Received: 9 August 2020 | Revised: 2 September 2020 | Accepted: 7 September 2020

DOI: 10.1002/cpe.6104

WILEY

RESEARCH ARTICLE

An uncertain future: Predicting events using conditional event evolutionary graph

Jianqi Gao[1] | **Xiangfeng Luo**[1,2] | **Hao Wang**[1,2]

[1]School of Computer Engineering and Science, Shanghai University, Shanghai, China

[2]Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai, China

Correspondence
Xiangfeng Luo and Hao Wang, School of Computer Engineering and Science, Shanghai 200444, China.
Email: luoxf@shu.edu.cn (X. L.) and wang-hao@shu.edu.cn (H. W.)

Funding information
Ministry of Industry and Information Technology project of the Intelligent Ship Situation Awareness System, Grant/Award Number: MC-201920-X01; National Natural Science Foundation of China, Grant/Award Number: 61991415; National Natural Science Foundation of China, Grant/Award Number: 91746203

Summary

Event evolutionary graph (EEG) reflects sequential and causal relations between events, which is of great value for event prediction. However, lacking event context in the EEG raises the problems of direction uncertainty and low accuracy when making predictions. In this article, we propose a conditional event evolutionary graph (CEEG) to deal with these problems. CEEG extends EEG with an additional four types of event context, including state, cause, sub-type, and object. We first extract event context by matching the input with self-adaptive semantic templates and generalize the context for each event. To identify the evolution direction, we treat it as a binary classification problem and calculate the event transition probability for each direction given the generalized context. Experimental results show that CEEG has a strong ability to generate better event evolutionary paths compared with NAR, EEM, and other non-context-based methods.

KEYWORDS

conditional event evolutionary graph, event context, event prediction, event transition probability

1 | INTRODUCTION

Nowadays, with the rapid popularization of the Internet, we can easily access to a variety of news websites to know what has been ongoing on the earth. Understanding the daily events and how its evolution described in the text is essential to many artificial intelligence applications, such as event prediction and intention recognition. For example, "Company X was investigated by the China Securities Regulatory Commission (CSRC) for inflating profits and forced to delist." By reading this sentence, we know the pattern of event evolution: *inflated profit→investigation→delist*. To help the decision-maker understand the blueprint of the entire event and assist decision making, EEG is proposed.[1,2] EEG is a directed acyclic graph (DAG), in which nodes represent the events and edges represent the evolutionary relationships between events. However, EEG has some disadvantages: (1) the uncertainty of the event evolution direction; and (2) edge weights between events are difficult to calculate accurately. The main reason is lacking of necessary context for the event nodes. For example, for event investigation, the investigation department typically divides into supervision department, judicial branch, and legal operation department. Different departments will produce different investigation results. The relationships and directions of event evolution change with the event context.

EEG can be divided into temporal-based EEG[1] and logic-based EEG.[2] Previous research[3-6] generates temporal-based EEG as follows: First, structurally dividing news events into hierarchical structures which contains a topic, event, and story. Topic detection and tracking (TDT) technique is used to discover and track event. Second, clustering news events that belong to the same topic by using the topic models[7-9] and the clustering algorithms.[10-13] Finally, the similarities of the event relationship are calculated and determine the direction of event evolution by time sequence. However, the method itself lacks fine-grained analysis of events and is unable to extract the exact time of the event, resulting in poor performance in predicting future events. Figure 1 shows an example of the temporal-based EEG of involving in *company X*. There are totally five events and six event

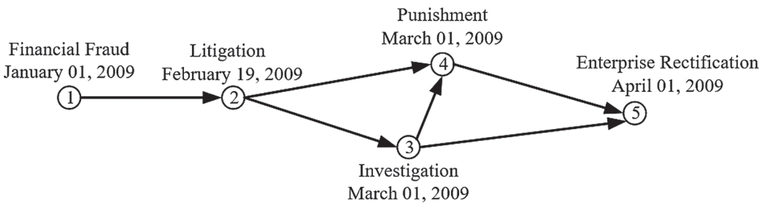


FIGURE 1 An example of temporal-based EEG

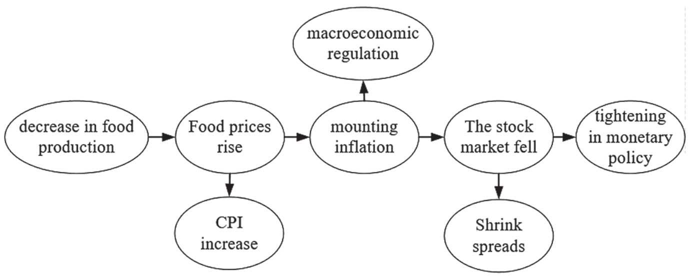


FIGURE 2 An example of logic-based EEG

evolution relationships. For example, there is an event evolution relationship from "Company X realized a false increase in profits." (i.e., Event 1) to "Shareholders filed a lawsuit in court" (i.e., Event 2). Following the event evolution relationship, we will find the terminal event is "Company X started a comprehensive enterprise rectification" (i.e., Event 5). Through temporal-based EEG, we can clearly understand how events are developing along the timeline.

Number	News	Time	Event Type
①	Company X inflated profits	January 01, 2009	Financial Fraud
②	Shareholders filed a lawsuit in court	February 19, 2009	Litigation
③	Security Regulatory Commission filling investigation	March 01, 2009	Investigation
④	Security Regulatory Commission will issue a punishment result	March 01, 2009	Punishment
⑤	Company X started a comprehensive enterprise rectification	April 01, 2009	Enterprise Rectification

Different from the above methods in the way of constructing temporal-based the EEG, logic-based EEG is an event logic knowledge base, which reveals evolutionary relationship of real world events. Figure 2 shows an example of logic-based EEG, we can see "mounting inflation" usually follows by "macroeconomic regulation" and "The stock market fell," which reflects the commonsense of event evolution patterns or people's activities in the real world. This kind of knowledge is very important in event prediction, risk estimation and making decision.

Li et al.[14] and Ding et al.[2] propose a robust pipeline to build logic-based EEG from a large-scale unstructured corpus as follows: event pair extraction, event relation classification, event direction classification, and event transition probability computation. Through the above steps, they build a logic-based EEG that better reveals the sequential and causal event relationship. However, these methods still extract wrong evolution directions and output unreliable event relationship weights, they ignore that the event evolution direction changes with the context. In this article, we further propose CEEG to make full use of the contextual information of the event, and apply it to the task of event prediction. As shown in Figure 3, our method can be split into six phases: event standardization, event extraction, event generalization, event pair generation, event transition type classification, and event transition probability computation. The main contributions of this article are summarized as follows:

- We propose a conditional event evolutionary graph (CEEG), which can mine the context of an event, improve the accuracy of event relationship calculation, and predict event evolution direction.
- We design a method to extract event and its context under the framework of CEEG. To represent the event context, we generalize the context, which can solve the problem of difficulty in event context extraction.
- We treat the event relation and direction classification as two classification tasks, and calculate event transition probability under different generalized context, which can predict future event between events with a different context.

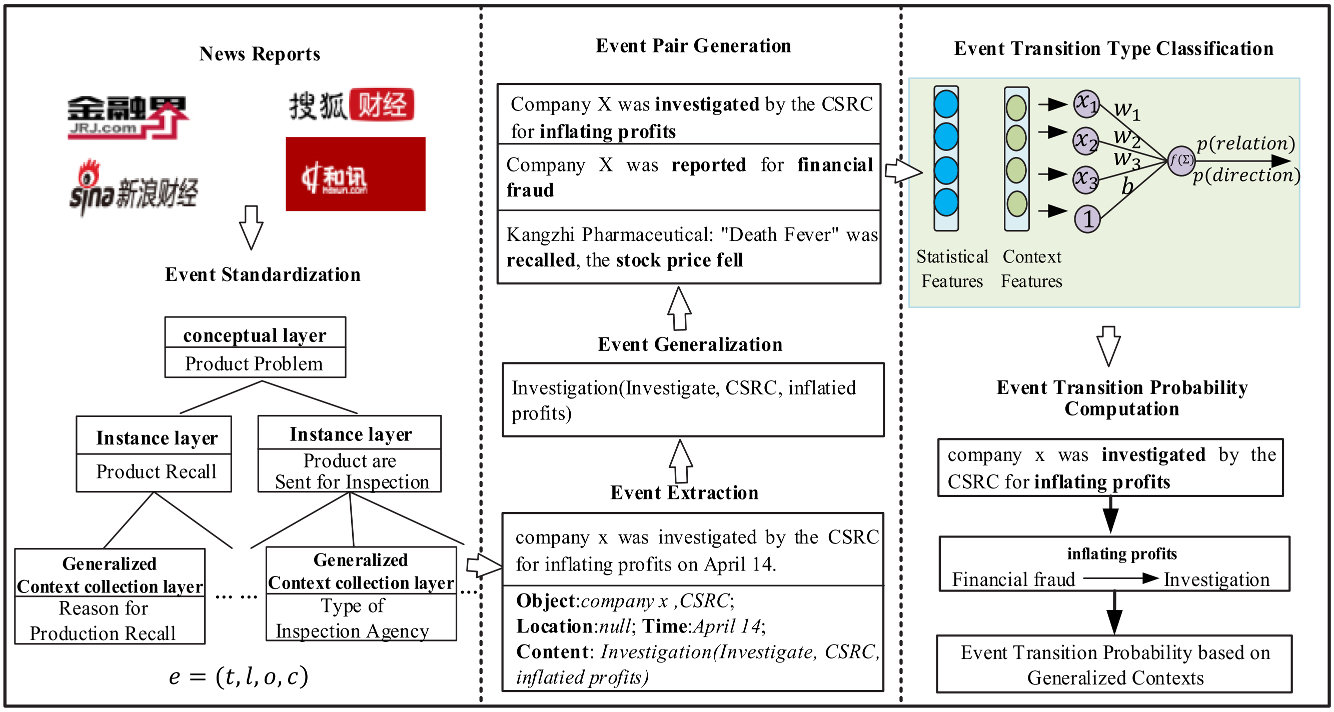


FIGURE 3 Overview of the CEEG framework

2 | RELATED WORK

Our approach is divided into two phases: event extraction and event evolution. We will briefly introduce in the next section.

2.1 | Event extraction

Event extraction involves three kinds of methods, including feature-based methods, pattern-based methods, and neural-based methods. For feature-based methods, Huang et al.[15] proposed a bootstrapping solution relying on lexical features for event extraction. To make use of sentence contextual information, Huang et al.[16] proposed a joint typing framework to improve the accuracy of event detection and extraction by making use of abstract meaning representation (AMR) and distributional semantics. However, feature-based methods require a lot of human engineering. In recent years, deep learning has achieved great success in the field of natural language processing. The text is represented by word embedding, which is fed into neural networks. Chen et al.[17] proposed a Dynamic Multi-pooling Convolutional Neural Network for event extraction. Nguyen et al.[18] explored recurrent neural networks (RNNs) in the event extraction task. To capture the sentence structure information, Sha et al.[19] proposed a dependency bridge recurrent neural network (dbRNN) for event extraction, and achieved competitive results compared with previous work.

However, both feature-based methods and neural-based methods require a lot of manual annotation, which makes it unsuitable for the construction of EEG. Compared with the above methods, pattern-based methods are more concise and can be flexibly changed according to different application scenarios.

2.2 | Event evolution graph

Event evolution graph (EEG) mainly consists of two parts, including temporal-based EEG[1] and logic-based EEG.[2]

For generating temporal-based EEG, Yeh et al.[20] applied correlation and event clustering over multiple evolving streams. Pei et al.[21] employed two incremental computation algorithms to maintain clusters and track event evolution. However, these methods focus on the event evolution through a sliding window, they cannot track entire complex events spanning over a long time. To solve these problems, Yang et al.[5] employed content similarity, temporal proximity, and document distributional proximity to measure whole event relationship. Lu et al.[22] applied multiple similarities to make a fine-grained analysis of the entire event relationship. Furthermore, Huang et al.[23] used three kinds of event relationships, including co-occurrence dependence analysis, event reference analysis, and temporal proximity analysis for modeling how event evolve from one to another.

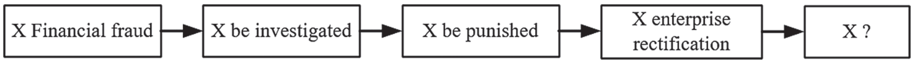


FIGURE 4 Example of statistical script learning on company

Temporal-based EEG employs a time sequence to judge event evolution direction. It cannot reflect the logical succession of event evolution, and time extraction in a news report may be not accurate, resulting in low accuracy and predictive ability of the generated EEG.

Unlike temporal-based EEG, logic-based EEG treats event evolution as commonsense knowledge. The most relevant research on logic-based EEG is statistical script prediction, which predicts the future event from candidate events given a series of events has happened. An example is shown in Figure 4. Chambers et al.[24] attempted to predict future events by using pairwise mutual information (PMI) to learn narrative event chains. Pichotta et al.[25] showed that prediction accuracy can be improved by modeling multi-argument events. Besides statistical methods, the neural network has been applied to this kind of task. Modi et al.[26] employed a neural network model which relies on distributed compositional representations for events script prediction, and had a better effect on the narrative cloze task compared with count-based counterparts. Wang et al.[27] proposed a novel dynamic memory network for script event prediction, which considered the temporal order and event relations. Li et al.[28] achieved the best script event prediction performance by constructing a narrative EEG. However, statistical script learning can only predict future activities on the event chain, it cannot reflect the logical relationship among multiple events in a domain.

Li et al.[14] and Ding et al.[2] generated logic-based EEG from large-scale unstructured corpus through three steps: event generalization, event relation classification, and event transition probability computation. EEG made by this method is more applicable and can be applied to down-stream applications such as statistical script learning, intention recognition, and question answering. Unlike logic-based EEG which organize event evolution into a commonsense knowledge base. CEEG calculated the event transition probability given the generalized context and have stronger predictive ability.

3 | MODEL AND ALGORITHM

In this section, we will briefly introduce the framework of CEEG as shown in Figure 3.

3.1 | Event standardization

Event: An event is something that happens at specific time and places, we define it as a four-tuple $e = (t, l, o, c)$. Elements in the tuple represent time, location, object, and content, respectively.

Event evolution: Event evolution is the transitional development process of related events, which is a sequential pattern: $e_0 \rightarrow e_1 \rightarrow e_2 \ \dots \ \rightarrow e_{n-1} \rightarrow e_n$, and $e_i \rightarrow e_j$ means e_j is evoluted from e_i.

Content: Content is a summary description of the event, it is a collection of trigger words. Event content is the most crucial part in the event elements. For the convenience of content extraction, we divide the event content into three layers, including the conceptual layer, instance layer and generalized context collection layer. The conceptual layer is a macro description of the event. The instance layer is a fine-grained description of event. The last layer is the generalized context collection layer. The generalized context collection contains four types of content elements, including state, cause, sub-type, and object. The division of the three layers is shown in Figure 5.

- State indicates the different phase of the event during event evolution. For example, the rectification event can be divided into three phases: before rectification, during rectification, and finish rectification.
- Cause generally describes the reason that leads to the current event, such as business loss, environmental pollution, and insolvency may cause termination of production.
- Sub-type denotes a detailed description of the event. For example, financial fraud can be divided into inflated profits, fraudulent issuance, counterfeit data, and so on.
- Object is event participation. For example, “Company X inflated profits,” the object is *company X*.

3.2 | Event extraction using extended trigger word sets

For the convenience of event extraction, we use pattern-based method to extract domain events, as pattern-based method has low recall in domain event extraction. To effectively extract events, we use extended trigger word sets to increase the recall of event extraction.

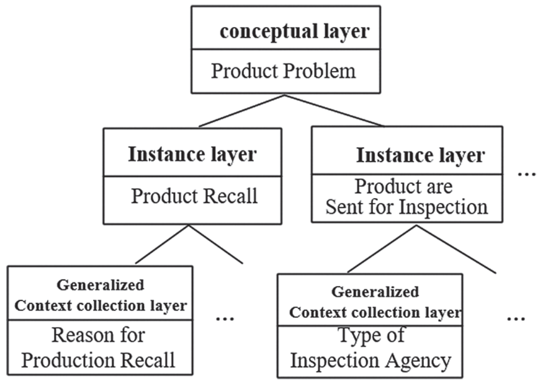


FIGURE 5 Example of three-layer division of event content

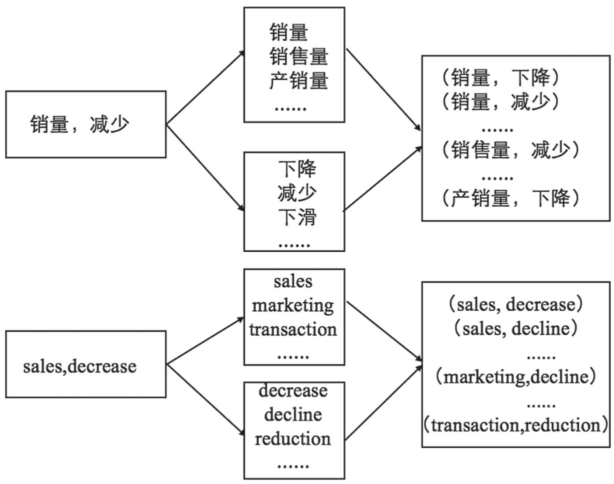


FIGURE 6 Example of event trigger words set generation

Given phrase description of related event. We use word2vec[29,30] to search the synonym for each word. Then extended trigger word sets are employed to generate the final trigger word set. Extended trigger word sets is a synonyms combination of different word which can be represented as

$$E = TW_1 \otimes TW_2 \otimes TW_3 \cdots\cdots \otimes TW_n, \tag{1}$$

where $TW = (tw_1, tw_2, tw_3, \dots, tw_n)$ denotes synonym set, $\otimes$ is the cartesian product of two synonym sets. An example of event trigger word sets generation by using extended trigger word sets is shown as Figure 6.

Then we can extract specific events by using adaptive semantic template based on the following formula:

$$dt(tw_i, tw_j) = |l_{tw_j} - l_{tw_i}|, dt(tw_i, tw_j) \leq \lambda \text{ and } tw_i \subset TW, \tag{2}$$

l_{tw_i} and l_{tw_j} denote the position of word in the text, $dt(tw_i, tw_j)$ is the distance of word tw_i and word tw_j, λ is the threshold.

3.3 | Event generalization and event pair generation

Event generalization means that the abstract of an event can be described clearly with only a few words. An event can be normalized into a four-tuple representation, and we do not need to care about the specific time and place of the event. For object, we retain the indirect object and omit the subject. For event content, we can divide the event into the third generalized context collection layer according to the event context. Therefore, for the purpose of generating CEEG, we constrain event generalization as $e_g = (s, i, r)$, each element in e_g is event triggers, indirect object, and event cause. For example, "Company X was investigated by the CSRC for inflating profits." We keep the indirect object *CSRC* and ignore the subject *company X*. Then the event can be generalized as *investigation=(investigate, CSRC, inflated profits).*

TABLE 1 The feature used for relation and direction classification

Count features T	Count feature description	Ratio features R	Ratio feature description
T_1	$C_1 : count(ab)$	R_1	$C_{21}/C_1, C_{22}/C_1$
T_2	$C_{21} : count(a \rightarrow b)$	R_2	$C_1/C_{31}, C_1/C_{32}$
	$C_{22} : count(b \rightarrow a)$		
T_3	$C_{31} : count(a)$	R_3	$C_1/C_{41}, C_1/C_{42}$
	$C_{32} : count(b)$		$C_1/C_{43}, C_1/C_{44}$
T_4	$C_{41} : count(verb\ a)$	R_4	
	$C_{42} : count(object\ a)$		$C_{41}/C_{31}, C_{42}/C_{31}$
	$C_{42} : count(verb\ b)$		$C_{43}/C_{32}, C_{44}/C_{32}$
	$C_{42} : count(object\ b)$		
Text Feature	Keywords selected by information gain and bi-gram based on keywords		

We use the method in Section 3.2 to extract events, and pick out the sentences with multiple event description, then event pairs are built for event evolution analysis. For example, if the event extracted in a sentence is (a, b, c, d), the event pairs constructed are combinations of $(a, b), (a, c), (a, d), (b, c), (b, d), (c, d)$.

3.4 | Event relation and direction classification

Given an event pair candidate (a, b), we need to determine whether there is a relationship between two events, then it is necessary to judge the evolution direction if there exist a relationship between two events.

We model the event relationship and direction recognition as two classification problem[2].[14] For event relation classification: it is a positive sample if event a is related to b. Otherwise, it is a negative sample. For event direction classification: it is a positive sample if $frequency(a \rightarrow b) > frequency(b \rightarrow a)$, otherwise it is a negative sample. Finally, we use the smote algorithm[31] to over-sample the negative samples as the number of positive and negative samples are unbalanced.

We build two types of features for the above two supervised classification tasks, including statistical features and text features. Statistical features contain counting features and ratio features, which are used to map the relationship between event type and event triggers in the two supervised classification tasks. Text features mainly includes keyword features and bi-gram features, which is to capture the contextual information in the sentence. The report of features is shown in Table 1.

$T_1 - T_4$ are counting features, $R_1 - R_4$ are ratio features. $count(ab)$ is the co-occurrences of event a and event b , $count(a \rightarrow b)$ is the frequency that event a appears before event b, $count(verb\ a)$ is the number of verb in the trigger word of event a, $count(object\ a)$ is the number of nouns in the trigger word of event a. For text feature, we calculate the information gain of each word, and sort each word by weight, finally top-k words are selected as keywords.

3.5 | Event transition probability based on generalized context

If event a evolves to event b , event a and event b must have some relationship under some context. For example, "Company x was investigated by CSRC for inflating profits." The news contains two event types, event (*complaint*, *investigation*), and event (*fraud*, *finical fraud*). The event evolved from financial fraud to investigation under the generalized context inflated profits. From the example, we can see that the evolution patterns can be expressed as: Finical Fraud $\xrightarrow{inflatedprofit}$ Investigation.

Given event evolution direction $a \rightarrow b_i$ and its general context c, we establish the relationship between two events under general contexts as follows:

$$P(b_i|a, c) = \frac{N(a \rightarrow b_i|c)}{\sum_{i=1}^{n-1} N(a \rightarrow b_i|c)} \quad (a \neq b_i), \tag{3}$$

where c is the general context collection, $N(a \rightarrow b_i|c)$ is the co-occurrence frequency of event a evolved to event b under general context c.

TABLE 2 Data used for event relation and direction classification

Name	Positive data	Negative data	Total
Relation data	2803	433	3236
Direction data	2537	266	2803

TABLE 3 Data used for event evolution evaluation

Data description	Values
Total number of events	14
Total number of event description	1077
Total number of company	8
Total number event evolution relationship	82

4 | EXPERIMENT AND EVALUATION

In this section, we do experiments on real datasets and compare our method with another three baselines.

4.1 | Dataset

We crawled a large scale of Chinese finical news reports from Hexun[1] and Jinrongjie,[2] and split the article by full point, semicolon, and exclamation point. We take the negative events of company as application background, and predefined 32 kinds of negative news event such as failure of reorganization, stop production, break the law, and so on. Then language technology platform (LTP)[32] and a predefined lexicon are used to extract company and institution. We collected sentences that contain company, and extracted the event for each sentence through adaptive semantic template. Finally, we got 370,521 sentences containing event trigger words and 63,007 sentences with multiple event description.

For event relation and direction classification, we pick out annotation data satisfy the following two conditions:

- The co-occurrence frequency of event pairs f satisfies $f \geq 6$.
- The number of words in the sentences n satisfies $n \leq 20$.

We invite three annotators to annotate the filtered data. Each annotator needs to determine whether there is a relationship between two events. If it does, they need to give the evolution direction. In other words, the relationship between direction data d and relation data r is: $d \subset r$. The data information is shown in Table 2.

For event evolution analysis, we collect news reports belongs to a company over a while, and divide news into different categories according to the event type, the final evaluation data obtained after filtering out the noise of each event. Each annotator needs to annotate the event evolution path for news event of each company. The final set collects the annotations given by all annotators. The data information is shown in Table 3.

4.2 | Evaluation

For event relation and direction analysis, we use three-layer fully connected network (3-FCN), gradient boosting decision tree (GBDT), support vector machine (SVM), and logistic regression (LR) for the above two supervised classification tasks. Accuracy (acc), Precision (P), Recall (R), and F1-Measure (F_1) are used as evaluation metrics. The optimal combination of classifier and features are obtained after 10-fold cross-validation.

For event evolution analysis, we compared our method with another three methods, and the baselines are summarized as follows:

- *Our method (CEEG)*: Given the current event a and its context c, we use transition probability $P(b_i|a, c)$ to predict probability that event a evolved to event b. The transition probability will be replaced if $P(b_i|a, c') > P(b_i|a, c)$ when the context of event changes.
- *EEM*: Yang et al.[5] apply content similarity, temporal proximity, and document distributional proximity to measure the relation between events. As it is difficult to extract time accurately, we only use content similarity and distributional document proximity as the relationship measurement and take the annotated direction as the evolution direction of the method.

[1]http://stock.jrj.com.cn
[2]http://stock.hexun.com

- *NAR*: Huang et al.[23] use co-occurrence dependence analysis, event reference analysis, and temporal proximity analysis to measure the event relationship. We use co-occurrence dependence analysis and event reference analysis to analyze the relationship between events, and take the annotated direction as the evolution direction of the method.
- *Method without general context (EEG)*: We only calculate the probability of an event and ignore the generalization context. Method without general context is calculated as follows:

$$P(b_i|a) = \frac{N(a \rightarrow b_i)}{\sum_{i=1}^{n-1} N(a \rightarrow b_i)} \quad (a \neq b_i). \tag{4}$$

We use precision (P'), recall (R'), and F1-Measure (F_1') as our evaluation metric of event evolution. The metric can be defined as follows:

$$P' = \frac{T \cap A}{A}, \quad R' = \frac{T \cap A}{T}, \quad F_1' = \frac{2PR}{P+R}, \tag{5}$$

where T is the truth set annotated by annotators. A is the result generated by the algorithm.

5 | EXPERIMENT RESULTS AND ANALYSIS

5.1 | Event relationship and direction classification analysis

Table 4 shows the experimental results for event relation classification, from which we find that GBDT, LR, and 3-FCN achieve very competitive performance. Compared with LR, GBDT, and 3-FCN, SVM achieves poor results. It may be that SVM is not sensitive to high-dimensional and sparse features. Besides, we can see the performance of classifiers with statistical features are better than those with text features. We explored all combinations of two kinds of features to find the best feature set for different classifiers, and we find that LR with statistical and text features achieves the best performance with a 0.983 accuracy, 0.98 precision, 0.98 recall, and 0.98 F_1.

Table 5 shows the experimental results for event evolution direction classification, from which we can see GBDT with statistical and text features achieve the best result (0.91 accuracy, 0.91 precision, 0.91 recall, and 0.91 F_1). Still, SVM achieves poor results. We find that the performance of LR, SVM, and 3-FCN with text features are better than those with statistical features. This means that text features are more important than statistical features for event evolution direction classification.

For the feature importance of event relation classification, we use information gain, random forest and Chi-square test to calculate the importance of each feature, and normalize each result in [0,1]. The average result is shown in Figure 7. We find that the statistical features are the most

TABLE 4 Result of event relation classification

Features	Classifier	Acc	*P*	*R*	F_1
Statistical feature T	LR	0.930	0.930	0.920	0.920
	SVM	0.885	0.890	0.880	0.880
Statistical feature T+R	GBDT	0.963	0.960	0.960	0.960
	3- FCN	0.960	0.960	0.960	0.960
Text	LR	0.922	0.930	0.920	0.920
	SVM	0.802	0.850	0.800	0.800
	GBDT	0.960	0.960	0.960	0.960
	3- FCN	0.950	0.950	0.950	0.950
T+R+Text	LR	**0.983**	**0.980**	**0.980**	**0.980**
	SVM	0.856	0.890	0.860	0.850
	GBDT	**0.975**	**0.980**	**0.980**	**0.980**
	3- FCN	0.950	0.950	0.950	0.950

Note: The first four rows are the optimal feature combinations of statistical features when given the classifier. The last four rows are the optimal feature combinations of statistical features and text feature when given the classifier.

TABLE 5 Event relation classification results

Features	Classifier	Acc	*P*	*R*	F_1
Statistical Feature T+R	LR	0.801	0.810	0.800	0.800
	SVM	0.781	0.790	0.780	0.780
	GBDT	0.855	0.850	0.850	0.850
	3- FCN	0.820	0.830	0.820	0.820
Text	LR	0.835	0.870	0.830	0.830
	SVM	0.818	0.820	0.820	0.820
	GBDT	0.847	0.850	0.850	0.850
	3- FCN	0.880	0.890	0.880	0.880
T+R+Text	LR	0.846	0.870	0.840	0.840
	SVM	0.808	0.840	0.810	0.800
	GBDT	**0.910**	**0.910**	**0.910**	**0.910**
	3- FCN	0.860	0.880	0.860	0.860

Note: The first four rows are the optimal feature combinations of statistical features when given the classifier. The last four rows are the optimal feature combinations of statistical features and text feature when given the classifier.

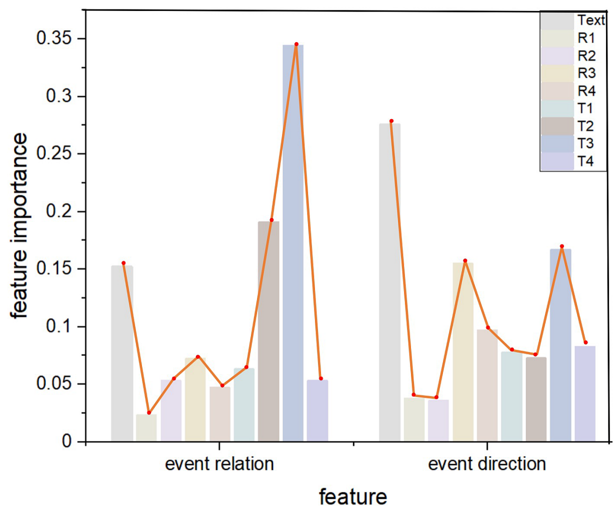


FIGURE 7 The importance of each feature calculated for event relation and direction classification

important feature. Experimental results in Table 4 verify this conclusion. For the feature importance of event evolution direction classification. We can see the most crucial feature is text feature, which is consistent with the results of Table 5. We further analyze the top-200 features, and find most features are keywords in the generalized context set. It also proves the importance of generalized context sets for event evolution analysis.

Event relationship and direction classification are important for CEEG construction. We can see the accuracy of event relationship classification is higher than direction classification, the main reason is that if event *a* evolves to event *b*, there must be relationship and evolutionary context between two event. Therefore, event direction classification is more complicated. On the other hand, statistical features and text features are two kinds of important features for event relationship and direction classification, and most of these two kinds of features are keywords in the generalized context set. That means event relationship and context are two indispensable conditions for event evolution.

5.2 | Event evolution analysis

Table 6 shows the result of our method compared with another three baselines. It can be seen from the table that the overall performance of our approach in precision and F_1' values are better than other methods, and 0.04 lower than NAR in recall. As EEM and NAR take the annotated direction as the evolution direction, it further proves the effectiveness of our method.

TABLE 6 Our method compared with another three baselines

Method	P'	R'	F'_1
EEM ($\lambda = 0.3$)	0.700	0.858	0.771
NAR ($\lambda = 0.23$)	0.724	**0.958**	0.824
Method without context (EEG, $\lambda = 0.03$)	0.690	0.917	0.787
Our method (CEEG, $\lambda = 0.04$)	**0.750**	0.917	**0.825**

We can find the performance of NAR is better than EEM. EEM applies content similarity and document distributional proximity to measure the relationship between events, which only consider the keywords and the distribution of documents, and it ignores the relationship between events. NAR uses event reference analysis and co-occurrence dependence analysis to measure the event relationship, which grasps the relationship and contextual features between events. However, this method ignores the event evolution patterns in the large-scale corpus, and the technique itself lacks fine-grained analysis of event context. We can see that our method outperforms NAR in precision and F'_1, it means that the generalized context can grasp the key features of event evolution context. Besides, our approach has the ability to predict future events. EEM and NAR can only mine event evolutionary along a timeline.

Number	Event	Generalized context	Event description
①	litigation	Litigation type: Civil action	Since February 19, 2009, shareholders have filed a lawsuit in the court
②	investigation	Investigation agency: CSRC	CSRC filed an investigation
③	Administrative penalty	Penalty agency: CSRC	The CSRC imposed a penalty on company X
④	Company rectification		In April 2009, Company X began a comprehensive company rectification.
⑤	Performance decline	Profit decline: Profit decline	Shareholder losses,company X profit declines.
⑥	Project negative	Contract issues: contract fraud	Reported on April 16, 2009, Company X contract fraud
⑦	Financial fraud	Financial fraud: inflated profits	On January 1, 2009, company X fraud, and inflated profits.

In addition, our method outperforms the method without context in precision and F'_1. CEEG framework transforms the event evolution into transition probabilities between events under different pervasive context. Compared with the technique without context, it can grasp the contextual information of event evolution and filter out some noise.

Different thresholds λ result in different evolutionary results. To find the best threshold, we experiment as shown in Figure 8. With the increase of λ, the weak evolutionary relationships are removed. We can see the recall and F'_1 decrease as the accuracy increase. When $\lambda = 0.04$, we get the highest F'_1 value. $\lambda = 0.04$ is chosen as the threshold in our method.

Figure 9 shows the event evolution of *company X* generated by our method under different generalized context, where nodes represent events and edges represent the evolutionary relationship. There are eight events in the example, we can see the event evolution start from finical fraud and

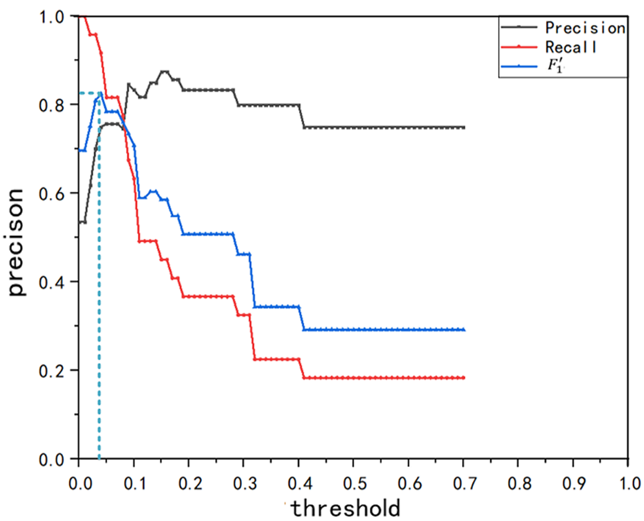
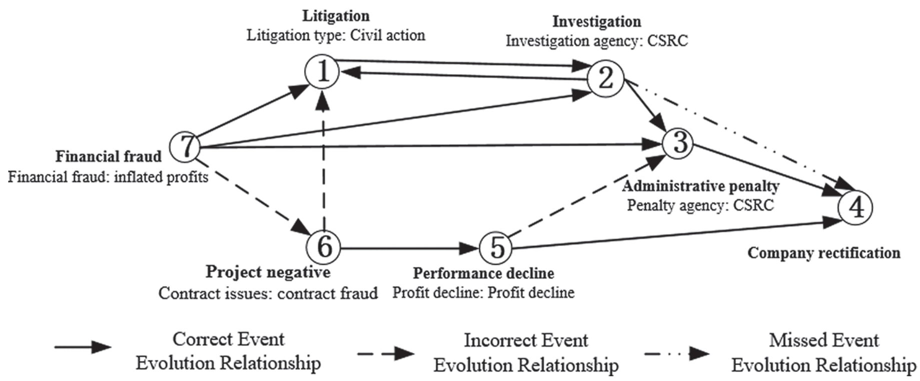


FIGURE 8 Event evolution result after removing weak evolutionary relationships with different threshold λ

FIGURE 9 Event evolution of *company X* generated by our method



end up with enterprise rectification. Different from temporal-based EEG and logic-based EEG, CEEG can well extract the context of event evolution, and predict future event under different context. We can see the rectification of the *company X* is evolved from CSRC investigation, CSRC penalty, and profit decline. Mutual evolution between litigation and investigation under the condition of civil action. Through CEEG, we can see the details of the event evolution path, which is of great value for event prediction, decision-making, and risk estimation.

Still, CEEG have some incorrect event evolution relationship and missed event evolution relationship. This is mainly because pattern-based method extract some wrong events, the boundary of some event types is difficult to distinguish, and pre-defined event types need some improvement.

6 | CONCLUSION AND FUTURE WORK

In this article, we present a novel framework of CEEG, which can effectively make use of context information to classify event relationship and determine event evolution direction. For the convenience of extracting events and their corresponding contexts, we first standardize the event and define the generalized context as containing four types of contextual elements. Then we use extended trigger word sets to improve the recall of event extraction. The event relation and direction classification is treated as two-phase supervised classification problem. Experimental results show that CEEG can consider rich event contexts to determine event evolution path, which can significantly improve the accuracy of downstream tasks such as event prediction.

To the best of our knowledge, there are few studies on EEG, and almost no practical application in real downstream tasks. This article shows extensive experiments to verify the effectiveness of our method in event prediction tasks. However, the method still has the following limitations: (1) Event standardization requires certain industry experience and manual participation. (2) Due to the complexity of Chinese, the accuracy of event and its corresponding context extraction needs to be further improved. In the future, we will also solve these problems to improve the efficiency and robustness of the algorithm.

ACKNOWLEDGMENTS

The research reported in this article was supported in part by the Natural Science Foundation of China under the grant No.91746203, Ministry of Industry and Information Technology project of the Intelligent Ship Situation Awareness System under the grant No. MC-201920-X01, and National Natural Science Foundation of China under the grant No.61991415.

ORCID

Jianqi Gao https://orcid.org/0000-0002-9840-7866

REFERENCES

1. Mei Q, Zhai CX. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. Paper presented at: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. Chicago, IL; 2005:198-207.
2. Ding X, Li Z, Liu T, Liao K. ELG: an event logic graph; 2019. arXiv preprint arXiv:1907.08015.
3. Aggarwal CC, Han J, Wang J, Yu PS. A framework for on-demand classification of evolving data streams. *IEEE Trans Knowl Data Eng*. 2006;18(5):577-589.
4. Nallapati R, Feng A, Peng F, Allan J. Event threading within news topics. Paper presented at: Proceedings of the 13th Association for Computing Machinery(ACM) International Conference on Information and Knowledge Management. Washington, DC; 2004:446-453.
5. Yang CC, Shi X, Wei C-P. Discovering event evolution graphs from news corpora. *IEEE Trans Syst Man Cybern Part A Syst Humans*. 2009;39(4):850-863.
6. Allan J. *Topic Detection and Tracking: Event-Based Information Organization*. Berlin, Germany: Springer Science & Business Media; 2012.
7. Becker H, Naaman M, Gravano L. Beyond trending topics: real-world event identification on twitter. In: 5th International Association for the Advancement of Artificial Intelligence(AAAI) Conference on Weblogs and Social Media; 2011.

8. Allan J, Carbonell JG, Doddington G, Yamron J, Yang Y. Topic detection and tracking pilot study final report; 1998.
9. Phuvipadawat S, Murata T. Breaking news detection and tracking in Twitter. Paper presented at: Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. Toronto, Canada; 2010:120-123.
10. Li D, Becchi M. Deploying graph algorithms on gpus: an adaptive solution. Paper presented at: Proceedings of the 2013 IEEE 27th International Symposium on Parallel and Distributed Processing. Cambridge, MA; 2013:1013-1024.
11. Long R, Wang H, Chen Y, Jin O, Yu Y. Towards effective event detection, tracking and summarization on microblog data. Paper presented at: Proceedings of the International Conference on Web-age Information Management. Wuhan, China; 2011:652-663.
12. Weng J, Lee BS. Event detection in twitter. Paper presented at: Proceedings of the 5th Association for the Advancement of Artificial Intelligence (AAAI) International Conference on Weblogs and Social Media. Catalonia, Spain; 2011.
13. Zhou X, Chen L. Event detection over twitter social media streams. *VLDB J*. 2014;23(3):381-400.
14. Li Z, Zhao S, Ding X, Liu T. Eeg: knowledge base for event evolutionary principles and patterns. Paper presented at: Proceedings of the Chinese National Conference on Social Media Processing. Beijing, China; 2017:40-52.
15. Huang R, Riloff E. Bootstrapped training of event extraction classifiers. Paper presented at: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. Avignon, France; 2012:286-295.
16. Huang L, Cassidy T, Feng X, et al. Liberal event extraction and event schema induction. Paper presented at: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany; 2016:258-268.
17. Chen Y, Xu L, Liu K, Zeng D, Zhao J. Event extraction via dynamic multi-pooling convolutional neural networks. Paper presented at: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Beijing, China; 2015:167-176.
18. Nguyen TH, Cho K, Grishman R. Joint event extraction via recurrent neural networks. Paper presented at: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego, CA; 2016:300-309.
19. Sha L, Qian F, Chang B, Sui Z. Jointly extracting event triggers and arguments by dependency-bridge rnn and tensor-based argument interaction. Paper presented at: Proceedings of the 32nd Association for the Advancement of Artificial Intelligence (AAAI) international Conference on Artificial Intelligence. New Orleans, LA; 2018.
20. Yeh M-Y, Dai B-R, Chen Ming- S. Clustering over multiple evolving streams by events and correlations. *IEEE Trans Knowl Data Eng*. 2007;19(10):1349-1362.
21. Lee P, Lakshmanan LVS, Milios EE. Incremental cluster evolution tracking from highly dynamic network data. Paper presented at: Proceedings of the 2014 IEEE 30th International Conference on Data Engineering. Chicago, IL; 2014:3-14.
22. Lu Z, Yu W, Zhang R, Li J, Wei H. Discovering event evolution chain in microblog. Paper presented at: Proceedings of the 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems. New York, NY; 2015:635-640.
23. Huang D, Hu S, Cai Y, Min H. Discovering event evolution graphs based on news articles relationships. Paper presented at: Proceedings of the 2014 IEEE 11th International Conference on e-Business Engineering. Guangzhou, China; 2014:246-251.
24. Chambers N, Jurafsky D. Unsupervised learning of narrative event chains. *Paper presented at: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*. Columbus, OH; 2008;789-797.
25. Pichotta K, Mooney R. Statistical script learning with multi-argument events. Paper presented at: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics. Gothenburg, Sweden; 2014:220-229.
26. Modi A. Event embeddings for semantic script modeling. Paper presented at: Proceedings of The 20th ACL's Special Interest Group on Natural Language Learning (SIGNLL) Conference on Computational Natural Language Learning. Berlin, Germany; 2016:75-83.
27. Wang Z, Zhang Y, Chang CY. Integrating order information and event relation for script event prediction. Paper presented at: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark; 2017:57-67.
28. Li Z, Ding X, Liu T. Constructing narrative event evolutionary graph for script event prediction; 2018. arXiv preprint arXiv:1805.05081.
29. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space; 2013. arXiv preprint arXiv:1301.3781.
30. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*. Lake Tahoe, NV: Curran Associates, Inc; 2013:3111-3119.
31. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res*. 2002;16:321-357.
32. Che W, Li Z, Liu T. Ltp: a Chinese language technology platform. *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Beijing, China: Coling 2010 Organizing Committee; 2010:13-16.

How to cite this article: Gao J, Luo X, Wang H. An uncertain future: Predicting events using conditional event evolutionary graph. *Concurrency Computat Pract Exper*. 2021;33:e6104. https://doi.org/10.1002/cpe.6104