Knowledge-Enriched Event Causality Identification via Latent Structure Induction Networks

Pengfei Cao[1,2], **Xinyu Zuo**[1,2], **Yubo Chen**[1,2], **Kang Liu**[1,2], **Jun Zhao**[1,2],
Yuguang Chen[3] and **Weihua Peng**[3]
[1]National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Beijing Baidu Netcom Science Technology Co., Ltd
{pengfei.cao, xinyu.zuo, yubo.chen, kliu, jzhao}@nlpr.ia.ac.cn,
{chenyuguang, pengweihua}@baidu.com

Abstract

Identifying causal relations of events is an important task in natural language processing area. However, the task is very challenging, because event causality is usually expressed in diverse forms that often lack explicit causal clues. Existing methods cannot handle well the problem, especially in the condition of lacking training data. Nonetheless, humans can make a correct judgement based on their background knowledge, including *descriptive knowledge* and *relational knowledge*. Inspired by it, we propose a novel **L**atent **S**tructure **I**nduction **N**etwork (LSIN) to incorporate the external structural knowledge into this task. Specifically, to make use of the descriptive knowledge, we devise a *Descriptive Graph Induction* module to obtain and encode the graph-structured descriptive knowledge. To leverage the relational knowledge, we propose a *Relational Graph Induction* module which is able to automatically learn a reasoning structure for event causality reasoning. Experimental results on two widely used datasets indicate that our approach significantly outperforms previous state-of-the-art methods.

1 Introduction

Event causality identification (ECI) aims to identify causal relation of events in texts. For example, in the sentence "The *earthquake* generated a *tsunami*.", an ECI model should be able to identify a causal relationship that holds between the two mentioned events, i.e., *earthquake* $\xrightarrow{\text{cause}}$ *tsunami*. ECI is an important task in natural language processing (NLP) area and can support many NLP applications, such as machine reading comprehension (Berant et al., 2014), process extraction (Thalappillil Scaria et al., 2013) and future event prediction (Radinsky et al., 2012; Hashimoto et al., 2014).

Identifying event causal relation is inherently challenging, because event causality is usually expressed in diverse forms that often lack explicit clues indicating its existence. For example in Figure 1, the sentence has no explicit clue indicating the causal relation between "*global warming*" and "*tsunami*". In this scenario, models can resort to a large amount of labeled data to learn diverse causal expressions. However, existing ECI datasets are very small. For example, the largest dataset EventStoryLine (Caselli and Vossen, 2017) only contains 258 documents, which is not sufficient to train neural network models (Liu et al., 2020). Consequently, models cannot thoroughly understand the text and possibly make a wrong prediction. Nonetheless, humans could make a correct judgement, because humans have the background knowledge about the two events. To be more specific, humans not only know what the two events are, but also know the connection between them. Fortunately, existing knowledge bases (KBs) usually contain the **Descriptive Knowledge** of events and **Relational Knowledge** between events, which can be regarded as the background knowledge to enhance ECI models. In this paper, we focus on how to incorporate these two kinds of external knowledge into the task.

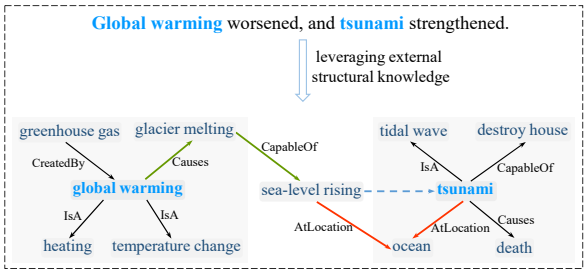

Figure 1: An example of leveraging the external structural knowledge for ECI task. The dashed arrow indicates a missing link in the knowledge base.

Descriptive Knowledge: The external knowl-

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 4862–4872
August 1–6, 2021.

edge base contains the descriptive or explanatory information about events, which can be called the descriptive knowledge of events. It usually consists of one-hop neighbors of events. This kind of knowledge is able to help the model better understand what the mentioned event is. For example in Figure 1, the descriptive knowledge associated with "global warming" includes *(global warming, IsA, temperature change)*, *(global warming, CreatedBy, greenhouse gas)* and so on. If the model can make use of such knowledge, it is obvious that the model can better understand the meaning of the event itself than using only the given text. Therefore, incorporating the descriptive knowledge is very helpful for this task. However, when leveraging this kind of knowledge, we find two critical challenges: (1) As shown in Figure 1, the descriptive knowledge forms a sub-graph. How to effectively encode the graph-structured knowledge is a very challenging problem; (2) The knowledge base is incomplete (Wang et al., 2020), which will inevitably cause the descriptive knowledge of some events cannot be obtained from the KB. Thus, the model should have the ability to obtain and encode such knowledge, even if it does not exist in the KB.

Relational Knowledge: The external knowledge base contains connections between events, which can be referred as the relational knowledge between events. It is usually defined by the multi-hop path between two events. This kind of knowledge can provide useful information for event causality reasoning, especially when the text lacks causal clues. For example in Figure 1, the relational knowledge between the two events is "*global warming*" $\xrightarrow{Causes}$ "*glacier melting*" $\xrightarrow{CapableOf}$ "*sea-level rising*" $\xrightarrow{AtLocation}$ "*ocean*" $\xleftarrow{AtLocation}$ "*tsunami*". Apparently, compared with only using text information, utilizing the relational knowledge can provide ample evidence for the model to judge the causality between "*global warming*" and "*tsunami*". However, two challenges exist when using the relational knowledge: (1) The multi-hop path may miss some potentially useful relations. For example in Figure 1, the fact *(sea-level rising, Causes, tsunami)* is described in the wikipedia page of "sea-level rising"[1], while it is not annotated in the KB; (2) Not all the knowledge on the path is related to causality, such as *(sea-level rising, AtLocation, ocean)*. Therefore, directly reasoning along the multi-hop path structure may not be optimal. The model should be able to learn a more reasonable structure for capturing potentially useful information and reducing the impact of irrelevant knowledge.

In this paper, we propose a novel method termed as **L**atent **S**tructure **I**nduction **N**etwork (LSIN) to overcome aforementioned challenges. Specifically, we devise a *Descriptive Graph Induction* module to make use of the descriptive knowledge. The module first adopts a hybrid method of retrieval and generation to obtain the descriptive knowledge, and then utilizes the information aggregation technique to encode the graph-structured knowledge. Meanwhile, we propose a *Relational Graph Induction* module to leverage the relational knowledge. The module first treats the reasoning structure as a latent variable and learns it in an end-to-end fashion. Then, the module performs event causality reasoning based on the induced structure. Experimental results on two widely used datasets demonstrate that our model substantially outperforms previous state-of-the-art methods.

Our contributions are summarized as follows:

- We propose a novel Latent Structure Induction Network (LSIN) to leverage the external structural knowledge. To our knowledge, we are the first to use both the descriptive knowledge and relational knowledge for this task.
- To exploit the descriptive knowledge, we devise a descriptive graph induction module. To utilize the relational knowledge, we propose a relational graph induction module.
- Experimental results on two widely used datasets indicate that our proposed approach significantly outperforms previous state-of-the-art methods.

2 Related Work

Event causality identification (ECI) is a very important task in natural language processing area, which has attracted extensive attention in the past few years. Early studies for the task are feature-based methods which utilize lexical and syntactic features (Riaz and Girju, 2013; Gao et al., 2019), explicit causal patterns (Beamer and Girju, 2009; Do et al., 2011; Hu et al., 2017), and statistical causal associations (Riaz and Girju, 2014; Hashimoto et al., 2014; Hu and Walker, 2017; Hashimoto, 2019) for the task. With the development of deep learning, neural

[1] https://en.wikipedia.org/wiki/Sea_level_rise

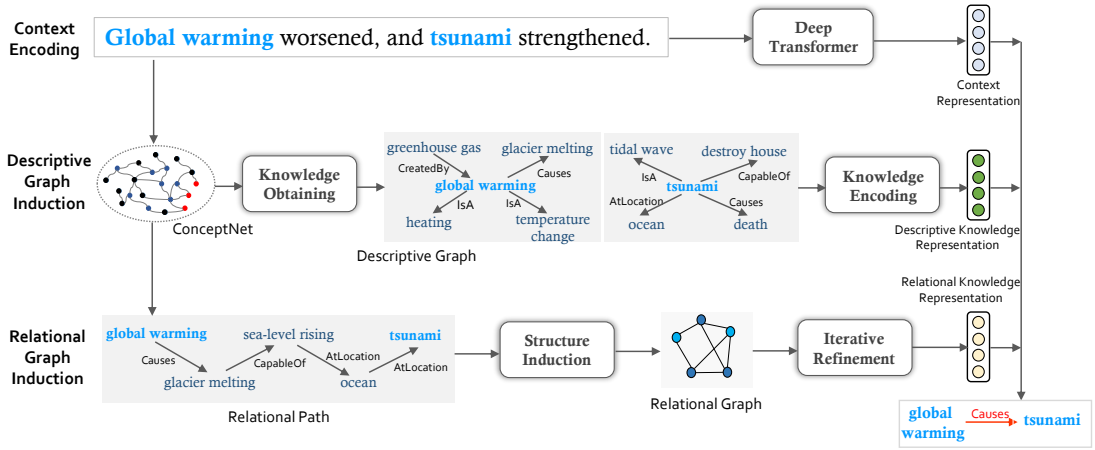

Figure 2: The architecture of our proposed latent structure induction network for event causality identification.

network-based methods have been proposed for the task and achieved the state-of-the-art performance (Kruengkrai et al., 2017; Kadowaki et al., 2019; Liu et al., 2020; Zuo et al., 2020). Liu et al. (2020) propose a mention masking generalization method and also consider the external structural knowledge. The very recent work (Zuo et al., 2020) propose a data augmentation method to alleviate the data lacking problem for the task. Regarding datasets construction, Mirza (2014) annotates the Causal-TimeBank dataset about event causal relations in the TempEval-3 corpus. Caselli and Vossen (2017) construct a dataset called EventStoryLine for event causality identification. Despite many efforts for this task, most existing methods typically train the models on manually labeled data solely, rarely considering the external structural knowledge. As a result, these methods cannot handle well the cases where there is no explicit causal clue.

Although Liu et al. (2020) leverage the descriptive knowledge to enrich event representations, they directly retrieve the descriptive knowledge from the KB. Therefore, their method cannot handle the cases where there is no knowledge about the event in the KB. In addition, they ignore the relational knowledge between events. By contrast, our method can not only generate the descriptive knowledge when it cannot be retrieved from the KB, but also leverage the relational knowledge. To our knowledge, we are the first to simultaneously make use of the descriptive knowledge and relational knowledge for this task.

3 Methodology

Following previous works (Ning et al., 2018; Liu et al., 2020), we formulate ECI as a binary classification problem. For every pair of events in a sentence, we predict whether a causal relation holds. Figure 2 schematically visualizes our approach, which consists of three major components: (1) *Context Encoding* (§3.1), which encodes the input sentence and outputs contextualized representations; (2) *Descriptive Graph Induction* (§3.2), which first obtains the corresponding descriptive knowledge for each event, and then encodes the graph-structured knowledge; (3) *Relational Graph Induction* (§3.3), which automatically induces a reasoning structure and performs causality reasoning on the induced structure. We will illustrate each component in detail.

3.1 Context Encoding

Given a sentence with a pair of events (denoted as e_1 and e_2), the context encoding module aims to extract context features, which takes the sentence as input and outputs the context representations. Our context encoder is based on the Transformer architecture (Vaswani et al., 2017). We adopt the BERT (Devlin et al., 2019) to encode the input sentence,[2] which has achieved the state-of-the-art performance for ECI task (Liu et al., 2020; Zuo et al., 2020). After using BERT encoder to compute the contextual representations of the entire sentence, we concatenate representations of [CLS], e_1 and e_2 as the context representation regarding to the event pair (e_1, e_2), namely

$$\boldsymbol{F}_C^{(e_1,e_2)} = \boldsymbol{h}_{[\text{CLS}]} \oplus \boldsymbol{h}_{e_1} \oplus \boldsymbol{h}_{e_2}, \quad (1)$$

[2]Note that the encoder is not our focus in this paper. In fact, other models like convolutional neural networks and long short-term memory networks can also be as encoders.

where $\oplus$ indicates the concatenation operation. $\boldsymbol{h}_{[\text{CLS}]} \in \mathbb{R}^d$, $\boldsymbol{h}_{e_1} \in \mathbb{R}^d$ and $\boldsymbol{h}_{e_2} \in \mathbb{R}^d$ are representations of [CLS], e_1 and e_2, respectively. d is the output hidden size of BERT model.

3.2 Descriptive Graph Induction

3.2.1 Knowledge Obtaining

Given e_1 and e_2, we adopt a hybrid method of *retrieval* and *generation* to obtain their descriptive knowledge, respectively. The descriptive knowledge forms a sub-graph which is called *Descriptive Graph* (denoted as G_d). For this paper, we prefer CONCEPTNET (Speer et al., 2017) as the external KB, which contains abundant semantic knowledge of concepts. We take e_1 as an example to illustrate the knowledge obtaining procedure:

(1) If the descriptive knowledge can be retrieved from the KB, we adopt the retrieval method. Our method first grounds e_1 to a concept via matching the event mention with the tokens of concepts in CONCEPTNET. We enhance the matching approach with some rules, such as soft matching with lemmatization and filtering of stop words. The grounded concept is called zero-hop concept. Then, our method grows zero-hop concept with one-hop concepts. The zero-hop concept, one-hop concepts and all relations between them form the descriptive graph for e_1 (denoted as G_{d_1}).

(2) If the descriptive knowledge cannot be retrieved from the KB, we adopt the generation method. Our method employs the pre-trained model, COMET (Bosselut et al., 2019), which is originally proposed for the knowledge base completion. Specifically, COMET is obtained by fine-tuning GPT (Radford et al., 2018) on CONCEPTNET. The input of COMET is the head event and candidate relation, and the output is the tail event. The relation types are the same as the ones used in Bosselut et al. (2019). By leveraging COMET, we can generate the descriptive graph G_{d_1} for e_1.

In the same way, we can also construct the descriptive graph G_{d_2} for e_2.

3.2.2 Knowledge Encoding

Graph neural networks have been widely used to encode graph-structured data (Lin et al., 2019; Yang et al., 2019), as they are able to effectively collect relevant evidence based on an information aggregation scheme. In addition, many works show that relational graph convolutional networks (R-GCNs) (Schlichtkrull et al., 2018) usually over-parameterize the model and cannot effectively utilize multi-hop relational information (Zhang et al., 2018; Lin et al., 2019). We thus apply GCNs (Kipf and Welling, 2017) to encode the related descriptive knowledge of e_1 and e_2.

Formally, given a descriptive graph G_d (i.e., G_{d_1} or G_{d_2}) with n_d nodes (i.e., concepts), which can be represented with an $n_d \times n_d$ adjacency matrix $\boldsymbol{A}^d$. If there is a connection between node i and node j, the $\boldsymbol{A}^d_{ij}$ is set to 1. For the node i at the l-th layer, the convolution computation can be defined as follows:

$$\boldsymbol{u}_i^{(l)} = \rho\Big(\sum_{j=1}^{n_d} \boldsymbol{A}^d_{ij} \boldsymbol{W}_u^{(l)} \boldsymbol{u}_j^{(l-1)} + \boldsymbol{b}_u^{(l)}\Big), \quad (2)$$

where $\boldsymbol{W}_u^{(l)}$ and $\boldsymbol{b}_u^{(l)}$ are the weight matrix and bias vector for the l-th layer, respectively. ρ is an activation function (e.g., ReLU). $\boldsymbol{u}_i^{(0)} \in \mathbb{R}^d$ is the initial representation of the i-th node obtained by the pre-trained model (i.e., BERT). To consider context information when encoding descriptive knowledge, we use the $\boldsymbol{h}_{e_1}$ and $\boldsymbol{h}_{e_2}$ obtained in Section 3.1 as the initial representations of events.

After the knowledge encoding, the representations of e_1 and e_2 in descriptive graphs are denoted as $\boldsymbol{u}_{e_1}$ and $\boldsymbol{u}_{e_2}$, respectively. We concatenate them as the descriptive knowledge representation:

$$\boldsymbol{F}_D^{(e_1,e_2)} = \boldsymbol{u}_{e_1} \oplus \boldsymbol{u}_{e_2}. \quad (3)$$

3.3 Relational Graph Induction

3.3.1 Multi-Hop Path Obtaining

Given e_1 and e_2, our model first retrieves the multi-hop path between the two events from CONCEPTNET. We refer to the multi-hop path as *Relational Path*. Since shorter connections between two concepts could mean stronger relevance (Lin et al., 2019), our model exploits the shortest path between the two events as the relational path. We represent the CONCEPTNET as a graph, and then use NetworkX toolkit[3] to get the shortest path between the two events. When there are multiple shortest paths, we randomly select one path for avoiding information redundancy.

3.3.2 Structure Induction

To capture potentially useful information and reduce the impact of irrelevant knowledge on the relational path, our model treats the reasoning structure as a latent variable and induces it with the

[3] https://networkx.org

input of the relational path, which can be shown in Figure 2. We call the induced reasoning structure as *Relational Graph* (denoted as G_r). The structure induction module is built based on the structured attention (Kim et al., 2017). We use a variant of Kirchhoff's Matrix-Tree Theorem (Koo et al., 2007; Nan et al., 2020) to learn the graph structure.

Formally, the nodes of relational graph are the concepts on the relational path. The initialized representation of each node is obtained via the pre-trained model (i.e., BERT). The representation of the i-th node is denoted as $\boldsymbol{m}_i \in \mathbb{R}^d$. We first calculate the pair-wise unnormalized attention score $\boldsymbol{s}_{ij}$ between the i-th node and the j-th node:

$$\boldsymbol{s}_{ij} = (\tanh(\boldsymbol{W}_p \boldsymbol{m}_i))^T \boldsymbol{W}_b (\tanh(\boldsymbol{W}_c \boldsymbol{m}_j)), \quad (4)$$

where $\boldsymbol{W}_p$ and $\boldsymbol{W}_c$ are weights matrixes. $\boldsymbol{W}_b$ are the weights for the bilinear transformation. Next, we compute the root score $\boldsymbol{s}_i^r$ which represents the unnormalized probability of the i-th node to be selected as the root node of the structure:

$$\boldsymbol{s}_i^r = \boldsymbol{W}_r \boldsymbol{m}_i, \quad (5)$$

where $\boldsymbol{W}_r \in \mathbb{R}^{1 \times d}$ is the weight for linear transformation. Suppose the graph G_r has n_r nodes, we first assign non-negative weights $\boldsymbol{P} \in \mathbb{R}^{n_r \times n_r}$ to the edges of the induced relational graph:

$$\boldsymbol{P}_{ij} = \begin{cases} 0, & \text{if } i = j \\ \exp(\boldsymbol{s}_{ij}), & \text{otherwise,} \end{cases} \quad (6)$$

where $\boldsymbol{P}_{ij}$ is the weight of the edge between the i-th and the j-th node. Then, following Koo et al. (2007), we define the Laplacian matrix $\boldsymbol{L} \in \mathbb{R}^{n_r \times n_r}$ of G_r, and its variant $\hat{\boldsymbol{L}} \in \mathbb{R}^{n_r \times n_r}$, respectively:

$$\boldsymbol{L}_{ij} = \begin{cases} \sum_{k=1}^{n_r} \boldsymbol{P}_{kj}, & \text{if } i = j \\ -\boldsymbol{P}_{ij}, & \text{otherwise,} \end{cases} \quad (7)$$

$$\hat{\boldsymbol{L}}_{ij} = \begin{cases} \exp(\boldsymbol{s}_i^r), & \text{if } i = 1 \\ \boldsymbol{L}_{ij}, & \text{otherwise.} \end{cases} \quad (8)$$

We use $\boldsymbol{A}_{ij}^r$ to denote the marginal probability of the edge between the i-th node and the j-th node, which can be computed as follows:

$$\begin{aligned} \boldsymbol{A}_{ij}^r = &(1 - \delta_{1,j}) \boldsymbol{P}_{ij} [\hat{\boldsymbol{L}}^{-1}]_{ij} \\ &- (1 - \delta_{i,1}) \boldsymbol{P}_{ij} [\hat{\boldsymbol{L}}^{-1}]_{ji}, \end{aligned} \quad (9)$$

where δ is the Kronecker delta (Koo et al., 2007) and $\cdot^{-1}$ denotes matrix inversion. $\boldsymbol{A}^r$ can be regarded as a weighted adjacency matrix of the graph G_r. Finally, $\boldsymbol{A}^r$ is fed into the iterative refinement for event causality reasoning.

3.3.3 Iterative Refinement

After obtaining the relational graph structure, we perform event causality reasoning on the induced structure. To better capture potential reasoning clues, we adopt the densely connected graph convolutional networks (DCGCNs) (Guo et al., 2019), which allows training a deeper reasoning model. The convolution computation of each layer is:

$$\boldsymbol{v}_i^{(l)} = \rho(\sum_{j=1}^{n_r} \boldsymbol{A}_{ij}^r \boldsymbol{W}_v^{(l)} \boldsymbol{g}_j^{(l)} + \boldsymbol{b}_v^{(l)}), \quad (10)$$

where $\boldsymbol{g}_j^{(l)}$ is the concatenation of the initial node representation and the node representations produced in layers $1, \ldots, l-1$, namely $\boldsymbol{g}_j^{(l)} = \boldsymbol{m}_j \oplus \boldsymbol{v}_j^{(1)} \oplus \cdots \oplus \boldsymbol{v}_j^{(l-1)}$.

The induced structure at once is relatively shallow (Liu et al., 2019; Nan et al., 2020) and may not be optimal for causality reasoning. Therefore, we iteratively refine the induced structure to learn a more informative structure. We stack N blocks (each block is structure induction and DCGCNs reasoning) of this module to induce the structure N times. Intuitively, as the structure gets more refined, the structure is more reasonable.

After the iterative refinement, the representations of e_1 and e_2 are denoted as $\boldsymbol{v}_{e_1}$ and $\boldsymbol{v}_{e_2}$, respectively. We concatenate them as the relational knowledge representation:

$$\boldsymbol{F}_R^{(e_1,e_2)} = \boldsymbol{v}_{e_1} \oplus \boldsymbol{v}_{e_2}. \quad (11)$$

3.4 Model Prediction and Training

We concatenate the context representation, descriptive knowledge representation and relational knowledge representation as the final representation:

$$\boldsymbol{F}_{e_1,e_2} = \boldsymbol{F}_C^{(e_1,e_2)} \oplus \boldsymbol{F}_D^{(e_1,e_2)} \oplus \boldsymbol{F}_R^{(e_1,e_2)}. \quad (12)$$

To make the final prediction, we perform a binary classification by taking $\boldsymbol{F}_{e_1,e_2}$ as input:

$$\boldsymbol{p}_{e_1,e_2} = \text{softmax}(\boldsymbol{W}_s \boldsymbol{F}_{e_1,e_2} + \boldsymbol{b}_s). \quad (13)$$

For training, we adopt cross entropy as the loss function:

$$J(\Theta) = -\sum_{s \in \mathcal{D}} \sum_{\substack{e_i, e_j \in E_s \\ e_i \neq e_j}} \boldsymbol{y}_{e_i,e_j} \log(\boldsymbol{p}_{e_i,e_j}), \quad (14)$$

where Θ denotes the model parameters. s denotes a sentence in the training set $\mathcal{D}$. E_s is the set of events in sentence s. $\boldsymbol{y}_{e_i,e_j}$ is a one-hot vector representing the gold label between e_i and e_j.

4 Experiments

4.1 Datasets and Evaluation Metrics

We evaluate our proposed method on two widely used datasets, including EventStoryLine (Caselli and Vossen, 2017) and Causal-TimeBank (Mirza et al., 2014). For EventStoryLine, the dataset contains 258 documents, 5,334 events in total, and 1,770 of 7,805 event pairs are causally related. For Causal-TimeBank, the dataset contains 184 documents, 6,813 events, and 318 of 7,608 event pairs are causally related. We conduct the 5-fold and 10-fold cross-validation on the EventStoryLine dataset and Causal-TimeBank dataset respectively, same as previous methods to ensure fairness. Following previous works (Choubey and Huang, 2017; Gao et al., 2019), we adopt Precision (P), Recall (R) and F1-score (F1) as evaluation metrics.

4.2 Parameter Settings

In our implementations, our method uses the HuggingFace's Transformers library[4] to implement the uncased BERT base model, which has 12-layers, 768-hidden, and 12-heads. The learning rate is initialized as 2e-5 with a linear decay. We use the Adam algorithm (Kingma and Ba, 2015) to optimize model parameters. The batch size is set to 20. The number of induction blocks (i.e., N) is set to 2. The dropout of GCN is set to 0.3. Due to the sparseness of positive examples, we adopt a negative sampling strategy for training. The negative sampling rate is 0.6 and 0.7 for the EventStoryLine and Causal-TimeBank, respectively. We utilize CONCEPTNET 5.0 as the external knowledge base.

4.3 Baselines

We compare the proposed approach LSIN with previous state-of-the-art methods:

Feature-based methods: (1) Mirza and Tonelli (2014), which proposes a data driven method with causal signals for the task; (2) Mirza (2014), which employs a verb rule based model with data filtering and causal signals enhancement; (3) Choubey and Huang (2017), which proposes a sequence model exploring complex handcrafted features for the task; (4) Gao et al. (2019), which utilizes a logistic regression classifier with the integer linear programming to model causal structure for the task.

Neural network-based methods: (1) Cheng and Miyao (2017), which proposes a dependency path based bidirectional long short-term memory network (BiLSTM) that models the context between two event mentions for causal relation identification; (2) KMMG (Liu et al., 2020), which proposes a mention masking generalization method and also utilizes the external knowledge; (3) KnowDis (Zuo et al., 2020), which proposes a knowledge enhanced distant data augmentation method to alleviate data lacking problem.

Methods	P(%)	R(%)	F1(%)
BERT	36.9	56.0	44.5
Cheng and Miyao (2017)	34.0	41.5	37.4
Choubey and Huang (2017)	32.7	44.9	37.8
Gao et al. (2019)	37.4	55.8	44.7
KnowDis (Zuo et al., 2020)	39.7	**66.5**	49.7
KMMG (Liu et al., 2020)	41.9	62.5	50.1
LSIN (Ours)	**47.9**	58.1	**52.5***

Table 1: Experimental results on the EventStoryLine dataset. Bold denotes best results. * denotes a significance test with p=0.05.

Methods	P(%)	R(%)	F1(%)
BERT	38.8	44.1	41.3
Mirza and Tonelli (2014)	67.3	22.6	33.9
Mirza (2014)	**69.0**	31.5	43.2
KMMG (Liu et al., 2020)	36.6	55.6	44.1
KnowDis (Zuo et al., 2020)	42.3	**60.5**	49.8
LSIN (Ours)	51.5	56.2	**52.9***

Table 2: Experimental results on the Causal-TimeBank dataset. Bold denotes best results. * denotes a significance test with p=0.05.

4.4 Overall Results

Since some baselines are evaluated either on the EventStoryLine dataset or the Causal-TimeBank dataset, the baselines used for the two datasets are different. Table 1 and Table 2 show the results on the EventStoryLine and Causal-TimeBank, respectively. From the tables, we can observe that:

(1) Our method outperforms all the baselines by a large margin on the two datasets. For example, compared with the state-of-the-art model KnowDis (Zuo et al., 2020), our method LSIN

[4] https://github.com/huggingface/transformers

Methods	P(%)	R(%)	F1(%)
BERT	36.9	56.0	44.5
BERT+DK	41.8	51.9	46.3
BERT+RK	46.1	55.4	50.3
BERT+DK+RK	**47.9**	**58.1**	**52.5**

Table 3: Experimental results by using different kinds of knowledge on the EventStoryLine dataset. "DK" and "RK" refer to "descriptive knowledge" and "relational knowledge", respectively.

Methods	P(%)	R(%)	F1(%)
Liu et al. (2020)	**44.5**	39.3	41.8
DGI-Retrieval	40.0	46.1	42.8
DGI-Generation	39.3	51.3	44.5
DGI-Hybrid	41.8	**51.9**	**46.3**

Table 4: Comparison between the different methods for using the descriptive knowledge on the EventStoryLine dataset. "DGI" refer to "descriptive graph induction".

achieves 2.8% and 3.1% improvements of F1-score on the EventStoryLine and Causal-TimeBank, respectively. It indicates that our proposed method is very effective for this task.

(2) Compared with the state-of-the-art model KMMG (Liu et al., 2020), our method achieves 6.0% improvements in terms of Precision score on the EventStoryLine. The reason may be that our method utilizes the relational knowledge between events for causality reasoning, which can improve the confidence of event causality prediction.

(3) Our method improves upon the BERT model by 8.0% and 11.6% in terms of F1-score on the two datasets, respectively. This suggests that only using the annotated training data is not enough to tackle the task. Moreover, it also indicates that our method is able to effectively leverage the external structural knowledge for ECI task.

(4) The BERT model achieves comparable performance with complex feature-based methods such as Gao et al. (2019) on the EventStoryLine dataset, which indicates that the BERT is able to extract useful text features for the task.

4.5 Effectiveness of External Structural Knowledge

We validate the effectiveness of external structural knowledge for this task. Based on the BERT model, we leverage the descriptive knowledge via descriptive graph induction module, and the relational knowledge via relational graph induction module. The results are shown in Table 3. We have two important observations:

(1) Based on the BERT model, incorporating these two kinds of knowledge can both improve performance. Moreover, simultaneously using these two kinds of knowledge can further improve the performance. It indicates that the external structural knowledge is very effective for this task.

(2) The performance improvement of using the relational knowledge is more obvious than that of using the descriptive knowledge, achieving 4.0% improvements in terms of F1-score. We guess that the relational knowledge can provide more clues for event causality reasoning.

4.6 Effectiveness of Descriptive Graph Induction

To verify the effectiveness of descriptive graph induction module, we compare our method with the state-of-the-art model (Liu et al., 2020). Liu et al. (2020) first retrieve the descriptive knowledge, and then transfer the knowledge into a sequence. Finally, they adopt the BERT to encode the knowledge. The results are listed in Table 4. In the table, "DGI-Retrieval", "DGI-Generation" and "DGI-Hybrid" denote obtaining the descriptive knowledge via retrieval, generation and hybrid method, respectively. Overall, we can observe that:

(1) The DGI-Hybrid model significantly outperforms Liu et al. (2020), achieving 4.5% improvements of F1-score. Moreover, even if we use the same retrieval method as Liu et al. (2020), our model still achieves better result. It indicates the descriptive graph induction module can better take advantage of the descriptive knowledge.

(2) Compared with Liu et al. (2020), the DGI-Hybrid model achieves great improvements in terms of Recall score (i.e., improving 12.6%). The reason is that our method can automatically generate the descriptive knowledge, when the knowledge cannot be retrieved from the KB.

4.7 Effectiveness of Relational Graph Induction

To validate the effectiveness of the relational graph induction module, we compare our method with other three baselines. The three baselines are illustrated as follows:

(1) **LSTM-based Reasoning**, which regards the relational path as a sequence and employs LSTM

Methods	P(%)	R(%)	F1(%)
LSTM-based	43.0	54.5	48.1
Fixed Graph-based	43.1	56.5	48.9
Attention-based	46.3	55.0	50.3
LSIN (Ours)	**47.9**	**58.1**	**52.5**

Table 5: Comparison between the different methods for leveraging the relational knowledge on the EventStoryLine dataset.

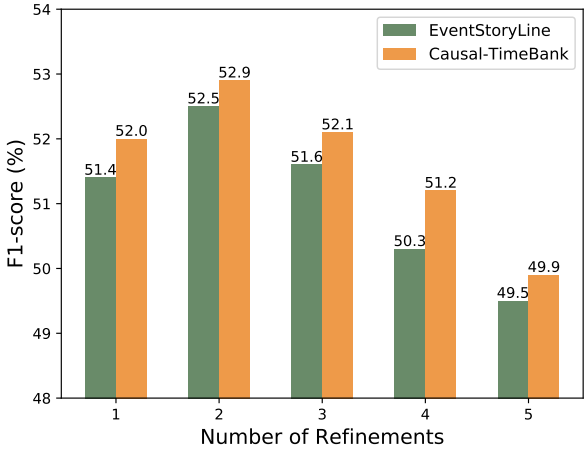

Figure 3: F1-score for different number of refinements (i.e., N) on the EventStoryLine dataset and Causal-TimeBank dataset, respectively. The number of refinements is ranging from 1 to 5.

to encode it; (2) **Fixed Graph-based Reasoning**, which regards the relational path as a graph. Its nodes are concepts on the path and edges only exist between adjacent concepts; (3) **Attention-based Reasoning**, which uses the self-attention to encode the relational path for modeling the dependencies between arbitrary two concepts.

The results are shown in Table 5. From the results, we can observe that:

(1) Our method LSIN outperforms the three methods by a large margin. For example, compared with LSTM-based reasoning method, our method achieves 4.4% improvements of F1-score. This empirically confirms using induced relational graph structure is more effective than directly using the relational path for causality reasoning.

(2) Compared with Fixed Graph-based reasoning method, our method achieves 3.6% improvements of F1-score. It indicates that our method is able to effectively capture the potentially useful information and reduce the impact of irrelevant knowledge on the relational path.

Examples	BERT	LSIN
a) Indonesia **earthquake**: over 200 **injured** in Aceh province ...	✗	✓
b) The **fights** erupted in Flatbush, and 46 were **arrested** at Wednesday ...	✗	✓

Table 6: Results of case study where bold denotes the two event pair. ✓ and ✗ denote a correct and incorrect prediction, respectively.

4.8 Impact of the Number of Refinements

We investigate the effect of the refinement on the overall performance. We plot the overall F1-score varying with the number of refinements in Figure 3. From the figure, we can observe that:

(1) Our method LSIN yields the best performance in the second refinement. Compared with the first induction, the second refinement achieves 1.1% improvements of F1-score on the EventStoryLine dataset. This indicates that the proposed LSIN is able to induce more reasonable reasoning structures by iterative refinement.

(2) When the number of refinements is too large, the performance on the two datasets stops increasing or even decreases due to over-fitting.

4.9 Case Study

We conduct case study to further verify the effectiveness of our method. Table 6 shows several cases showing the outputs of BERT and our method LSIN. From the results, we can observe that the BERT model cannot handle the cases where there is no causal clue. By contrast, our method can make correct predictions by leveraging the external structural knowledge. For the second example in Table 6, although the text has no clue indicating the existence of causality between "*fights*" and "*arrested*", there is the relational knowledge between the two events in the KB, namely "*fight*" $\xrightarrow{\text{HasSubevent}}$ "*hurt someone else*" $\xrightarrow{\text{HasSubevent}}$ "*get arrested*". Our method can make use of the relational knowledge to make a correct prediction. The two examples qualitatively demonstrate our method can effectively leverage the external knowledge for ECI task.

5 Conclusion

In this paper, we propose a novel latent structure induction network (LSIN) to leverage the external structural knowledge for ECI task. To make use of the descriptive knowledge, we devise a descrip-

tive graph induction module to obtain and encode the graph-structured descriptive knowledge. To utilize the relational knowledge, we propose a relational graph induction module to induce a more reasonable reasoning structure for causality reasoning. Experimental results on two widely used datasets indicate that our approach substantially outperforms previous state-of-the-art methods.

Acknowledgments

We thank anonymous reviewers for their insightful comments and suggestions. This work is supported by the National Key Research and Development Program of China (No. 2020AAA0106400), and the National Natural Science Foundation of China (No. 61806201). This work is also supported by Beijing Academy of Artificial Intelligence (BAAI2019QN0301) and the fund of the joint project with Beijing Baidu Netcom Science Technology Co., Ltd.

References

Brandon Beamer and Roxana Girju. 2009. Using a bigram event model to predict causal potential. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 430–441.

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D. Manning. 2014. Modeling biological processes for reading comprehension. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1510. Association for Computational Linguistics.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779. Association for Computational Linguistics.

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*, pages 77–86. Association for Computational Linguistics.

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional LSTM over dependency paths. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1–6. Association for Computational Linguistics.

Prafulla Kumar Choubey and Ruihong Huang. 2017. A sequential model for classifying temporal relations between intra-sentence events. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1796–1802. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics.

Quang Do, Yee Seng Chan, and Dan Roth. 2011. Minimally supervised event causality identification. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 294–303. Association for Computational Linguistics.

Lei Gao, Prafulla Kumar Choubey, and Ruihong Huang. 2019. Modeling document-level causal structures for event causal relation identification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1808–1817. Association for Computational Linguistics.

Zhijiang Guo, Yan Zhang, Zhiyang Teng, and Wei Lu. 2019. Densely connected graph convolutional networks for graph-to-sequence learning. *Transactions of the Association for Computational Linguistics*, 7:297–312.

Chikara Hashimoto. 2019. Weakly supervised multilingual causality extraction from Wikipedia. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2988–2999. Association for Computational Linguistics.

Chikara Hashimoto, Kentaro Torisawa, Julien Kloetzer, Motoki Sano, István Varga, Jong-Hoon Oh, and Yutaka Kidawara. 2014. Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 987–997. Association for Computational Linguistics.

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*, pages 52–58. Association for Computational Linguistics.

Zhichao Hu and Marilyn Walker. 2017. Inferring narrative causality between event pairs in films. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 342–351. Association for Computational Linguistics.

Kazuma Kadowaki, Ryu Iida, Kentaro Torisawa, Jong-Hoon Oh, and Julien Kloetzer. 2019. Event causality recognition exploiting multiple annotators' judgments and background knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 5816–5822. Association for Computational Linguistics.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. 2017. Structured attention networks. In *5th International Conference on Learning Representations, 2017, Conference Track Proceedings*. OpenReview.net.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, 2015, Conference Track Proceedings*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, 2017, Conference Track Proceedings*. OpenReview.net.

Terry Koo, Amir Globerson, Xavier Carreras, and Michael Collins. 2007. Structured prediction models via the matrix-tree theorem. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 141–150. Association for Computational Linguistics.

Canasai Kruengkrai, Kentaro Torisawa, Chikara Hashimoto, Julien Kloetzer, Jong-Hoon Oh, and Masahiro Tanaka. 2017. Improving event causality recognition with multiple background knowledge sources using multi-column convolutional neural networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3466–3473. AAAI Press.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. KagNet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2829–2839. Association for Computational Linguistics.

Jian Liu, Yubo Chen, and Jun Zhao. 2020. Knowledge enhanced event causality identification with mention masking generalizations. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, 2020*, pages 3608–3614. ijcai.org.

Yang Liu, Ivan Titov, and Mirella Lapata. 2019. Single document summarization as tree induction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1745–1755. Association for Computational Linguistics.

Paramita Mirza. 2014. Extracting temporal and causal relations between events. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 10–17. Association for Computational Linguistics.

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)*, pages 10–19. Association for Computational Linguistics.

Paramita Mirza and Sara Tonelli. 2014. An analysis of causality between events and its relation to temporal information. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2097–2106. Dublin City University and Association for Computational Linguistics.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1546–1557. Association for Computational Linguistics.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018. Joint reasoning for temporal and causal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2278–2288. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Kira Radinsky, Sagie Davidovich, and Shaul Markovitch. 2012. Learning causality for news events prediction. In *Proceedings of the 21st World Wide Web Conference, 2012*, pages 909–918. ACM.

Mehwish Riaz and Roxana Girju. 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*, pages 21–30. Association for Computational Linguistics.

Mehwish Riaz and Roxana Girju. 2014. In-depth exploitation of noun and verb semantics to identify causation in verb-noun pairs. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 161–170. Association for Computational Linguistics.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4444–4451. AAAI Press.

Aju Thalappillil Scaria, Jonathan Berant, Mengqiu Wang, Peter Clark, Justin Lewis, Brittany Harding, and Christopher D. Manning. 2013. Learning biological processes with global constraints. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1710–1720. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 5998–6008.

Peifeng Wang, Nanyun Peng, Filip Ilievski, Pedro Szekely, and Xiang Ren. 2020. Connecting the dots: A knowledgeable path generator for commonsense question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4129–4140. Association for Computational Linguistics.

Hsiu-Wei Yang, Yanyan Zou, Peng Shi, Wei Lu, Jimmy Lin, and Xu Sun. 2019. Aligning cross-lingual entities with multi-aspect information. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4431–4441. Association for Computational Linguistics.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215. Association for Computational Linguistics.

Xinyu Zuo, Yubo Chen, Kang Liu, and Jun Zhao. 2020. KnowDis: Knowledge enhanced data augmentation for event causality detection via distant supervision. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1544–1550. International Committee on Computational Linguistics.