Joint Learning Event-specific Probe and Argument Library with Differential Optimization for Document-Level Multi-Event Extraction

Jianpeng Hu [1], Chao Xue [2], Chunqing Yu [1], Jiacheng Xu [1], Chengxiang Tan [1]
[1] College of Electronic and Information Engineering, Tongji University
[2] College of Software, Beihang University
hujianpeng@tongji.edu.cn, xuechao@buaa.edu.cn
{chunqingyu, 2011263, jerrytan}@tongji.edu.cn

Abstract

Document-level multi-event extraction aims to identify a list of event types and corresponding arguments from the document. However, most of the current methods neglect the fine-grained difference among events in multi-event documents, which leads to event confusion and missing. This is also one of the reasons why the recall and F1-score of multi-event recognition are lower compared to single-event recognition. In this paper, we propose an event-specific probe-based method to sniff multiple events by querying each corresponding argument library, which uses a novel probe-label alignment method for differential optimization. In addition, the role contrastive loss and probe consistent loss are designed to fine-tune the fine-grained role differences and probe differences in each event. The experimental results on two general datasets show that our method outperforms the state-of-the-art method in the F1-score, especially in the recall of multi-events.

1 Introduction

The purpose of event extraction is to identify event triggers with specific types from unstructured text and extract arguments related to events. In practice, multiple events are often described in one document, and some common arguments and triggers are shared. These events may not have obvious triggers and their arguments are scattered in multiple sentences. Therefore, the document-level multi-event extraction without event triggers has been widely concerned by scholars (Xu et al., 2022; Zhang et al., 2022).

Existing methods (Wang et al., 2023; Yang et al., 2021; Zhu et al., 2021a; Zheng et al., 2019) work well when there is only one event in the document. When there are multiple events in one document, they cannot accurately find all the events. An important reason for this phenomenon is that most event records in a document tend to describe several events of the same type, and they have similar arguments (further discussion in Appendix A). We define these event records as **homogeneous events**, which have only fine-grained differences. As shown in Fig. 1, we show three homogeneous event records in a multi-event document, which are all events of "EquityPledge" and have mostly the same arguments.

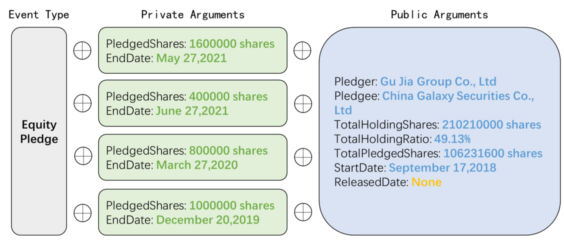

Figure 1: An example of three event records in one document. Each event record is divided into three parts: event type, private arguments, and public arguments. Black bold font is the type of event, and black normal font is the role name of the event record. Private and public arguments are marked green and blue, respectively. The missing role of the event framework is marked yellow.

Unlike the shared events proposed by Gao et al. (2022), it focuses on event-level features for contrastive learning and ignores the fine-grained argument level. Some previous works (Sheng et al., 2021; Wang et al., 2023) have noticed a similar problem of overlapping arguments, but they mainly focus on the problem of one argument playing different roles, which does not solve the event confusion caused by one argument playing the same role in multiple events in homogeneous events. Other methods (Liang et al., 2022; Wang et al., 2023; Xu et al., 2021) often failed to pay attention to the fine-grained differences between multiple events in the document, resulting in these homogeneous events being recognized as one event record or causing confusion in event roles.

To identify multiple events in documents more

Findings of the Association for Computational Linguistics: NAACL 2025, pages 714–726
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

accurately, we propose a method based on event-specific probes and argument libraries to extract multi-events from documents differentially. When the model makes inferences, the probe is regarded as a query of the corresponding role library, using the filling-in method based on the role framework to map the argument and role detected by the probe. During training, to distinguish arguments in homogeneous events at a finer granularity, the arguments that appear in only one event record are defined as private arguments of this event, and those that appear in multiple event records of the same event type are defined as public arguments (as shown in Fig. 1). Obviously, private arguments are the source of fine-grained differences between different event records. Based on the characteristics of private and public arguments, we propose a probe-label alignment optimization algorithm to refine the probe group corresponding to each ground-truth event label or group. By applying differential optimization algorithms to jointly learn probes and libraries, the event confusion caused by the constant change of probe-label correspondence in the training process is effectively reduced. The probe embedding centered on private argument can also better notice the fine-grained differences between event records.

In addition to the main training objectives, we add two auxiliary loss functions to fine-tune the parameters of the model during training, which makes the model learn the differences in event records detected by different probes. These two loss functions fine-tune the model's understanding of different granularity differences from the perspective of probe and role library respectively. Specifically, at the probe level, we use the probe consistent loss to ensure the consistency of probes in terms of privacy and publicity and to distinguish the macro difference between probes in label correspondence. At the role library level, we design the role contrastive loss to help the model understand the microscopic differences of arguments corresponding to the same role of different events.

To sum up, the contributions of this paper mainly include the following three aspects:

- We propose a document-level multi-event extraction framework and optimization algorithm based on event-specific probe and argument library to make the model better extract multiple events from documents.
- We design probe consistent loss and role contrastive loss to help the model understand the differences between multiple event records from both macro and micro perspectives.
- The experimental results on two general datasets show that the comprehensive performance of our model is higher than previous state-of-the-art methods, especially on the recall of multiple events.

2 Related Work

To solve document-level event extraction, a large number of methods have been proposed recently, which can be divided into generative model-based methods and discriminant model-based methods.

Based on generative models, this task is modeled as a question-answering system (Huang and Jia, 2021), reading comprehension (Liu et al., 2021) sequence conversion (Huang et al., 2021), etc. As a classic method, Lu et al. (2021); Hsu et al. (2022); Lin et al. (2022) proposed a sequence-to-structure generation paradigm. Another major strategy of the generative method is to introduce external knowledge(Zhou et al., 2022), which includes AMR information (Li et al., 2022), external knowledge base (Hsu et al., 2023), knowledge of different datasets (Zhang et al., 2023), etc.

Based on discriminant models, this task can be understood as a table-filling task (Zhu et al., 2021b; Wan et al., 2023a; Wang et al., 2022), in which each event is represented as a combination of an argument list and event type. Formally, the event extraction problem can be modeled as link prediction (Yang et al., 2023), subgraph division (Wan et al., 2023b), role classification (Liu et al., 2023b), etc. Most of these methods (Van Nguyen et al., 2022; Ren et al., 2022a; Cao et al., 2022) can perform well in documents with a small number of events, but they cannot extract multiple events differently. To face this challenge, some approaches (Yang et al., 2021; Wang et al., 2023; Liang et al., 2022) proposed event query and role query to extract events in parallel, but they lack fine-grained difference optimization on roles. Ren et al. (2022b) uses comparative learning of roles and arguments to augment the information about the roles corresponding to the arguments, while we distinguish homogeneous events by performing contrastive learning between arguments of multiple events. In summary, we pay more attention to the modeling of the difference between multiple events in the same document.

3 Methodology

3.1 Problem Definition

T and R ($|R| = k$) are defined as the type set and role set of events, respectively. Therefore, an event record e is mathematically defined as a combination of an event type $t \in T$ and a series of roles $\{r_i\} \in R$, with each role corresponding to a specific argument a_j. Formally, given a document D, our goal is to detect every event record $e = (t, \{(r_i, a_j)\})$ in D without event trigger. If r_i is not mentioned in D, then the argument corresponding to r_i is $None$. The set of all event records in D is recorded as $E = \{e\}$. The overview of our method is shown in Fig. 2.

3.2 Entity Recognition

Our first step is to identify entities from D that can be viewed as candidate arguments. This step is framed as a traditional BIO (Beginning and Inside of an entity span, and Other tokens) sequence labeling problem. BERT is used as an encoder for sentences in D and the token embedding of BERT output is passed through a MLP to compute the loss function L_{bio} of the BIO sequence annotation. We employ average pooling for all token embeddings that make up an entity a_i to obtain the hidden representation of entities.

With this step, the model gets the set $\{a_i\}_{i=1}^{n}$ of all entities and the set $C = \{c_i\}_{i=1}^{m}$ of all CLS tokens at the beginning of each segment in D, where n is the number of entities and m is the number of CLS tokens. Since the same entity appearing at different locations in a document may be related to different events, we attach position embedding to that hidden representation and input it into Feed Forward Network (FFN) to obtain $h_{a_i} \in R^{1\times d}$. In the same way, we get the embedding of CLS token h_{c_i}. This position embedding is the absolute position of the entity throughout the document.

3.3 Joint Learning Event Representation

In this section, we use event-specific probes $\{q_i\}$ and argument libraries $\{A_i\}$ to joint learning event representations in documents.

Event-specific Probe To optimize a dedicated probe for each event record, context information associated with each entity is captured by the entity globalization encoder. The globalized entity embeddings are represented as generic probes of the document D. Based on the definition of event record, the number of entities is usually greater than the number of event records described in D. As a result, the number of probes is usually much larger than the number of events. The probe can fully discover all events in D without a fixed upper limit. Each entity may be a potential private argument, which determines the difference between each event and other events, so each event can correspond to a specific probe in the optimization process.

Specifically, the Multi-Head Attention (MHA) mechanism is employed to implement this entity globalization encoder. The query, key, and value of MHA are defined as $Q_{a_i} = h_{a_i}$, $K_c = V_c = \{h_c\}_{c\in C}$. Based on different attention coefficients, MHA can compute different global context information for entities, which are included in CLS embeddings. The globalized embedding of entity information is calculated as:

$$\hat{h}_{a_i} = MHA(Q_{a_i}, K_c, V_c) \quad (1)$$

The global and individual entity embedding are concatenated and then fed into an FFN. We then get the initialized representation of the probe:

$$h_{q_i} = FFN([\hat{h}_{a_i}; h_{a_i}]) \quad (2)$$

The total number of probes in a document is equal to the total number of entities, which can make the model tend to discover more event records in a document.

Event-specific Argument Library Each entity may play a different role in different events. To differentially encode the same entities across events, we build a library of arguments for each probe. Specifically, we use probes as queries and entity mentions as keys and values to encode entities in event-specific argument libraries.

$$h_{q_i,\bar{a}_j} = MHA(h_{q_i}, \{h_a\}_{a\in\bar{a}_j}, \{h_a\}_{a\in\bar{a}_j}) \quad (3)$$

where $\bar{a}_j$ denotes the set of mentions for entity a_j. In practice, there are often missing roles for event objects described in documents. For the identification of missing roles in the event framework, we add a virtual argument to each library. Specifically, the virtual argument is implemented based on CLS embedding from each segment. When a role is populated with CLS, it means that the probe has not explored the role's argument after scanning all the arguments in the document. The virtual argument in library A_i is computed as follows:

$$h_{q_i,cls} = MHA(h_{q_i}, \{h_c\}_{c\in C}, \{h_c\}_{c\in C}) \quad (4)$$

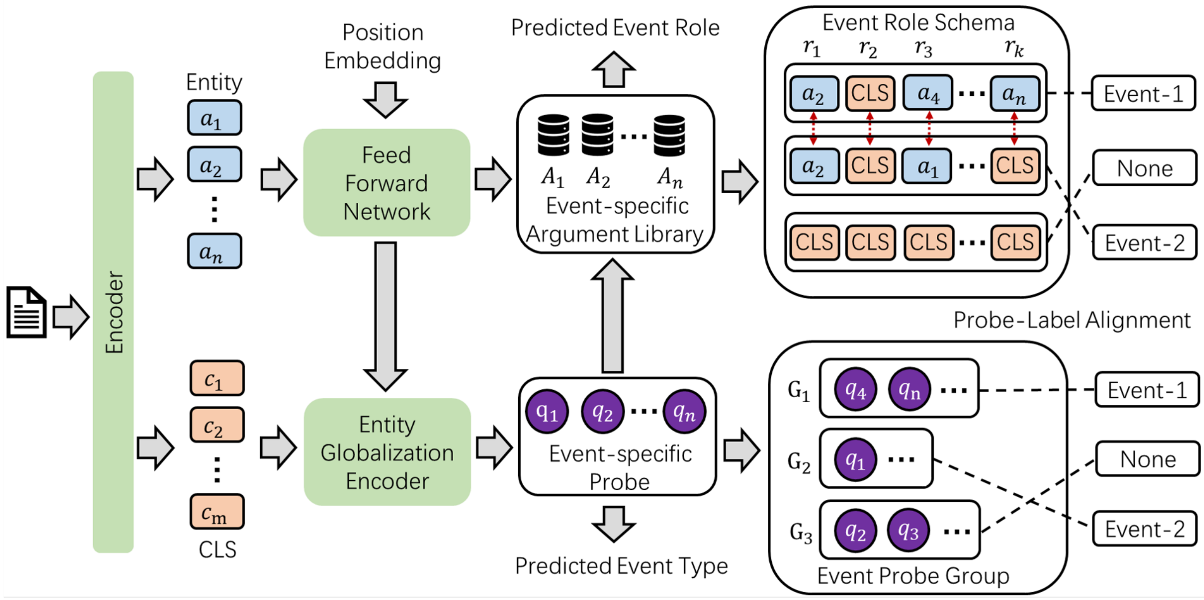

Figure 2: The framework of our method. The black dotted line represents the correspondence between the probe groups and ground-truth labels obtained by the probe-label alignment algorithm. G_3 is aligned with None because the probes in G_3 correspond to a public argument or useless argument. Since probes q_4 and q_n correspond to a private argument of Event-1, G_1 is aligned with Event-1. And the alignment between G_2 and Event-2 is the same as G_1. The red dotted line represents the argument for calculating role contrastive loss, where the corresponding argument pairs are positive if they are the same and negative if they are different. Each row of arguments in the Event Role Schema represents one complete event record.

We take $[h_{q_i}; h_{q_i,\bar{a}_j}]$ and $[h_{q_i}; h_{q_i,cls}]$ as the atomic element constituting the event-specific argument library $A_i \in R^{(n+1)\times 2d}$.

Event Inference After obtaining q_i and A_i , we use a MLP layer to identify the types of events:

$$p_{t,q_i} = softmax(MLP(h_{q_i})) \quad (5)$$

where p_{t,q_i} is the event type probability distribution of q_i. The None label is added to indicate that this probe did not detect any events. To identify the role-argument pairs of events, we use MLP on A_i corresponding to q_i and transpose the resulting matrix. Then the softmax function is computed on the row vectors of $h_{r,q_i} \in R^{k\times(n+1)}$:

$$\begin{aligned} h_{r,q_i} &= MLP(A_i)^T \\ p_{r_j,q_i} &= softmax(h_{r_j,q_i}) \end{aligned} \quad (6)$$

where h_{r_j,q_i} is the row vector of h_{r,q_i} with respect to r_j. If the event type detected by q_i is not None, p_{r_j,q_i} indicates the probability distribution of role r_j in the event detected by q_i. Each step of the above event representation learning can be done in a single matrix operation. Therefore, increasing the number of probes within a certain range will not unduly increase the model inference and training time.

3.4 Differential Optimization

In this section, we present the optimization algorithms for probes and argument libraries.

3.4.1 Probe-Label Alignment Optimization

Recent optimization algorithms for multi-event extraction learn event representations by minimizing the Hausdorff distance between the set of predicted events and gold events. Since the mapping is done from full set to full set, this will lead to instability of the optimization objective during the training process. To constrain this instability and optimize the sniffing ability of probes, the probe-label alignment algorithm is designed, which differentially optimizes the embedding of probes and arguments corresponding to events.

Before discussing the probe-label alignment algorithm, we define the distance between probe q_l and ground-truth event label e_i as:

$$d(q_l, e_i) = CE(p_{t,q_l}, t) + \frac{1}{k}\sum_{j=1}^{k} CE(p_{r_j,q_l}, a_{r_j}) \quad (7)$$

where t is the ground-truth event type of e_i and a_{r_j} is the ground-truth argument of role r_j in e_i, $CE(\cdot)$ denotes the cross-entropy loss. The smaller

$d(q_l, e_i)$ is, the more accurately probe q_l sniffs the entire event record e_i.

We add the definition of useless argument: the entities that have not appeared in all event records. Then, The arguments are divided into three categories: private, public, and useless. Obviously, private arguments are the key to distinguishing an event record from other homogeneous event records in one document, i.e., each group of private arguments corresponds to one event record. Since probes originate from entity embeddings, we can partition the group of probes based on the different types of arguments.

Based on the above definition, all probes are divided into subgroups that correspond to different gold event records. Specifically, the group of probes corresponding to all private arguments of an event is mapped to the record of that event. If there are no private arguments in an event, the group of probes corresponding to the arguments with the least number of occurrences in other events is chosen as the mapping source for that event. This allows each gold event record to have a corresponding probe during training. For useless arguments and unused public arguments, we classify their corresponding probes into another group, which is labeled None. From this, we generate multiple probe subgroups $G = \{G_{e_i}\}_{i=1}^{|E|} \cup G_{None}$, where $|E|$ is the number of events in D, G_{e_i} and G_{None} is the probe subgroup corresponding to the gold event record e_i and None event record respectively.

After obtaining the correspondence between subgroups and gold event records, we minimize the distance between each pair based on Eq.(7), which allows the probes to learn the ability to extract the corresponding event in the document. Specifically, we first determine the correspondence within each probe subgroup G_j to the gold event record e_i:

$$Q_{G_{e_i}} = \{(q_l, e_i)|min(d(q_l, e_i)), q_l \in G_{e_i}\} \quad (8)$$

where each q_l can only have one corresponding e_i, which can prevent confusion caused by multi-objective optimisation of q_l. For other probes in this subgroup, we denote them as $O_{G_{e_i}} = \{q_l|(q_l, \cdot) \notin Q_{G_{e_i}} \cap q_l \in G_{e_i}\}$ and correspond to the event record e_{None}. The event loss function is computed as:

$$L_{G_{e_i}} = \sum_{(q_l,e_i)\in Q_{G_{e_i}}} d(q_l, e_i) + \sum_{q_l\in O_{G_{e_i}}} d(q_l, e_{None})$$
$$L_e = \sum_{i=1}^{|E|} L_{G_{e_i}} + \sum_{q_l\in G_{None}} d(q_l, e_{None}) \quad (9)$$

where $L_{G_{e_i}}$ denotes the loss of all probes in subgroup G_{e_i} sniffing the gold event record e_i. If multiple ground-truth labels correspond to the same probe groups, we combine these labels into one set $\bar{e}$ and calculate the correspondence $Q_{G_{\bar{e}}} = \{(q_l, e_i)|min(d(q_l, e_i)), q_l \in G_j\}_{e_i\in\bar{e}}$. Then use the above formula to calculate $L_{G_{\bar{e}}}$.

3.4.2 Role Contrastive Loss

For the model to notice role differences between these homogeneous events, the role contrastive loss is designed between multiple events of the same document. Specifically, for the same role of two events with the same type, it is viewed as a positive pair if the arguments filling the role are the same, and a negative pair if the arguments are different. The contrastive loss is computed only for roles included in the role framework. Since Virtual arguments are employed to populate the event frame when the document does not describe the corresponding argument for a role, two events with missing arguments for the same role are considered to be a positive pair. In summary, the contrastive loss is calculated as:

$$f(e_i, e_j) = \sum_{u=1, A_{i,u}=A_{j,u}}^{k} exp(\frac{sim(A_{i,u}, A_{j,u})}{\tau})$$
$$g(e_i, e_j) = \sum_{u=1, A_{i,u}\neq A_{j,u}}^{k} exp(\frac{sim(A_{i,u}, A_{j,u})}{\tau})$$
$$L_{rc} = -\frac{1}{N}\sum_{e_i,e_j\in E} log\frac{f(e_i, e_j)}{f(e_i, e_j) + g(e_i, e_j)} \quad (10)$$

where N is the number of event pairs in E, τ denotes temperature coefficient, $A_{i,u}$ is the embedding of the argument corresponding to role r_u in A_i, $sim(\cdot, \cdot)$ denotes the cosine similarity. Through the constraint of L_{rc}, if one argument corresponds to the same roles in e_i and e_j, the corresponding embedding in A_i and A_j will be more and more similar, otherwise, the similarity will be lower and lower. This allows the model to better recognize differences in role granularity for homogeneous events.

3.4.3 Probe Consistent Loss

According to the above definition, each probe corresponds to a private, public, or useless argument. Therefore, the probes in the model can also be divided into these three categories. Different types of probes represent different corresponding ground-truth event labels in the probe-label alignment algorithm. If the probe representation of the same type is inconsistent, it may cause frequent changes in label alignment and instability of the optimization target during training.

To make the model learn the type of probes from embedding and maintain consistency of the same type, the probe consistency loss is introduced to assist the model training, where probe representations will be updated during learning. Specifically, it is a three-classification task, and an MLP network is employed to map the probe embedding:

$$L_{pc} = \sum_{i=1}^{n} CE(MLP(q_i), y_i) \tag{11}$$

where y_i is the label of probe category.

3.4.4 Objective Function

The final loss is a weighted sum of the above loss functions:

$$L = \lambda_{bio} L_{bio} + \lambda_e L_e + \lambda_{rc} L_{rc} + \lambda_{pc} L_{pc} \tag{12}$$

The settings of each hyperparameter in the loss function are detailed in Appendix B.

4 Experiments

4.1 Experimental Setup

Datasets Consistent with previous work (Wang et al., 2023), our experiment will be conducted on two common datasets and be compared to the state-of-the-art approaches. (1) ChFinAnn (Zheng et al., 2019) [1] is a document-level event extraction dataset that includes 32,040 documents. And 71% of these documents contain only one event and 29% contain multiple events. (2) DuEE-Fin (Han et al., 2022) [2] has 11,900 documents. There are 67% of single-event samples and 33% of multi-event samples.

[1]https://github.com/dolphin-zs/Doc2EDAG/blob/master/Data.zip

[2]https://aistudio.baidu.com/aistudio/competition/detail/46/0/task-definition

Evaluation Metrics The evaluation metrics we employed are the same as (Zheng et al., 2019). The micro-average role precision, recall, and F1-score between the predicted event and the selected ground-truth event are used to test the effectiveness of the models. During code copying, we discovered that the measurement code for Wang et al. (2023) ignores the case where the same argument occurs multiple times in one event. This bug has been fixed and we retested its performance on each dataset. The training method and hyper-parameters are given in Appendix B.

Baselines Six baselines are introduced to compare our method (EPAL). **Doc2EDAG** (Zheng et al., 2019) proposes an end-to-end model that can generate entity-based directed acyclic graphs to implement document-level event extraction. **DE-PPN** (Yang et al., 2021) introduces a multi-granular, non-autoregressive decoder to extract structured events from the document in a parallel way. To reduce computational resource consumption, **PT-PCG** (Zhu et al., 2021a) combines the event arguments in a non-autoregressive decoding method with pruned complete graphs. **GIT** (Xu et al., 2021) is an improvement on Doc2EDAG, which adds additional path information to assist the classification task when generating graphs. **ReDEE** (Liang et al., 2022) considers that the relation information of event arguments is important for solving the cross-sentence multi-event problem, and proposes a relation-augmented attention transformer to capture relation dependence. **ProCNet** (Wang et al., 2023) proposes the proxy nodes to extract the events in the document and uses the Hausdorff distance to optimize the difference between the ground-truth events and the events identified by the proxy node.

4.2 Main Results

Table 1 shows the comparison between our model and baselines on the ChFinAnn and DuEE-Fin datasets, respectively. In addition, the evaluation results for each event type are discussed in Appendix C. The results of the baselines in the table are either from paper (Wang et al., 2023) or from open-source code. Based on the experimental results, we can intuitively find that all models perform better on single-event documents than on multi-event documents, which shows that it is more challenging for the model to accurately extract multiple events from documents.

Model	P.	R.	F1	F1(S.)	F1(M.)
ChFinAnn					
Doc2EDAG	82.7	75.2	78.8	83.9	67.3
DE-PPN	83.7	76.4	79.9	85.9	68.4
PTPCG	83.7	75.4	79.4	88.2	–
GIT	82.3	78.4	80.3	87.6	72.3
ReDEE	**83.9**	79.9	81.9	88.7	74.1
ProCNet	83.6	78.1	80.8	87.5	73.5
EPAL (Ours)	83.1	**83.5**	**83.4**	**89.7**	**76.6**
DuEE-Fin					
Doc2EDAG	67.1	60.1	63.4	69.1	58.7
DE-PPN	69.0	33.5	45.1	54.2	21.8
PTPCG	71.0	61.7	66.0	–	–
GIT	69.8	65.9	67.8	73.7	63.8
ReDEE	77.0	72.0	74.4	78.9	70.6
ProCNet	**79.3**	71.4	75.1	80.1	71.0
EPAL (Ours)	77.3	**75.5**	**76.4**	**81.2**	**72.7**

Table 1: Overall precision (P.), recall (R.), and F1-score (F1) on ChFinAnn and DuEE-Fin. F1(S.) and F1(M.) denote scores under Single-event(S.) and Multi-event(M.) sets.

As an early method, Doc2EDAG only uses a directed acyclic graph to detect events, which has limited performance and high time complexity in the training process. Although parallel coding is employed in DE-PPN to greatly reduce the training time, when faced with more complex datasets (such as DuEE-Fin), the recall decreases by about 30%. Through the performance of GIT, it can be found that the improvement of Doc2EDAG by path expansion method mainly increases recall by 3-5%. ReDEE extends the coding method of the transformer to achieve better results on precision, but it does not notice the differences between events, resulting in a relatively conservative exploration of new events. ProCNet uses randomly initialized proxy nodes for global matching learning, which makes the correspondence between proxy and label unstable during training and finally leads to the decline of the F1-score. It can also observed that EPAL gives the best overall F1-score, outperforming the best baseline by 1-3% on ChFinAnn and DuEE-Fin. It is worth noting that the recall of EPAL is 4-5% higher than that of the model in the same period, which indicates that EPAL has a stronger event discovery ability. Because the labeled events in the event extraction dataset tend to have high precision and low recall, it means that some ground-truth events in the dataset are not marked (Zheng et al., 2019). EPAL may have

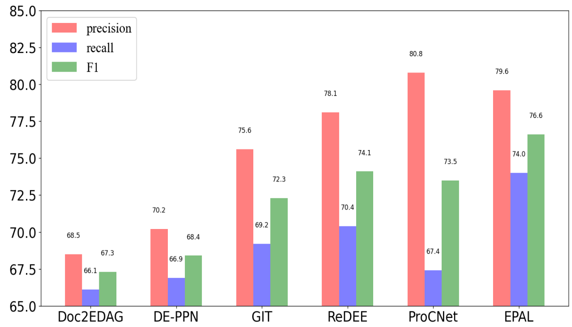

Figure 3: Precision, recall and F1-score of multi-event documents on ChFinAnn.

found some unmarked events in the dataset, resulting in a slightly lower precision of EPAL compared with other models. We will analyze the multi-event extraction capability of EPAL in detail in section 4.3.

4.3 Multi-event Extraction Capability

To validate EPAL's ability to extract multiple events from a document, the precision, recall, and F1-score of the models for multiple events on ChFinAnn are shown in Fig. 3. Obviously, EPAL has the highest Recall and F1-score on multi-event documents, which suggests that it can discover more events and their arguments in a document. Compared to the increase in recall, precision has decreased by only 1.2%, so EPAL has a higher F1-score than previous models on multi-event documents.

As shown in Fig. 3, the improvement of F1-scores of the previous models on multiple events is mainly due to the improvement of precision, while the improvement of recall is small. These methods are more concerned with accurately identifying one single event record in a document and therefore can have significant improvements on single-event documents. But in reality, a document tends to describe multiple events, which makes them have lower recall in multi-event documents. Because they do not notice the differences between different events in multi-event documents, baselines tend to extract homogeneous events as one event. EPAL detects homogeneous events by different probes and finely distinguishes them through the event-specific argument Library, which improves the recall of multiple events by 4-8%. Compared with the model of the same period, the precision and recall of EPAL are more balanced, which means that the model has reached a relatively stable level in predicting the number of events aggressively and conservatively.

Appendix D makes a qualitative error analysis and specifically shows the ability of our model to distinguish homogeneous events.

4.4 Training Time

We compared the training time of EPAL and the five baselines, using GeForce RTX 4060Ti 16GB. The average training time for each model is recorded in Table 2. DuEEFin contains less data than ChFinANN, so all models are faster to train on DuEEFin. From the point of view of the decoding strategy of the method, the non-autoregressive method has a significant time advantage over the autoregressive method.

EPAL tends to extract multiple events in one document synchronously using a multi-probe strategy without additional computational cost from the graph structure, which makes it possible for the model to accurately extract multiple events with a short training time. In summary, EPAL has a similar training time to the SOAT method in terms of time complexity, which suggests that the extraction strategy of multiple probes and the probe-label alignment algorithm have less impact on the computational time of the model.

Model	ChFinAnn	DuEE-Fin
Doc2EDAG	3 : 36	0 : 43
DE-PPN	1 : 26	0 : 12
PTPCG	0 : 21	0 : 05
GIT	3 : 34	1 : 03
ReDEE	6 : 42	5 : 53
ProCNet	0 : 46	0 : 09
EPAL (Ours)	0 : 43	0 : 08

Table 2: The GPU time (hh:mm) of each epoch in average.

4.5 Ablation Study

-L_{rc} and -L_{pc} We remove the two auxiliary losses L_{rc} and L_{pc} in the objective function. It can be observed from Table 3 that they mainly affect the precision part of the F1-score. And L_{rc} has a greater impact on precision, mainly because the loss of role contrast can help the model understand the differences of arguments in different events. In contrast, since the model can learn the consistency information of probes through the classification of event types, which is low-level information compared with argument classification, L_{pc} has less impact than L_{rc}.

-Alignment The probe-label alignment optimization is replaced by Hausdorff distance minimization, which is equivalent to viewing all probes as a group to align with the ground-truth labels during training. The non-specific alignment method is easy to lead to the instability of the corresponding relationship during optimization and the chaos of multiple events learned by the model, we can observe that the optimization method has a great impact on recall. See Appendix D for a detailed discussion of specific examples.

-Role filling We replace the method of taking arguments from the library to fill the role framework with the method of directly classifying the arguments in each library according to the number of roles. The experimental results show that this replacement will lead to a large degree of performance degradation, especially the multi-event precision and recall. The reason for this phenomenon is that the role-filling method can better model the dependencies between arguments corresponding to the same role. Directly classifying the arguments in the library will not be able to model this dependency, and will cause multiple arguments to be assigned to the same role.

Model	P.	R.	F1	F1(S.)	F1(M.)
EPAL	83.1	83.5	83.4	89.7	76.6
-L_{rc}	80.9	83.8	82.3	88.4	76.0
-L_{pc}	82.2	83.4	82.8	88.7	76.8
-Alignment	82.3	79.0	80.6	88.1	72.6
-Role filling	80.1	73.7	76.6	89.4	61.7

Table 3: Ablation study on ChFinAnn.

5 Conclusion

In this paper, we first define the concepts of private and public arguments in multi-event documents and focus on the extraction strategy of homogeneous events. A document-level multi-event extraction method and optimization strategy are proposed based on event-specific probes and argument libraries, which can learn multiple events differently and notice their fine-grained differences in arguments. In our experiments, EPAL outperforms the state-of-the-art method and has a relatively high multi-event recall.

Limitations

In each document, the number of event-specific probes is equal to the number of entities in the document. Most of these probes may be redundant, meaning that the detected event type is None. To further optimize the computational efficiency of the model, it may be necessary to screen the entities used to initialize the probe. Using probes selected by some filtering algorithms for event sniffing can further improve the precision of EPAL.

Due to using only CLS embeddings for interaction between sentence segments, EPAL has limited cross-paragraph learning ability. One possible improvement approach is to model segments and tokens using heterogeneous graphs and use graph neural networks to aggregate information from adjacent segments. However, this will result in significant computational overhead, which needs a trade-off between efficiency and accuracy.

References

Hu Cao, Jingye Li, Fangfang Su, Fei Li, Hao Fei, Shengqiong Wu, Bobo Li, Liang Zhao, and Donghong Ji. 2022. Oneee: A one-stage framework for fast overlapping and nested event extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1953–1964.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Xinya Du, Sha Li, and Heng Ji. 2022. Dynamic global memory for document-level argument extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5264–5275.

Jun Gao, Wei Wang, Changlong Yu, Huan Zhao, Wilfred Ng, and Ruifeng Xu. 2022. Improving event representation via simultaneous weakly supervised contrastive learning and clustering. pages 3036–3049.

Cuiyun Han, Jinchuan Zhang, Xinyu Li, Guojin Xu, Weihua Peng, and Zengfeng Zeng. 2022. Duee-fin: A large-scale dataset for document-level event extraction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 172–183. Springer.

Yuxin He, Jingyue Hu, and Buzhou Tang. 2023. Revisiting event argument extraction: Can EAE models learn better when being aware of event co-occurrences? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12542–12556, Toronto, Canada. Association for Computational Linguistics.

Hendrycks. 2022. Gaussian error linear units (gelus). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2776–2789.

I Hsu, Zhiyu Xie, Kuan-Hao Huang, Prem Natarajan, Nanyun Peng, et al. 2023. Ampere: Amr-aware prefix for generation-based event argument extraction model. *arXiv preprint arXiv:2305.16734*.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Degree: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1890–1908.

Kung-Hsiang Huang, Sam Tang, and Nanyun Peng. 2021. Document-level entity-based extraction as template generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5257–5269.

Yusheng Huang and Weijia Jia. 2021. Exploring sentence community for document-level event extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 340–351.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Haochen Li, Tong Mo, Hongcheng Fan, Jingkun Wang, Jiaxi Wang, Fuhao Zhang, and Weiping Li. 2022. Kipt: knowledge-injected prompt tuning for event detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1943–1952.

Sha Li, Heng Ji, and Jiawei Han. 2021a. Document-level event argument extraction by conditional generation. In *2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 894–908. Association for Computational Linguistics (ACL).

Sha Li, Heng Ji, and Jiawei Han. 2021b. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,{NAACL-HLT} 2021*, volume 2021.

Yuan Liang, Zhuoxuan Jiang, Di Yin, and Bo Ren. 2022. Raat: Relation-augmented attention transformer for relation modeling in document-level event extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4985–4997.

Jiaju Lin, Qin Chen, Jie Zhou, Jian Jin, and Liang He. 2022. Cup: Curriculum learning based prompt tuning for implicit event argument extraction.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 7999–8009.

Jian Liu, Yufeng Chen, and Jinan Xu. 2021. Machine reading comprehension as data augmentation: A case study on implicit event argument extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2716–2725.

Jian Liu, Chen Liang, Jinan Xu, Haoyan Liu, and Zhe Zhao. 2023a. Document-level event argument extraction with a chain reasoning paradigm. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9570–9583, Toronto, Canada. Association for Computational Linguistics.

Wanlong Liu, Shaohuan Cheng, Dingyi Zeng, and Qu Hong. 2023b. Enhancing document-level event argument extraction with contextual clues and role relevance. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12908–12922, Toronto, Canada. Association for Computational Linguistics.

Xiao Liu, He-Yan Huang, Ge Shi, and Bo Wang. 2022. Dynamic prefix-tuning for generative template-based event extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5216–5228.

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

Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 2021. Text2event: Controllable sequence-to-structure generation for end-to-end event extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2795–2806.

Yubo Ma, Zehao Wang, Yixin Cao, Mukai Li, Meiqi Chen, Kun Wang, and Jing Shao. 2022. Prompt for extraction? paie: Prompting argument interaction for event argument extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6759–6774. Association for Computational Linguistics.

Yubing Ren, Yanan Cao, Fang Fang, Ping Guo, Zheng Lin, Wei Ma, and Yi Liu. 2022a. Clio: Role-interactive multi-event head attention network for document-level event extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2504–2514.

Yubing Ren, Yanan Cao, Fang Fang, Ping Guo, Zheng Lin, Wei Ma, and Yi Liu. 2022b. Clio: Role-interactive multi-event head attention network for document-level event extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2504–2514.

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

Minh Van Nguyen, Bonan Min, Franck Dernoncourt, and Thien Nguyen. 2022. Joint extraction of entities, relations, and events via modeling inter-instance and inter-label dependencies. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4363–4374.

Qizhi Wan, Changxuan Wan, Keli Xiao, Rong Hu, Dexi Liu, and Xiping Liu. 2023a. Cfere: Multi-type chinese financial event relation extraction. *Information Sciences*, 630:119–134.

Qizhi Wan, Changxuan Wan, Keli Xiao, Dexi Liu, Chenliang Li, Bolong Zheng, Xiping Liu, and Rong Hu. 2023b. 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 (Volume 1: Long Papers)*, pages 10481–10492, Toronto, Canada. Association for Computational Linguistics.

Sijia Wang, Mo Yu, Shiyu Chang, Lichao Sun, and Lifu Huang. 2022. Query and extract: Refining event extraction as type-oriented binary decoding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 169–182.

Xinyu Wang, Lin Gui, and Yulan He. 2023. Document-level multi-event extraction with event proxy nodes and hausdorff distance minimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10118–10133, Toronto, Canada. Association for Computational Linguistics.

Yinyi Wei, Shuaipeng Liu, Jianwei Lv, Xiangyu Xi, Hailei Yan, Wei Ye, Tong Mo, Fan Yang, and Guanglu Wan. 2022. Desed: Dialogue-based explanation for sentence-level event detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2483–2493.

Runxin Xu, Tianyu Liu, Lei Li, and Baobao Chang. 2021. Document-level event extraction via heterogeneous graph-based interaction model with a tracker. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3533–3546.

Runxin Xu, Peiyi Wang, Tianyu Liu, Shuang Zeng, Baobao Chang, and Zhifang Sui. 2022. A two-stream amr-enhanced model for document-level event argument extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5025–5036.

Zhiyang Xu, Jay Yoon Lee, and Lifu Huang. 2023. Learning from a friend: Improving event extraction via self-training with feedback from Abstract Meaning Representation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10421–10437, Toronto, Canada. Association for Computational Linguistics.

Hang Yang, Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Taifeng Wang. 2021. 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 (Volume 1: Long Papers)*, pages 6298–6308.

Yuqing Yang, Qipeng Guo, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. An AMR-based link prediction approach for document-level event argument extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12876–12889, Toronto, Canada. Association for Computational Linguistics.

Kaihang Zhang, Kai Shuang, Xinyue Yang, Xuyang Yao, and Jinyu Guo. 2023. What is overlap knowledge in event argument extraction? APE: A cross-datasets transfer learning model for EAE. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 393–409, Toronto, Canada. Association for Computational Linguistics.

Senhui Zhang, Tao Ji, Wendi Ji, and Xiaoling Wang. 2022. Zero-shot event detection based on ordered contrastive learning and prompt-based prediction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2572–2580.

Shun Zheng, Wei Cao, Wei Xu, and Jiang Bian. 2019. 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)*, pages 337–346.

Jie Zhou, Qi Zhang, Qin Chen, Liang He, and Xuan-Jing Huang. 2022. A multi-format transfer learning model for event argument extraction via variational information bottleneck. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1990–2000.

Tong Zhu, Xiaoye Qu, Wenliang Chen, Zhefeng Wang, Baoxing Huai, Nicholas Jing Yuan, and Min Zhang. 2021a. Efficient document-level event extraction via pseudo-trigger-aware pruned complete graph. *arXiv preprint arXiv:2112.06013*.

Tong Zhu, Xiaoye Qu, Wenliang Chen, Zhefeng Wang, Baoxing Huai, Nicholas Jing Yuan, and Min Zhang. 2021b. Efficient document-level event extraction via pseudo-trigger-aware pruned complete graph.

A Private Argument Distribution

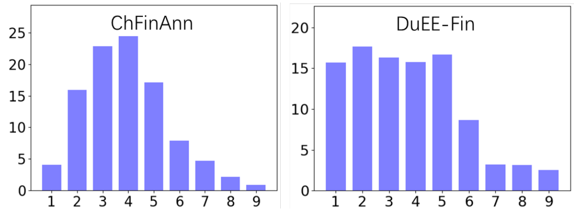

Figure 4: The distribution of private argument in ChFinAnn and DuEE-Fin. The abscissa represents the number of private arguments in an event. The ordinate represents the proportion of events with x private arguments in all events.

The number distribution of private arguments contained in multi-event documents of ChFinAnn and DuEE-Fin is shown in Fig. 4, respectively. Through the histogram, we can find that most of the multiple event records described in one document are homogeneous, that is, their event types and arguments tend to be consistent. Especially in DuEE-Fin, the number of events with only one different argument from other events in the same document accounts for about 15%.

B Implementation Detail

The BERT-base (Devlin et al., 2018) in Roberta setting (Liu et al., 2019) is used as the sequence labeling model. We set the maximum length of the sentence to 512, beyond which it is divided into multiple paragraphs, and attached CLS tokens at the beginning of each paragraph. The embedding size of BERT is 768 and the output size of FFN is 512, i.e. $d = 512$. The head of the attention mechanism is set to 8. GELU (Hendrycks, 2022) is employed as the activation function in our model. The optimizer we chose is Adam (Kingma and Ba, 2014) with a learning rate of $1e-5$ and a batch size of 32. In the role contrastive loss, we set the temperature coefficient $\tau = 0.5$. And the weights in the total loss L is set to $\lambda_e = 1$ and $\lambda_{bio} = \lambda_{rc} = \lambda_{pc} = 0.01$, which is the optimal result obtained from many experiments in $[0.001, 0.01, 0.1, 1, 10]$. Theoretically, L_e is the main loss for event recognition and the rest are auxiliary task losses. Therefore L_e has the highest weight in the whole loss function. We run the model 3 times with training epoch 100 on one NVIDIA 4060Ti GPU and selecting the best checkpoint.

Model	EF	ER	EU	EO	EP
Doc2EDAG	70.2	87.3	71.8	75.0	77.3
DE-PPN	73.5	87.4	74.4	75.8	78.4
GIT	73.4	90.8	74.3	76.3	77.7
ReDEE	74.1	90.7	75.3	**78.1**	80.1
ProCNet	71.4	92.5	67.3	66.7	79.8
EPAL (Ours)	**74.8**	**93.4**	**76.3**	77.3	**81.5**

Table 4: F1-score of 5 event types on ChFinAnn.

C Per-Event-Type Results

Table 4 and Table 5 show the evaluation results of 5 and 13 event types on ChFinAnn and DuEE-Fin, respectively. EPAL performs better than baseline methods in most classes, but the results in EO, CF, SI, and EC classes are slightly lower than those of other methods. Based on data analysis, we identify two main reasons for this phenomenon:

(1) The samples in these classes tend to have long contexts. EPAL only employs the CLS tokens of each paragraph and the attention mechanism of the entities to interact with each other but does not specifically encode the overall context, resulting in poor performance of the model on these classes. However, ProCNet uses a graph neural network to mine the overall entity information, and GIT models long path information for embeddings, which makes them have a higher F1-score for these samples.

(2) The role information of unrecognized samples in these classes is severely missing and some samples even have only one argument. EPAL infers that these events are none or identifies some unnecessary arguments. For these samples, ReDEE introduces the relation-enhanced attention mechanism, so that the model can filter out some unnecessary tokens and has higher recognition efficiency.

D Case Study

Figure 5 shows an error case for EPAL. We attach ProCNet and EPAL without probe-label alignment optimization as a comparison. We find that ProCNet recognizes only two of the four event records, which is because it cannot adequately detect the fine-grained differences between all the event records in a document. In contrast, EPAL can more accurately discover all the event records in a document, where the probes for discovering the event records are also derived from the private arguments of the corresponding events. If we re-

Model	WB	FL	BA	BB	CF	CL	SD	SI	SR	RT	PR	PL	EC
Doc2EDAG	60.0	78.3	50.6	40.1	63.2	51.5	50.7	52.9	83.7	51.2	64.8	61.7	51.2
DE-PPN	50.7	62.7	41.3	21.4	36.3	23.0	32.9	31.3	67.8	25.8	42.1	36.3	23.4
GIT	58.8	77.6	56.6	44.7	68.5	55.1	58.8	**71.2**	86.4	45.0	66.4	71.3	53.8
ReDEE	72.2	81.2	58.9	53.4	76.7	56.7	68.2	56.6	90.6	49.9	75.0	77.8	56.6
ProCNet	75.9	85.1	67.5	61.2	**77.0**	49.6	69.5	60.5	90.0	60.3	75.5	77.1	**62.0**
EPAL (Ours)	**76.3**	**86.1**	**68.3**	**69.3**	76.1	**59.1**	**74.5**	57.5	**91.3**	**71.1**	**76.8**	**78.2**	61.2

Table 5: F1-score of 13 event types on DuEE-Fin.

Gold event record	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 11970000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 24th October 2017 EndDate: None ReleasedDate: 6 September 2018	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 4000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 6 September 2018 EndDate: None ReleasedDate: None	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 5000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 6 September 2018 EndDate: None ReleasedDate: None	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 14030000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: None TotalHoldingRatio: None TotalPledgedShares: None StartDate: 13th December 2017 EndDate: None ReleasedDate: None
Predict event record (ProCNet)	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 11970000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 24th October 2017 EndDate: None ReleasedDate:None	Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 4000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: None TotalPledgedShares: 154530000 shares StartDate: 6 September 2018 EndDate: None ReleasedDate: None		
Predict event Record (EPAL)	**Probe from**: 11970000 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 11970000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 24th October 2017 EndDate: None ReleasedDate: 6 September 2018	**Probe from**: 4000000 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 4000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: 6 September 2018 EndDate: None ReleasedDate: None	**Probe from**: 177173451 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 5000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 42.37% TotalPledgedShares: None StartDate: 6 September 2018 EndDate: None ReleasedDate: None	**Probe from**: 14030000 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 14030000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: None TotalHoldingRatio: 48.58% TotalPledgedShares: None StartDate: 13th December 2017 EndDate: None ReleasedDate: None
Predict event record (EPAL without alignment)	**Probe from**: 4000000 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 4000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: 154530000 shares StartDate: None EndDate: None ReleasedDate: 6 September 2018	**Probe from**: 5000000 shares Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 4000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: None TotalPledgedShares: 154530000 shares StartDate: 6 September 2018 EndDate: None ReleasedDate: 6 September 2018	**Probe from**: 6 September 2018 Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 5000000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: 177173451 shares TotalHoldingRatio: 48.58% TotalPledgedShares: None StartDate: 6 September 2018 EndDate: None ReleasedDate: None	**Probe from**: Zheshang Securities Co. Pledger: Zhejiang Golden Eagle Group Co. PledgedShares: 14030000 shares Pledgee: Zheshang Securities Co. TotalHoldingShares: None TotalHoldingRatio: 3.85 TotalPledgedShares: None StartDate: 14th September 2017 EndDate: None ReleasedDate: None

Figure 5: Error case study with incorrect arguments colored in red. The first line represents the gold event record of the document. The second row represents the event records inferred by ProCNet. The third and fourth rows represent the event records inferred by EPAL and EPAL without probe-label alignment optimization, respectively.

move the probe-label alignment optimization in EPAL, we can notice a clear misalignment of the probes corresponding to the detected event records and more errors in the event arguments they extract. This suggests that EPAL can more accurately detect multiple events in a document through the optimization of the alignment algorithm and the auxiliary loss function.

For the errors in EPAL, we found that EPAL tends to confuse numerical types of arguments. This may be because there are more numerical arguments in financial news. Numerical information about different events without special handling may cause errors in the model's computation of contextual numerical information dependence. In the future, we will focus our work on the differentiation of different event-recorded numerical information in documents.