

# Incorporating Causal Connective Prediction to Improve Event Causality Identification with Generated Explanations

Jinzhao Cheng, Sheng Xu, Peifeng Li\*, and Qiaoming Zhu

<sup>1</sup> School of Computer Science and Technology, Soochow University, Suzhou, China  
{20225227034, sxu}@stu.suda.edu.cn, {pfli, qmzhu}@suda.edu.cn

**Abstract.** Event Causality Identification (ECI) aims to predict the causal relation for a pair of events in text. Previous work has often combined fine-tuning with specific classifiers, which contradicts the pre-trained task of the model and fails to utilize the knowledge within the pre-trained language model (PLM). Additionally, event causality identification is a complex inference task, and relying solely on sample content makes it challenging to establish an effective inference process. To tackle these two issues, we propose a new prompt-based approach for ECI, which includes a new task, causal connective prediction, and the use of explanations generated by a large-scale language model (LLM) to enhance event causality identification. Initially, we direct the LLM to produce natural language explanations of target event pairs to aid prompt generation. These explanations assist the model in comprehending events and their correlation. Additionally, we develop a task for predicting causal connectives to guide the reasoning process. Furthermore, we introduce a tensor matching mechanism to capture the semantic interaction of events in context, supporting our two prompt tasks. Our experimental results on two benchmark datasets demonstrate that our method outperforms state-of-the-art models in the sentence-level ECI task.

**Keywords:** Event Causality Identification, Prompt-based Learning, Causal Connective Prediction.

## 1 Introduction

Event causality identification (ECI) is a crucial natural language processing (NLP) task that aims to identify causality between events in text. The event causality identification task supports various NLP applications, such as machine reading comprehension [1], future event prediction [2, 3] and question answering [4].

Identifying the event causality is inherently challenging, because it usually requires a complex reasoning process. Most causal samples have no explicit causal clues, so the model that solely utilizes the content of original text is difficult to identify the causal relation. As shown in Fig. 1. The sentence has no explicit causal clues to identify the event causality: *recommendation*  $\rightarrow$  *die*. and when we consider the meanings of “recommendation” and “die” themselves, there is also no strong semantic correlation.

In this scenario, if we obtain information about these two events in context outside the annotated data, we will make it easier to determine their causal relation. Inspired by Chain of Thought (CoT) prompting [5], which guides the model to establish effective arithmetic reasoning. We can utilize the ability of the large-scale language model (e.g., GPT-3.5) to generate relevant natural language explanations to simplify the inference process.

**Example:** A Norwalk Superior Court judge on Thursday upheld a jury's **recommendation** that convicted cop killer Jose Luis Orozco **die** for his crimes .

**Explanation:** **Recommendation** means that the jury has suggested or advised that Jose Luis Orozco should receive the death penalty for his crimes. **Die** means to cease living or to pass away. In this context, the die is the sentencing of Jose Luis Orozco to death for his crimes as recommended by the jury.

**Fig. 1.** Example of causality. There is causal relation between **recommendation** and **die**. The explanation generated by LLM contains information about two events.

Most existing methods regard ECI as a classification task. However, the ECI dataset is relatively small. Therefore, it is difficult for the model to fully understand the text using traditional fine-tuning methods, which limits the predictive performance of the model. To address this problem, various methods have been proposed to leverage external knowledge [6, 7] or intrinsic knowledge of pre-trained models [8]. Specifically, Liu et al. [6], and Cao et al. [7] attempt to utilize external knowledge base to introduce commonsense knowledge to identify causality. Shen et al. [8] utilize potential causal knowledge to help causality identification. Although these methods have achieved significant results, they have two limitations. First, introducing external structured knowledge inevitably brings noise, especially when there is relatively limited annotated data, which will bring a significant negative impact on the performance of the model. Second, the event causality identification task requires complex reasoning process. It is challenging for the model to understand semantics and to transform them into causal reasoning solely with original text.

In this paper, we propose a new prompt-based method for the ECI task, which can leverage natural language explanations related to events to effectively improve the performance of the model. First, we leverage GPT-3.5 to generate natural language explanations according to the predefined prompts, including event trigger explanation and event correlation explanation. In order to avoid exceeding the maximum encoding length of the PLM, we set the maximum length for two types of explanations. As shown in Fig. 1, the explanations will provide more information about the target events and

assist model in determining causal relation. Secondly, it is crucial for our judgement that whether causal connectives are present (because, lead to, etc.) in the text. However, most of the samples do not contain causal connectives. Therefore, we can explicitly construct this condition via a prompt template without considering the presence of connectives in the original sample. Based on the above idea, we introduce causal connective prediction as an auxiliary task to mine potential causal cues for the model. In addition, due to the performance of prompt-based method relies on the construction of verbalizer. Therefore, we select the less ambiguous connectives corresponding to each class based on the causal signals in the annotated data and external knowledge to implement answer mapping. Finally, considering that two events with causal relation contain more common semantic features. Therefore, we introduce tensor matching mechanism [9] to capture the semantic interaction of the events in context to optimize the final hidden state.

## 2 Related Work

### 2.1 Event Causality Identification

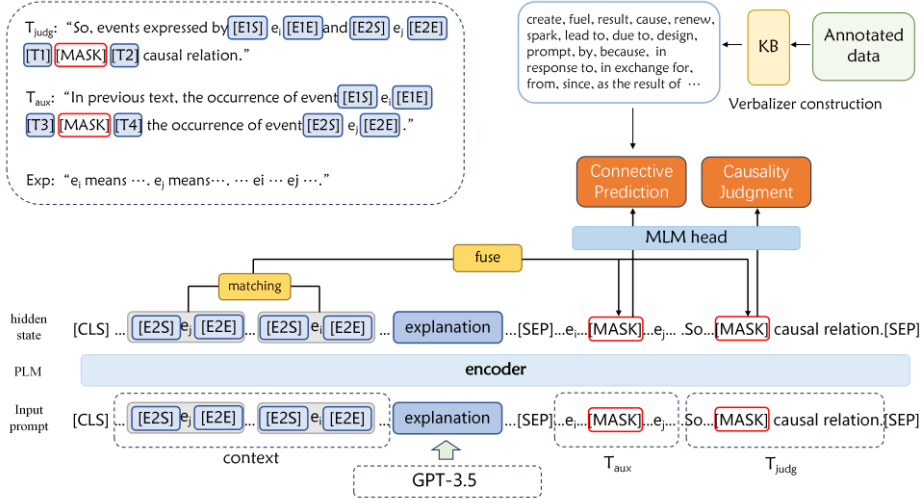
Early work utilizes different feature-based methods to identify events causal relation [10-13]. With the development of deep learning, most of studies prefer to utilize neural-based method for this task and achieve state-of-the-art performance [6, 14]. However, the amount of annotated data related to causality is relatively small in the existing datasets, which inevitably affects the predictive performance of the model. To alleviate this problem, some studies attempt to introduce external commonsense knowledge to enrich labeled data and trigger semantics. Specifically, Zuo et al. [14] effectively expand data scale with distantly supervised labeled training data. Cao et al. [7] construct graph module to encode different external structural knowledge. Although structured external knowledge can enhance the semantic understanding of events, it inevitably introduces noise. Based on the above considerations, we introduce the natural language explanation to capture more information about events.

### 2.2 Prompt-based Methods

Prompt-based methods [16] transform classification tasks into a cloze-style MLM problem, which aligns the downstream task with pre-trained task. Han et al. [17] apply logic rules to construct prompts with multiple sub-prompts for text classification task. Xu et al. [9] achieve event coreference reasoning by constructing auxiliary tasks for event types and argument compatibility. For event causality identification, Shen et al. [8] leverage two derivative prompts to model explicit causality and implicit causality separately. Compared to [8], our method not only constructs rich actual label words, but also generates natural language explanations to help the model simplify causal inference.

### 3 Method

Formally, given an instance  $x = (S, (e_i, e_j))$ , where  $S$  is a sentence and  $(e_i, e_j)$  is a pair of events in  $S$ , we need to predict whether there is a causal relation. Following previous work [6, 14], we formulate the ECI as a binary classification, so our final prediction is one of  $\{exist, not\_exist\}$ , representing there is or no causality respectively. The overall framework of our model is shown in Fig. 2.



**Fig. 2.** The overall framework of our prompt-based method for ECI, where KB is the external knowledge base.

#### 3.1 Overview

In this paper, we transform ECI into a cloze-style MLM problem by using the template  $T(\cdot)$  and the verbalizer  $v$ . The  $v: Y \rightarrow V$ , represents the mapping from the label space of the ECI task to the token space of the PLM. Specifically, we insert the [MASK] token into template  $T(\cdot)$  and send it into PLM to learn the context representation of the mask position. Then, we utilize a mask language model (MLM) head to get confidence score of the tokens in the vocabulary, and calculate the normalization probability of special words as follows.

$$p(y|x) = p([MASK] = v(y)|T(x)) = \frac{\exp P(v(y)|T(x))}{\sum_{i=1}^k \exp P(v(i)|T(x))} \quad (1)$$

where the  $v(y)$  and the  $k$  represent the token filling the mask position of the template  $T(x)$  and the length of the label word space, respectively.  $P(t|T(x))$  represents the confidence score of the token  $t$  that predicted by the MLM head of PLM.

### 3.2 Explanation Generation

Due to the poor performance of directly using LLMs to identify events causality [18], we only generate explanations that can assist the model in reasoning. For an instance  $x = (S, (e_i, e_j))$ , we generate two types of explanations with GPT-3.5, including 1) event trigger explanation (ETE) and 2) event correlation explanation (ECE). Specifically, ETE includes the target trigger meaning and the description of the event expressed by the trigger. ECE refers to the correlation between two events in the context, which is a premise for identifying the causal relation. In order to prompt LLM to generate desired explanations, we design two types of prompts as follows.

- *Prompt1*: Explain the event triggered by ‘ $t_i/t_j$ ’ in the following sentence: ‘ $S$ ’ return the explanation in the form of  $t_i/t_j$  means ... as a sentence.
- *Prompt2*: Explain whether exist correlation between event ‘ $t_i$ ’ and event ‘ $t_j$ ’ or not in the following sentence: ‘ $S$ ’.

where  $t_i$  and  $t_j$  represent the triggers expressing for  $e_i$  and  $e_j$  respectively.  $S$  represents the sentence where events  $e_i$  and  $e_j$  are located. We insert event markers  $\langle e1 \rangle / \langle e2 \rangle$  into the  $S$  to represent the start / end of two target event triggers  $t_i$  and  $t_j$ . In addition, we also limit the length of the explanation returned by GPT-3.5 by mentioning it in the prompt and setting argument `max_tokens`. After error analysis, we found that although concise explanations are required, LLM tends to over-explain and produce disruptive content. In some cases, this explanation can even lead to a drop in the performance of the model. Therefore, we truncate ETE to ensure that there is no intersection with other events in the obtained explanations as much as possible. For ECE, we remove fixed format correlation judgements and retain specific explanations. Finally, we concatenate both explanations for the construction of the input sequence.

### 3.3 Auxiliary Task: Causal Connective Prediction

Considering that ECI is a complex task requiring reasoning, we construct an auxiliary prompt task to build the association between the semantic understanding and identification by mining potential causal clues. As shown below, we convert this task into a mask language prediction problem and set  $Y_{aux} = \{na, forward\ causality, backward\ causality\}$  as the auxiliary task label according to the direction of the causal expression. Note that their output is only for assistance rather than for the final prediction. The format of the auxiliary task template is as follows.

- $T_{aux}$ : the occurrence of event [E1S] $e_i$ [E1E] [T1] [MASK] [T2] the occurrence of event [E2S] $e_j$ [E2E].

Similar to PTR [17], [T1] / [T2], [E1S] / [E1E], and [E2S] / [E2E] will then be added to the vocabulary as learnable tokens to make the template dynamically adapt to the task during the training process.

For verbalizer construction, the key issues are to select label words related to the direction of the causal expression and to establish an association between explicit and implicit causal relations. From these two perspectives, we first extract causal signals for each causal sample from the annotated data. If it is not present, we skip it. We then apply lemmatization and filtering to all the words or phrases obtained, so that each of them can be filled in the middle of the template. After that, we divide words into two categories based on their semantics. For the ‘*na*’ class, we choose parallel connectives, such as *and*, *or*, etc. Finally, inspired by [19], we choose Related Words as our external knowledge base to further enrich the label words for each class.

After obtaining all the label words, we establish a mapping from one label to multiple label words, and the sum of the normalized probabilities of all label words corresponding to each label is used as the label probability. The advantage of this method is that it allows for higher coverage for label words. However, its disadvantage is that for label words with more than one token, we can only use the average score as an approximate result. In the training stage, we calculate the cross-entropy loss  $L_a$  between the predicted probability distribution of label and golden causal direction label. During the prediction stage, we do not need the predicted result of this mask position. The connective prediction template will mine the potential causal associations based on the memory of the training stage. Benefiting from the attention mechanism, PLM can focus more on connective information when encoding the context to guide the final causal judgment.

### 3.4 Joint Reasoning

To better guide the model to infer the relation of the events and obtain the contextual representation, we insert special tokens [E1S] / [E1E] and [E2S] / [E2E] on both sides of the target events. Due to we treat the event causality identification as a binary classification problem, so we design the final causal judgement template  $T_{judg}$  as:

- $T_{judg}(x)$ : So, the events expressed by [E1S] $e_i$ [E1E] and [E2S] $e_j$ [E2E] [T3] [MASK] [T4] causal relation.

We set the label words  $V_y = \{exist, no\_exist\}$  to represent a causal relation and no causal relation, respectively. The *no\_exist* contains multiple tokens, and it is impossible to find this word from the vocabulary to satisfy a single mask, so we create a virtual label word for it using the semantic verbalizer [20, 21]. For label word *no\_exist*, we take 'not exist' as the semantic description and get tokenized sequence  $\{t_1, t_2 \dots, t_m\}$ . Then we initialize the embedding of the virtual label word  $v$  as follows.

$$E(v) = \frac{1}{m} \sum_{i=1}^m E(t_i) \quad (2)$$

where  $E(\cdot)$  is the token embedding table. Then, we expand the MLM head of PLM with the virtual label word ‘*no\_exist*’. After completing the above work, we concatenate the context, explanations, and two templates in order and then send them into the PLM for sequence encoding.

According to the previous work [9], the tensor matching mechanism can effectively capture semantic interactions between events, which can help optimize our prompt tasks. Therefore, we introduce these semantic matching operations to update the mask position representation by fusing interactive features. Suppose that the event trigger of  $e_i$  is tokenized into  $k$  tokens. Specifically, we first apply the attention mechanism on top of the hidden vectors  $h_j$  of the  $j$ -th tokens of the trigger in context to obtain the representation vector  $t_i$  of event mention  $e_i$  as follows.

$$t_i = \sum_{j=1}^k a_j h_j \quad (3)$$

$$a_i = \frac{\exp(w_s^T h_i)}{\sum_{j=1}^k \exp(w_s^T h_j)} \quad (4)$$

where  $w_s$  is model parameter that can be optimized during the training stage. Considering that two events with causal relations contain more common semantic features, their representation vectors will have greater similarity [22]. Therefore, we select the multi-perspective cosine similarity  $MultiCos(\cdot)$  and the element-wise product as matching operations to capture the semantic interactions between event mentions in context as follows.

$$I(x_i, x_j) = [x_i \odot x_j || MultiCos(x_i, x_j)] \quad (5)$$

$$I_{trig} = I(t_i, t_j) \quad (6)$$

where  $I(\cdot)$  represents the interaction vector, and the symbols  $\cdot || \cdot$  and  $\odot$  represent concatenation and element-wise production operations, respectively. After obtaining the trigger matching features  $I_{trig}$ , we utilize it to update the final representation vectors of two mask tokens in the connective prediction template  $T_{aux}$  and the causal judgement template  $T_{judg}$ . The fusion formula is as follows.

$$\tilde{h}_{[mask]}^{Aux} = [h_{[mask]}^{Aux} || I_{trig}] W_a \quad (7)$$

$$\tilde{h}_{[mask]}^{judg} = [h_{[mask]}^{judg} || I_{trig}] W_j \quad (8)$$

where  $h_{[mask]}^{Aux}$  and  $h_{[mask]}^{judg}$  are representation vectors of the mask position corresponding to the causal connective prediction and causality judgement tasks, respectively, and  $W_a, W_j$  are parameter matrices responsible for transforming the tensor dimension into the hidden size of the PLM. Finally, we send the updated representation vectors of mask tokens to the MLM head of the PLM. During training, we use the cross-entropy function to calculate the losses of two mask tasks. Afterwards, we update the model parameters through the back-propagation algorithm. Conveniently, we denote the causality judgement loss as  $L_j$  and the causal connective prediction loss as  $L_a$ . We use a weighted approach to obtain the final loss  $L$  as follows.

$$L = \lambda_1 L_j + \lambda_2 L_a \quad (9)$$

where  $\lambda_1$  and  $\lambda_2$  represent the balance coefficient.

## 4 Experimentation

### 4.1 Experimental Settings

**Datasets and Metrics.** Following previous work [7, 8], we evaluate our methods on two benchmark datasets for ECI, i.e., EventStoryLine (ESL) v0.9 [23] and Causal-TimeBank (CTB) [24]. EventStoryLine v0.9 involves 258 documents, 22 topics, 4,316 sentences, 5,334 event mentions, and 1,770 of 7,805 event mention pairs that are causally related. Following [7, 8], we use the documents of the last two topics as the development set, and we conduct 5-fold cross-validation on the remaining document. Causal-TimeBank [24] contains 184 documents, 6,813 events, and 318 of 7,608 event mention pairs annotated with causal relations. We perform a 10-fold cross-validation evaluation for Causal-TimeBank. For evaluation metrics, we adopt Precision (P), Recall (R) and F1-score (F1).

**Parameter Settings.** We choose the RoBERTa-base with open pre-trained parameters, which includes 12-layers, 768-hidden, and 12-heads, as our encoder. We add all learnable tokens to a vocabulary with 768-dimensional embedding. We choose the gpt-3.5-turbo-instruct as our explanation generator. For the tensor matching operations, following previous work [9], we set the matching dimension and perspective number to 64 and 128, respectively, and set the tensor factorization parameter to 4. We set the epoch and batch size to 20 and 4, respectively, and apply a learning rate of  $2e-5$  to update the parameters. We set  $\lambda_1$  and  $\lambda_2$  to 0.5 and 0.5 respectively for balancing the two types of losses. To alleviate the imbalance in sample proportion, we adopt a negative sampling rate of 0.6 in the training.

### 4.2 Baselines

We compare the proposed method with two types of existing methods, namely, feature-based methods, PLM-based methods. For the ESL dataset, we adopted following methods as baselines.

- **LR+** and **ILP** [25], which consider the causal structures at the document level.

For Causal-TB, we choose the following baselines:

- **VR-C** [26], a verb rule-based model based on lexical information and causal signals.

Furthermore, we compare our method with the methods based on PLM as follows.

- **LSIN** [7], a method that constructs two types of graphs using external knowledge to strengthen event semantics;
- **DPJL** [8], a method that introduces two derivative prompt tasks to support causal reasoning;



- **SemSIn** [27], a method leveraging AMR to model explicit semantic structures and explore implicit associations;
- **Base Prompt**, an MLM that only includes our final causal judgement task.

### 4.3 Experimental Results

Table 1 and Table 2 show the results of our method and other baselines on the datasets ESL and CTB, respectively. We can see that our method achieves the best F1 score on both datasets. This demonstrates the effectiveness of our proposed method in the sentence-level event causality identification task.

**Table 1.** Comparison of different methods on EventStoryLine v0.9. The best results are highlighted in **bold**, and the second-best results are underlined.

model	P	R	F1
LR+	37.0	45.2	40.7
ILP	37.4	55.8	44.7
LSIN	47.9	58.1	52.5
DPJL	<u>65.3</u>	<u>70.8</u>	<u>67.9</u>
SemSIn	64.2	65.7	64.9
Base prompt	64.4	<b>71.9</b>	<u>67.9</u>
<b>Ours</b>	<b>69.1</b>	70.2	<b>69.6</b>

**Table 2.** Comparison of different methods on Causal-TimeBank.

model	P	R	F1
VR-C	<u>69.0</u>	31.5	43.2
LSIN	51.5	56.2	52.9
DPJL	63.6	<u>66.7</u>	64.6
SemSIn	52.3	65.8	58.3
Base prompt	67.4	66.4	<u>66.7</u>
<b>Ours</b>	<b>70.5</b>	<b>68.3</b>	<b>68.4</b>

After further comparison, we find that the prompt-based method significantly outperforms the traditional fine-tuning method, especially when there is relatively little training data, such as CTB. This is attributed to the fact that prompt-based learning aligns with the pre-trained task and fully utilizes the potential knowledge within the PLM. Benefiting from prompt-based learning and multiple derivative prompt tasks, the performance of DPJL [8] improves significantly. However, due to the limited content of original text, relying on this information alone makes it difficult to understand the relationship between two events. In contrast, we design two simpler prompt tasks to establish the causal reasoning process, incorporating intuitive natural language explanations generated by GPT-3.5 to guide the model. Furthermore, we construct different

verbalizers to bridge the gap between reasoning and judgement. Compared to the soft verbalizer used in DPJL [8], the label words can more intuitively reflect the meaning expressed by the template.

#### 4.4 Ablation Study

To investigate the contributions of different components of our method, we conducted ablation study. (1) -Exp: removing explanations; (2) -Aux: removing the auxiliary causal connective prediction template; (3) -TM: removing the tensor matching mechanism.

**Table 3.** Results of the ablation study on EventStoryLine.

model	P	R	F1
<b>-Exp</b>	66.2	<b>71.5</b>	68.7
<b>-Aux</b>	68.7	67.6	68.1
<b>-TM</b>	66.3	70.8	68.2
<b>Full</b>	<b>69.1</b>	70.2	<b>69.6</b>

**Table 4.** Results of the ablation study on Causal-TimeBank.

model	P	R	F1
<b>-Exp</b>	66.7	<b>70.1</b>	67.2
<b>-Aux</b>	62.5	69.8	65.6
<b>-TM</b>	<b>71.1</b>	66.9	68.2
<b>Full</b>	70.5	68.3	<b>68.4</b>

Table 3 and Table 4 show that removing any component will decrease the performance of the model. The most significant decrease is when we remove the auxiliary causal connective prediction template, indicating that the auxiliary causal connective prediction template effectively improves causal inference. It is worth noting that compared to the ESL dataset, the performance of the model in the CTB dataset shows a large drop. This may be due to the fact that when constructing label words for auxiliary prompt task, we rely on the causal signals in the annotated data. Almost half of causal samples in the CTB have explicit causal signals. Benefiting from these causal signals, the auxiliary prompt task can directly obtain causal cues to guide reasoning.

## 5 Conclusion

In this paper, we design a prompt to transform ECI into a mask language prediction task. In this way, the event modeling and causal judgement can be performed simultaneously based on a shared context. In addition, we introduce an auxiliary prompt task, namely, causal connective prediction, to explicitly show the reasoning process of ECI.

We also leverage the generative capability of LLM to generate explanations to assist in performing causal reasoning. The experimental results on the ESL and CTB datasets show that our method outperforms previous SOTA methods in sentence-level ECI. Despite the effectiveness of our approach, it still suffers from an obvious shortcoming. Due to the difficulty in determining the quality of each explanation, some simple samples may make incorrect predictions after adding explanations. In future work, how to utilize higher quality explanations is the focus of our future work.

**Acknowledgments.** The authors would like to thank the two anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (Nos. 62276177 and 62376181), and Key R&D Plan of Jiangsu Province (BE2021048).

## References

1. Berant, J., Srikumar, V., Chen, P.-C., Vander Linden, A., Harding, B., Huang, B., Clark, P., Manning, C. D. (2014). Modeling biological processes for reading comprehension. In EMNLP 2014, pp. 1499--1510.
2. Radinsky, K., Davidovich, S., Markovitch, S. (2012) Learning causality for news events prediction. In WWW 2012. pp. 909--918.
3. Hashimoto, C., Torisawa, K., Kloetzer, J., Sano, M., Varga, J., Oh, J.-H., Kidawara, Y. (2014). Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In ACL 2014, pp. 987--997.
4. Oh, J.H., Torisawa, K., Kruengkrai, C., Iida, R., Kloetzer, J. (2017). Multi-column convolutional neural networks with causality attention for why-question answering. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17), pp. 415--424.
5. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. arXiv preprint arXiv: 2201.11903.
6. Liu, J., Chen, Y.B., Zhao, J. (2020). Knowledge enhanced event causality identification with mention masking generalizations. In IJCAI 2020, pp. 3608--3614.
7. Cao, P.-F., Zuo, X. Y., Chen, Y. B., Liu, K., Zhao, J., Chen, Y. G, Peng, W. H. (2021). Knowledge-enriched event causality identification via latent structure induction networks. In ACL 2021, pp. 4862--4872.
8. Shen, S.R., Zhou, H., Wu T.T., Qi, G.L. (2022). Event Causality Identification via Derivative Prompt Joint Learning. In COLING 2022, pp. 2288--2299.
9. Xu, S., Li, P.F., Zhu, Q.M. (2023). CorefPrompt: Prompt-based event coreference resolution by measuring event type and argument compatibilities. In EMNLP 2023, pp. 15440--15452.
10. Do, Q., Chan, Y.S., Roth, D. (2011). Minimally supervised event causality identification. In EMNLP 2011, pp. 294--303.
11. Riaz, M., Girju, R. (2013). Toward a better understanding of causality between verbal events: Extraction and analysis of the causal power of verb-verb associations. In Proceedings of the SIGdial Meeting on Discourse and Dialogue 2013 Conference, pp. 21--30.
12. Zhichao Hu, Elahe Rahimtoroghi, and Marilyn Walker. (2017). Inference of fine-grained event causality from blogs and films. In Proceedings of the Events and Stories in the News Workshop, pp. 52--58.

13. Hashimoto, C. (2019). Weakly supervised multilingual causality extraction from Wikipedia. In EMNLP 2019, pp. 2988--2999.
14. Xinyu Zuo, Yubo Chen, Kang Liu, and Jun Zhao. (2020). KnowDis: Knowledge enhanced data augmentation for event causality detection via distant supervision. In COLING 2020, pp. 1544--1550.
15. Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. (2013). Abstract meaning representation for sembanking. In Proceedings of the 7th linguistic annotation workshop and interoperability with discourse, pp. 178--186.
16. Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing*
17. Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2022. Ptr: Prompt tuning with rules for text classification. *AI Open*, 3:182--192.
18. Jinglong Gao, Xiao Ding, Bing Qin, Ting Liu. 2023. Is ChatGPT a good causal reasoner? A comprehensive evaluation. In EMNLP 2023. pp. 11111--11126.
19. Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, Maosong Sun. 2022. Knowledgeable prompt-tuning: incorporating knowledge into prompt verbalizer for text classification. In *ACL 2022*, pp. 2225--2240.
20. Lu Dai, Bang Wang, Wei Xiang, Yijun Mo. 2022. Bi-directional iterative prompt-tuning for event argument extraction. In EMNLP 2022. pp. 6251--6263.
21. Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022. Knowprompt: Knowledge aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web Conference 2022*, pp. 2778--2788.
22. Sifan Wu, Ruihui Zhao, Yefeng Zheng, Jian Pei, Bang Liu. 2023. Identify event causality with knowledge and analogy. In *AAAI 2023*, pp. 13745--13753.
23. Tommaso Caselli and Piek Vossen. (2017). The event StoryLine corpus: A new benchmark for causal and temporal relation extraction. In *Proceedings of the Events and Stories in the News Workshop*, pp. 77--86.
24. Paramita Mirza, Rachele Sprugnoli, Sara Tonelli, and Manuela Speranza. 2014. Annotating causality in the TempEval-3 corpus. In *Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL)*, pp. 10--19.
25. Lei Gao, Prafulla Kumar Choubey, and Ruihong Huang. 2019. Modeling document-level causal structures for event causal relation identification. In *NAACL 2019*, pp.1808--1817.
26. Paramita Mirza. (2014). Extracting temporal and causal relations between events. In *Proceedings of the ACL 2014 Student Research Workshop*, pp. 10--17.
27. Zhilei Hu, Zixuan Li, Xiaolong Jin, Long Bai, Saiping Guan, Jiafeng Guo, and Xueqi Cheng. (2023). Semantic structure enhanced event causality identification. *arXiv preprint arXiv:2305.12792*.