents [21–23]. Word2vec [24], a word vector method proposed by the Google team, allows semantically similar texts to acquire similar embedding representations. The GloVe model [25] leverages co-occurrence matrices to capture word semantics comprehensively. On the other hand, the ELMo model [26] dynamically adjusts word vectors based on context, effectively addressing the issue of polysemous words; however, it does not fully exploit contextual information. The BERT model [27], which utilizes the Bidirectional Transformer [28] language model, combines contextual semantics with a masking approach for training, resulting in more expressive word vectors. Moreover, Liu et al. [29] used local features of arguments to enhance role classification, marking the first time that entity recognition and argument extraction were studied as a joint learning task.

Although the aforementioned methods have advanced trigger word recognition and argument extraction tasks [30], they fall short in capturing semantic associations between arguments and trigger words [31], as well as between trigger words and event types within a sentence. Relying solely on local features such as word-level semantics [32] is inadequate, necessitating the acquisition of contextual semantic representations at the sentence level. Chen et al. [33] proposed dynamic multi-pooled Convolutional Neural Networks (CNN) [34] to extract sentence-level clues, employing dynamic pooling layers to preserve more information about event trigger words and arguments. Nguyen et al. [35] learned sentence representations using a structure based on bidirectional Recurrent Neural Networks (RNN) [36], utilizing memory vectors and memory matrices to store information related to trigger words, arguments, and their dependencies. Liu et al. [37] extracted multiple candidate trigger words and arguments from sentences, establishing connections through syntactic relations and employing the Graph Attention Network (GAT) to model graph information. This approach successfully addresses the issue of sequential modeling models' inefficiency in capturing long-range dependencies. To tackle the problem of entities assuming different roles in different events, Yang et al. [38] separated the prediction of arguments and prediction of argument roles tasks, effectively resolving role overlapping. Inspired by Machine Reading Comprehension (MRC) research, EE based on the MRC framework has garnered increasing attention. Du et al. [39] extracted trigger words and arguments by defining problem templates for trigger words and event roles in the form of fragment extraction. Liu et al. [5] introduced an unsupervised question generation method that avoids generating semantically insufficient template-based questions. Zhou et al. [40] employed a dyadic question and answer approach, enabling the model to comprehend the semantics of roles. Questions were formulated not only about argument roles but also about the arguments themselves. This MRC approach addresses data scarcity and facilitates

parameter sharing when extracting different arguments. Thus, existing EE research predominantly focuses on datasets containing trigger words. However, in certain chapters, events may lack obvious trigger words or even lack trigger words altogether. Consequently, the commonly used trigger word-based SEE approach does not perform effectively in DEE [41].

Compared to SEE, DEE faces two major challenges: dealing with widely distributed arguments and recognizing multiple events [42]. As a result, extracting event arguments in a single sentence often leads to incomplete results. To overcome this challenge, DEE models need to have a comprehensive understanding of the connections between various levels of information in a document, as well as the features extracted from event arguments [43]. Yang et al. [44] proposed the DCFEE model, which extracts trigger words and arguments in a sentence-by-sentence manner. They used CNN to classify each sentence as critical or non-critical. To obtain complete event arguments, they proposed an argument complementation strategy that retrieves arguments from surrounding sentences to complement the ones in the key event sentence. However, this method is simplistic, and it fails to eliminate errors in the entity recognition stage. Zheng et al. [45] transformed argument recognition into a path expansion subtask based on the directed acyclic graph of entities to solve the problems of argument dispersion and multiple events. However, relying solely on the fusion of Transformer sentences and entities is insufficient, and the Transformer model struggles to capture internal dependencies when event arguments appear in different sentences. Xu et al. [46] introduced the Tracker module to model the relationship between events, store decoded event information, and decode information for other event arguments. Yang et al. [47] proposed a multi-granularity decoder to extract all events in parallel. They also need to be capable of combining relevant arguments across sentences and recognizing multiple event types in a document. Researchers are beginning to use GCN to reason about intra- and inter-sentence relationships and have made some progress in extracting sentence relationships. To address these challenges, DEE requires models that can integrate document-level information while capturing multiple events across multiple sentences [48,49]. Huang et al. [50] converted each document into an undirected graph based on sentence relationships, dividing the graph into subgraphs representing sentence communities. Event classification and argument recognition tasks are then performed within each sentence community. Hu et al. [51] aimed to identify role arguments of a specific event type in a document. They employed a role-knowledge oriented approach to enhance the interaction between roles and templates. Overall, these approaches address the challenges of DEE by incorporating various strategies such as sentence classification, argument complementation, path expansion, and role-knowledge orientation. While some approaches have attained notable success, existing methods for DEE tasks rely on predefined event role orders for argument detection without considering the correlations between event roles or overlooking the overall information of the document. As a result, there is still a need for improvement in effectively capturing cross-sentence event relationships.

3. Approach

The structure of the model in this paper is shown in Figure 2 and consists of four main components: 1) Entity Extraction. Sentences of a document are fed into the encoder to obtain a contextual representation, and then the entity information is extracted through the Conditional Random Field (CRF) [52] layer; 2) Heterogeneous Graph for Entity Interactions. A global heterogeneous graph is constructed, including document nodes (D), sentence nodes (S) and entity nodes (E), so as to realize

richer interaction representations between different relationship nodes. The global interaction information between them can be captured based on GCN. 3) Event Types Detection. After obtaining document-aware representations of entities and sentences, a Transformer classifier is introduced to detect event types and perform multi-label categorization; 4) Reciprocal Feedback Argument Extraction. The decoding module is used to extract the event records, sort on the already extracted argument roles, utilize this feedback process to determine the arguments efficiently, and store the information of the event records into the global storage.

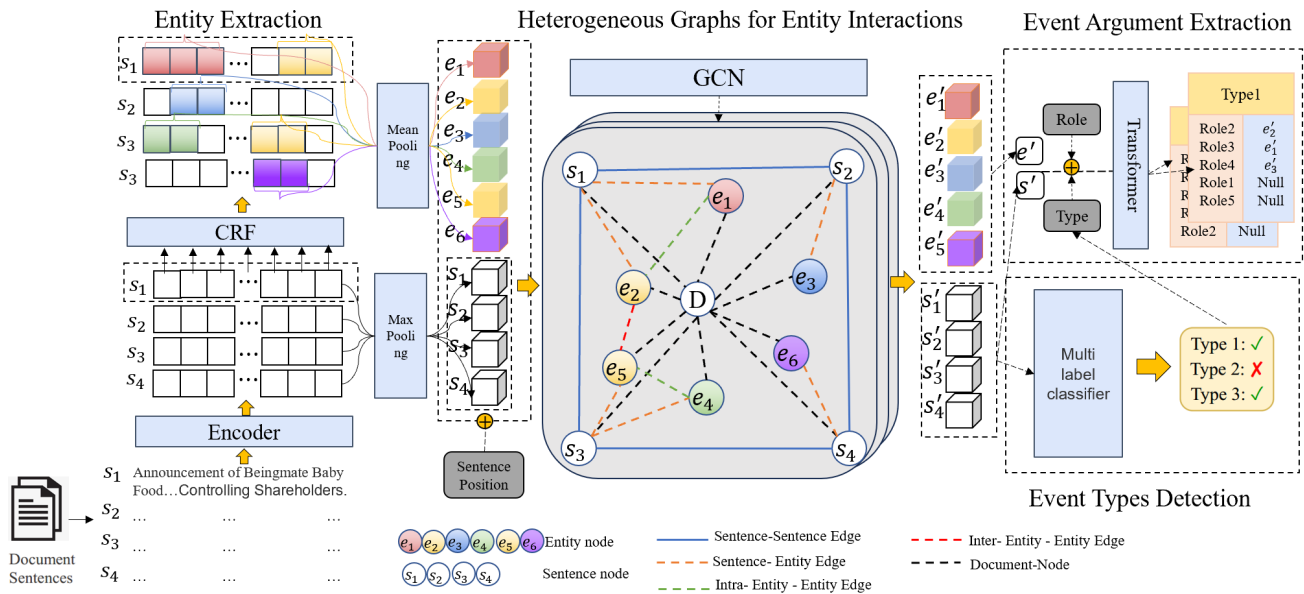


Figure 2. The overall architecture of the proposed model.

3.1. Entity extraction

In this paper, the model represents each document as a sequence of sentences. First, each document is divided into multiple sentences $D = \{s_1, s_2, \dots, s_N\}$. And entity recognition can be considered as a sequence tagging task. In this paper, we use the model transformer to encode the documents and obtain the contextual information embeddings for both sentences and the entire documents:

$$\{T_{w_{i,1}}, T_{w_{i,2}}, T_{w_{i,3}}, \dots, T_{w_{i,j}}\} = \text{Transformer}(S_1, S_2, S_3, \dots, S_N) \quad (1)$$

where $T_{w_{i,j}}$ is a vector sequence of document transformations, $w_{i,j} \in \mathbb{R}^{d_w}$, i is the number of tag types, j is the length of sentences, d_w is the embedding size, \mathbb{R} is the trainable matrix, and N is the maximum number of sentences. The word representation of Transformer is the sum of corresponding tags and positional embeddings, and the range of entities and their types can be accurately identified by way of labeling BIO tags. However, when the Transformer model is applied to perform the sequence labeling task, the highest rated tags may be misclassified. To cope with these problems, this paper adopts the CRF layer for entity recognition. Incorporating the CRF layer into the model of this paper

aims to improve the effectiveness of document entity recognition. In this stage, firstly, embedding representations are provided for each word. Then the state features of the sequence are learned and the scores are input into the CRF layer to obtain the transfer score matrix \mathbf{T} ($\mathbf{T} \in \mathbb{R}^{m \times n}$), the operation is as follows:

$$\mathbf{T} = \text{CRF}(T_{w_{i,1}}, T_{w_{i,2}}, \dots, T_{w_{i,j}}) \quad (2)$$

Then the possible tag sequence scores are calculated:

$$\text{score} = \sum_{i=1}^m F_{i,y_i} + \sum_{i=1}^{n-1} T_{y_i,y_{i+1}} \quad (3)$$

where F_{i,y_i} is the score of label y_i for the i -th label in the sequence, and $T_{y_i,y_{i+1}}$ denotes the score of the transition from label y_i to y_{i+1} . For training, we minimize the following losses:

$$\text{Loss}_{ner} = -\sum_{s \in D} \log P(y_s | s) \quad (4)$$

where y_s is the golden tag sequence of s , P is the score of the golden tag sequence, and s is the predicted tag sequence. In order to obtain the best probabilistic results for full text sentences, this paper applies the Viterbi algorithm, by which we can decode the labeled sequences with maximum probability.

3.2. Heterogeneous graph for entity interactions

In DEE tasks, event arguments are often scattered between multiple sentences. In order to create associations between different sentences and the entities within them to model the complex interactions between different mentions in a document and to enhance the connection between document and entity information. This is essential to solve the problem of long-distance dependencies between entities within a document. We construct a heterogeneous graph in this study to enable cross-sentence information transfer so that the model can understand the context more comprehensively. This heterogeneous graph consists of entity nodes and sentence nodes. For entity nodes, since an entity e may contain multiple tokens, an average pooling strategy is used to obtain an initialized representation of that entity node:

$$h_e = \text{MeanPooling}(\{t_i\}_{i \in e}) \quad (5)$$

where h_e denotes the entity node. Similarly, for a sentence node, the initialized representation of the sentence node is obtained by using the maximum pooling strategy for the tokens in the sentence and adding the position code of the sentence:

$$h_{s_i} = \text{MaxPooling}(\{t_j\}_{j \in s_i}) + \text{SentPos}(s_i) \quad (6)$$

where h_{s_i} denotes the sentence node. When constructing edges, the following rules are used to constitute 5 types of edges: 1) Sentence-Sentence Edge (S-S Edge): connects all sentence nodes. By establishing a long-distance dependency between any two independent sentences in a document, the S-S Edge captures the relationship between sentences. 2) Sentence-Entity Edge (S-E Edge): connects sentence nodes with entity nodes within the same sentence. The S-E Edge models the context of entity mentions in the sentence by connecting the sentence to all entity mentions within it. 3) Intra-Entity-Entity Edge (Intra-E-E Edge): connects all entity nodes within the same sentence. By linking different

entity mentions within a sentence, the Intra-E-E Edge indicates that these mentions may be related to the same event. 4) Inter- Entity-Entity Edge (Inter-E-E Edge): connects mentions of the same entity in different sentences. It helps track all occurrences of a specific entity in a document, facilitating the extraction of long document events. 5) Document-Node (D-N Edge): connects all other nodes to the document node through document edges. By enabling the document node to pay attention to information from all other nodes, it facilitates interaction between documents, sentences, and entity mentions. Moreover, centering on document nodes allows for better modeling of long-distance dependencies. GCN is applied on the global graph to aggregate features from the neighborhood. Given node u at layer l -th, the operation of graph convolution:

$$h_u^{(l+1)} = \sigma \left(\sum_{k \in K} \sum_{v \in N_k(u)} \frac{1}{c_{u,k}} W_k^{(l)} h_v^{(l)} + b_k^{(l)} \right) \quad (7)$$

where K are different types of edges, $W_k^{(l)} \in \mathbb{R}^{d \times d}$ and $b_k^{(l)} \in \mathbb{R}^d$ are trainable parameters, $h_v^{(l)}$ denotes the representation of the v -th word after the l -th layer of GCN, and $N_k(u)$ denotes the neighbors of node u connected in the k -th type of edges. σ is the Relu activation function. Different layers of GCN express different abstraction levels of features. Therefore, in order to cover all levels of features, the model in this paper connects the hidden states of each level to form the final representation of node u :

$$h_u = [h_u^{(0)}; h_u^{(1)}; \dots; h_u^{(L)}] \quad (8)$$

where L is the number of layers of the GCN, and $h_u^{(0)}$ is the initial representation of the node u shown. After this stage, a document-level context-aware entity representation $E = [e'_1, \dots, e'_{N_e}] \in \mathbb{R}^{d_m \times N_e}$ and a sentence representation $S = [s'_1, \dots, s'_{N_s}] \in \mathbb{R}^{d_m \times N_s}$ are obtained, N_e is the number of different entity mentions number and N_s is the number of sentences.

3.3. Event types detection

Multiple events can exist in a single document, making it more appropriate to consider this task as a multi-label classification problem. Global feature vectors alone may not fully capture the granularity of the semantics in a document. Therefore, we adopt an approach that combines local and global features generated by the transformer model. We add a fully connected layer at the end to enhance the prediction accuracy of the classifier for text classification. Specifically, the input text first goes through an embedding layer. Then, it is passed into a multi-layer Transformer encoder for feature extraction, which generates hidden state representations for all time steps. These hidden state representations are then converted into a fixed-size feature vector using a global pooling layer. Finally, classification is performed by the fully connected layer. The event type classifier model's structure is depicted in Figure 3. The classifier takes a text file as input. The output from the transformer is received by the classification header, which generates the predicted category labels.

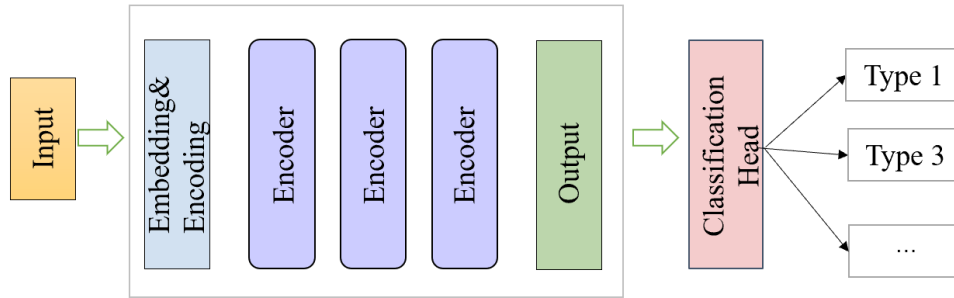


Figure 3. Architecture diagram of transformer classification model.

After the previous step, the feature matrix S of the sentence is obtained, and the multi-headed attention mechanism is used to further explore the perceptual degree of the sentence to the event:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (9)$$

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_n)W^0 \quad (10)$$

where the perceived degree of the event type is used as query and the feature matrix S of the sentence is used as both key and value. Denoting the output of each sentence S after the multi-headed self-attentive mechanism as $M = \{m_{i,1}, m_{i,2}, \dots, m_{i,N_s}\}$, the vector representation of the document can be obtained as:

$$D_i = MaxPooling\{m_{i,1}, m_{i,2}, \dots, m_{i,N_s}\} \quad (11)$$

Afterwards, a multi-layer transformer network can be used to obtain the document vector with the full exchange of text information. After the computation of multi-headed attention on S , the computed results are fed into the classifier for classification. Denote V the set of all event types, for any event type $v \in V$, a trainable fully connected layer W_v is defined to classify the document vector $D = \{D_1, D_2, \dots, D_N\}$. The probability of triggering event type v is $p^{(v)} \in \mathbb{R}^{N_t \times 2}$.

$$p^{(v)} = \log softmax(D \cdot W_v) \in \mathbb{R}^{N_t \times 2} \quad (12)$$

where $W_v \in \mathbb{R}^{d_m}$ is a trainable parameter. A $label_{t,v} \in \mathbb{R}^{N_t \times 1}$ can be generated for all $v \in V$ using the real situation of events. Finally, the cross-entropy loss function of the event type detection module is:

$$Loss_{detect} = \sum_{v \in V} \frac{1}{N_t} \sum_{i=1}^{N_d} - \ln \left(P_{i, label_{t,v}}^{(v)} \right) \quad (13)$$

3.4. Reciprocal Feedback Argument Extraction

Multiple events can exist in a document, and each event may have multiple event arguments. Furthermore, the same argument can serve as a role for different events. However, existing methods largely overlook the relationships and interactions among these multiple events. Previous approaches extract the arguments either simultaneously or in a pre-defined role order, without considering the

impact of the extraction order on the roles of the arguments.

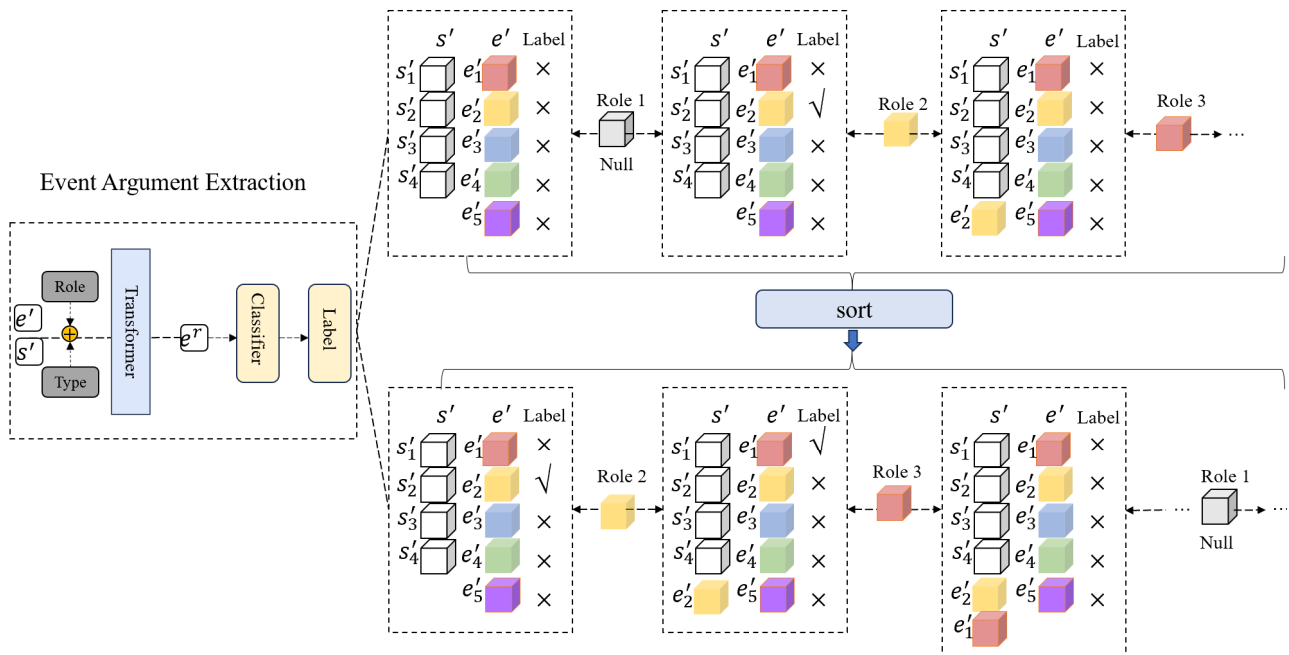


Figure 4. Reciprocal Feedback Argument Extractor.

We propose the Reciprocal Feedback Argument Extraction strategy to extract event arguments, as illustrated in Figure 4. This strategy involves sorting the already extracted argument roles in descending order, traversing the sorted arguments, and processing the corresponding event roles. The feedback process is carried out iteratively. In each iteration, we provide forward feedback based on the knowledge of the previously extracted argument roles, as well as backward feedback based on newly acquired information. This iterative process facilitates the transfer and interaction of information between the argument roles, thereby enhancing the decision-making and extraction accuracy of the arguments. Feedback representation of the argument roles:

$$r_i^c = \frac{\exp(W_b \cdot \tanh(W_a[h_i; u_c]))}{\sum_{j=1}^n \exp(W_b \cdot \tanh(W_a[h_j; u_c]))} \quad (14)$$

where h_i, h_j are each hidden layer state, using the trainable vector u_c to represent the features of its roles, r_i^c is the attention score corresponding to the list of theoretical argument roles, and W_a, W_b are the trainable matrices.

The updated list of roles is obtained by sorting according to the updated scores $r_i' = \{r_1', r_2', \dots, r_n'\}$. In this way, in each iteration, the roles are sorted according to their scores. It can ensure that the high scoring argument roles are prioritized in the next iteration, thus further improving the decision and extraction accuracy of the arguments. In identifying the argument of the k role of the event, a new representation of the entity is obtained for each entity by adding the role name embedding:

$$\bar{E} = E + Role_k \quad (15)$$

where $Role_k$ refers to the embedding of the k role name, and \bar{E} is the new entity feature matrix. Then we decode event arguments by performing path extensions on the argument role extraction. Also,

in order to model the dependencies between events and extract event records for specific event types, we decode the records with an expanded tree. Starting from the virtual root node, multiple branches might be generated during the node expansion due to the possible existence of multiple eligible entities for the event roles. In this way, each path can be considered as a set of arguments of this event. For an event-argument path consisting of a sequence of entities, the entities in the route are stitched together to obtain a representation of the path $P_i = [E_{i_1}, \dots, \dots, E_{i_e}]$, which is encoded using the Bi-directional Long Short-Term Memory (BiLSTM) [53], adding the embedding transformation of the event type into a vector T and a sentence feature vector S , which are stored in global storage G , which is shared among the different event types. They are then stitched together and input into Transformer to obtain a new entity feature matrix $E' \in \mathbb{R}^{d_m \times |e|}$.

$$[E', S', P_i', G'] = \text{Transformer}([\bar{E}, S, P_i, G]) \quad (16)$$

For each argument role domain R under each event type, the model predefines a trainable fully connected layer W_r to bifurcate the candidate arguments composed of all entities \bar{E} and determine whether to extract the candidate arguments into the current argument role domain.

$$P_e = \text{FFN}(E') \quad (17)$$

Minimize the following losses during training:

$$\text{Loss}_{record} = - \sum_{n \in N_Q} \sum_{t=1}^{|e|} \log P(y_t^n | n) \quad (18)$$

where N_Q is the set of nodes in the path and y_t^n refers to the gold label.

3.5. Training

The total loss of the model in this paper is shown:

$$\text{Loss}_{all} = \lambda_1 \text{Loss}_{ner} + \lambda_2 \text{Loss}_{detect} + \lambda_3 \text{Loss}_{record} \quad (19)$$

where $\lambda_1, \lambda_2, \lambda_3$ are hyperparameters, Loss_{ner} is the loss of entity extraction, Loss_{detect} is the loss of event type detection, Loss_{record} is the loss of event record extraction, and then the model is optimized as a whole by selecting an optimizer and setting a reasonable learning rate to train the model to achieve the document-level event extraction task.

4. Results and discussion

4.1. Data set

We utilize the individually constructed cosmetic event information from the cosmetic domain to create our experimental dataset, named COSM. The self-constructed dataset consists of three predefined event types: Adverse Reactions (AR), Invest (IV), and Cooperate (CP). Table 1 provides detailed information about the COSM dataset. In addition, we also utilize a public dataset in this paper, which is derived from the Chinese document-level financial event dataset ChFinAnn [35]. This dataset, provided by Doc2EDAG, comprises 32,040 documents and 35 event elements. These event elements span five types of events: Equity Freeze (EF), Equity Repurchase (ER), Equity Underweight (EU),

Equity Overweight (EO), and Equity Pledge (EP). Approximately 30% of the articles involve multiple events. To conduct our experiments, we divide the dataset into a training set, a validation set, and a test set in an 8:1:1 ratio.

Table 1. COSM dataset description.

Event	AR	IV	CP
Statistics	862	1335	1104

4.2. Experimental environment

In this paper, the following parameter settings are used on both datasets. The encoder for the sentence-level entity extraction model is an 8-layer transformer. The sentence input length is set to 128, with a total of 64 sentence inputs. The dimension of the hidden layer and the fully connected layer are 768 and 1024. Additionally, the model utilized 3 GCN layers. The training batch size is set to 64, and the training process run for 100 epochs. The learning rate is set to 10^{-4} , λ_1 is set to 0.05, λ_2 and λ_3 are set to 1. The optimizer used in this study is Adam [54], with a learning rate of 3×10^{-5} .

The following evaluation criteria were used in this paper. Specifically, for all golden events in each chapter, the predicted events with the same event type and the highest number of correct roles and arguments were found using a non-relaxation approach. This is used as the model's prediction result to calculate precision (P), recall (R), and F1 measure (F1 score). Since the event type usually includes multiple actors, the Micro-F1 value at the actor level is calculated as the final metric.

To determine the optimal parameter values, the relationship between epochs and loss values is depicted in Figure 5. Throughout the training process, the model demonstrates commendable performance on both the ChFinAnn and COSM datasets, with notable improvements achieved after approximately 90 epochs. As the number of epochs increases further, the training and validation loss values gradually stabilize. This observation suggests that our model has successfully converged to a stable state.

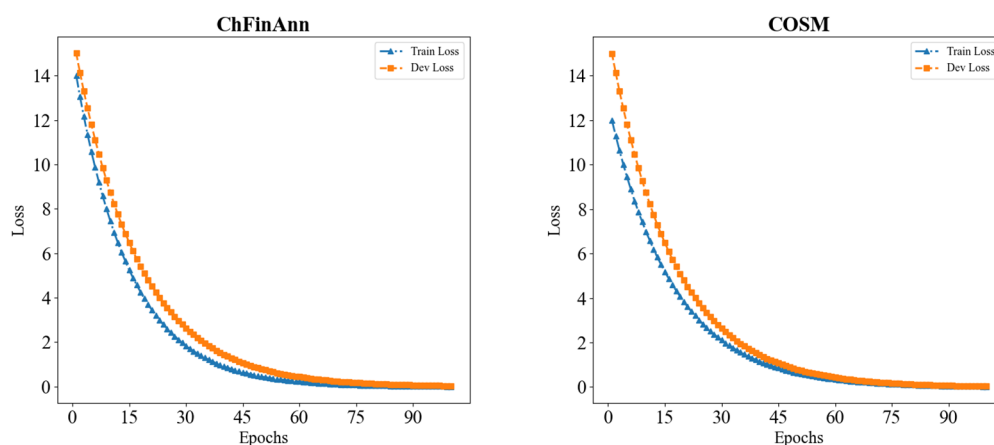


Figure 5. The loss curves of ChFinAnn and COSM datasets.

4.3. Experimental results

4.3.1. Single event and multi-event results

In order to validate the performance of our model, we performed a comparative analysis of several baseline models of the DEE task:

- DCFEE [44] reduces the DEE task to a SEE task by extracting arguments from specific core sentences while looking for missing arguments in neighboring sentences. The model is divided into two versions: DCFEE-S, which extracts arguments from a single key sentence, and DCFEE-M, which generates various potential combinations of arguments based on distance.

- Doc2EDAG [45] uses different converters to obtain sentence and entity embeddings and fuse entity and sentence information to convert the theorem recognition task into an entity-based path extension task that populates the event table.

- GreedyDec: A simple decoding baseline model of Doc2EDAG that only greedily populates an event table entry by using identified entity roles to verify the necessity of end-to-end modeling.

- GIT [46]: The model designs a Heterogeneous Graph Interaction Network to describe the global interactions between sentences. In order to reduce the complexity of extracting relevant events, an additional Tracker module is introduced to record the extracted events.

- IATRF (ours): Using Heterogeneous Graph to strengthen the connection between document and entity information, entity and sentence representations with document-level context-awareness are obtained by GCN. Reciprocal Feedback Argument Extraction is introduced to achieve argument extraction, and a Transformer classifier is used for multi-label classification.

For the validation of the effect of multiple events at the document level, we categorize the ChFinAnn dataset into single events (S.) and multiple events (M.). We label the event types based on the document index, and when a document involves only one event type, we label it as a single event. On the contrary, if the document involves multiple events of the same or different types, we label it as a multiple event. See Table 2 for experimental results.

Table 2. F1 scores on single-record (S.) and multi-record (M.) sets.

Model	EF		ER		EU		EO		EP		Overall	
	S.	M.	S.	M.	S.	M.	S.	M.	S.	M.	S.	M.
	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)
DCFEE-S	55.7	38.1	83.0	55.5	52.3	41.4	49.2	43.6	62.4	52.2	69.0	50.3
DCFEE-M	45.3	40.5	76.1	50.6	48.3	43.1	45.7	43.3	58.1	51.2	63.2	49.4
Greedy-Dec	74.0	40.7	82.2	50.0	61.5	35.6	63.4	29.4	78.6	36.5	77.8	37.0
Doc2EDAG	79.7	63.3	90.4	70.7	74.7	63.3	76.1	70.2	84.3	69.3	81.0	67.4
GIT	81.9	65.9	93.0	71.7	82.0	64.1	80.9	70.6	85.0	73.5	87.6	72.3
IATRF	83.5	67.2	94.8	73.1	83.6	65.5	82.5	72.0	86.4	74.8	88.6	73.7

From the above tables we can conclude the following:

- It is evident that the F1 scores for both IATRF and the comparison models are lower for multi-event extraction compared to single-event extraction. This suggests that extracting multiple events in document-level event extraction poses a greater challenge, resulting in significantly lower overall performance for all models.

• In terms of performance improvement, IATRF exhibits a 1.0% and 1.4% increase in F1 scores for single and multiple events, respectively. The entity-oriented approach utilized in this paper incorporates GCN to model document, sentence, and entity features. This methodology effectively captures long-range dependencies between different nodes in a document and aggregates interaction information through GCN, ultimately yielding context-aware representations of entities and sentences at the document level. Consequently, it performs better in multi-event extraction scenarios.

• By employing the Reciprocal Feedback Argument Extraction strategy, IATRF demonstrates improved F1 scores compared to the GIT model for both single and multiple events across various event types. This strategy leverages previously extracted argument knowledge to determine the roles of challenging arguments individually. Additionally, the incorporation of event role-specific information and event type information into the entity representation allows for sharing of such information across event types, leading to more accurate event role predictions. Furthermore, the introduction of a Transformer classifier enhances event detection accuracy, particularly when extracting events in multi-event records.

4.3.2. Results for different event types and cross-sentence event records

This experiment compares the five baseline models described with the IATRF model across all event types, and the results are presented in Table 3.

Table 3. F1 scores for the five event types in the ChFinAnn dataset.

Model	EF (%)	ER (%)	EU (%)	EO (%)	EP (%)	Overall (%)
DCFEE-S	46.7	80.0	47.5	46.7	56.1	60.3
DCFEE-M	42.7	73.3	45.8	44.6	53.8	56.6
Greedy-Dec	57.7	79.4	51.2	50.0	54.2	61.0
Doc2EDAG	71.0	88.4	69.8	73.5	74.8	77.5
GIT	73.4	90.8	74.3	76.3	77.7	80.3
IATRF	74.9	92.8	75.7	77.8	79.3	81.9

To verify the effectiveness of IATRF in capturing cross-sentence information. The average number of sentences involved in the records of each document in the dataset ChFinAnn is first calculated sorted in ascending order. Then they are divided into four groups of equal size I/II/III/IV. The experimental results are shown in Table 4.

Table 4. F1 scores on four sets with growing average number of involved sentences for records.

Model	I (%)	II (%)	III (%)	IV (%)
DCFEE-S	64.6	70.0	57.7	52.3
DCFEE-M	54.8	54.1	51.5	47.1
Greedy-Dec	67.4	68.0	60.8	50.2
Doc2EDAG	79.6	82.4	78.4	72.0
GIT	81.9	85.7	80.0	75.7
IATRF	83.5	87.3	81.6	77.2

The conclusions drawn from the above tables are as follows:

- Table 3 demonstrates the superior performance of IATRF compared to the baselines. The proposed approach consistently outperforms all the baselines, exhibiting an overall F1 improvement of 1.6% when compared to GIT. Specifically, IATRF achieves F1 scores improvements of 1.5%, 2.0%, 1.4%, and 1.5% across the five event types EF, ER, EU, EO, and EP, respectively. The success of IATRF can be attributed to its effective modeling of global interactions and interdependencies.

- Table 4 reveals that when compared to the GIT model, IATRF exhibits F1 scores improvements of 1.6%, 1.6%, 1.6%, and 1.5% for event records I to IV, respectively. By incorporating GCN, IATRF effectively avoids the challenge of extracting event information across multiple sentences and enhances the extraction of exchanged text information between sentence events by considering contextual semantics. The five types of edges constructed by IATRF play a vital role in extracting event records involving multiple sentences, consequently improving the efficacy of event extraction.

4.3.3. Experimental results on the Cosm dataset

To validate the comprehensiveness of our proposed model, we performed experiments on our self-constructed dataset COSM and compared it with the five baseline models. The experimental results are presented in Table 5.

Table 5. Experimental results on the Cosm dataset.

Model	P (%)	R (%)	F1 (%)
DCFEE-S	59.4	57.3	58.1
DCFEE-M	62.2	59.7	61.8
Greedy-Dec	61.3	60.2	59.9
Doc2EDAG	74.8	71.9	73.4
GIT	81.1	80.3	79.8
IATRF	83.6	81.4	82.5

The experimental results demonstrate that our model also achieves superior performance on the self-constructed dataset COSM. Notably, there is a substantial increase in P scores and R scores by 2.5% and 1.1% respectively. Moreover, we observe a remarkable improvement of 2.7% in F1 scores. This further validates the effectiveness of our proposed method in extracting opinion information about cosmetic events from social media.

4.4. Ablation study

4.4.1. Event record extraction experiments

In this section, we set up two different ablation experiments to verify the effect of IATRF. The “-Path” experiment indicates the removal of path information from the event record. The “-Global storage” experiment clears the information interaction among event records of different types. By employing these specific settings, we are able to assess the influence of path information records and the global storage module on event record extraction.

Table 6. Performance of event record extraction experiment.

Model	P (%)	R (%)	F1 (%)	S. (%)	M. (%)
IATRF	83.9	80.0	81.9	88.6	73.7
-Path	83.2	79.5	81.3	87.9	72.9
-Global storage	82.9	78.2	80.5	87.5	72.2

As indicated in Table 6, there has been an overall increase of 0.7% and 1.0% in P scores, a boost of 0.5% and 1.8% in R scores, and an improvement of 0.6% and 1.4% in F1 scores, respectively. This outcome underscores the effectiveness of storing records from various event types and facilitating global queries that can be shared across these types. By expanding the event record through path information, we enable better event role prediction and achieve more successful event extraction from multiple event records.

4.4.2. Experiments with different classifiers

We conduct ablation experiments to verify the effectiveness of the Transformer classifier on overall performance. In the “Only_Attention” experiment, we rely solely on the self-attentive mechanism as a classifier. By doing so, we are able to evaluate the importance of the Transformer classifier for event classification performance and assess how well the model, relying only on the self-attention mechanism, performed on this task. In the “Only_Sigmoid” experiment, we use only the Sigmoid as the activation function while keeping the other components unchanged. This allows us to study the effect of using only the Sigmoid activation function on event classification performance. Through this analysis, we aim to understand the advantages and disadvantages of different activation functions for this task. Lastly, in the “Only_Linear” experiment, we use only the linear model as a linear classifier, without any activation function, while leaving the other components unchanged. This experiment enables us to evaluate the impact of a linear model without any activation function on event classification performance.

Table 7. Experiments on the performance of different classifiers.

Model	P (%)	R (%)	F1 (%)	S. (%)	M. (%)
IATRF	83.9	80.0	81.9	88.6	73.7
Only_Attention	82.9	78.4	80.7	87.6	72.2
Only_sigmoid	82.3	78.4	80.3	87.6	72.3
Only_Linear	81.9	76.8	79.5	86.9	70.7

According to the results presented in Table 7, it is evident that the inclusion of the transformer classifier yields the best performance, with overall F1 scores of 1.2%, 1.6%, and 2.4% higher than the other three classifiers, P scores of 1.0%, 1.6%, and 2.0%, and R scores of 1.6%, 1.6%, and 3.2%, respectively, and especially for multiple event types, the F1 scores are all higher than the other three classifiers. These findings highlight the pivotal role played by the transformer classifier in achieving event detection. By leveraging its ability to classify events with multiple labels, the predictive capability of the transformer classifier is significantly enhanced, resulting in more effective event detection.

4.4.3. Experiments with different sorting rules

In this section, we establish various ordering rules. Previous approaches have either extracted arguments simultaneously or followed a predefined role order, without taking into account the impact of the extraction order on argument recognition. As illustrated in Figure 6, where “Regular” represents sorting in a fixed order, this ablation experiment clearly demonstrates the influence of argument role extraction order on argument recognition. It can be seen that overall P, R, and F1 scores improved by 2.0%, 3.1%, and 2.5% compared to sorting according to a fixed order, and that F1 scores improved by 1.9% and 2.9% in single and multiple events, respectively. Therefore, by introducing the Reciprocal Feedback Argument Extraction strategy and ranking the already-extracted argument roles, this bidirectional feedback process empowers us to effectively identify argument roles using the relationships between arguments, thereby enhancing the accuracy of argument extraction.

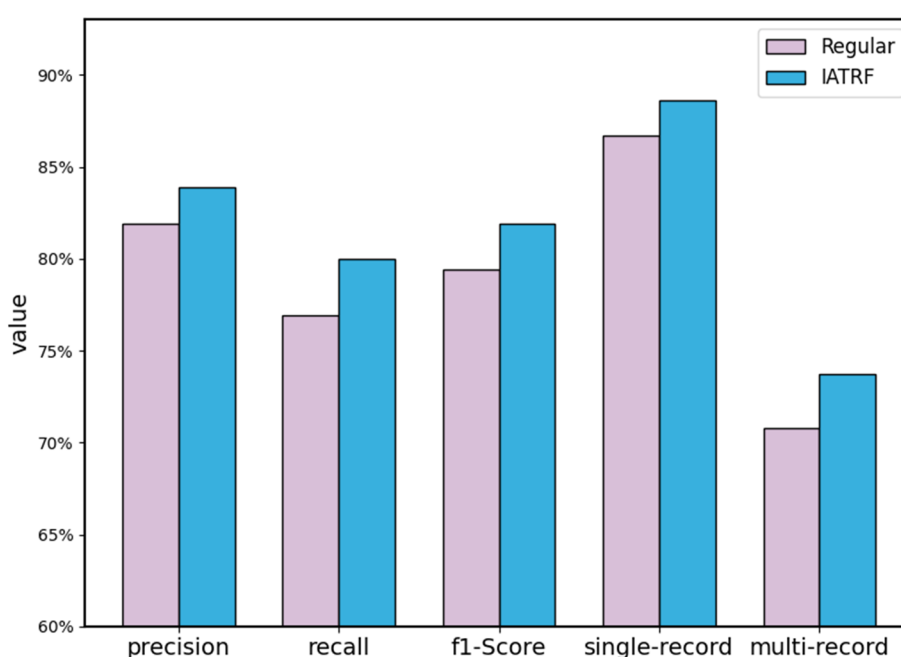


Figure 6. Experiments on different sorting rules.

4.4.4. Graphical Neural Network ablation experiment

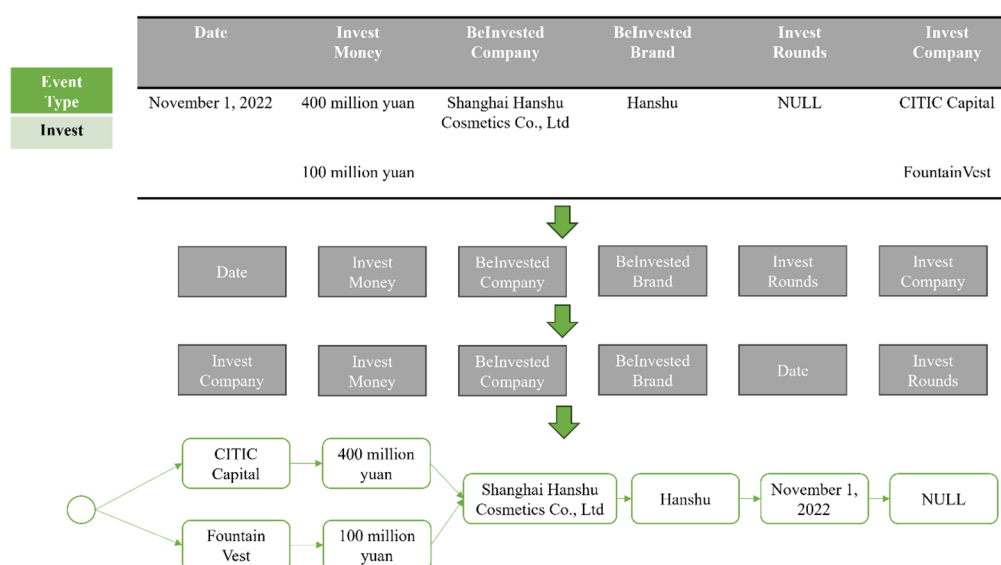
To investigate the impact of the heterogeneous graph, we conduct experiments by progressively removing one type of edge and ultimately eliminating the entire GCN. The results, presented in Table 8, demonstrate varying improvements in F1 values for different event records, labeled as I to IV. It is evident that the removal of the GCN leads to a significant decrease of 2.5% in records involving multiple sentences. This highlights the crucial role of the heterogeneous graph in enhancing cross-sentence event extraction. Furthermore, the five types of edges play a vital role in effectively extracting textual information. The incorporation of document nodes, sentence nodes, entity nodes, and the construction of these diverse edges within the graph neural network strengthen the connection between document and entity information. This enables the model to better capture long-range document-aware representations, incorporate semantic context information, and ultimately enhance overall performance.

Table 8. F1 scores on ablation study for heterogeneous graph interaction network.

Model	F1 (%)	I (%)	II (%)	III (%)	IV (%)
IATRF	81.9	83.5	87.3	81.6	77.2
-(S-S Edge)	80.5	82.5	87.1	79.5	74.9
-(S-E Edge)	80.6	81.8	85.5	80.5	76.5
-(Intra-E-E Edge)	80.8	82.9	85.8	80.2	75.2
-(Inter-E-E Edge)	80.6	82.8	85.9	79.2	75.6
-(D-N Edge)	80.5	82.5	87.1	79.5	74.9
-Graph	79.9	81.7	85.8	79.6	74.7

5. Case study and discussion

This section presents a case study highlighting the functionality of our model. Following event type detection, we utilize an ordered spanning tree to decode documents that contain multiple event records. This enables us to extract event records of a specific type. Additionally, we perform parameter role recognition by dynamically adjusting the detection order based on the obtained vector representation and the labels of all candidate entities in the text. As illustrated in Figure 7, the event type identified in the document is “Invest” and resulting in the generation of two event records. The process of populating event records involves the analysis of six argument roles. Among them, “Invest Company” and “Invest Money” have two event arguments, while “BeInvested Company”, “BeInvested Brand” and “Date” each have one event argument. On the other hand, “Invest Rounds” does not possess any event arguments. During the event record population process, priority is given to identifying parameter roles associated with a higher number of parameters. Subsequently, emphasis is gradually shifted towards roles with fewer event parameters. In each iteration, we help identify a smaller number of argument parameters based on the parameters of the extracted argument roles. Such a dynamic ordering process allows information to be passed and interacted with each other between the argument roles.

**Figure 7.** Cases generated by event records.

From the experimental results, we have made the following observations: Our model uses document nodes to effectively integrate information from the entire document. We also employ transformer classifiers, which allow for a more precise determination of the document's topic. Moreover, when the dataset contains numerous event types, we implement dynamic ordering of argument roles for event parameter extraction. This involves giving priority to the more easily recognizable argument roles. This approach significantly enhances the model's ability to handle complex documents.

6. Conclusions

To enhance the precision of event detection, we utilize heterogeneous graphs to represent the interactions between various entities in a document and capture its perceptual features. Our incorporation of the transformer classifier augments event classification precision. Additionally, we introduce the Reciprocal Feedback Argument Extraction strategy, tailored to dynamically order argument roles. This approach leverages extracted argument knowledge to aid in identifying challenging argument roles that are difficult to recognize independently, thereby improving argument recognition accuracy. We undertook experiments in three distinct dimensions to attest to our approach's efficacy: single event, multi-event, and cross-sentence event records, juxtaposing our findings against the benchmark model. This advantage is also demonstrated on the COSM dataset demonstrating the best results. Empirical findings reveal that IATRF surpasses the comparative model in performance across all event categorizations. It also solves the problems of meta-dispersion and multiple events in public opinion events more effectively.

In our future research, we will consider syntactic structure and semantic roles in the context of linguistic features of opinion events. Semantic roles play a crucial role in identifying predicates within a sentence and determining their associated arguments. By assigning role labels to each argument, we can unveil the semantic relationships between different components within the sentence. To conduct a thorough event extraction study of opinion news, we will employ the method of semantic role analysis. This involves extracting the structure of the sentence based on predicate verb-centered argument elements, transforming them into specific semantic roles, and matching them with the event elements. This comprehensive approach aims to enhance the model's performance by improving event extraction capabilities.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by the National Key Technology R&D Program of China (Grant No. 2021YFD2100605); the Natural Science Foundation of China (Grant No. 62006008); Project of Beijing Municipal University Teacher Team Construction Support Plan (Grant No. BPHR20220104); the Humanity and Social Science Youth Foundation of Ministry of Education of China (Grant No. 20YJCZH229); the IFLYTEK University Intelligent Teaching Innovation Research Special Project (Grant No. 2022XF055).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. X. Wu, J. Wu, X. Fu, J. Li, P. Zhou, X. Jiang, Automatic knowledge graph construction: A report on the 2019 icdm/icbk contest, in *2019 IEEE International Conference on Data Mining (ICDM)*, (2019), 1540–1545. <https://doi.org/10.1109/ICDM.2019.00204>
2. Z. Chen, H. Yu, J. Li, X. Luo, Entity representation by neighboring relations topology for inductive relation prediction, in *PRICAI 2022: Trends in Artificial Intelligence*, Springer, (2022), 59–72. https://doi.org/10.1007/978-3-031-20865-2_5
3. C. Y. Liu, C. Zhou, J. Wu, H. Xie, Y. Hu, L. Guo, CPMF: A collective pairwise matrix factorization model for upcoming event recommendation, in *2017 International Joint Conference on Neural Networks (IJCNN)*, (2017), 1532–1539. <https://doi.org/10.1109/IJCNN.2017.7966033>
4. L. Gao, J. Wu, Z. Qiao, C. Zhou, H. Yang, Y. Hu, Collaborative social group influence for event recommendation, in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, (2016), 1941–1944. <https://doi.org/10.1145/2983323.2983879>
5. J. Liu, Y. Chen, K. Liu, W. Bi, X. Liu, Event extraction as machine reading comprehension, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2020), 1641–1651. <https://doi.org/10.18653/v1/2020.emnlp-main.128>
6. F. Li, W. Peng, Y. Chen, Q. Wang, L. Pan, Y. Lyu, et al., Event extraction as multi-turn question answering, in *Findings of the Association for Computational Linguistics: EMNLP 2020*, (2020), 829–838. <https://doi.org/10.18653/v1/2020.findings-emnlp.73>
7. X. Ma, J. Wu, S. Xue, J. Yang, C. Zhou, Q. Z. Sheng, et al., A comprehensive survey on graph anomaly detection with deep learning, *IEEE Trans. Knowl. Data Eng.*, **2021** (2021). <https://doi.org/10.1109/tkde.2021.3118815>
8. L. Li, L. Jin, Z. Zhang, Q. Liu, X. Sun, H. Wang, Graph convolution over multiple latent context-aware graph structures for event detection, *IEEE Access*, **8** (2020), 171435–171446. <https://doi.org/10.1109/access.2020.3024872>
9. Y. Diao, H. Lin, L. Yang, X. Fan, D. Wu, Z. Yang, et al., FBSN: A hybrid fine-grained neural network for biomedical event trigger identification, *Neurocomputing*, **381** (2020), 105–112. <https://doi.org/10.1016/j.neucom.2019.09.042>
10. W. Yu, M. Yi, X. Huang, X. Yi, Q. Yuan, Make it directly: Event extraction based on tree-LSTM and Bi-GRU, *IEEE Access*, **8** (2020), 14344–14354. <https://doi.org/10.1109/access.2020.2965964>
11. L. Huang, H. Ji, K. Cho, C. R. Voss, Zero-shot transfer learning for event extraction, *arXiv preprint*, (2017), arXiv:1707.01066. <https://doi.org/10.48550/arXiv.1707.01066>
12. W. Shi, F. Li, J. Li, H. Fei, D. Ji, Effective token graph modeling using a novel labeling strategy for structured sentiment analysis, in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, (2022), 4232–4241.
13. Y. Wang, N. Xia, X. Luo, H. Yu, Event extraction based on the fusion of dynamic prompt information and multi-dimensional features, in *2023 International Joint Conference on Neural Networks (IJCNN)*, (2023), 1–9. <https://doi.org/10.1109/IJCNN54540.2023.10191308>

14. Z. Zhao, H. Yu, X. Luo, J. Gao, X. Xu, S. Guo, Ia-icgcn: Integrating prior knowledge via intra-event association and inter-event causality for chinese causal event extraction, in *Artificial Neural Networks and Machine Learning–ICANN 2022*, (2022), 519–531. https://doi.org/10.1007/978-3-031-15931-2_43
15. H. Zhang, D. Zhang, Z. Wei, Y. Li, S. Wu, Z. Mao, et al., Analysis of public opinion on food safety in Greater China with big data and machine learning, *Curr. Res. Food Sci.*, **6** (2023), 100468. <https://doi.org/10.1016/j.crfs.2023.100468>
16. M. Siegrist, C. Hartmann, Consumer acceptance of novel food technologies, *Nat. Food*, **1** (2020), 343–350. <https://doi.org/10.1038/s43016-020-0094-x>
17. M. Zuo, Y. Wang, W. Dong, Q. Zhang, Y. Cai, J. Kong, Visual description augmented integration network for multimodal entity and relation extraction, *Appl. Sci.*, **13** (2023), 6178. <https://doi.org/10.3390/app13106178>
18. W. Lu, D. Roth, Automatic event extraction with structured preference modeling, in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, (2012), 835–844.
19. H. Fei, Y. Ren, D. Ji, Boundaries and edges rethinking: An end-to-end neural model for overlapping entity relation extraction, *Management*, **57** (2020), 102311. <https://doi.org/10.1016/j.ipm.2020.102311>
20. T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, *arXiv preprint*, (2016), arXiv:1609.02907. <https://doi.org/10.48550/arXiv.1609.02907>
21. K. Shalini, H. B. Ganesh, M. A. Kumar, K. Soman, Sentiment analysis for code-mixed Indian social media text with distributed representation, in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, (2018), 1126–1131. <https://doi.org/10.1109/ICACCI.2018.8554835>
22. R. Zhao, K. Mao, Fuzzy bag-of-words model for document representation, *IEEE Trans. Fuzzy Syst.*, **26** (2018), 794–804. <https://doi.org/10.1109/tfuzz.2017.2690222>
23. L. Zhang, S. Wang, B. Liu, Deep learning for sentiment analysis: A survey, *WIREs Data Min. Knowl. Discovery*, **8** (2018), e1253. <https://doi.org/10.1002/widm.1253>
24. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, *arXiv preprint*, (2013), arXiv:1301.3781. <https://doi.org/10.48550/arXiv.1301.3781>
25. J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2014), 1532–1543. <https://doi.org/10.3115/v1/D14-1162>
26. M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, et al., Deep contextualized word representations, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (2018), 2227–2237.
27. J. Devlin, M. W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint*, (2018), arXiv:1810.04805. <https://doi.org/10.48550/arXiv.1810.04805>
28. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, (2023), arXiv:1706.03762. <https://doi.org/10.48550/arXiv.1706.03762>
29. S. Liu, Y. Chen, S. He, K. Liu, J. Zhao, Leveraging framenet to improve automatic event detection, in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, (2016), 2134–2143. <https://doi.org/10.18653/v1/P16-1201>

30. Y. Hong, J. Zhang, B. Ma, J. Yao, G. Zhou, Q. Zhu, Using cross-entity inference to improve event extraction, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (2011), 1127–1136.
31. H. Fei, F. Li, B. Li, D. Ji, Encoder-decoder based unified semantic role labeling with label-aware syntax, in *Proceedings of the AAAI Conference on Artificial Intelligence*, (2021), 12794–12802. <https://doi.org/10.1609/aaai.v35i14.17514>
32. J. Li, H. Fei, J. Liu, S. Wu, M. Zhang, C. Teng, et al., Unified named entity recognition as word-word relation classification, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **36** (2022), 10965–10973.
33. Y. Chen, L. Xu, K. Liu, D. Zeng, J. Zhao, Event extraction via dynamic multi-pooling convolutional neural networks, in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, (2015), 167–176. <https://doi.org/10.3115/v1/P15-1017>
34. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE*, **86** (1998), 2278–2324. <https://doi.org/10.1109/5.726791>
35. T. H. Nguyen, K. Cho, R. Grishman, Joint event extraction via recurrent neural networks, in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (2016), 300–309. <https://doi.org/10.18653/v1/N16-1034>
36. K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, et al., Learning phrase representations using RNN encoder-decoder for statistical machine translation, *arXiv preprint*, (2014), arXiv:1406.1078. <https://doi.org/10.48550/arXiv.1406.1078>
37. X. Liu, Z. Luo, H. Huang, Jointly multiple events extraction via attention-based graph information aggregation, *arXiv preprint*, (2018), arXiv:1809.09078. <https://doi.org/10.18653/v1/D18-1156>
38. S. Yang, D. Feng, L. Qiao, Z. Kan, D. Li, Exploring pre-trained language models for event extraction and generation, in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (2019), 5284–5294. <https://doi.org/10.18653/v1/P19-1522>
39. X. Du, C. Cardie, Event extraction by answering (almost) natural questions, *arXiv preprint*, (2020), arXiv:2004.13625. <https://doi.org/10.48550/arXiv.2004.13625>
40. Y. Zhou, Y. Chen, J. Zhao, Y. Wu, J. Xu, J. Li, 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*, (2021), 14638–14646. <https://doi.org/10.1609/aaai.v35i16.17720>
41. A. P. B. Veyseh, M. Van Nguyen, F. Derroncourt, B. Min, T. Nguyen, Document-level event argument extraction via optimal transport, in *Findings of the Association for Computational Linguistics: ACL 2022*, (2022), 1648–1658. <https://doi.org/10.18653/v1/2022.findings-acl.130>
42. Y. Ren, Y. Cao, F. Fang, P. Guo, Z. Lin, W. Ma, et al., CLIO: Role-interactive Multi-event Head Attention Network for Document-level Event Extraction, in *Proceedings of the 29th International Conference on Computational Linguistics*, (2022), 2504–2514.
43. F. Wang, F. Li, H. Fei, J. Li, S. Wu, F. Su, et al., Entity-centered cross-document relation extraction, in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, (2022), 9871–9881. <https://doi.org/10.48550/arXiv.2210.16541>

44. H. Yang, Y. Chen, K. Liu, Y. Xiao, J. Zhao, Dcfee: A document-level chinese financial event extraction system based on automatically labeled training data, in *Proceedings of ACL 2018, System Demonstrations*, (2018), 50–55. <https://doi.org/10.18653/v1/P18-4009>
45. S. Zheng, W. Cao, W. Xu, J. Bian, Doc2EDAG: An end-to-end document-level framework for Chinese financial event extraction, in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (2019), 337–346.
46. R. Xu, T. Liu, L. Li, B. Chang, Document-level event extraction via heterogeneous graph-based interaction model with a tracker, *arXiv preprint*, (2021), arXiv:2105.14924. <https://doi.org/10.48550/arXiv.2105.14924>
47. H. Yang, D. Sui, Y. Chen, K. Liu, J. Zhao, T. Wang, Document-level event extraction via parallel prediction networks, in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, (2021), 6298–6308. <https://doi.org/10.18653/v1/2021.acl-long.492>
48. Q. Wan, C. Wan, K. Xiao, D. Liu, C. Li, B. Zheng, et al., Joint document-level event extraction via token-token bidirectional event completed graph, in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, (2023), 10481–10492. <https://doi.org/10.18653/v1/2023.acl-long.584>
49. J. Li, K. Xu, F. Li, H. Fei, Y. Ren, D. Ji, 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*, (2021), 1359–1370.
50. Y. Huang, W. Jia, Exploring sentence community for document-level event extraction, in *Findings of the Association for Computational Linguistics: EMNLP 2021*, (2021), 340–351. <https://doi.org/10.18653/v1/2021.findings-emnlp.32>
51. R. Hu, H. Liu, H. Zhou, Role knowledge prompting for document-level event argument extraction, *Appl. Sci.*, **13** (2023), 3041. <https://doi.org/10.3390/app13053041>
52. J. Lafferty, A. McCallum, F. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, in *Proceedings of the Eighteenth International Conference on Machine Learning*, (2001), 282–289.
53. Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF models for sequence tagging, *arXiv preprint*, (2015), arXiv:1508.01991. <https://doi.org/10.48550/arXiv.1508.01991>
54. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint*, (2014), arXiv:1412.6980. <https://doi.org/10.48550/arXiv.1412.6980>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)