Toward Consistent and Informative Event-Event Temporal Relation Extraction

Xiaomeng Jin
University of Illinois Urbana-Champaign
xjin17@illinois.edu

Haoyang Wen
Carnegie Mellon University
hwen3@cs.cmu.edu

Xinya Du
University of Texas at Dallas
xinya.du@utdallas.edu

Heng Ji
University of Illinois Urbana-Champaign
hengji@illinois.edu

Abstract

Event-event temporal relation extraction aims to extract the temporal order between a pair of event mentions, which is usually used to construct temporal event graphs. However, event graphs generated by existing methods are usually *globally inconsistent* (event graphs containing cycles), *semantically irrelevant* (two unrelated events having temporal links), and *context unaware* (neglecting neighborhood information of an event node). In this paper, we propose a novel event-event temporal relation extraction method to address these limitations. Our model combines a pretrained language model and a graph neural network to output event embeddings, which captures the contextual information of event graphs. Moreover, to achieve global consistency and semantic relevance, (1) event temporal order should be in accordance with the norm of their embeddings, and (2) two events have temporal relation only if their embeddings are close enough. Experimental results on a real-world event dataset demonstrate that our method achieves state-of-the-art performance and generates high-quality event graphs.

1 Introduction

Event-event temporal relation extraction aims to extract the temporal order between a pair of event mentions in natural language text (i.e., an event is BEFORE or AFTER another event), which is essential for constructing temporal event graphs. Event-event temporal relation extraction enables researchers to understand the dynamics of complex events, and benefits a variety of downstream tasks, including event graph construction (Li et al., 2018), future event prediction (Li et al., 2021; Du et al., 2022; Wang et al., 2022; Jin et al., 2022), question answering (Souza Costa et al., 2020; Wang et al., 2021), and summarization (Glavaš and Šnajder, 2014).

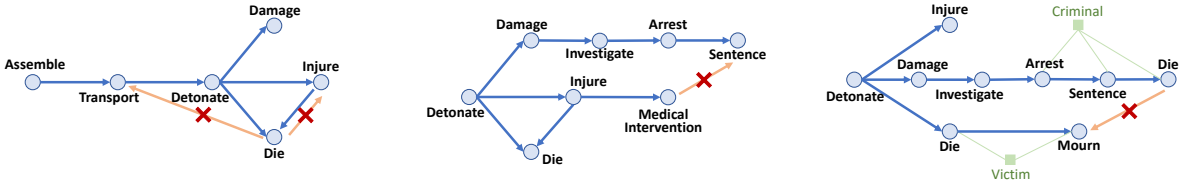
Researchers have proposed many methods (Dligach et al., 2017; Han et al., 2020; Wen and Ji, 2021) to tackle this challenging task. Previous work usually formulates the problem as a pairwise classification task (Dligach et al., 2017; Han et al., 2020; Wen and Ji, 2021). However, they have three major issues when applied to constructing the temporal event graph:

(1) *Global inconsistency*. Local pairwise classification is likely to introduce conflicting predictions when constructing temporal event graphs. Figure 1a shows an example of conflicting local predictions, where yellow links (e.g., DIE → INJURE) conflict with blue links (e.g., DETONATE → INJURE). Although previous work can address conflicts through inference methods such as Integer Linear Programming (Bramsen et al., 2006; Han et al., 2019), resolving this issue directly in temporal relation extraction models yet receives limited attention. (2) *Semantic irrelevance*. Existing methods output a predicted temporal relation for any two given atom events, regardless of their semantic relevance. For example, as shown in Figure 1b, given two events MEDICAL INTERVENTION and SENTENCE, existing models will predict that there is a temporal link from MEDICAL INTERVENTION to SENTENCE. Though it is very likely that MEDICAL INTERVENTION happens before SENTENCE in a real bombing event, those two events have no direct semantic relation, which makes the predicted temporal link semantically irrelevant.

(3) *Context unawareness*. Events with sharing arguments are usually closely related in a temporal event graph, which provides valuable information about the nature of a particular event (Vo and Bagheri, 2019). As shown in Figure 1c, CRIMINAL (rather than VICTIM) is shared by SENTENCE event and DIE event, so it is not likely that the MOURN event follows the DIE (yellow link). However, existing work considers information from candidate event pairs only, while ignoring those rich connections among other related events.

In this paper, we propose a new event-event tem-

Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023), pages 23–32



(a) Global inconsistency (b) Semantic irrelevance (c) Context unawareness

Figure 1: Limitations of existing event-event temporal relation extraction methods. Yellow links are incorrect predictions, but existing methods are prone to making such mistakes. (a) Global inconsistency. The yellow links will introduce cycles to the event graph and make the event graph globally inconsistent. (b) Semantic irrelevance. The yellow link is semantically irrelevant because its two endpoint events have no direct relevance. (c) Context unawareness. The yellow link is incorrect because its DIE event shares an argument with the SENTENCE event (rather than the MOURN event), indicating that this DIE event is associated with criminal rather than victim.

poral relation extraction approach that addresses the above limitations of existing methods. The goal of our approach is to learn event representations that are *globally consistent*, *semantically relevant*, and *context-aware*. As shown in Figure 2, given an input document as well as the entity mentions, we first use off-the-shelf information extraction tools (Du et al., 2022) to extract arguments of events. We then use a pretrained language model (PLM, Devlin et al., 2018) to encode events/arguments and get their PLM-based embeddings. To allow events to be aware of their contextual information, we construct an initial event graph consisting of events/arguments as nodes and event-argument links as edges, then use a graph neural network (GNN, Scarselli et al., 2008) to aggregate neighborhood information for each node iteratively and get their GNN-based embeddings. The PLM-based and GNN-based embeddings are combined together as the final embeddings of events.

Moreover, to ensure that the learned event embeddings are globally consistent and semantically relevant, we hypothesize that the event embedding space should be geometrically meaningful, in which event embeddings satisfy the following two rules: (1) The temporal order of events is in accordance with the norm of event embeddings. Specifically, if event A happens before event B, then the embedding norm of event A should be smaller than that of event B. (2) There exists a temporal link between two events if and only if their embeddings are close enough to each other in the event embedding space. Specifically, if events A and B are connected by a temporal edge (either A happens before B or after B), then the distance between A's and B's embedding should be smaller than a predefined threshold, and vice versa. The first rule ensures that the constructed event graph is globally consistent, and the second rule ensures that there will be a temporal link between two events only if they are semantically relevant. We implement these two rules in our model by minimizing a corresponding margin-based loss w.r.t the model parameters, thus the whole model can be trained in an end-to-end fashion.

We conduct experiments on the Event Story Line dataset (Caselli and Vossen, 2017). The experimental results demonstrate that our proposed method achieves state-of-the-art performance on event-event temporal relation extraction. We also show that compared with baseline methods, event graphs generated by our method are globally consistent and semantically relevant.

In summary, our contributions are as follows:

- We review the literature on event-event temporal relation extraction thoroughly, and observe that a well-behaved event temporal relation extraction method should be *globally consistent*, *semantically relevant*, and *context aware*.
- Methodologically, we use graph neural networks to process event graphs and learn event representations, which enables the model to learn event embeddings that are *context aware*. Moreover, we use the distance between event embeddings as the criterion for judging the existence of event temporal edges, and use the norm of event embeddings as the criterion for determining the direction of event temporal edges, which enables our model to be *globally consistent* and *semantically relevant*.
- We conduct extensive experiments on event-event temporal relation extraction task, and the results demonstrate that our proposed method achieves substantial improvements over state-of-the-art baseline methods.

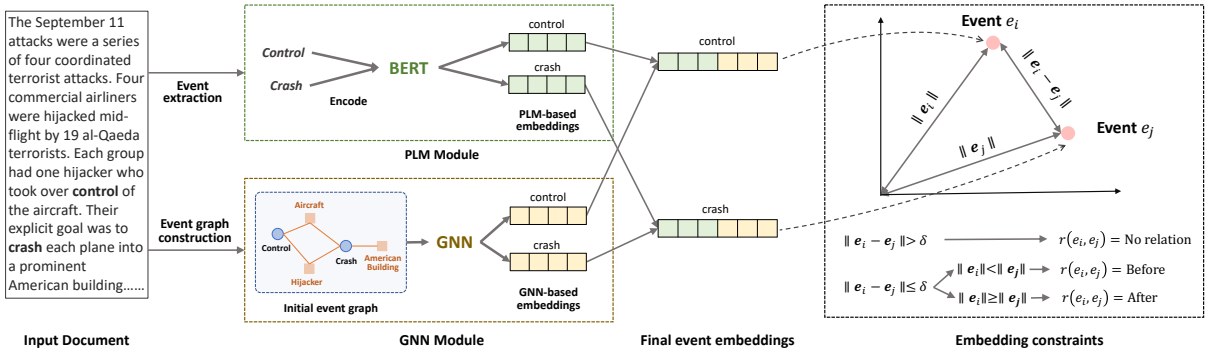


Figure 2: The architecture of our model. Given an input document, our model uses a pretrained language model (PLM) to encode event mentions and get their PLM-based embeddings, and uses a graph neural network (GNN) to aggregate contextual information on the initial event graph and get their GNN-based embeddings. The PLM-based and GNN-based embeddings are combined together as the final embeddings of events. To achieve global consistency and semantic relevance, we hypothesize that the event embedding space is geometrically meaningful by imposing two constraints on event embeddings. See Section 3.3 for details.

2 Problem Formulation

The event-event temporal relation extraction problem is formulated as follows. Given a document, we use $\{e_1, e_2, \cdots\}$ to denote the set of event mentions, and $\{a_1, a_2, \cdots\}$ to denote the set of argument mentions. An event node e_i and an argument a_j are connected by a link if a_j is an argument of e_i. The event mentions are obtained from the gold standard annotations for the dataset. The arguments and event-argument links can be obtained by applying off-the-shelf information extraction (IE) tools (Luan et al., 2019; Wen et al., 2021) to the input document. We also assume they are given as our input.

Our task is to predict the temporal relation between a pair of events (e_i, e_j), which can be BEFORE, AFTER, or NO RELATION. The predicted results can then be used to construct a temporal event graph, in which each node represents an event, and each directed edge $e_i \rightarrow e_j$ represents a temporal relation indicating that event e_i happens before event e_j (or equivalently, event e_j happens after event e_i). If the relation type between e_i and e_j is predicted as NO RELATION, then there is no edge between the two event nodes in the temporal event graph. Our goal is to propose a *globally consistent* and *semantically relevant* event-event temporal relation extraction method, so that the generated temporal event graph is *valid* (no conflict), *concise* (only related events can be connected), and *meaningful* (temporal links should be aware of the meaning of event nodes).

3 Our Approach

The overall framework of our proposed approach is shown in Figure 2. In this section, we will introduce each part of the framework in detail.

3.1 PLM Module

Pretrained language models (PLMs) are usually trained on a large corpus and thus is able to encode words as vector representations while preserving their semantics. Following most existing methods, we feed an input document into a PLM[1] first to obtain an initial vector representation for each token in the document. Specifically, for a document with a sequence of tokens $\{w_1, w_2, \cdots\}$, we first sum the token, segment, and positional embeddings for each token to compute its initial input representation $\{\boldsymbol{h}_1^0, \boldsymbol{h}_2^0, \cdots\}$, and then compute an output representation for each PLM layer l:

$$\{\boldsymbol{h}_1^l, \cdots\} = \text{PLM-Layer}(\{\boldsymbol{h}_1^{l-1}, \cdots\}) \quad (1)$$

for $l = 1, \cdots, L$, where PLM-Layer$(\cdot)$ is a single PLM encoder layer, whose parameters are initialized using a pretrained model, and L is the number of PLM layers. We suggest readers to refer to Devlin et al. (2018) for technical details of these layers.

The representation of event mention e_i output by the last PLM layer is denoted as $\boldsymbol{h}_i^L$. If an event mention consists of multiple tokens, we simply average the embeddings of all tokens in this event mention. Finally, we use an MLP with two hidden

[1] In our case, we use BERT (Devlin et al., 2018).

layers to compute the final PLM-based representation of event e_i:

$$\boldsymbol{h}_{e_i}^{\text{PLM}} = \text{MLP}\big(\boldsymbol{h}_i^L\big). \quad (2)$$

3.2 GNN Module

Note that in a temporal event graph, an event is usually closely related to its contextual events, which share common argument entities with the given event. Contextual events provide valuable information about the nature of a particular event and help improve the performance of temporal link prediction. As shown in Figure 1c, The contextual events of the right DIE (i.e., SENTENCE, ARREST) and the contextual events of the left DIE (i.e., MOURN) indicate that they are associated with criminal and victim, respectively, so the right DIE event should not be followed by a MOURN event.

To let our model be aware of contextual event information, we first construct an initial event graph where nodes represent event mentions and argument mentions extracted from the given input document, and edges represent event-argument links. Then we use Graph Neural Networks (GNNs, Kipf and Welling, 2017) to perform message passing on the initial event graph and learn event representations. Specifically, for an initial event graph G, we use $\boldsymbol{s}_i^k$ to denote the representation of node $i \in G$ at iteration k (which can be either an event or an entity). Then the node representation is updated by aggregating its neighborhood information:

$$\boldsymbol{s}_i^k = \sigma\Big(\mathbf{W}^k \sum\nolimits_{j \in \mathcal{N}(i) \cup \{i\}} \alpha_{ij} \boldsymbol{s}_j^{k-1}\Big) \quad (3)$$

for $k = 1, \cdots, K$, where K is the number of GNN layers.

For each node i, $\mathcal{N}(\cdot)$ denotes the set of its neighbors [2]. $\alpha_{ij} = 1/\sqrt{|\mathcal{N}(i)| \cdot |\mathcal{N}(j)|}$ is the weight coefficient. $\boldsymbol{s}_i^0$ is initialized as the embedding of event mention e_i (i.e., $\boldsymbol{s}_i^0 = \boldsymbol{h}_i^L$), which is provided by PLM in Section 3.1.

The output of the GNN's last layer is taken as the final GNN-based representation of event e_i:

$$\boldsymbol{s}_{e_i}^{\text{GNN}} = \boldsymbol{s}_i^K. \quad (4)$$

3.3 Globally Consistent and Semantically Relevant Event Representation

After obtaining the PLM-based and GNN-based event representations, we concatenate these two types of embeddings for each event and get the final event embedding:

$$\boldsymbol{e}_i = \text{CONCAT}\left(\boldsymbol{h}_{e_i}^{\text{PLM}}, \boldsymbol{s}_{e_i}^{\text{GNN}}\right). \quad (5)$$

To predict the temporal relation between a pair of events, a straightforward way is to train a classifier on their embeddings, for example, an MLP that takes the concatenation of two event embeddings as input and outputs labels of BEFORE, AFTER, or NO RELATION. However, the trained classifier is not guaranteed to be globally consistent (no cycle in event graphs) and semantically relevant (temporal links only exist between events that are closely related), which makes the predicted temporal event links invalid and irrelevant.

To address these issues, we hypothesize that the event embedding space should be geometrically meaningful, and event embeddings should satisfy the following two constraints:

- *The temporal order of events is in accordance with the norm* [3] *of event embeddings.* Specifically, if there is a temporal link from event e_i to event e_j, then the length of event e_i's embedding should be smaller than the length of event e_j's, embedding:

$$e_i \rightarrow e_j \Rightarrow \|\boldsymbol{e}_i\| < \|\boldsymbol{e}_j\|. \quad (6)$$

  It is clear to see that event graphs will be cycle-free under the above constraint. Otherwise, assume that there is cycle $e_i \rightarrow e_j \rightarrow \cdots \rightarrow e_i$, then according to Eq. (6), we have $\|\boldsymbol{e}_i\| < \|\boldsymbol{e}_j\| < \cdots < \|\boldsymbol{e}_i\|$, which is impossible.

- *There exists a temporal relation between two events if and only if their embeddings are close enough in the event embedding space*, since we assume that a temporal relation is meaningful only if the two events are semantically related. Specifically, if events e_i and e_j are connected by a temporal edge (e_i happens either before or after e_j), then the distance between e_i's and e_j's embeddings should be less than a threshold δ that is a pre-defined real positive number, and vice versa:

$$e_i \rightarrow e_j \vee e_j \rightarrow e_i \Leftrightarrow \|\boldsymbol{e}_i - \boldsymbol{e}_j\| < \delta. \quad (7)$$

  Under the constraint in Eq. (7), two events can be connected by a temporal link only if their

[2] To alleviate the sparsity issue of event graphs, all edges in event graphs are treated undirected when counting neighbors.

[3] We use L2 norm in this paper.

embeddings are close to each other, which discourages the model from predicting a temporal link for two events that are distant in the timeline. In this way, our model will learn to output a "minimal" temporal event graph that preserves its essential chronological structure.

3.4 Model Training and Inference

Training. Each training document consists of a set of event temporal links used for training the model. According to the ground-truth label of an event temporal link, the loss function is designed as follows:

- If $r(e_i, e_j) = \text{BEFORE}$, i.e., event e_i happens before event e_j, then the loss term for event pair (e_i, e_j) is

$$L_{ij}^{\text{BEFORE}} = \big[\|\boldsymbol{e}_i\| - \|\boldsymbol{e}_j\|\big]^+ + \big[\|\boldsymbol{e}_i - \boldsymbol{e}_j\| - \delta\big]^+,$$

  where the function $[\cdot]^+ = \max(\cdot, 0)$. The first term encourages the embedding length of e_i to be smaller than e_j, and the second term encourages the distance between e_i's and e_j's embedding to be less than δ.

- Similarly, if $r(e_i, e_j) = \text{AFTER}$, i.e., event e_i happens after event e_j, then the loss term for event pair (e_i, e_j) is

$$L_{ij}^{\text{AFTER}} = \big[\|\boldsymbol{e}_j\| - \|\boldsymbol{e}_i\|\big]^+ + \big[\|\boldsymbol{e}_j - \boldsymbol{e}_i\| - \delta\big]^+.$$

- Otherwise, if $r(e_i, e_j) = \text{NO RELATION}$, i.e., there is no explicit temporal ordering between e_i and e_j, then the loss term for event pair (e_i, e_j) is

$$L_{ij}^{\text{NO-REL}} = \big[\delta - \|\boldsymbol{e}_i - \boldsymbol{e}_j\|\big]^+,$$

  which encourages the distance between e_i's and e_j's embedding to be larger than δ.

The total loss function of our model is therefore as follows:

$$\begin{aligned} L = \sum_{D \in \mathcal{D}} \sum_{(e_i, e_j) \in D} \Big(& \mathbb{1}[r(e_i, e_j) = \text{BEFORE}] L_{ij}^{\text{BEFORE}} \\ & + \mathbb{1}[r(e_i, e_j) = \text{AFTER}] L_{ij}^{\text{AFTER}} \\ & + \mathbb{1}[r(e_i, e_j) = \text{NO RELATION}] L_{ij}^{\text{NO-REL}} \Big), \end{aligned}$$

where $\mathcal{D}$ is the training dataset, and $D \in \mathcal{D}$ is a training document. The whole model can thus be trained by minimizing the above total loss using gradient-based optimization methods.

# train/val/test documents	206 / 26 / 26
# avg events / document	12.6
# avg arguments / document	30.1
# avg relations / document	21.4

Table 1: Statistics of the Event StoryLine Corpus.

Inference. In the inference stage, to predict the temporal relation between two events e_i and e_j, we first calculate the event embeddings of e_i and e_j using the PLM module and GNN module in our model, then output the label of (e_i, e_j) according to the following criteria:

$$r(e_i, e_j) = \begin{cases} \text{BEFORE}, \text{if } \|\boldsymbol{e}_i - \boldsymbol{e}_j\| < \delta \wedge \|\boldsymbol{e}_i\| < \|\boldsymbol{e}_j\|, \\ \text{AFTER}, \text{if } \|\boldsymbol{e}_i - \boldsymbol{e}_j\| < \delta \wedge \|\boldsymbol{e}_i\| \geq \|\boldsymbol{e}_j\|, \\ \text{NO RELATION}, \text{if } \|\boldsymbol{e}_i - \boldsymbol{e}_j\| \geq \delta. \end{cases}$$

4 Experiments

4.1 Datasets

We conduct experiments on Event StoryLine Corpus (Caselli and Vossen, 2017), which contains 258 documents on 22 calamity topics.[4] It consists of human-annotated event temporal links: RISING_ACTION, which means the former event happens earlier than and implicitly enables the later event, or FALLING_ACTION, which means the former event happens later than and is the outcome/effect of the later event. We map RISING_ACTION to BEFORE and FALLING_ACTION to AFTER in our method.

The statistics of the dataset are summarized in Table 1. We split the documents into train, validation, and test sets. There are also entity annotations in each document including location and person. We use these entity mentions as argument nodes in the initial event graph construction.

4.2 Baseline Methods

We compare our method with the following event-event temporal relation extraction methods:

- **BERT+MLP.** Given two events e_i and e_j, we use BERT base model to encode each event and get their embeddings $\boldsymbol{h}_{e_i}^{\text{BERT}}$ and $\boldsymbol{h}_{e_j}^{\text{BERT}}$. Then the temporal relation between e_i and e_j is computed by $r(e_i, e_j) = \text{MLP}\big(\text{CONCAT}(\boldsymbol{h}_{e_i}^{\text{BERT}}, \boldsymbol{h}_{e_j}^{\text{BERT}})\big)$.

[4] We do not conduct experiments on another popular dataset MATRES (Ning et al., 2018) because a large portion of the annotated temporal edges in MATRES are redundant and semantically irrelevant.

Methods	Accuracy			Consistency	
	Precision	Recall	F_1	SCR	CCR
BERT+MLP	0.617 ± 0.013	0.655 ± 0.017	0.633 ± 0.016	0.214 ± 0.020	0.130 ± 0.018
GNN+MLP	0.629 ± 0.010	0.663 ± 0.014	0.644 ± 0.011	0.286 ± 0.014	0.170 ± 0.016
Wen and Ji (2021)	**0.692** ± 0.017	0.618 ± 0.022	0.652 ± 0.019	0.754 ± 0.026	0.481 ± 0.028
Our method	0.633 ± 0.014	**0.719** ± 0.019	**0.673** ± 0.016	**1.000** ± 0.000	**0.626** ± 0.020
Ablations					
- w/o GNN	0.699 ± 0.018	0.613 ± 0.015	0.651 ± 0.017	1.000 ± 0.000	0.592 ± 0.024
- w/o PLM	0.505 ± 0.023	0.684 ± 0.016	0.585 ± 0.020	1.000 ± 0.000	0.513 ± 0.017

Table 2: The results of ternary classification (BEFORE, AFTER, or NO RELATION). The best results are highlighted in bold. SCR and CCR mean "Simple Consistency Rate" and "Correct Consistency Rate", respectively.

Methods	Accuracy			Consistency	
	Precision	Recall	F_1	SCR	CCR
BERT+MLP	0.020 ± 0.015	0.662 ± 0.012	0.038 ± 0.014	0.246 ± 0.013	0.127 ± 0.011
GNN+MLP	0.018 ± 0.012	**0.683** ± 0.018	0.035 ± 0.016	0.352 ± 0.010	0.137 ± 0.016
Liu et al. (2021)	0.419	0.625	0.501	-	-
Our method	**0.596** ± 0.016	0.632 ± 0.009	**0.618** ± 0.013	**1.000** ± 0.000	**0.470** ± 0.017
Ablations					
- w/o GNN	0.552 ± 0.017	0.572 ± 0.022	0.565 ± 0.019	1.000 ± 0.000	0.431 ± 0.013
- w/o PLM	0.571 ± 0.019	0.585 ± 0.024	0.580 ± 0.022	1.000 ± 0.000	0.368 ± 0.014

Table 3: The results of binary classification (HAVE RELATION or NO RELATION).

- **GNN+MLP.** This is similar to BERT+MLP, except that we use GNN to encode each event and get their embeddings. Specifically, the temporal relation between e_i and e_j is computed by $r(e_i, e_j) = \text{MLP}\big(\text{CONCAT}(\boldsymbol{h}_{e_i}^{\text{GNN}}, \boldsymbol{h}_{e_j}^{\text{GNN}})\big)$.
- **Wen and Ji (2021)** propose a joint model for event-event temporal relation classification. It is the state-of-the-art event-event temporal relation extraction approach, which adopts a stack-propagation framework to incorporate relative event time prediction for temporal relation classification.
- **Liu et al. (2021)** propose an event causality identification model. It is an event-event causal relation identification model that uses a mechanism called event mention masking generalization. Note that this model performs a causality existence prediction on Event StoryLine Corpus. To make a fair comparison with this baseline, we modify our model output to binary classification. Specifically, the relation between two events i and j is decided by the distance between two event embeddings $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$: If $\|\boldsymbol{e}_i - \boldsymbol{e}_j\| < \delta$, then $r(e_i, e_j) =$ HAVE RELATION, otherwise $r(e_i, e_j) =$ NO RELATION.

In addition, to examine the effectiveness of using GNN to learn contextual information, we conduct ablation study and design the following reduced version of our model:

- Our method without GNN module, which uses the PLM-based embedding as the event embedding. Instead of Eq. (5), the final embedding of event e_i is $\boldsymbol{e}_i = \boldsymbol{h}_{e_i}^{\text{PLM}}$.
- Our method without PLM module, which uses the GNN-based embedding as the event embedding, i.e., the final embedding of event e_i is $\boldsymbol{e}_i = \boldsymbol{h}_{e_i}^{\text{GNN}}$.

4.3 Experimental Setup

Dataset preprocessing. Our model predicts the relation between events, which is a classification task. The ground truth annotation only includes the relation type BEFORE and AFTER, without NO RELATION. To compare with baselines, we randomly select negative samples from all event pairs that are not in the annotation set, and label them as NO RELATION. The number of negative samples is one half of annotated event pairs for each document to ensure that labels are balanced. To compare with Liu's method (Liu et al., 2021), we merge the BEFORE and AFTER labels to HAVE RELATION, and treat all negative pairs as NO RELATION.

To construct the initial input graph, we first include event nodes which represent event mentions in the ground truth annotations. In addition, there are also annotations of person and location spans in the ground truth annotations. We add the annotations as argument nodes in the initial input graph.

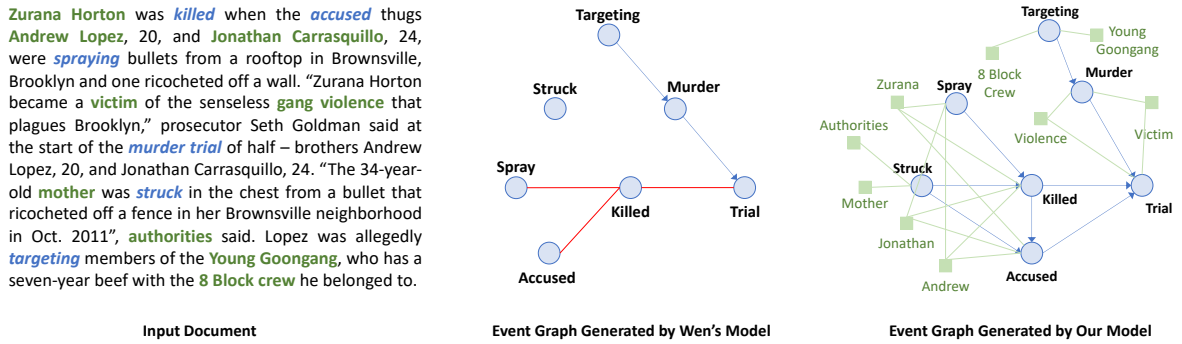


Figure 3: Case study on the quality of generated temporal event graphs. Given the input document, the event graph generated by Wen's model (Wen and Ji, 2021) is drawn in the middle, in which more than one half of edges are inconsistent (highlighted in red). Moreover, it fails to identify the edge STRUCK → KILLED since it does not consider argument information. In contrast, our model predicts all edges precisely and consistently.

We connect an event node and an argument node as an event-argument relation if they belong to the same sentence or consecutive sentences.

Evaluation metrics. We use the following metrics to evaluate our model and baseline methods:

- **Accuracy**. We use $Precision$, $Recall$, and F_1 score to evaluate performance of our model and baseline methods. We report our averaged test performance on 5 random seeds.
- **Consistency**. Note that whether a temporal relation extraction model satisfies global consistency greatly affects its practical reliability. To investigate the global consistency of our model as well as baselines, we exchange the two events in an input event pair, and feed the reversed event pair into the model and obtain the prediction. For a pair of events (e_i, e_j), the consistent prediction of its reversed pair should be

$$r(e_j, e_i) = \begin{cases} \text{BEFORE}, & \text{if } r(e_i, e_j) = \text{AFTER}, \\ \text{AFTER}, & \text{if } r(e_i, e_j) = \text{BEFORE}, \\ \text{NO-REL}, & \text{if } r(e_i, e_j) = \text{NO-REL}. \end{cases}$$

We count the number of event pairs in the test set whose reversed pair has the consistent prediction with the original pair, and define the Simple Consistency Rate (SCR) as

$$\text{SCR} = \frac{\text{\# consistent event pairs}}{\text{\# all event pairs}}.$$

Note that SCR does not consider the model's prediction accuracy. Therefore, we define the Correct Consistency Rate (CCR) as

$$\text{CCR} = \frac{\text{\# consistent and correct event pairs}}{\text{\# all event pairs}}.$$

Hyperparameter Settings. For the GNN module, we use a three-layer GCN as the encoder, whose dimensions of hidden layers are 256, 128, and 16, respectively. For the PLM module, we use BERT base model uncased (Devlin et al., 2018) and the dimensions of the MLP hidden layers in Eq. (2) are 128 and 16, respectively. The learning rate is 10^{-5}, the number of training epochs is 200, and δ is set to 16.

4.4 Results and Analysis

Comparison with baseline methods. The results of ternary and binary classification are reported in Tables 2 and 3, respectively. It is clear that our method achieves substantial gains over all baseline methods in both classification tasks. Specifically, the F1 score of our method surpass the the best baseline method by 2.1% and 11.2% in ternary and binary classification, respectively. This demonstrates that utilizing contextual information of event graphs and preserving the global consistency as well as semantic relevance are essential to improving the performance of event temporal relation extraction.

It is also worth noticing that the Simple Consistency Rate of all baseline methods are significantly lower than our method. Moreover, the Correct Consistency Rate is much lower than the Simple Consistency Rate. This is because these models do not take into account the global consistency during training and thus causing conflicts in prediction results. In contrast, our method is theoretically guaranteed to be globally consistent.

Ablation study. The results of the ablation study are shown in Tables 2 and 3. We observe a sub-

stantial performance degradation after removing the GNN module or PLM module from our model. The result demonstrates that both GNN module and PLM module are essential to learning high-quality event representations, since PLM provides general sense of events while GNN explicitly utilizes contextual information in event graphs.

Case study. The example temporal event graphs generated by Wen's model (Wen and Ji, 2021) and our model are drawn in Figure 3. The input document is shown on the left of Figure 3, where texts in blue are event mentions and texts in orange are the annotated named entities (arguments). The graph in the middle is the temporal event graph predicted by Wen's model. We use blue links to denote consistent temporal edges and red links to denote inconsistent temporal edges according to the prediction of Wen's model. We observe that the prediction of Wen's model has inconsistency problem since more than one half of the predicted temporal links are inconsistent. Specifically, the prediction of r(SPRAY, KILLED) is BEFORE whereas the prediction of r(KILLED, SPRAY) is NO RELATION. This is because Wen's model does not consider the consistency issue, thus causes conflicts in its generated temporal event graph. In addition, Wen's model fails to identify the relation between STRUCK and KILLED.

The graph on the right is predicted by our model. The additional green rectangles are arguments. As opposed to the middle graph, all the predictions by our model are correct and consistent. An important reason is that our model takes the contextual information of event graphs into account. For example, there are three named entities connecting STRUCK and KILLED (i.e., JONATHAN, ANDREW, and ZURANA), which provides valuable information to identify the temporal relation between them.

5 Related Work

Event-event temporal relation extraction can be viewed as a classification task that predicts the relation type between two event mentions. In general, existing event-event temporal relation extraction methods can be classified into two categories: traditional rule-based methods and neural network based methods.

The traditional rule-based methods apply linguistic rules to the features extracted from documents to predict the relation between a given event pair. For example, Laokulrat et al. (2013) propose a system that uses a rule-based approach as baseline to determine temporal links and a machine learning classifier to filter out baseline candidates. Chambers et al. (2014) design a sieve-based architecture CAEVO that applies a sequence of temporal relation classifiers to label event-event temporal relations. This supports a combination of both rule-based and machine learned classifiers. However, these rule-based methods require substantial manual design of rules, which greatly limits their usage in practice. Moreover, rules are usually not comprehensive enough to capture the complex event-event temporal relations.

Another line of related work focuses on the neural network based methods, which extracts event-event temporal relations via deep neural networks and pre-trained language models. For example, Wang et al. (2020) introduce a joint constrained learning framework that incorporates contextual features encoded with pre-trained language models and external knowledge from commonsense knowledge bases. Wen and Ji (2021) adopt a stack-propagation framework to combine relative time prediction and event-event temporal relation classification. However, they do not consider global consistency and semantic relevance of the generated event graphs.

6 Conclusion and Future Work

In this paper, we propose a *globally consistent*, *semantically relevant*, and *context aware* event-event relation extraction framework, which addresses the limitations of existing methods. Our model uses a pretrained language model module and graph neural network module to jointly represent event graphs. In addition, we make the event embedding space geometrically meaningful by imposing two constraints on event embeddings: event temporal order should be in accordance with event embedding norm, and event temporal relations should only exist between events whose embeddings are close enough. Experiments demonstrate that our method significantly outperforms baselines by generating accurate and globally consistent temporal event graphs.

In the future, we aim to incorporate external background knowledge and commonsense knowledge into our framework. We also plan to make use of the generated temporal event graphs in downstream tasks, such as future event prediction and question answering.

Limitations

In the current design setting, our proposed model is only able to classify temporal relations between event pairs into one of three classes: BEFORE, AFTER, and NO RELATION. Our model should be more practically useful if we can extend it to predict more relation types in addition to temporal relations, such as PARENT-CHILD and CAUSE-CAUSED_BY relations. We believe that our model is able to make such extension without too much modification.

In addition, as mentioned in the previous section, our model does not make use of any external knowledge, e.g., commonsense knowledge of event temporal relations. Our framework should be more powerful to deal with domain-specific articles if utilizing such knowledge in the framework.

Ethical Considerations

We acknowledge that our work is aligned with the *ACL Code of the Ethics* (Gotterbarn et al., 2018) and will not raise ethical concerns. We do not use sensitive datasets/models that may cause any potential issues. The design, implementation, and evaluation of our proposed method are robust and secure.

References

Philip Bramsen, Pawan Deshpande, Yoong Keok Lee, and Regina Barzilay. 2006. Inducing temporal graphs. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 189–198. Citeseer.

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.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dmitriy Dligach, Timothy Miller, Chen Lin, Steven Bethard, and Guergana Savova. 2017. Neural temporal relation extraction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 746–751.

Xinya Du, Zixuan Zhang, Sha Li, Pengfei Yu, Hongwei Wang, Tuan Manh Lai, Xudong Lin, Ziqi Wang, Iris Liu, Ben Zhou, et al. 2022. Resin-11: Schema-guided event prediction for 11 newsworthy scenarios. In *Proc. 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL2022) System Demonstration Track*.

Goran Glavaš and Jan Šnajder. 2014. Event graphs for information retrieval and multi-document summarization. *Expert systems with applications*, 41(15):6904–6916.

DW Gotterbarn, Bo Brinkman, Catherine Flick, Michael S Kirkpatrick, Keith Miller, Kate Vazansky, and Marty J Wolf. 2018. Acm code of ethics and professional conduct.

Rujun Han, Qiang Ning, and Nanyun Peng. 2019. Joint event and temporal relation extraction with shared representations and structured prediction. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 434–444. Association for Computational Linguistics.

Rujun Han, Yichao Zhou, and Nanyun Peng. 2020. Domain knowledge empowered structured neural net for end-to-end event temporal relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5717–5729.

Xiaomeng Jin, Manling Li, and Heng Ji. 2022. Event schema induction with double graph autoencoders. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2013–2025.

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.

Natsuda Laokulrat, Makoto Miwa, Yoshimasa Tsuruoka, and Takashi Chikayama. 2013. Uttime: Temporal relation classification using deep syntactic features. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 88–92.

Manling Li, Sha Li, Zhenhailong Wang, Lifu Huang, Kyunghyun Cho, Heng Ji, Jiawei Han, and Clare Voss. 2021. Future is not one-dimensional: Graph

modeling based complex event schema induction for event prediction. *arXiv preprint arXiv:2104.06344*.

Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Constructing narrative event evolutionary graph for script event prediction. *arXiv preprint arXiv:1805.05081*.

Jian Liu, Yubo Chen, and Jun Zhao. 2021. Knowledge enhanced event causality identification with mention masking generalizations. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3608–3614.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. *arXiv preprint arXiv:1904.03296*.

Qiang Ning, Hao Wu, and Dan Roth. 2018. A multi-axis annotation scheme for event temporal relations. *arXiv preprint arXiv:1804.07828*.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80.

Tarcísio Souza Costa, Simon Gottschalk, and Elena Demidova. 2020. Event-qa: A dataset for event-centric question answering over knowledge graphs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3157–3164.

Duc-Thuan Vo and Ebrahim Bagheri. 2019. Extracting temporal event relations based on event networks. In *European Conference on Information Retrieval*, pages 844–851. Springer.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020. Joint constrained learning for event-event relation extraction. *arXiv preprint arXiv:2010.06727*.

Hongwei Wang, Zixuan Zhang, Sha Li, Jiawei Han, Yizhou Sun, Hanghang Tong, Joseph P. Olive, and Heng Ji. 2022. Schema-guided event graph completion. In *arXiv*.

Jiexin Wang, Adam Jatowt, Michael Färber, and Masatoshi Yoshikawa. 2021. Improving question answering for event-focused questions in temporal collections of news articles. *Information Retrieval Journal*, 24(1):29–54.

Haoyang Wen and Heng Ji. 2021. Utilizing relative event time to enhance event-event temporal relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10431–10437.

Haoyang Wen, Ying Lin, Tuan Lai, Xiaoman Pan, Sha Li, Xudong Lin, Ben Zhou, Manling Li, Haoyu Wang, Hongming Zhang, et al. 2021. Resin: A dockerized schema-guided cross-document cross-lingual cross-media information extraction and event tracking system. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 133–143.