7-2023

CHEER: Centrality-aware high-order event reasoning network for document-level event causality identification

Meiqi CHEN

Yixin CAO
Singapore Management University, yxcao@smu.edu.sg

Yan ZHANG

Zhiwei LIU

CHEER: Centrality-aware High-order Event Reasoning Network for Document-level Event Causality Identification

Meiqi Chen[1], **Yixin Cao**[2], **Yan Zhang**[1], **Zhiwei Liu**[3],
[1] Peking University [2] Singapore Management University [3] Meituan
meiqichen@stu.pku.edu.cn

Abstract

Document-level Event Causality Identification (DECI) aims to recognize causal relations between events within a document. Recent studies focus on building a document-level graph for cross-sentence reasoning, but ignore important causal structures — there are one or two "central" events that prevail throughout the document, with most other events serving as either their cause or consequence. In this paper, we manually annotate central events for a systematical investigation and propose a novel DECI model, CHEER, which performs high-order reasoning while considering event centrality. First, we summarize a general GNN-based DECI model and provide a unified view for better understanding. Second, we design an Event Interaction Graph (EIG) involving the interactions among events (e.g., coreference) and event pairs, e.g., causal transitivity, *cause(A, B)* $\wedge$ *cause(B, C)* $\Rightarrow$ *cause(A, C)*. Finally, we incorporate event centrality information into the EIG reasoning network via well-designed features and multi-task learning. We have conducted extensive experiments on two benchmark datasets. The results present great improvements (5.9% F1 gains on average) and demonstrate the effectiveness of each main component.

1 Introduction

Event Causality Identification (ECI) aims at identifying causal relations between events within texts. It is a fundamental NLP task and beneficial to various applications, such as question answering (Shi et al., 2021; Sui et al., 2022) and future event forecasting (Hashimoto, 2019; Bai et al., 2021). In terms of the text length, events may occur within the same sentence (SECI) or span across the entire document (DECI). DECI is more practical than SECI but suffers from the lack of clear causal indicators, e.g., causal words *because*.

Recent DECI works often build a document-level graph for cross-sentence reasoning, but ignore important causal structures. Tran Phu and Nguyen (2021) take events as nodes and extract linguistic/discourse relations as edges. Then, they apply Graph Neural Network (GNN) to enhance event/node embeddings with their neighbors for final causality prediction. To avoid noisy and exhaustive relation extraction, ERGO (Chen et al., 2022) instead takes each event pair as nodes and leverages GNN on the relational graph for high-order causal transitivity, e.g., *cause(A, B)* $\wedge$ *cause(B, C)* $\Rightarrow$ *cause(A, C)*. However, some useful prior event relations such as coreference are discarded. Moreover, we observe a causal information loss from document to graph. Not all events are equally important. There are one or two "central" events that prevail throughout the document, and other events are either to explain their cause or the consequence (Gao et al., 2019). As shown in Figure 1, event *FIRE* is the central event. It is mentioned several times (i.e., coreferences *blaze* and *fire*), causing almost all the other events (e.g., *collapsed* and *repairs*).

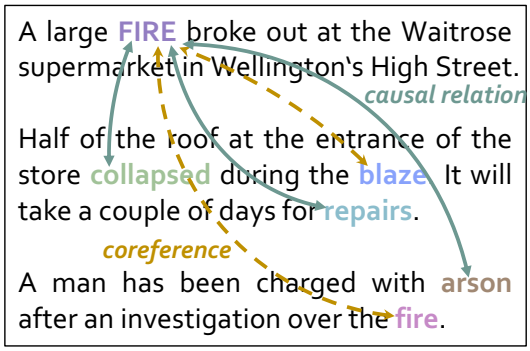

Figure 1: An example of DECI. Solid green lines denote target causal relations and dashed yellow lines denote coreference. *FIRE* is the central event in this document.

In this paper, we propose to consider the above causal structures while leveraging the reasoning power of GNN. To do so, we highlight the following questions:

- How to identify central events? Are they recognizable?

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 10804–10816

- How to effectively consider such causal structures for cross-sentence reasoning?

To address the issues, we manually annotate central events in the public dataset EventStoryLine (Caselli and Vossen, 2017) and propose a novel DECI model, **C**entrality-aware **H**igh-order **E**v**E**nt **R**easoning network (**CHEER**). We first summarize a general GNN-based DECI model for better understanding. Then, we design an Event Interaction Graph (EIG) that involves interactions between events and among event pairs (i.e., high-order relations). Finally, we incorporate event centrality information into the EIG reasoning network via well-designed features and multi-task learning.

In specific, for the first challenge, we preserve centrality information into event embeddings using two measures: (i) position centrality to maintaining the order of sentences where events are located, and (ii) degree centrality that counts the number of prior relations of each event. The motivation is that a central event usually summarizes the main content at the beginning and almost all the other events are relevant to it. Then, we use the centrality-aware event embeddings for central event prediction. Evaluated on our central event annotations, we found that this centrality modeling method is feasible and effective, with potential for further improvement.

For the second challenge, based on the general GNN-based DECI model, our proposed EIG unifies both event and event-pair graphs, so that we can reason over not only available causal structures but also high-order event relations. Particularly, there are three types of edges. First, two event pair nodes shall be connected if they share a common event, so that their relational information can be fused for transitivity. Second, we connect event nodes to their corresponding event pair nodes to enhance event embeddings with high-order reasoning. Moreover, the edge types will be further distinguished according to whether the event node is a central event or not. Third, EIG is also scalable to prior event relations (e.g., coreference) that connect event nodes if available.

Our contributions can be summarized as follows:

- We propose to consider causal structures (i.e., event centrality and coreference) and manually annotate central events for investigation.
- We design an EIG and propose a novel DECI framework CHEER for effective reasoning at the document level.
- Extensive experiments on two benchmark datasets validate the effectiveness of CHEER (5.9% F1 gains on average).

2 Related Work

2.1 Sentence-level ECI

Early feature-based methods explore different resources for causal expressions, such as lexical and syntactic patterns (Riaz and Girju, 2013, 2014b,a), causality cues or markers (Do et al., 2011; Hidey and McKeown, 2016), temporal patterns (Ning et al., 2018), statistical information (Hashimoto et al., 2014; Hu et al., 2017), and weakly supervised data (Hashimoto, 2019; Zuo et al., 2021b). Recently, some methods have leveraged Pre-trained Language Models (PLMs) for the ECI task and have achieved promising performance (Kadowaki et al., 2019; Liu et al., 2020; Zuo et al., 2020). To deal with implicit causal relations, Cao et al. (2021) incorporate external knowledge from ConceptNet (Speer et al., 2017), and Zuo et al. (2021a) learn context-specific causal patterns from external causal statements.

2.2 Document-level ECI

Following the success of sentence-level natural language understanding, many tasks are extended to the entire document, such as relation extraction (Yao et al., 2019), natural language inference (Yin et al., 2021), and event argument extraction (Ma et al., 2022). DECI poses new challenges to cross-sentence reasoning and the lack of clear causal indicators. Gao et al. (2019) propose a feature-based method that uses Integer Linear Programming (ILP) to model the global causal structures. DSGCN (Zhao et al., 2021) uses a graph inference mechanism to capture interaction among events. RichGCN (Tran Phu and Nguyen, 2021) constructs an even graph and uses GCN (Kipf and Welling, 2017) to capture relevant connections. However, noise may be introduced in the construction of edges and the interdependency among event pairs is neglected. ERGO (Chen et al., 2022) builds a relational graph and model interaction between event pairs. Although intuitive, some meaningful event relations such as coreference are ignored. Compared with them, CHEER could capture high-order interactions among event pairs automatically while being compatible with prior event relations. Moreover, we consider the centrality of events to conduct global reasoning.

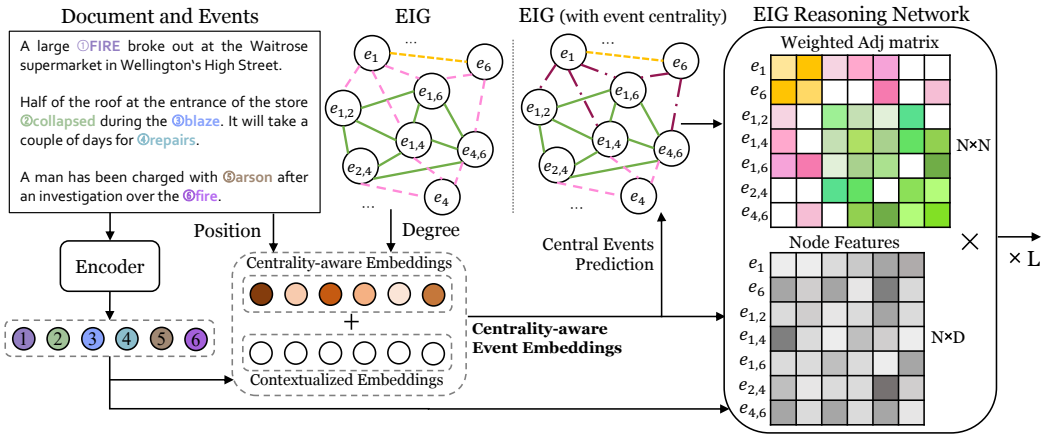

Figure 2: An overview of our proposed Centrality-aware High-order Event Reasoning Network (CHEER).

3 Methodology

Given document $\mathcal{D}$ and all its events, DECI is to predict whether there is a causal relation between any two event mentions e_i and e_j in $\mathcal{D}$. As shown in Figure 2, our proposed CHEER includes four main components: (1) **Document Encoder** to encode the document and output contextualized representations of events; (2) **Event Interaction Graph** that builds a graph including event nodes and event pair nodes for document-level reasoning. (3) **Event Centrality Incorporation** that incorporates event centrality information through two aspects. (4) **EIG Reasoning Network** that improves the quality of event and event pair representations by conducting inference over EIG, and then combines two types of node embeddings for final classification.

3.1 Document Encoder

Given document $\mathcal{D} = [x_t]_{t=1}^{L_\mathcal{D}}$ where $\mathcal{D}$ can be of any length $L_\mathcal{D}$, the document encoder aims to output the contextualized document and event representations. Almost arbitrary PLMs can serve as the encoder. In this paper, we leverage pre-trained BERT (Devlin et al., 2019) as a base encoder to obtain the contextualized embeddings. Following conventions (Chen et al., 2022), we add special tokens at the start and end of $\mathcal{D}$ (i.e., "`[CLS]`" and "`[SEP]`"), and insert additional special tokens "`<t>`" and "`</t>`"" at the start and end of all the events to mark the event positions. Then, we have:

$$H = [h_1, h_2, ..., h_{L_\mathcal{D}}] = \text{Encoder}([x_1, x_2, ..., x_{L_\mathcal{D}}]), \quad (1)$$

where $h_i \in \mathbb{R}^d$ is the output embedding of token x_i. Then, we use the embedding of the token "`[CLS]`" for document representation and the embedding of the token "`<t>`" for event representation.

Considering BERT's original limits that it cannot handle documents longer than 512, we leverage a dynamic window mechanism to deal with it. Specifically, we divide $\mathcal{D}$ into several overlapping spans according to a specific step size and input them into BERT separately. For the same event occurring in different spans, we calculate the average of all the embeddings of the corresponding token "`<t>`" to obtain the final event representation h_{e_i} for event i.

3.2 Event Interaction Graph

Our EIG could not only performs high-order inference among event pairs but also be compatible with prior event relations. Specifically, given all the events of document $\mathcal{D}$, we formulate EIG as: $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges. There are two types of nodes in $\mathcal{V}$: the nodes for a single event $\mathcal{V}_1$ and the nodes to represent a pair of events $\mathcal{V}_2$. Each node in $\mathcal{V}_2$ is constructed by combining any two events of $\mathcal{D}$.

For global inference, we introduce three main types of edges in $\mathcal{E}$: (1) (Event pair) - (event pair) edges $\mathcal{E}_1$ for two event pairs that share at least one event, e.g., the green line of (*FIRE*, *collapsed*)-(*collapsed*, *repairs*) in Figure 2, which is motivated by the causal transitivity described in Introduction; and (2) Event - (event pair) edges $\mathcal{E}_2$ for an event pair and its corresponding two events, e.g., the pink line of *FIRE*-(*FIRE*, *collapsed*) in Figure 2. (3) Event - event edges $\mathcal{E}_3$ for prior event relations obtained by external knowledge or tools (this type of edge is optional). Take coreference edges as an example (the yellow line of *FIRE-fire* in Figure 2), they are helpful for causal reasoning, since there is no causal relation between coreference events themselves. Moreover, coreference events shall have the

same causal relations between other events, which is so-called *coreference consistency*. Therefore, both coreference consistency and causal transitivity can be regarded as a kind of high-order reasoning.

3.3 Event Centrality Incorporation

Considering the centrality of events is based on the motivation that the central event should play a more important role in global inference. In this section, we introduce two aspects for incorporating event centrality information into our model. First, we propose centrality-aware event embeddings, which could be used to predict whether an event is a central event. Obtained the contextualized event embeddings h_{e_i} output by the document encoder, we perform the following two different centrality encoding modules:

Position Centrality Encoding which assigns each event an embedding vector $c_{\text{pos}} \in \mathbb{R}^d$ according to which sentence the event locates in the document. We initialize the vector randomly for each position. The motivation is central events often appear in the front of the document to summarize the core gist. For example, in Figure 2, the first sentence of the document outlines the main context of story and contains the central event *FIRE*.

Degree Centrality Encoding which assigns each event an embedding vector $c_{\text{deg}} \in \mathbb{R}^d$ according to the degree of its corresponding event node in EIG. We initialize the vector randomly for each degree. Intuitively, central events are throughout the document with many repeated mentions. Thus, central events will have a greater degree. For example in Figure 2, the degree of central event *FIRE* is greater than that of event *collapsed*, due to it has two coreference events *blaze* and *fire*.

As the centrality encoding is applied to each event, we directly add it to the event contextualized embeddings. Formally, for an event e_i and its corresponding embedding h_{e_i}, the final centrality-aware event embeddings is obtained by:

$$c_{e_i} = h_{e_i} + c_{\text{pos}(e_i)} + c_{\text{deg}(e_i)}, \tag{2}$$

where $c_{\text{pos}}, c_{\text{deg}}$ are obtained by the position and degree centrality encoding of e_i, respectively.

Central Events Prediction and EIG Enhancement Once obtained the centrality-aware event embeddings, we use them to predict whether an event is a central event: $p_{e_i} = f\left(c_{e_i}\mathbf{W}_c\right)$, where f denotes the sigmoid function, $\mathbf{W}_c \in \mathbb{R}^{d\times 1}$ is the parameter weight matrix. if p_{e_i} is greater than 0.5, we will regard e_i as a central event. Then, we increase the type of edges in $\mathcal{E}$: we further divide the event - (event pair) edges into *central* event - (event pair) edges $\mathcal{E}_{21}$ and *normal* event - (event pair) edges $\mathcal{E}_{22}$, and so does the event-event edges. In this way, the interaction of central events on EIG could have more of a special influence.

Central Events Annotation We manually annotate central events on the public dataset EventStoryLine to investigate the effect of centrality. In specific, we annotate central events considering the following rules: (1) the central events should be the focus of the story; (2) almost all other events described in the document should be related to it; (3) the coreference of central events will be regarded as central events, too; (4) on the premise of expressing the main content of the document correctly and completely, the number of central events should be as small as possible. According to the rules, we have three annotators to complete the task. Each document was annotated by two junior annotators independently. If the answers of the two annotators were inconsistent, a senior annotator checked the answers and made the final decision. The average inter-annotator agreement is 86.4% (Cohen's kappa). For 258 documents of EventstoryLine, we get 352 central events, of which 166 documents have one central event, 90 documents have two central events, and only 2 documents have three central events (these documents have more than 30 sentences and introduce several independent events). Then, we use the labels to train the model to predict central events:

$$\mathcal{L}_1 = -\sum_{e_i\in\mathcal{D}} \log(p_{e_i}). \tag{3}$$

More analysis can be seen in Section 4.5.

3.4 EIG Reasoning Network

In this section, we first describe a general GNN-based DECI model, then instantiate our implementation by considering causal structures. Finally, we provide a unified view for better understanding and discussing existing models.

A General GNN-based DECI Model To predict whether there is a causal relation between events e_i and e_j, we concatenate "`[CLS]`" embeddings of the document, the event features z_i, z_j, event pair features z_k, and define the probability of being

causal relation as follows:

$$p_{e_{i,j}} = f\left([h_{\texttt{[CLS]}}||z_i||z_j||z_k]\,\mathbf{W}_p\right), \quad (4)$$

where f denotes the softmax function, $\|$ denotes concatenation, $\mathbf{W}_p$ is the parameter weight matrix. Event-related features are typically initialized with contextualized embeddings via PLM in Section 3.1 and enhanced through L-layer GNN reasoning. The l-th layer takes a set of node embeddings $\mathbf{Z}^{(l)} \in \mathbb{R}^{N\times d_{\text{in}}}$ as input, and outputs a new set of node embeddings $\mathbf{Z}^{(l+1)} \in \mathbb{R}^{N\times d_{\text{out}}}$, where $N = |\mathcal{V}_1| + |\mathcal{V}_2|$ is the number of nodes, d_{in} and d_{out} are the dimensions of input and output embeddings, respectively. Formally, the output of the l-th layer for node v_i can be written as:

$$z_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}_i} g\left(z_i^{(l)}, z_j^{(l)}\right)\right), \quad (5)$$

where σ denotes non-linearity, $\mathcal{N}_i$ denotes the set that contains all the first-order neighbors of v_i, g denotes how to aggregate neighborhood information. By stacking multiple layers L, multi-hop reasoning could be reached.

EIG Reasoning Network Instantiation

Event & Event-pair Features For an event node e_i, we directly take the centrality-aware event embeddings for its initialization:

$$z_i^{(0)} = c_{e_i}\mathbf{W}_t, v_i \in \mathcal{V}_1, \quad (6)$$

where 0 denotes the initial state for the following neural layers, $\mathbf{W}_t \in \mathbb{R}^{d\times 2d}$ is a parameter weight matrix to make event nodes be the same size as the following event pair nodes for efficient computing.

As for an event pair node $(e_i, e_j) \rightarrow v_k$, we concatenate their corresponding two contextualized event embeddings as the event pair node features:

$$z_k^{(0)} = [h_{e_i}\|h_{e_j}], v_k \in \mathcal{V}_2, \quad (7)$$

EIG Reasoning It is intuitive that different types of edges represent various semantics contributing differently to the causality prediction. To handle this heterogeneity issue, EIG Reasoning Network incorporates the edge features with a self-attention mechanism during aggregation. Specifically, let T denote the number of edge types in EIG. We incorporate the edge features and learn a scalar γ_t $(1 \le t \le T)$ for each different type of edge to measure their importance:

$$\gamma_t = r_t\mathbf{W}_r, \quad (8)$$

where $r_t \in \mathbb{R}^{1\times d}$ is the edge feature specified by the edge type t, $\mathbf{W}_r \in \mathbb{R}^{d\times 1}$ is parameter vector according to t. In this way, we could adaptively adjust the interaction strength between two adjacent nodes by weighing different types of connections with γ_t. γ_t will be automatically learned.

Figure 2 illustrates an example of the entire process of CHEER (here we take a sub-graph of EIG for brevity). Different colors of edges indicate different connection types in EIG. Edges with the same color (i.e., the same edge type) will use the same γ_t. Each layer has its own set of $\gamma_t^{(l)}$. Then we could instantiate the aggregation function g as:

$$g\left(z_i^{(l)}, z_j^{(l)}\right) = f(\gamma_t^{(l)} + \alpha_{ij}^{(l)})(z_j^l\mathbf{W}_v^{(l)}), \quad (9)$$

where f denotes the softmax function, $\mathbf{W}_v^{(l)} \in \mathbb{R}^{d_{\text{in}}\times d_{\text{out}}}$ is the parametwer weight matrix. α_{ij} is computed by a shared self-attention mechanism (Vaswani et al., 2017) to measure the importance of neighbor j to i, where $\mathbf{W}_q, \mathbf{W}_k \in \mathbb{R}^{d_{\text{in}}\times d_{\text{out}}}$ are parameter weight matrices:

$$\alpha_{ij} = \frac{(z_i\mathbf{W}_q)(z_j\mathbf{W}_k)^T}{\sqrt{d_{\text{out}}}}. \quad (10)$$

As shown in Figure 2, the above process can be organized as a matrix multiplication to compute representations for all the nodes simultaneously through a weighted adjacency matrix. Denote $\mathbf{A}_{ij}$ as the (i,j)-element of the binary adjacency matrix $\mathbf{A}$, $\mathbf{A}_{ij}$ is 1 if there is an edge between nodes v_i and v_j or 0 otherwise. We could compute each entry of the edge-aware adjacency matrix as follows, where $\delta_{ij} = f(\gamma_t^{(l)} + \alpha_{ij}^{(l)})$ is the normalized weight:

$$\mathbf{A}_{ij}^{'(l)} = \delta_{ij}\mathbf{A}_{ij}, \quad (11)$$

Figure 2 shows that the corresponding neighbor node features are aggregated with different weights according to δ_{ij} to obtain the representation of the target node. Finally, the node representations of layer l can be obtained by:

$$\mathbf{Z}^{(l+1)} = \sigma\left(\mathbf{A}^{'(l)}\mathbf{Z}^{(l)}\mathbf{W}_v^{(l)}\right). \quad (12)$$

3.5 Training

Following ERGO (Chen et al., 2022), we adopt the focal loss (Lin et al., 2017) to alleviate the false-negative issue (i.e., the number of negative samples during training far exceeds that of positives). We adopt the β-balanced variant of focal loss, which

introduces a weighting factor β in $[0, 1]$ for the class "positive" and $1 - \beta$ for the class "negative". The loss function $\mathcal{L}_2$ can be written as:

$$\mathcal{L}_2 = -\sum_{e_i, e_j \in \mathcal{D}} \beta_{e_{i,j}} (1 - p_{e_{i,j}})^\tau \log(p_{e_{i,j}}), \quad (13)$$

where τ is the focusing hyper-parameter, β is a weighting hyper-parameter and its value is related to the ratio of positive and negative samples.

Besides, we find that predicting causal and coreference relations jointly brings benefits. A support point for this is that these two types of relations are mutually exclusive. Thus, we leverage the coreference information and perform a ternary classification training, i.e., to predict the label of each sample as a causal relation class, a coreference relation class, or no relation class (negative samples).

The final loss function combines event centrality and causality learning, where λ is a hyper-parameter:

$$\mathcal{L} = \lambda \mathcal{L}_1 + \mathcal{L}_2, \quad (14)$$

3.6 A Unified View of GNN-based DECI Methods

CHEER is a general framework that first constructs a document-level graph, then incorporates event centrality, and finally conducts reasoning on the graph. In this section, we discuss the difference between CHEER and previous GNN-based DECI methods. Note that only CHEER considers joint training, and we do not discuss loss function here.

(1) RichGCN (Tran Phu and Nguyen, 2021) has only event nodes and uses vanilla GCN's aggregation function: $g\left(z_i^{(l)}, z_j^{(l)}\right) = z_j^l \mathbf{W}_v^{(l)}$. By removing: i) event centrality incorporation, ii) event pair nodes and their relevant edges, iii) edge features and self-attention mechanism, CHEER could degenerate into RichGCN's framework.

(2) DSGCN (Zhao et al., 2021) has only event nodes and uses a combination of GCNs: $g\left(z_i^{(l)}, z_j^{(l)}\right) = \sum_{k=1}^{K} \alpha_k z_j^l \mathbf{W}_v^{(l,k)}$, where α_k denotes a feature filter. By removing: i) event centrality incorporation, ii) event pair nodes and their relevant edges, iii) edge features and modifying g accordingly, CHEER is scalable to DSGCN.

(3) ERGO (Chen et al., 2022) has only event-pair nodes and performs self-attention aggregation: $g\left(z_i^{(l)}, z_j^{(l)}\right) = f(\alpha_{ij}^{(l)})(z_j^l \mathbf{W}_v^{(l)})$. By removing i) event centrality incorporation, ii) event nodes and their relevant edges, and iii) edge features, CHEER could degenerate into ERGO's framework.

Therefore, by modifying the event centrality incorporation, the construction of EIG, and the aggregation function, CHEER can degenerate into different GNN-based DECI methods, and thus provide a unified view for better document-level reasoning.

4 Experiments

4.1 Experimental Setup

Datasets Details We evaluate CHEER on two widely used datasets. **EventStoryLine** (version 0.9) (Caselli and Vossen, 2017) contains 22 topics, 258 documents, and 5,334 events. Among them, 1,770 intra-sentence and 3,885 inter-sentence event pairs are annotated with causal relations. Following Gao et al. (2019), we group documents according to their topics. Documents in the last two topics are used as the development data, and documents in the remaining 20 topics are employed for 5-fold cross-validation. **Causal-TimeBank** (Mirza, 2014) contains 184 documents and 6,813 events. Among them, 318 event pairs are annotated with causal relations. Following Tran Phu and Nguyen (2021), we employ 10-fold cross-validation and only evaluate ECI performance for intra-sentence event pairs because the number of inter-sentence event pairs in Causal-TimeBank is quite small (i.e., only 18 pairs). EventStoryLine provides ground-truth event coreference chains, but Causal-TimeBank does not. To solve this, we have preprocessing steps on Causal-TimeBank. We first perform pre-training on EventStoryLine, and then use the pre-trained model to extract coreference data for Causal-TimeBank. We also use the Stanford CoreNLP toolkit (Manning et al., 2014) for a supplement. After the preprocessing steps, we add event-event coreference edges $\mathcal{E}_3$ to EventStoryLine and Causal-TimeBank. We perform a joint training in Section 3.5 on EventStoryLine. In evaluation, we only report and compare the prediction results of causal relations with baselines.

Implementation Details We set the dynamic window size in Section 3.1 to 256, and divide documents into several overlapping windows with a step size of 32. We implement our method based on the Pytorch version of Huggingface Transformer (Wolf et al., 2020). We use uncased BERT-base (Devlin et al., 2019) as the document encoder. We optimize our model with AdamW (Loshchilov and Hutter, 2019) using a learning rate of 2e-5 with a linear warm-up for the first 8% steps. We apply layer nor-

malization (Ba et al., 2016) and dropout (Srivastava et al., 2014) between the EIG reasoning network layers. We clip the gradients of model parameters to a max norm of 1.0. We perform early stopping and tune the hyper-parameters by grid search based on the development set performance: dropout rate $\in \{0.1, \mathbf{0.2}, 0.3\}$, focusing parameter $\tau \in \{0, 1, \mathbf{2}, 3\}$, weighting factor $\beta \in \{0.25, 0.5, \mathbf{0.75}\}$, loss weight $\lambda \in \{\mathbf{0.1}, 0.2\}$. Our model is trained on an NVIDIA RTX 2080 GPU with 24GB memory.

Evaluation Metrics We adopt Precision (P), Recall (R), and F1-score (F1) as evaluation metrics, same as previous methods (Tran Phu and Nguyen, 2021) to ensure comparability.

4.2 Baselines

We compare our proposed CHEER with various state-of-the-art SECI and DECI methods.

SECI Baselines (1) **KMMG** (Liu et al., 2020), a mention masking generalization method using extenal knowledge. (2) **KnowDis** (Zuo et al., 2020), a knowledge-enhanced distant data augmentation method to alleviate the data lacking problem. (3) **CauSeRL** (Zuo et al., 2021a), which learns context-specific causal patterns from external causal statements. (4) **LearnDA** (Zuo et al., 2021b), which uses knowledge bases to augment training data. (5) **LSIN** (Cao et al., 2021), which constructs a descriptive graph to leverage external knowledge.

DECI Baselines (1) **OP** (Caselli and Vossen, 2017), a dummy model that assigns causal relations to event pairs. (2) **LR+** and **LIP** (Gao et al., 2019), feature-based methods that construct document-level structures and use various types of resources. (3) **BERT (our implementation)** a baseline method that leverages dynamic window and event marker techniques. (4) **RichGCN** (Tran Phu and Nguyen, 2021), which constructs a document-level interaction graph and uses GCN to capture relevant connections. (5) **ERGO** (Chen et al., 2022), which builds a relational graph and model interaction between event pairs. We compare with its BERT-base implementation for fairness. Due to DSGCN (Zhao et al., 2021) does not provide results on benchmark datasets and does not release codes, we do not compare with it here.

4.3 Overall Results

Since some baselines can not handle the inter-sentence scenarios in EventStoryLine, and the number of inter-sentence event pairs in Causal-TimeBank is quite small (i.e., only 18 pairs). Thus we report the results of intra- and inter-sentence settings separately.

Model	EventStoryLine			Causal-TimeBank		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
OP	22.5	**98.6**	36.6	-	-	-
LR+	37.0	45.2	40.7	-	-	-
LIP	38.8	52.4	44.6	-	-	-
KMMG[○]	41.9	62.5	50.1	36.6	55.6	44.1
KnowDis[○]	39.7	66.5	49.7	42.3	60.5	49.8
LSIN[○]	47.9	58.1	52.5	51.5	56.2	53.7
LearnDA[○]	42.2	69.8	52.6	41.9	68.0	51.9
CauSeRL[○]	41.9	69.0	52.1	43.6	68.1	53.2
BERT[○]	47.8	57.2	52.1	47.6	55.1	51.1
RichGCN[○]	49.2	63.0	55.2	39.7	56.5	46.7
ERGO[○]	49.7	72.6	59.0	**58.4**	60.5	59.4
CHEER[○]	**56.9**	69.6	**62.6**	56.4	**69.5**	**62.3**

Table 1: Models' intra-sentence performance on EventStoryLine and Causal-TimeBank, the best results are in **bold** and the second-best results are underlined. [○] denotes models that use pre-trained BERT-base encoders. Overall, CHEER outperforms previous SOTA methods with a significant test at the level of 0.05.

Model	Inter-sentence			Intra + Inter		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
OP	8.4	**99.5**	15.6	10.5	**99.2**	19.0
LR+	25.2	48.1	33.1	27.9	47.2	35.1
LIP	35.1	48.2	40.6	36.2	49.5	41.9
BERT[○]	36.8	29.2	32.6	41.3	38.3	39.7
RichGCN[○]	39.2	45.7	42.2	42.6	51.3	46.6
ERGO [○]	43.2	48.8	45.8	46.3	50.1	48.1
CHEER[○]	**45.2**	52.1	**48.4**	**49.7**	53.3	**51.4**

Table 2: Model's inter and (intra+inter)-sentence performance on EventStoryLine.

Intra-sentence Evaluation From Table 1, we can observe that: (1) CHEER outperforms all the baselines by a large margin on both datasets, which demonstrates its effectiveness. (2) Compared with feature-based methods OP, LR+, and LIP, models using PLMs far boost the performance, which verifies that BERT could extract useful text features for the ECI task. We notice that OP achieves the highest Recall on EventStoryLine, which may be due to simply assigning causal relations by mimicking the textual order. This leads to many false positives and thus a low Precision.

Inter-sentence Evaluation From Table 2, we can observe that: (1) CHEER greatly outperforms all

Model	Intra	Inter	Intra + Inter
CHEER	62.6	48.4	51.4
w/o event centrality	60.3	46.3	49.3
w/o edge features	61.4	47.6	50.4
w/o coref	60.8	46.9	50.1

Table 3: F1 results of ablation study on EventStoryLine.

the baselines under both inter- and (intra+inter)-sentence settings. This demonstrates that CHEER can make better document-level inferences via our effective modeling over EIG. (2) the overall F1-score of the inter-sentence setting is much lower than that of the intra-sentence, which shows the challenge of DECI where events scatter in the document without clear causal indicators. Specifically, the BERT baseline could achieve competitive performance under the intra-sentence setting. However, it performs much worse than LIP, RichGCN, ERGO, and CHEER under inter-sentence settings, which indicates that a document-level structure or graph helps capture the global interactions for causal relation prediction.

4.4 Ablation Study

To analyze the effect of each main component proposed in CHEER, we consider evaluating the following ablated models on the EventStoryLine dataset. As shown in Table 3: (1) **Effect of Event Centrality** (w/o event centrality), which removes event centrality incorporation introduced in Section 3.3. Removing event centrality leads to information loss from the document to the graph. The performance degradation proves our contribution to preserving the event centrality information. (2) **Effect of Edge Features** (w/o edge features), which does not incorporate the edge features in Section 3.4 and thus the learnable scalar γ_t is removed in aggregation function. We can see that removing the edge-aware scalar clearly decreases the performance, which validates the necessity of capturing the semantic information of different edge features in EIG. (3) **Effect of Coreference** (w/o coref), which removes the $\mathcal{E}_3$ edges in EIG and does not use the ground-truth coreference chains as auxiliary training labels. The results indicate that the prior coreference information is helpful for the DECI task and supports us to unify event and event-pair graphs.

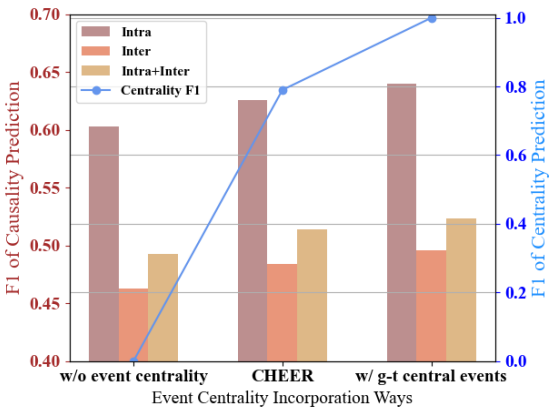

Figure 3: DECI performance of using different ways of event causality incorporation and the according F1 results of central event prediction.

4.5 Event Centrality Investigation

We further analyze the role of central events in the DECI task and the effect of our incorporation ways.

4.5.1 Role of Central Events

In Figure 3, the histograms represent the F1 results of CHEER under intra/inter/intra+inter settings on EventStoryLine. Three different groups represent three different ways of event causality incorporation, and the lines represent F1 results of central events prediction under three ways: (1) w/o event centrality, which removes the event centrality incorporation introduced in Section 3.3; (2) CHEER, the original incorporation way; (3) w/ g-t central events, which preserves centrality-aware event embeddings as event node features initialization but uses ground-truth central event labels to distinguish edge types. It can be seen that the F1 result of our central event classification reaches nearly 80%, which is feasible and still has space for improvement. We also observe that compared with using ground-truth labels, the inaccuracy of event centrality prediction limits the performance of DECI. Nevertheless, the performance of event centrality prediction could be higher by using more advanced encoding methods.

4.5.2 Case Study

In this section, we conduct a case study to further illustrate an intuitive impression of CHEER and choose the SOTA baseline ERGO for comparison. In Figure 3, we show a piece of text with five events, where *quake* is the central event (with a coreference *earthquake*) We notice that: (1) ERGO cannot

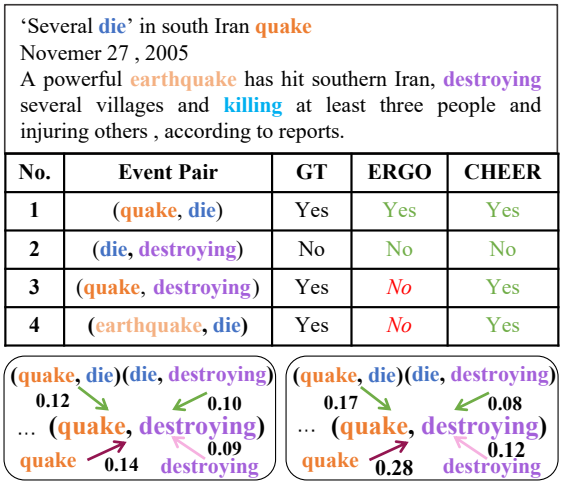

Figure 4: A case study of CHEER.

achieve the coreference consistency (No.1 and 4 event pairs), but CHEER could solve this explicitly by introducing prior relations and joint training. (2) ERGO could suffer from the false negative issue (No.3 event pair). For example when (*quake*, *destroying*) receives positive prediction from (*quake*, *die*) but negative prediction from (*die*, *destroying*), it tends to think the transitivity does not hold and outputs a wrong prediction. In contrast, CHEER blocks the propagation over these misleading paths by making central events take effect. 3) In the bottom graph, we visualize the normalized weights δ of Equation (11) with (left part) and without event centrality information (right part). For clarity, we only show some main nodes and edges here. We could see that when there is no event centrality incorporation, the δ values of neighboring nodes to (*quake*, *destroying*) are relatively even, which makes its prediction disturbed by negative paths, i.e., information from (*die*, *destroying*) node. When the event centrality is incorporated, (*quake*, *destroying*) pays more attention to the paths where central events are involved, i.e., *quake* node and (*quake*, *die*) node. Therefore, CHEER can learn more from such informative neighbors for the DECI task.

5 Conclusion

In this paper, we propose a novel centrality-aware high-order event reasoning network (CHEER) to conduct global reasoning for DECI. We first summarize a general GNN-based DECI model and provide a unified view for better understanding. Then we design an Event Interaction Graph (EIG) that involves prior event relations and high-order interactions among event pairs. Finally, we incorporate event centrality via well-designed features and multi-task learning. Extensive experiments show a great improvement of CHEER for both intra- and inter-sentence ECI on two benchmark datasets. Further analysis demonstrates the effectiveness of each main component.

Limitations

Although our modeling of event centrality is feasible and effective, there is still space for improvement. The performance of event centrality prediction could be higher by using more advanced encoding methods.

Besides, it is meaningful to further explore the interactions among various types of event relations. Existing datasets only cover limited relation types at once, and many works focus on the identification of causal relations alone. In this paper, although we further consider the effect of coreference relations and perform joint classification, there are still some other relations that can be explored, such as temporal relations, subevent relations, etc.

Acknowledgments

This work was supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant, as well as cash and in-kind contribution from the industry partner(s).

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *ArXiv preprint*, abs/1607.06450.

Long Bai, Saiping Guan, Jiafeng Guo, Zixuan Li, Xiaolong Jin, and Xueqi Cheng. 2021. Integrating deep event-level and script-level information for script event prediction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9869–9878, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pengfei Cao, Xinyu Zuo, Yubo Chen, Kang Liu, Jun Zhao, Yuguang Chen, and Weihua Peng. 2021. Knowledge-enriched event causality identification via latent structure induction networks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4862–4872, Online. 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, Vancouver, Canada. Association for Computational Linguistics.

Meiqi Chen, Yixin Cao, Kunquan Deng, Mukai Li, Kun Wang, Jing Shao, and Yan Zhang. 2022. Ergo: Event relational graph transformer for document-level event causality identification. *arXiv preprint arXiv:2204.07434*.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. 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, Edinburgh, Scotland, UK. 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, Volume 1 (Long and Short Papers)*, pages 1808–1817, Minneapolis, Minnesota. Association for Computational Linguistics.

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 (EMNLP-IJCNLP)*, pages 2988–2999, Hong Kong, China. 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 (Volume 1: Long Papers)*, pages 987–997, Baltimore, Maryland. Association for Computational Linguistics.

Christopher Hidey and Kathy McKeown. 2016. Identifying causal relations using parallel Wikipedia articles. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1424–1433, Berlin, Germany. 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, Vancouver, Canada. 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 (EMNLP-IJCNLP)*, pages 5816–5822, Hong Kong, China. Association for Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society.

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, IJCAI 2020*, pages 3608–3614. ijcai.org.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Paramita Mirza. 2014. Extracting temporal and causal relations between events. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 10–17, Baltimore, Maryland, USA. 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 (Volume 1: Long Papers), pages 2278–2288, Melbourne, Australia. Association for Computational Linguistics.

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 2013 Conference*, pages 21–30, Metz, France. Association for Computational Linguistics.

Mehwish Riaz and Roxana Girju. 2014a. 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 (SIGDIAL)*, pages 161–170, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Mehwish Riaz and Roxana Girju. 2014b. Recognizing causality in verb-noun pairs via noun and verb semantics. In *Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL)*, pages 48–57, Gothenburg, Sweden. Association for Computational Linguistics.

Jiaxin Shi, Shulin Cao, Lei Hou, Juanzi Li, and Hanwang Zhang. 2021. Transfernet: An effective and transparent framework for multi-hop question answering over relation graph. *ArXiv preprint*, abs/2104.07302.

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, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Yuan Sui, Shanshan Feng, Huaxiang Zhang, Jian Cao, Liang Hu, and Nengjun Zhu. 2022. Causality-aware enhanced model for multi-hop question answering over knowledge graphs. *Knowledge-Based Systems*, 250:108943.

Minh Tran Phu and Thien Huu Nguyen. 2021. 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*, pages 3480–3490, Online. 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, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational Linguistics.

Wenpeng Yin, Dragomir Radev, and Caiming Xiong. 2021. DocNLI: A large-scale dataset for document-level natural language inference. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4913–4922, Online. Association for Computational Linguistics.

Kun Zhao, Donghong Ji, Fazhi He, Yijiang Liu, and Yafeng Ren. 2021. Document-level event causality identification via graph inference mechanism. *Information Sciences*, 561:115–129.

Xinyu Zuo, Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Weihua Peng, and Yuguang Chen. 2021a. Improving event causality identification via self-supervised representation learning on external causal statement. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2162–2172, Online. Association for Computational Linguistics.

Xinyu Zuo, Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Weihua Peng, and Yuguang Chen. 2021b. LearnDA: Learnable knowledge-guided data augmentation for event causality identification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3558–3571, Online. 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, Barcelona, Spain (Online). International Committee on Computational Linguistics.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract & 1 Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

4 Experiments

☑ B1. Did you cite the creators of artifacts you used?
4.1 Experimental Setup

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
4.1 Experimental Setup

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
4.1 Experimental Setup

C ☑ Did you run computational experiments?

4 Experiments

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
No response.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
4.1 Experimental Setup

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
4.3 Overall Results

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
4.1 Experimental Setup

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

3.3 Event Centrality Incorporation

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
3.3 Event Centrality Incorporation

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.