Learning Event Logic Graph Knowledge for Credit Risk Forecast

Yifan Wang ★ Jibin Wang ★ Jing Shang ★ Zhuo Chen ★ Xuelian Ding★
Heyuan Wang☆
★ China Mobile Information Technology Co., Ltd.
☆ School of Computer Science, Peking University

Abstract

With the development of event knowledge graph technology, researchers have solved the singleness problem of event graph based on temporal relationship by constructing event logic graph, but have not integrated the multiple relationships among events with time series data for trend prediction. In addition, due to the impact of COVID-19, corporate credit risks have been gradually exposed in recent years, and defaults have occurred frequently. The technology of event graph and event logic graph is mostly used for event schema induction, script induction, etc., but abundant graph knowledge is not well exploited for forecast task.

To fill this gap, we construct an event logic graph by extracting various types of event relationships, such as causal relationship, sequential relationship, parallel relationship, and reversal relationship. Different types of edges among events are used to represent different relationships. Combined with the time series of corporate credit bonds, a temporal convolutional network driven by event logic graph is built, and applied to forecast corporate credit risk.

We extract structured events from financial news, construct event logic graph and learn the graph knowledge. Then, the event logic graph embedding is combined with time series of bonds to forecast whether the corporate will default. Experiments show that the proposed method outperforms baseline methods in forecasting credit risk.

1 Introduction

Events contain a large number of internal composition structures (such as participants, time, place, etc.) and external associations (such as causal relations, sequential relations, parallel relations, reversal relations, etc.). The news information platform has accelerated the spread of information among various social groups. How to perceive hot events and sort out the context among them has become the key to tracking sudden social turmoil, epidemic diseases, credit defaults and other events.

Most of the existing intelligent news system construct the event graph based on temporal relationship, but there are complex internal and external correlations among events. The cascade reactions brought about by different event relationships are disparate. We can assist decision-making in many fields such as natural disasters, aviation safety, and public health, in the method of extracting events and identifying relationships from news texts.

There are three key technologies in applying event graph to forecasting: event detection, event relationship extraction, and trend forecast.

In terms of event detection , topic detection and text clustering are traditional technical methods, using BoW model and LDA model to represent the topic of the text. These methods have the following problems: 1. Clustering algorithms need to determine the number of clusters in advance to achieve event classification. 2. The unbalanced distribution of event clusters increases the difficulty of clustering.

In terms of event relationship extraction , the traditional solution is to implement topic discovery through article clustering, and construct the relationship among events based on sequence of events. However, how many relationship types among events is still a controversial issue, and identifying different relations from text is challenging.

In terms of trend forecast , most of the current research based on event context is applied to smart news in the aim of assist the public in visualizing hot spots. A small number of financial quantification teams extract structured event tuples from financial news, and use knowledge graph technology to correlate discrete event tuples with each other. Event embeddings are obtained by training event tuples and knowledge graph triples. On this basis, the method of multichannel connection is adopted, the price vector and event embedding are used as the input of model to forecast the stock price. However, such methods do not use the multiple relationship among events for forecast.

Our contribution can be summarized as follows:

Extract multiple event relationships, such as sequential relationships, causal relationships, reverse relationships and parallel relationships. Construct event logic graph using events as nodes and multiple event relationships as edges.

Build a temporal convolutional network model driven by event logic graph, which integrates news events and corporate credit bond transaction data to forecast credit risks. The comparative experiments and ablation experiments show that our approach outperforms the baseline methods, and multiple relationship has better performance in trend forecast than the single relationship.

DOI reference number: 10.18293/SEKE2023-222

2 Related Work

2.1 Event Extraction

The research and application of event extraction mainly rely on machine learning and deep learning. Based on the existing models, the following two common problems are summarized:

1.How to learn the semantic representation of events from the given text: mining effective features is the key to the model. Early methods designed fine-grained features such as lexical, syntactic, and kernel-based features. Neural networks have been tried in this tasks, including CNNs and Transformers. Due to the complexity of event structures, recent studies have begun to use additional information such as entities, document-level information, and syntactic structures.

2.How to extract events across sentences or at document level: Current research mainly focuses on the sentence level, with the basic assumption that events are manifested in sentences. Compared with sentence-level event extraction, document-level event extraction needs to consider more complex issues: including parameter dispersion, multi-event expression, etc.

2.2 Event Relation Extraction

Event Co-reference Resolution is to confirm whether multiple event elements belong to the same event, which is treated as a classification or ranking problem. Machine learning models are widely used in this field, such as decision tree classifier[1], information propagation models[2] and multi-loss neural models[3]. These models focus on understanding the context of two events.

Event Causality Extraction is often viewed as a classification task. Existing models generally complete classification under the premise of supervision, the key point is how to extract clues and how to represent the semantics of causality. Extracting effective clues of contextual event causality requires the use of various text features, including syntactic features[4], lexical features, explicit causal patterns[5], statistical causality, etc.

Temporal Relation Extraction is mainly based on the TimeML format[6], and most methods solve it as a classification problem. Early ETE models usually relied on temporal rules[7]. Some researchers also used temporal context features to build models and extract temporal relationships based on machine learning[8]. In addition, neural network models are widely used, such as the classic CNNs, LSTMs methods[9] and the BiLSTM model based on dependent paths[10]. In addition, some more refined improved methods achieve more accurate extraction results[11].

2.3 Event Knowledge Graph

Gottschalk and Demidova[12] designed and implemented the Event Knowledge Graph(EKG). At present, events and their temporal relationships are mostly distributed in entity-centric knowledge graphs and artificially curated semi-structured resources. The proposal of EKG promotes a global view of events and temporal relationships. Ding et al.[13] proposed Event Logic Graph(ELG). Nodes in ELG represent events, and edges represent relationships among events. ELG reveals the development and evolution process of objective events.

Zhang et al.[14]proposed a large-scale event knowledge graph (ASER) for discovering real-world activity knowledge. ASER defines a brand new knowledge graph, in which each vertex is the basic element of an event, and the event relationship is the connection a hyperedge of several vertices. In addition, some researchers try to construct commonsense knowledge graphs around events, such as Event2Mind, GLUCOSE, ATOMIC.

2.4 Applications in Trend Forecasting

The event graph is centered on events, describing event information and the relationship among events. Based on this, technologies such as event prediction and trend prediction are realized. By analyzing the development process of historical events, it is possible to predict future events. Many researchers use contextualized events such as event skeletons in different fields to predict trends, such as the evolution process of natural disasters, event evolution analysis, stock price prediction, etc.

3 Method

3.1 Explicit Event Relationship Extraction

We unify the extraction methods of various event relationships to make the processing of news corpus more efficient. Inspired by Chang et al., this paper adopts the combination of explicit relationship trigger words and pattern recognition. Different from Chang's approach, in addition to establishing temporal relations, this paper formulates explicit trigger words and linguistic rules for causal, parallel, and reversal relations. We set different types of relational triples (preword, postword, type) for pattern matching and event tuple constructing, where preword and postword do not necessarily appear at the same time, such as (”due to”, ”resulting in”, causal).

We constructed the corresponding syntactic patterns, and extracted event tuples (text, type, preword, prepart, postword, postpart) of various relations. ”type” is the type of event relationship, ”text” is the original text, ”preword” and ”postword” represent the event relationship trigger words obtained by pattern matching, the prepart and postpart are the event content before and after the relationship trigger words.

3.2 Construction of Event graph

The event graph we constructed uses events as nodes and the relationship among events as edges. Both "event" and "relation" are obtained from the event-relation tuple.

Take the news of ICBC as an example, "Regulations have tightened the reporting standards, leading to a general decline in the stock of public offerings sold by securities companies." The event relationship tuple can be obtained through the explicit event relationship extraction method (type="causal",preword=null, prepart="Regulations have tightened the reporting standards", postword="leading to", postpart="The stock of public offerings sold by securities companies has generally dropped"). Then set an edge whose type is "causal" between the two events. The construction methods of sequential relationship, parallel relationship and reverse relationship are the same.

3.3 ELGTCN

When forecasting the credit risk of corporates, it is also necessary to combine with financial time series data, such as the lowest transaction price of bonds. This paper uses Event Logic Graph Driven Temporal Convolutional Networks (ELGTCN) to combine event logic graph with bond data in corporate credit risk forecast.

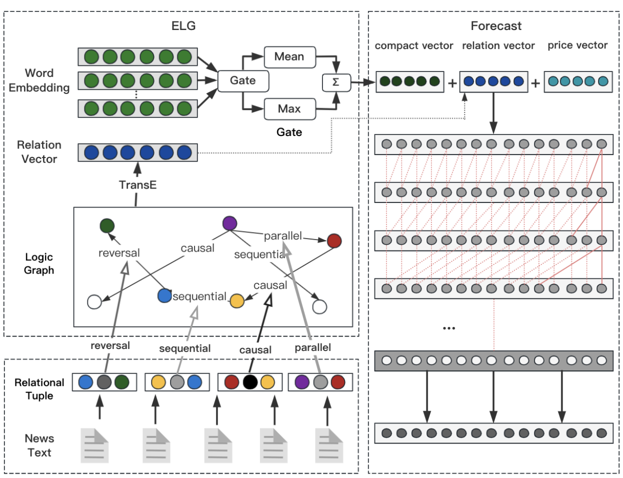

Figure 1. ERCDTCN

3.3.1 Representation of Original News

For the original news, we use word2vec to encode and obtain vector sequences. Calculate the maximum value of each dimension through the maximum pooling operation to capture the most significant attributes in the sequence. The average value of each vector sequence is calculated by the average pooling operation to obtain universal information. The max pooling and average pooling operations on vector sequences are two complementary ways. Finally, the results of the two operations are connected in parallel to obtain the representation of mean-max. The calculation process is as follows:

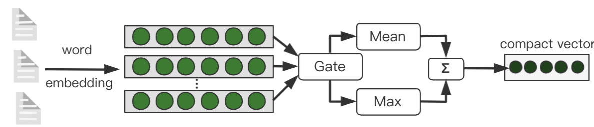

Figure 2. "mean-max" schematic diagram of control mechanism

$$
\begin{aligned}
&z_{\max}[i] = \max_t \boldsymbol{h}_{ti}^e \\
&z_{\text{mean}} = \frac{1}{N} \sum_t \boldsymbol{h}_t^e \\
&z = [z_{\max}, z_{\text{mean}}] \\
&z_{\max}^d = z_{\max} \otimes \sigma\left(\boldsymbol{W} \cdot z_{\max} + \boldsymbol{b}\right) \\
&z_{\text{mean}}^d = z_{\text{mean}} \otimes \sigma\left(\boldsymbol{W} \cdot z_{\text{mean}} + \boldsymbol{b}\right) \\
&\boldsymbol{h}_t^d = z_{\max}^d \oplus z_{\text{mean}}^d
\end{aligned} \tag{1}
$$

3.3.2 Representation Method of ELG

We transform the multiple relationship graph structure into a low-dimensional space vector by means of TransE[15]. Suppose there is an event relation triplet of (e_1, $relation$, e_2), where the vector of event e_1 is expressed as V_{e_1}, event e_2 The vector representation of V_{e_2}, the relationship between event pairs is $V_{relation}$. The ultimate goal is to map events and their relationships into k dimensional vectors. The essential idea is that if there is a relationship $relation$ between the event pair e_1 and e_2, then try to make $V_{e_1} + V_{relation} \approx V_{e_2}$, otherwise, keep as far away as possible.

In order to realize the representation learning of multiple relations, the relation r_i is modeled as a hyperplane w_{r_i} and a transfer vector d_{r_i}. First, project the two events on the square of w_{r_i}, and then calculate the distance on the projected hyperplane. That is, $V_{r_i}(e_1) = e_1 - w_{r_i}^{\top} e_1 w_{r_i}$, r is expressed as a transition vector d_{r_i}, $V_{r_i}(e_2) = e_2 - w_{r_i}^{\top} e_2 w_{r_i}$. Use the Euclidean distance between $V_{r_i}(e_1) + d_{r_i}$ and $V_{r_i}(e_2)$ as the difference between two events.

Multivariate event relations are projected to different hyperplanes, and different relations of events are represented by vector calculations between different hyperplanes. Set the distance function $d(V_{r_i}(e_1)+V_{r_i}, V_{r_i}(e_2))$, for all event pairs in the event logic graph, minimize the distance function $d(V_{r_i}(e_1) + V_{r_i}, V_{r_i}(e_2))$. Specifically, the loss function $\mathcal{L}$ is defined as follows, where r_i represents different event types, $i \leq 4$.

$$
\begin{aligned}
\mathcal{L} = &\sum_{(e_1, r_i, e_2) \in S} \sum_{(e_1', r_i, e_2') \in S'} \\
&[\lambda + d\left(V_{r_i}(e_1) + V_{r_i}, V_{r_i}(e_2)\right) \\
&-d\left(V_{r_i}(e_1)' + V_{r_i}, V_{r_i}(e_2)'\right)]_+
\end{aligned} \tag{2}
$$

Where $[x]_+$ represents the positive part of $[x]$, $S'_{(e_1,r,e_2)}$ represents the relational tuple whose head or tail is randomly replaced . Increase the distance function of non-existent event-pair relations by constructing error samples. The composition of $S'_{(e_1,r,e_2)}$ is as follows:

$$\begin{aligned} S'_{(e_1,relation,e_2)} &= \{(e_1{}', relation, e_2) \mid e_1{}' \notin E\} \\ &\cup \{(e_1, relation, e_2{}') \mid e_2{}' \notin E\} \end{aligned} \tag{3}$$

3.3.3 Representation of Time Series Data Related to Corporate Credit Debt

According to financial market experience, bond variables, corporate variables and macro variables have a certain early warning effect on corporate credit risk. Bond variables include minimum transaction price, coupon rate, etc. Corporate variables include return on total assets, proportion of long-term liabilities, etc. Macro variables include risk-free interest rates and the CSI 300 Index. We use the above information as time series data reflecting corporate credit risk. By splicing the vector representations of the above three kinds of data, the original news $\mathcal{N}$, the event logic graph $\mathcal{G}$, and the time series data $\mathcal{X}$ are fused (as shown on the right side of Figure 1), as the input for forecast.

3.3.4 ELGTCN for Forecast

Inspired by Bai et al. [16], the TCN model is used to prevent the look-ahead error that may exist in the application of time series data. Longer historical data can be looked back through deep networks augmented with dilated convolutions and residual layers.

We employ a 1D fully convolutional network (FCN) architecture, where each hidden layer has the same length as the input layer, and zero padding is added to keep subsequent layers the same length as the previous layer. In this way, the network can produce an output of the same length as the input. In addition, TCN uses causal convolution, and the output at time t is only convolved with elements at time t and earlier in the previous layer, so that there is no look-ahead bias.

The historical information that causal convolution can recall is linear with the depth of the network. The prediction of enterprise credit risk is a task that requires a long historical review. Therefore, we use the expansion convolution to achieve an exponentially large receptive field. Specifically, for a one-dimensional sequence input $x \in R^n$ and a filter $f : \{0, \ldots, k-1\} \rightarrow R$, the dilated convolution operation $\mathcal{F}$ on element s in the sequence is defined as:

$$\mathcal{F}(s) = (\mathbf{x} * df)(s) = \sum_{i=0}^{k-1} f(i) \cdot \mathbf{x}_{s-d \cdot i} \tag{4}$$

where d is the dilation factor, k is the filter size, and $s - d \cdot i$ represents the direction of the history. Therefore, the dilation factor is equivalent to introducing a fixed step size between every two adjacent filters. If the size of d is 1, the expansion convolution is an ordinary convolution kernel; if a larger expansion factor is used, the output result can represent a larger range of input data, that is, the receptive field of the convolutional network is larger.

Deep networks are prone to the problem of gradient disappearance or gradient explosion. At present, BN, regularization and other methods can be used to improve it, but it still cannot support too deep networks. Therefore, we use the residual module to realize the identity mapping of cross-layer connections, and learn the residual function $F(X) = H(X) - X$, that is, to learn the partial modification of the input X. Introducing the residual module can solve the problem of gradient disappearance.

The input of the ELGTCN model includes: bond-related time series data $\mathcal{X}$ (bond variables, corporate variables, macro variables), original news $\mathcal{N}$ and event logic graph $\mathcal{G}$. The data is normalized and mapped to a vector representation, where each vector p_t represents the fusion vector of the time series data, news corpus, and event logic graph of the bond trading day t.

$$\mathcal{P} = \{p_0, p_1, \cdots, p_{t-1}\} \tag{5}$$

Credit risk prediction can be abstracted as a binary classification problem. Considering that the liquidity of bonds is weaker than that of stocks, we choose to use one week as the step size. We use historical n days of company news and financial data to predict whether the company in the specified target $\mathcal{S} = \{s_1, \ldots, s_N\}$ has credit risk.

4 Experiment

4.1 Datasets and Compared Methods

The data required for the experiment include financial data and news texts, and the access methods include: wind database and news websites. All industrial credit bonds in China from January 2014 to December 2021 were extracted from the wind database, with a total of 36,702 samples. The financial news that can be periodically obtained through the Scrapy crawler framework involves more than 4,000 listed companies. Bond variables include the minimum transaction price, coupon rate, issue size, remaining maturity,

Table 1. Baseline Models with Different Inputs

Model	Input	
	Raw Data	Training Data
TCN	$\mathcal{X}$	TS vector
PVEB-TCN	$\mathcal{X} + \mathcal{N}$	TS vector + event embedding[1]
TDPVEB-TCN	$\mathcal{X} + \mathcal{N} + \mathcal{L}$	TS vector + event embedding[2]
KDTCN	$\mathcal{X} + \mathcal{N} + \mathcal{K}$	TS vector + knowledge embedding
ELGTCN	$\mathcal{X} + \mathcal{N} + \mathcal{G}$	TS vector + relation embedding

callability, putability, etc. Company changes include return on total assets, proportion of long-term liabilities, company size, whether it is a listed company, etc. Macro variables include risk-free interest rate, CSI 300 index.

4.2 parameters and comparison settings

Set time series data as $\mathcal{X}$, news text as $\mathcal{N}$, event knowledge graph as $\mathcal{K}$, event graph as $\mathcal{G}$. We select several commonly used baseline models and set the data types they contain as shown in Table 1.

Table 2. Evaluation Metrics

Model	Accuracy	Precision	Recall	F1-scare
TCN	68.88%	52.04%	93.30%	66.81%
PVEB-TCN	76.43%	69.60%	82.65%	75.57%
TDPVEB-TCN	78.99%	73.89%	83.37%	78.35%
KDTCN	69.43%	63.27%	76.52%	69.27%
ELGTCN	85.38%	84.83%	84.93%	84.88%

The baseline model has different inputs. In the first column, EB means event embedding, PV means bond time series data vector, TD means event graph driven, KDTCN is a stock prediction method driven by knowledge graph, ELGTCN is proposed for this paper model. event embedding^1 means that the event structure is extracted from the text and used directly for trend forecasting, event embedding^2 is used for trend forecasting after the time context of the event is generated, event knowledge embedding is the embedding of knowledge graph constructed by event elements, and event relation embedding is the embedding of event logic graph.

The training process uses the Adam optimizer. In order to avoid model overfitting, set the Batch Size to 5, the learning rate lr to 1×10^{-3}, and the Dropout Rate to 0.3.

4.3 Experimental Results

4.3.1 Visualization of Event Logic Graph

Taking "Shanghai Electric" as an example, we use hot news in 2021 to generate an event logic graph. It can be seen in Figure 3 that there are many examples of causal and reverse relationship, which is common in listed companies.

We visualized Liu et al.[17]'s event relationship extraction method based on knowledge enhancement. As shown in Figure 4, this method realizes the identification and extraction of causality, but the type of event relationship is relatively single and covers less content.

Comparing Figure 3 and Figure 4, it can be found that the method in this paper has advantages over the control method in terms of the diversity of event relationships, the comprehensiveness of event scope, and the integrity of event development.

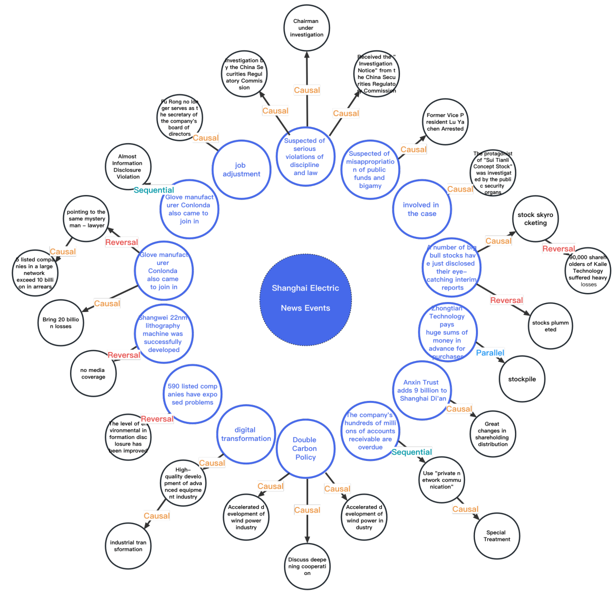

Figure 3. ELG

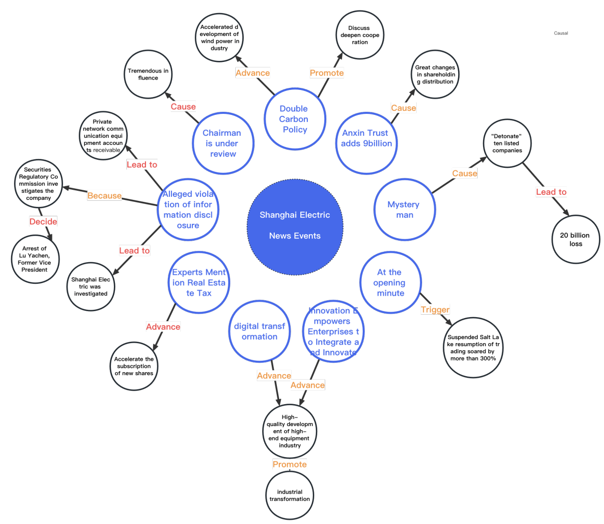

Figure 4. Knowledge-enhanced Event Extraction

4.3.2 Analysis of Evaluation Metrics

We compare the performance of the models under different inputs, and calculate the accuracy, precision, recall and F1-score of each model.

It can be seen from Table 2, PVEB-TCN incorporates the event information in the news corpus, and the performance of the model has improved. KDTCN builds a knowledge graph based on event elements and combines bond data for prediction, whose performance is lower than other models. ELGTCN has the best performance.

Table 3. Performance Evaluation of Ablation Experiments

Model	B/W	Accuracy	Presision	Recall	F1-score
TCN	**Best**	**68.88%**	**52.04%**	**93.30%**	**66.81%**
	Worst	66.02%	50.02%	88.29%	63.86%
TDPVEB Sequential	**Best%**	**72.13%**	**65.59%**	**82.89%**	**73.24%**
	Worst	69.45%	62.36%	81.00%	70.47%
CRPVEB Causal	**Best**	**72.98%**	**72.91%**	**71.39%**	**72.14%**
	Worst	63.22%	61.10%	67.64%	64.21%
PRPVEB Parallel	**Best**	**68.39%**	**77.04%**	**60.83%**	**67.98%**
	Worst	64.08%	66.73%	58.99%	62.62%
IRPVEB Reversal	**Best**	**71.12%**	**62.26%**	**76.03%**	**68.46%**
	Worst	67.12%	58.36%	71.24%	64.16%
ELGTCN	**Best**	**85.38%**	**84.83%**	**84.93%**	**84.88%**
	Worst	82.32%	79.35%	83.34%	81.30%

In order to reflect that multiple relationships are more advantageous than single relationships, we separately use sequential relationships, causal relationships, parallel relationships and reversal relationships as the basis for trend prediction. Table 3 shows the best performance and worst performance of single event relationship and compound event relationship in forecast. Through the experimental results, it can be found that the indicators of ELGTCN that integrates multiple event relationships are the best.

5 Conclusion

We design temporal convolutional networks driven by event logic graphs(ELGTCN). We constructed a graph of event by identifying multiple relationships, and applied it to the forecasting of corporate credit risk. First, we used a combination of relational triggering and pattern matching to extract multiple event relations from news text. Secondly, when constructing the graph structure, events were regarded as nodes and event relations were regarded as edges. Finally, ELGTCN was designed to integrate the event logic graph with time series data, used for corporate credit risk forecast. Experiments show that the method outperforms baseline methods in forecasting default risks.

References

[1] A. Cybulska and P. Vossen, "Translating granularity of event slots into features for event coreference resolution," in *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, 2015.

[2] Z. Liu, J. Araki, E. H. Hovy, and T. Mitamura, "Supervised within-document event coreference using information propagation." in *LREC*, 2014, pp. 4539–4544.

[3] X. Zuo, Y. Chen, K. Liu, and J. Zhao, "Event co-reference resolution via a multi-loss neural network without using argument information," *Science China(Information Sciences)*, 2019.

[4] M. Tran Phu and T. H. Nguyen, "Graph convolutional networks for event causality identification with rich document-level structures," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 3480–3490. [Online]. Available: https://aclanthology.org/2021.naacl-main.273

[5] X. Zuo, P. Cao, Y. Chen, K. Liu, J. Zhao, W. Peng, and Y. Chen, "Improving event causality identification via self-supervised representation learning on external causal statement," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, Aug. 2021, pp. 2162–2172. [Online]. Available: https://aclanthology.org/2021.findings-acl.190

[6] J. Pustejovsky, "Timeml : Robust specification of event and temporal expressions in text," in *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS-5), 2003*, 2003.

[7] H. Tissot, G. Gorrell, A. Roberts, L. Derczynski, and M. Fabro, "Ufprsheffield: Contrasting rule-based and support vector machine approaches to time expression identification in clinical tempeval," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[8] A. Leeuwenberg and M. F. Moens, "Structured learning for temporal relation extraction from clinical records," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, 2017.

[9] D. Dligach, T. Miller, C. Lin, S. Bethard, and G. Savova, "Neural temporal relation extraction," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 746–751. [Online]. Available: https://aclanthology.org/E17-2118

[10] C. Fei and Y. Miyao, "Classifying temporal relations by bidirectional lstm over dependency paths," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2017.

[11] R. Han, Y. Zhou, and N. Peng, "Domain knowledge empowered structured neural net for end-to-end event temporal relation extraction," *arXiv preprint arXiv:2009.07373*, 2020.

[12] S. Gottschalk and E. Demidova, "Eventkg: A multilingual event-centric temporal knowledge graph," *Springer, Cham*, 2018.

[13] X. Ding, Z. Li, T. Liu, and K. Liao, "Elg: an event logic graph," *arXiv preprint arXiv:1907.08015*, 2019.

[14] H. Zhang, X. Liu, H. Pan, Y. Song, and C. W.-K. Leung, *ASER: A Large-Scale Eventuality Knowledge Graph*. New York, NY, USA: Association for Computing Machinery, 2020, p. 201–211. [Online]. Available: https://doi.org/10.1145/3366423.3380107

[15] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Curran Associates Inc.*, 2013.

[16] C. Lea, M. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," *IEEE Computer Society*, 2016.

[17] J. Liu, Y. Chen, and J. Zhao, "Knowledge enhanced event causality identification with mention masking generalizations," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, C. Bessiere, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2020, pp. 3608–3614, main track. [Online]. Available: https://doi.org/10.24963/ijcai.2020/499