DCT-Centered Temporal Relation Extraction

Liang Wang, Peifeng Li * and **Sheng Xu**
School of Computer Science and Technology Soochow University, Jiangsu, China
docy@vip.qq.com, pfli@suda.edu.cn, sxu@stu.suda.edu.cn

Abstract

Most previous studies on temporal relation extraction focus on extracting temporal relations among events and suffer from the issue of different forms of events, timexes and Document Creation Time (DCT) in a document. Moreover, DCT can act as a hub to semantically connect the other events and timexes. Unfortunately, previous work cannot fully use such critical and helpful information. To address the above issues, we propose a unified DCT-centered Temporal Relation Extraction model DTRE to identify temporal relations among events, timexes and DCT. Specifically, we first introduce sentence-style DCT to unify the expressions of event, timex and DCT. Then, we apply a DCT-aware graph to obtain their contextual structural representations. Furthermore, we propose a DCT-anchoring multi-task framework to jointly predict three tasks of temporal relation extraction in a batch. Finally, we provide a DCT-guided global inference to further enhance the global consistency among different relations. Experimental results on three popular datasets TBD, TDD-man and TDD-Auto show that our DTRE outperforms several SOTA baselines on E-E, E-T and E-D significantly.

1 Introduction

Temporal relation extraction focuses on the occurrence order (TLINK) of event mentions, time expressions (timexes) and Document Creation Time (DCT). Most previous studies only focus on the event-centered tasks and consider three TLINKs: event-event (E-E), event-timex (E-T), and event-DCT (E-D). As a crucial component of relation extraction, temporal relation extraction can help many downstream NLP tasks, such as question answering (Ning et al., 2020), summarization (Noh et al., 2020) and timeline construction (Li et al., 2021).

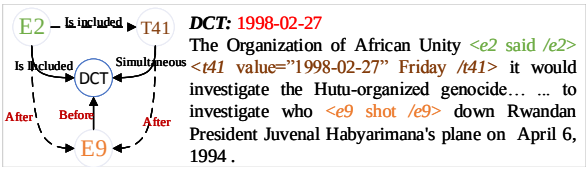

Figure 1: Examples of temporal relations among events, timex and DCT.

Most previous studies (Mathur et al., 2021; Liu et al., 2021) only focus on the single E-E task, ignoring the other E-T and E-D tasks. The main barricade is the hardness of combining the E-T or E-D task with the E-E task due to their different expression forms. Since most event mentions [1] are sentences or clauses, their rich information is helpful for a neural network model to identify the specific relation between two event mentions. However, timex and DCT (especially DCT) are only word-level or phrase-level tokens, and the information imbalance between events and timexes/DCT will lead to the issue that it is difficult for a neural network model to extract their correct temporal relation by a unified model.

As shown in Figure 1, identifying the temporal relation between the two long-distance event mentions *e2* and *e9* is challenging, even for humans. However, if we first recognize the *Is Included* and *Before* links of (*e2*, *DCT*) and (*e9*, *DCT*), then the *After* link of (*e2*, *e9*) will be much easier to obtain. Fortunately, identifying the temporal relation between event (or timex) and DCT is relatively simple for humans and pre-trained language models.

Since event, timex and DCT have different forms of expressions, most previous studies (Cheng and Miyao, 2017; Meng and Rumshisky, 2018) only focus on E-E or regarded E-E, E-T and E-D as three independent tasks, as we mentioned above. They often suffer from the issues of data scarcity

*Corresponding author

[1] An event mention refers to a phrase, clause or sentence within which an event is described.

Proceedings of the 29th International Conference on Computational Linguistics, pages 2087–2097
October 12–17, 2022.

and overfitting due to the small size of available datasets. Cheng et al. (2020) propose a model to fuse E-E, E-T and E-D into a unified model by introducing a learnable parameter-based DCT representation. However, they still suffer from two issues: 1) the different expressions of event, timex and DCT harm the information interaction among different tasks; 2) they ignore the importance of DCT to connect the events and timexes in a document.

To address the above issues, we propose a unified **D**CT-centered **T**emporal **R**elation **E**xtraction (**DTRE**) model to discover the temporal relations among events, timexes and DCT in a document. Specifically, we first introduce a sentence-style DCT representation to unify the expressive forms of event, timex, and DCT. Then, we generate a DCT-aware graph to obtain their contextual structural representations. Furthermore, we propose a DCT-anchoring multi-task learning framework to jointly predict three temporal relations (i.e., E-E, E-T and E-D). Finally, we provide a DCT-guided global inference mechanism to benefit from the high accuracies of the E-D and T-D tasks. Experimental results on three popular datasets TBD, TDD-man and TDD-Auto show that our DTRE outperforms several SOTA baselines on all three tasks E-E, E-T and E-D significantly. In summary, our contributions are as follows:

- We introduce a sentence-style DCT representation to unify the expressive forms of events, timexes and DCT;
- We propose a DCT-aware graph to obtain the contextual structural representations;
- We construct a DCT-anchoring multi-task learning framework to jointly predict three different temporal relations in one batch;
- We provide a DCT-guided global inference mechanism to further enhance the global consistency among different relations. In our future work, we will focus on constructing more effective DCT representation.

2 Related Work

TimeBank (Pustejovsky et al., 2003) is an early temporal relation corpus and its extended version TimeBank-Dense (TBD) (Cassidy et al., 2014) adopts a dense annotation scheme in a slide window within adjacent sentences, in which there are mainly five types of TLINKs: E-E, E-T, T-T, E-D and T-D. MATRES (Ning et al., 2018) only contains E-E relations and simplifies the relation labels with a higher inter-annotator agreement. TDDiscourse (Naik et al., 2019) is a discourse-level temporal relation dataset based on TBD, which also focuses on E-E temporal ordering.

Early work on temporal relation extraction (Chambers et al., 2007; Chambers and Jurafsky, 2008; Do et al., 2012; D'Souza and Ng, 2013; Chambers et al., 2014) focused on various linguistic features, including part-of-speech (POS), lexical and morphological features, dependency parsing information, etc. Recent work mainly focuses on the E-E task using neural networks. Liu et al. (2021) and Mathur et al. (2021) show that graph-based neural networks can help relieve informative sparsity between long-distance event mentions in discourse-level temporal relation extraction. Besides, a bunch of efforts focus on incorporating external resources to deal with the limited training resource, such as combining the pre-trained temporal-aware language model (Han et al., 2021), collecting the distantly-supervised examples (Zhao et al., 2021), applying the transfer learning methods to leverage complementary datasets (Ballesteros et al., 2020). Other methods seek to enhance global inference with structural constraints, i.e., relieving the transitive conflicts within temporal graphs (Ning et al., 2017; Han et al., 2019).

Only a few studies consider all three tasks. Early methods were mostly rule-based on event attributes (Chambers et al., 2014; Mirza and Tonelli, 2016), whose performance are deeply harmed by the vague relation, or simply transferred the neural architecture on E-E to E-D and E-T directly via input adjustment (Cheng and Miyao, 2017; Meng and Rumshisky, 2018). Motivated by the success of Multi-Task Deep Neural Network (Liu et al., 2019a) that leverages different supervised learning tasks with the shared contextual embeddings, Cheng et al. (2020) proposed a multi-category learning framework to joint E-E, E-T and E-D. Specially, they introduce a learnable vector to represent DCT as it does not explicitly occur in documents.

3 DTRE: DCT-Centered Temporal Relation Extraction

The temporal relations between event mentions are determined by their occurrence intervals (i.e., start and end points). However, in most real-world texts,

events' intervals are rarely explicitly mentioned, and then external knowledge or common sense is required for temporal reasoning. One important clue is "Had this event happened yet?" or "Is this a future event?", i.e., the E-D task, which is easier and helpful for the E-E and E-T tasks.

To fully exploit the DCT representation and its bridge function to connect events and timexes, we propose a DCT-centered Temporal Relation Extraction model DTRE to discover the temporal relations among events, timexes and DCT. Figure 2 illustrates the overview of our DTRE framework. We first combine the sentence-style DCT representation with the original document as the input of the pre-trained model (BERT or RoBERTa) to obtain the mention embeddings of the different types. Then, we build a DCT-Aware Graph DAG for each document to obtain the contextual structural representations of events, timexes, DCT, etc. Furthermore, we conduct a DCT-anchoring Multi-Task Learning framework DAML to jointly predict the three tasks of temporal relation extraction. Finally, we introduce a DCT-guided Global Inference mechanism DGI to our model according to the high accuracies of the E-D and T-D tasks.

3.1 Input Representation and Encoding

Different from most previous studies that only use event mentions as the input, we input a document with the annotated event mentions, timexes and DCT to our DTRE. Formally, the input is a document D consisting of a sentence set $S = \{s_1, \ldots, s_i, \ldots, s_k\}$, a token set $W = \{w_1, \ldots, w_i, \ldots, w_l\}$, an event set $E = \{e_1, \ldots, e_i, \ldots, e_m\}$, a timex set $T = \{t_1, \ldots, t_i, \ldots, t_n\}$, an entity set $V = \{v_1, \ldots, v_i, \ldots, v_p\}$ and a representation of the document creation time $DCT = \{t_{dct}\}$, where *k, l, m, n, p* represent the total number of sentences, tokens, event mentions, timexes and entity mentions in the document *D*, respectively. In this paper, we do not normalize the timexes and use their original values as the example in Figure 2.

Due to the different expression forms and the different amounts of tokens, it is a challenge to identify those E-T and E-D relations directly. Moreover, DCT does not explicitly occur in the document, making it hard to represent its semantics for temporal relation extraction. However, DCT is an anchor to connect those event mentions or timexes in a document-level temporal ordering graph. Hence, how to represent DCT is critical for our DTRE. To address this issue, we propose a novel DCT representation, which uses a generated sentence to express the token-level DCT. In detail, a sentence that contains DCT is used to represent DCT, and three forms (i.e., CreatN, CreatD, and CreatND) are shown in Table 1.

Type	DCT-indicator Sentence
CreatN	The document is **creating** *now*.
CreatD	The document is **creating** by {*date*}.
CreatND	The document is **creating** *now* by {*date*}.

Table 1: Various forms of DCT representation in sentence-style, where {*date*} is the specific DCT of a document.

The purpose of our sentence-style DCT representation is to make DCT have a similar sentence-based expression as events and timexes. Hence, DTRE can use a unified framework to extract the E-E, E-T and E-D relations simultaneously. Specifically, since most event triggers are verbs, we select "creating" to denote the occurrence of document creation. Moreover, timex has two types, i.e., absolute (e.g., "2022.10.10") and relative time (e.g., "yesterday") that are explicitly annotated in documents. Therefore, we also utilize date like "20221010" extracted from the raw corpus as well as "now" to denote DCT's value.

Finally, we insert the DCT-indicator sentence shown in Table 1 at the beginning of each document to form the input (an example is shown in Figure 2). Hence, all temporal mentions, i.e., events, timexes and DCT, explicitly occur in the input document. In this way we can establish the bridge of information interaction on DCT. In the input, we also use *DCT*, E_i and T_i to represent DCT, the *i*-th event mention, and timex, respectively.

Following previous work (Mathur et al., 2021; Liu et al., 2021) and for fair comparison in our evaluation, we also use BERT and RoBERTa as the pre-trained models to encode the input document and obtain the embeddings of the token set $\boldsymbol{H}_W = \{\boldsymbol{h}_{w1}, \ldots, \boldsymbol{h}_{wi}, \ldots, \boldsymbol{h}_{wl}\}$, the event set $\boldsymbol{H}_E = \{\boldsymbol{h}_{e1}, \ldots, \boldsymbol{h}_{ei}, \ldots, \boldsymbol{h}_{em}\}$, the timex set $\boldsymbol{H}_T = \{\boldsymbol{h}_{t1}, \ldots, \boldsymbol{h}_{ti}, \ldots, \boldsymbol{h}_{tn}\}$, the entity set $\boldsymbol{H}_V = \{\boldsymbol{h}_{v1}, \ldots, \boldsymbol{h}_{vi}, \ldots, \boldsymbol{h}_{vp}\}$, the sentence set $\boldsymbol{H}_S = \{\boldsymbol{h}_{s1}, \ldots, \boldsymbol{h}_{si}, \ldots, \boldsymbol{h}_{sk}\}$, the document $\boldsymbol{H}_D = \{\boldsymbol{h}_{[CLS]} | \boldsymbol{h}_{<s>}\}$, and the DCT set $\boldsymbol{H}_{DCT} = \{\boldsymbol{h}_{dct}\}$.

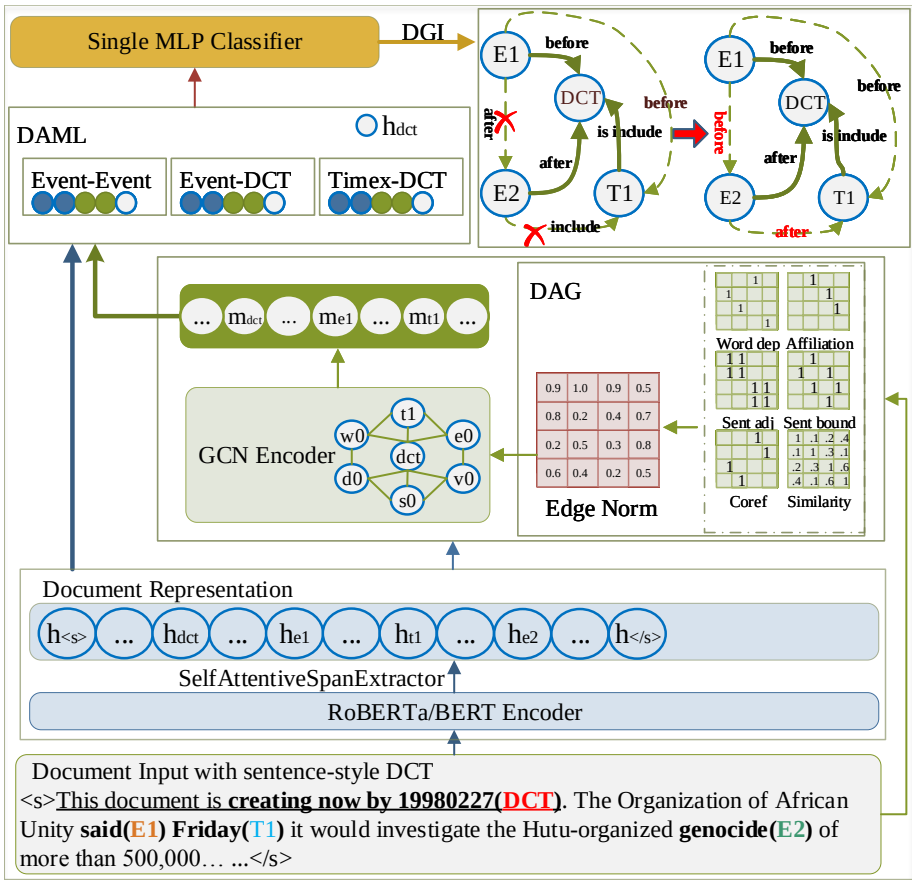

Figure 2: An overview of our proposed DTRE framework.

3.2 DCT-Aware Graph

To capture the structural and interactive information between the different types of temporal mentions, we introduce a DCT-aware graph $DAG = \{N, E\}$ to our model. In this way, we relieve the difference within three tasks to provide rich discourse-level temporal clues.

Our DAG is different from previous GCN models TIMERS (Mathur et al., 2021) and UCGraph (Liu et al., 2021). Specifically, TIMERS constructed three graphs (syntactic, time and rhetorical-aware graph) on events and timexes, while UCGraph built an uncertainty-guided graph on events. Different from their graph, our DAG is a DCT-aware fully-connected graph on events, timexes, and DCT. Moreover, our DAG is simpler than their graphs, because it does not need the edge prediction and optimization.

The node set N can be divided into three levels: token, mention, and discourse, i.e., the token $w_i \in W$, the entity mention $v_j \in V$, the event mention $e_k \in E$, the timex $t_l \in T$, the DCT t_{dct}, the sentence $s_r \in S$, and the document D. We use the embeddings $\boldsymbol{h}_{wi}, \boldsymbol{h}_{ei}, \boldsymbol{h}_{ti}$ and $\boldsymbol{h}_{dct}$ to represent the nodes w_i, e_i, t_i, and DCT, respectively. For nodes consisting of multiple tokens (e.g., entity mentions and sentences), a self-attention mechanism is applied over RoBERTa/BERT embeddings to obtain node representations following (Lee et al., 2017). For the document D, we take $\boldsymbol{h}_{[cls]}$ (BERT) or $\boldsymbol{h}_{<s>}$ (RoBERTa) as its representation. In addition, DAG is composed of six types of edges, i.e., the relations of the affiliation, sentence boundary, word dependency, sentence adjacency, entity coreference, and semantic similarity. We initialize six adjacency matrices ($\boldsymbol{A}^{af}$, $\boldsymbol{A}^{bd}$, $\boldsymbol{A}^{dp}$, $\boldsymbol{A}^{ad}$, $\boldsymbol{A}^{cf}$, $\boldsymbol{A}^{sm}$) to represent them in our graph DAG as follows, where $\boldsymbol{A}^{y} = \{a^{y}_{1,1}, ..., a^{y}_{i,j}, ..., a^{y}_{|N|,|N|}\}$ ($y \in \{af, bd, dp, ad, cf, sm\}$).

Affiliation. A token node w_i connects to its subordinate event/entity/timex node $o_j \in V \cup E \cup T$,and o_j connects to its respective sentence node $s_q \in S$. Each sentence node s_q connects to the document node D. Besides, if the *i*-th node in DAG connects to the *j*-th node, then we set $a^{af}_{i,j} = 1$; otherwise, $a^{af}_{i,j} = 0$, where $a^{af}_{i,j} \in \boldsymbol{A}^{af}$. In this way, we can capture structural information on the word, sentence and document levels.

Sentence Boundary. Entity and event mention pairs that occur in the same sentence usually own a strong relevance. If an event/entity/timex mention pair (o_i, o_j) occur in the same sentence, we set $a^{bd}_{i,j} = 1$; otherwise, $a^{bd}_{i,j} = 0$, where $a^{bd}_{i,j} \in \boldsymbol{A}^{bd}$.

Word Dependency. To encode the syntactic structure, two token nodes w_i and w_j are connected if they share a parent-child relation in a dependency tree, namely we set $a^{dp}_{i,j} = 1$; otherwise, $a^{dp}_{i,j} = 0$, where $a^{dp}_{i,j} \in \boldsymbol{A}^{dp}$.

Sentence Adjacency. If two sentence nodes s_i and s_j are adjacent, then we set $a^{ad}_{i,j} = 1$; otherwise, $a^{ad}_{i,j} = 0$, where $a^{ad}_{i,j} \in \boldsymbol{A}^{ad}$. In this way, the sentence ordering is retained in our DAG.

Entity Coreference. If two entity mention nodes v_i and v_j refer to the same real-world entity, then we set $a^{cf}_{i,j} = 1$; otherwise, $a^{cf}_{i,j} = 0$, where $a^{cf}_{i,j} \in \boldsymbol{A}^{cf}$. This type of edge can help identify the temporal relations between those long-distance event mention pairs.

Semantic Similarity. We compute the cosine similarity $c_{i,j} (0 < c_{i,j} \leq 1)$ between any two nodes n_i and n_j. We set $a^{sm}_{i,j} = c_{i,j}$, where $a^{sm}_{i,j} \in \boldsymbol{A}^{sm}$. In this way, we can capture the rich semantic information among events, timexes and DCT.

The above six matrices are sparse matrices and have the same dimensions. We apply an edge normalization step for the imputation of Graph Convolutional Network (GCN) after generating the above adjacency matrices as follows.

$$\begin{aligned} \boldsymbol{A} = \text{sigmoid}(&\boldsymbol{A}^{af} + \boldsymbol{A}^{bd} + \boldsymbol{A}^{dp} + \\ &\boldsymbol{A}^{ad} + \boldsymbol{A}^{cf} + \boldsymbol{A}^{sm}) \end{aligned} \tag{1}$$

Then the GCN model encodes the original node representations $\boldsymbol{H}_0 = \boldsymbol{H}_W \cup \boldsymbol{H}_V \cup \boldsymbol{H}_E \cup \boldsymbol{H}_T \cup \boldsymbol{H}_S \cup \boldsymbol{H}_D \cup \boldsymbol{H}_{DCT}$ and the adjacency matrix $\boldsymbol{A}$ through G layers as follows.

$$\boldsymbol{H}_I = \text{ReLU}(\boldsymbol{A} \cdot \boldsymbol{H}_{I-1} \cdot \mathbf{W}_I) \tag{2}$$

where $\mathbf{W}_I$ is the weight matrix for the I-th $(0 < I \leq G)$ layer. We denote the GCN out $\boldsymbol{H}_G = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{|N|}\}$.

3.3 DCT-Anchoring Multi-Task Learning

Most previous studies often suffer from data scarcity and overfitting due to the small size of the available datasets and the single E-E task. Cheng et al. (2020) proposed a multi-task learning model SEC that puts three tasks E-E, E-T and E-D into a batch to train, which addresses the issues. However, SEC still suffered from two other issues. The first is the different expressive forms of event, timex and DCT, making it difficult for a unified model to reveal the different types of temporal relations. The second is that they ignore the importance of DCT to connect the events and timexes in a document.

To address the above issues, we propose an efficient DCT-Anchoring Multi-task Learning (DAML) framework to unify the E-E, E-T and E-D tasks, which can enforce the events and timexes to pay more attention to their temporal order related to DCT, considering the highly credible E-D and T-D relations and their transitivity.

Firstly, our sentence-style DCT representation can not only make DCT occur in the document, but also erase the differences in expression among events, timexes and DCT. In this way, DAML can minimize the task distinction of E-E, E-T and E-D, and make them relatively close to each other. Hence, we train a single general classifier for all three tasks. Specifically, not like SEC that takes a fixed batch size, we organize all mention pairs in the same document into a single batch, which helps maintain global consistency in those densely annotated corpora for each prediction via the global relation anchored to DCT.

Thus, we represent each mention $n_i \in N$ by concatenating its original BERT or RoBERTa embedding $\boldsymbol{h}_i \in \boldsymbol{H}_0$ and GCN out $\boldsymbol{m}_i \in \boldsymbol{H}_G$, then the pair representation $\boldsymbol{d}_{i,j}$ for $n_i, n_j \in N$ is as follows.

$$\boldsymbol{d}_{i,j} = \text{concat}([\boldsymbol{h}_i; \boldsymbol{m}_i; \boldsymbol{h}_j; \boldsymbol{m}_j]) \tag{3}$$

Secondly, since DCT is the anchor to connect the relative events and timexes in a document, we incorporate the DCT representation $\boldsymbol{h}_{dct} \in \boldsymbol{H}_0$ into the classifier to enforce the E-E and E-T pairs noticing their temporal orders with DCT as follows.

$$P(r \mid n_i, n_j) = \text{softmax}(\text{MLP}([\boldsymbol{d}_{i,j}; \boldsymbol{h}_{dct}])) \tag{4}$$

where MLP is the single Multi-layer Perceptron classifier for all tasks. Then we calculate the cross-entropy loss for each task as follows.

$$\mathcal{L}_{\mathcal{T}} = -\sum_{n_i, n_j \in \mathcal{T}} \log P(r = r_{(n_i, n_j)} \mid n_i, n_j) \tag{5}$$

where $\mathcal{T} \in \{EE, ET, ED\}$ refers to one of the E-E, E-T and E-D tasks, and $r_{(n_i, n_j)}$ is the golden label for the mention pair (n_i, n_j). Finally, we combine the three task losses as follows, where α and β are trade-off parameters.

$$\mathcal{L} = \mathcal{L}_{EE} + \alpha \cdot \mathcal{L}_{ET} + \beta \cdot \mathcal{L}_{ED} \tag{6}$$

Dataset	E-E	E-T	E-D
TBD	6088	2001	1737
TDD-Man	6150	-	1221
TDD-Auto	38302	-	1221

Table 2: Statistics of the three datasets.

3.4 DCT-Guided Global Inference

Previous work applied different strategies for global consistent predictions, such as ILP constraints (Ning et al., 2017). Recent studies found that ILP constraints can improve consistency, while they maybe generally harm the F1 score (Liu et al., 2021).

In our multi-task framework DTRE, since the performance of E-D are high reliable (e.g., $F1 > 80$ on TBD and $F1 > 90$ on TDD), we propose a DCT-Guided Global Inference mechanism DGI to treat all of the E-D predictions as golden labels and use them to check whether those E-E and E-T instances obey transitivity in document-level. For example, if an event mention *e1* is before *DCT*, and *DCT* is before *e2*, then the label of *(e1, e2)* should be before. Specially, if the predicted label of *(e1, e2)* is vague, then we do not change it through DGI.

4 Experimentation

In this section, we first introduce the datasets and the experimental settings, and then report results on our proposed DTRE and baselines.

4.1 Datasets and Experimental Settings

We evaluate our DTRE on two popular datasets TimeBank-Dense (TBD) (Cassidy et al., 2014) and TDDiscourse (TDD) (Naik et al., 2019). TBD densely annotated 4 TLINKs (E-E, E-D, E-T and T-D) within an adjacent sentence slide window (as DCT does not explicitly occur in texts, each event or Timex has a temporal relation annotation with DCT). TBD has six types of labels, i.e. *before, after, include, is included, simultaneous*, and *vague*. There are 243 T-D instances (2%) in TBD, we do not distinguish T-D with E-D for simplicity in this paper. TDD is a discourse-level temporal ordering corpus and has five types of event temporal relations except for the vague relation in TBD, which makes the class distribution more balanced. TDD consists of two subsets: TDD-Man and TDD-Auto, which are manual and auto annotated, respectively. Since TDD does not annotate E-D relation, we additionally take the E-D examples in TBD (except those vague samples) into the training step, as TDD shares the same documents and event annotations with TBD. Table 2 shows the statistics of the three datasets.

We split the standard train/dev/test sets on TBD and TDD datasets following (Mathur et al., 2021) and report Precision (P), Recall (R), and micro-F1 scores. Since previous studies on TBD treat vague as none type (Liu et al., 2021) or positive type (Cheng et al., 2020) when calculating F1 scores, we report both (five types/six types) for fair comparison.

In DAG, we utilize SpaCy[2] to extract the entities and word dependency trees. Entity coreference resolution is obtained by neuralcoref[3] toolkit. We use cosine_similarity()[4] to obtain the cosine similarity between mentions. We tune all the hyperparameters on the development set. For the pre-trained encoder, we choose BERT-base and RoBERTa-large architecture following previous work for fair comparison. The number of MLP layer is set to 2, and the number of GCN layer is set to 2. The hidden dimension of GCN is set to 768 and 1024 for BERT and RoBERTa, respectively. The trade-off parameters α and β in Eq.6 are both set to 1.0.

4.2 Experimental Results

To evaluate the performance of our model DTRE, we conduct seven strong baselines for comparison as follows:

- **DP-RNN** (Cheng and Miyao, 2017): a model applied event pair's shortest dependency path (SDP) into Bi-LSTM, while it utilizes single event's DP branch for the E-D task;
- **GCL** (Meng and Rumshisky, 2018): a context-aware neural network with a uniform architecture for E-E, E-T and E-D;
- **SEC** (Cheng et al., 2020): a multi-task source event centric model that dynamically managed event representations across three TLINK types;
- **Rand** (Cheng et al., 2020): a multi-task model that RNN module is removed in **SEC**, which

[2] https://spacy.io/

[3] https://github.com/huggingface/neuralcoref/

[4] https://pytorch.org/docs/stable/nn.functional.html

Method	E-E	E-T	E-D
DP-RNN	-/52.9	-/47.1	-/54.6
GCL	-/57.0	-/48.7	-/48.9
SEC*	-/65.0	-/55.8	-/65.9
Rand*	63.0/61.4	60.2/54.8	75.9/65.2
ECONET†	66.8/-	-/-	-/-
TIMERS*	67.8/-	-/-	-/-
BERT*	62.2/59.7	49.4/49.0	73.8/69.4
RoBERTa†	62.4/59.8	51.5/49.8	76.3/72.1
DTRE*	69.2/68.4	64.9/62.1	77.7/73.6
DTRE†	**72.3/70.2**	**70.6/67.5**	**81.9/75.8**

Table 3: F1-score comparison of E-E, E-T and E-D on TBD. The figures before and after "/" refer to the results on five and six types, where "*" and "†" refer to the encoder BERT and RoBERTa, respectively.

uses randomly initialized learnable embeddings to represent DCT.

- **BERT**-based (Devlin et al., 2019) and **RoBERTa**-based (Liu et al., 2019b) Transformer: the models follow (Zhao et al., 2021) to conduct a pair-wise classification in which the E-D task utilizes our DCT representation;
- **UCGraph** (Liu et al., 2021): the first work introducing graph representation learning and uncertainty modeling to temporal relation extraction;
- **TIMERS** (Mathur et al., 2021): a graph-based method on the E-E task that merges syntactic, temporal, and rhetorical information;
- **ECONET** (Han et al., 2021): a pre-trained method on the E-E task using millions of raw temporal relative data.

Table 3 and Table 4 show the performance comparison of our model DTRE and the baselines on the datasets TBD, TDD-Man and TDD-Auto, respectively. It can be observed that our model DTRE outperforms all baselines on the three datasets significantly (significance test with $p < 0.05$).

Compared with the SOTA models TIMERS (E-E) and SEC (E-T and E-D) on TBD, DTRE improves the F1-score by 4.5, 11.7, and 9.9 on the three tasks E-E, E-T and E-D, respectively. Compared with the SOTA model TIMERS on TDD-Auto and TDD-Man, DTRE gains improvements (E-E) of 10.7 and 10.8 on F1 score, respectively.

	TDD-Man			TDD-Auto		
Method	P	R	F1	P	R	F1
UCGraph*	44.5	42.3	43.4	66.1	56.9	61.2
TIMERS*	43.7	46.7	45.5	64.3	72.7	71.1
BERT*	39.9	39.9	39.9	62.3	62.3	62.3
RoBERTa†	44.8	44.8	44.8	76.7	76.7	76.7
DTRE*	50.0	50.0	50.0	70.2	70.2	70.2
DTRE†	**56.3**	**56.3**	**56.3**	**81.8**	**81.8**	**81.8**

Table 4: Performance comparison of E-E on TDD-Man and TDD-Auto, where "*" and "†" refer to the encoder BERT and RoBERTa, respectively.

These results verify the effectiveness of our DTRE on extracting all kinds of temporal relations. Besides, compared with E-E of our DTRE in Table 3, E-T and E-D gain much higher improvements. This result further indicates that timexes and DCT are the critical clues for temporal relation extraction and our DCT representation is effective in DTRE.

In Table 3, the pre-trained models BERT and RoBERTa achieve similar performance on TBD, while RoBERTa outperforms BERT rapidly on both TDD-Man and TDD-Auto in Table 4. These results indicate that RoBERTa is better than BERT as encoder to extract the temporal relations among inter-sentence event mentions and RoBERTa works well on a large-scale training set (e.g., TDD-Auto).

5 Ablation Study

In this section, we conduct the ablation study of DTRE (RoBERTa-based) on TBD as examples. It is worth mentioning that BERT-based DTRE on TBD and TDD also show the similar results and we do not describe here for simplification.

5.1 Impacts of Different Modules

To verify the effectiveness of each module in DTRE, we conduct the experiments on the following variants and baseline:

- *w/o DAG*: we remove DAG and only use the original RoBERTa embeddings in pair representation;
- *w/o DAML*: we remove DAML and separately train each task;
- *w/o DGI*: we remove DGI from DTRE. The results are shown in Table 5.

Method	E-E	E-T	E-D
DTRE (RoBERTa)	72.3	70.6	81.8
w/o DAG	-3.5	-3.6	-1.6
w/o DAML	-5.4	-15.8	-2.5
w/o DGI	-0.6	-2.0	-

Table 5: F1 scores of DTRE and its variants on TBD.

Variant	E-E	E-T	E-D
Rand	63.0*	60.2*	75.9*
CreatN (*w/o* DCT_{feat})	64.2	62.3	74.3
CreatD (*w/o* DCT_{feat})	66.0	62.4	75.4
CreatND (*w/o* DCT_{feat})	66.3	63.7	76.8
CreatN + DCT_{feat}	65.8	64.5	76.3
CreatD + DCT_{feat}	66.5	65.2	78.1
CreatND+ DCT_{feat}	67.1	66.0	79.3

Table 6: Effects of different representation of DCT and the DCT-aware feature, where "+DCT_{feat}" denotes that adding $\boldsymbol{h}_{dct}$ in pair representation mentioned in Eq.4 and "*" denotes our re-implementations on five temporal types without the *vague* relation.

Resource	E-E	E-T	E-D
Single task	66.9	54.8	79.3
E-E&E-T	68.4	66.8	-
E-E&E-D	70.8	-	80.4
E-E&E-T&E-D	71.7	68.6	81.9

Table 7: Effect of training resources on TBD, where DGI is removed for fair comparison.

The results of DTRE and its variants on TBD are showed in Table 5. When we remove DAG (*w/o DAG*), the F1 scores of the E-E, E-T and E-D tasks decrease by 3.5, 3.6, and 1.6, respectively. This indicates the importance of the document structure for temporal reasoning, especially for the E-E and E-T tasks.

Removing DAML (*w/o DAML*) leads to the biggest performance deterioration for E-E, E-T and E-D by 5.4, 15.8, and 2.5, respectively. This indicates that the three tasks can complement each other in a unified framework. The significant improvements of E-E and E-T also show that utilizing DCT to anchor events and timexes is an effective way for temporal relation extraction, which can be regarded as a bridge to link event pair or event-timex pair.

Moreover, E-E and E-T tasks benefit from DCT-guided global inference (*w/o DGI*) with the gains of 0.6 and 2.0, although the transitivity is harmed by *vague* relation to some extent, which verifies that the E-D task can provide direct temporal clues for E-E and E-T.

5.2 Impacts of DCT Representations

Obviously, DCT can often provide explicit temporal information and be a bridge to link those events without temporal clues. To further analyze the impacts of the different DCT representations and our DCT-aware feature $\boldsymbol{h}_{dct}$ in pair representation (Eq.4), we adopt several DCT representation strategies in Table 1 and the results are shown in Table 6. Specifically, to compare with the existing multi-task learning model *Rand* (Cheng et al., 2020), we remove DAG and DGI from our DTRE, and only conduct the resource-shared multi-task learning with BERT-base settings for direct comparison.

As showed in Table 6, compared with *Rand*, our three strategies (*w/o* DCT_{feat}) improve the F1 scores of E-E and E-T significantly. Their randomly initialized learnable embeddings do not explicitly contain any DCT information, while our representation uses a sentence to let DCT explicitly occur in the document. This result indicates that DCT is an important hub to connect the events and timexes scattered in a document. Although our DCT-indicator sentences are simple, it also shows that all three strategies are effective, especially **CreatND** with the highest improvement.

We also introduce our DCT-aware feature $\boldsymbol{h}_{dct}$ to our model and the results in Table 6 indicate that it can boost all three tasks, especially E-D and E-T. In this way, we enforce the classifier to pay more attention to the related E-D relations when predicting E-E and E-T pairs and then can gain the improvement for all tasks.

5.3 Impacts of DAML

We also evaluate the impacts of DAML on different tasks. Intuitively, we remove one task from our DTRE (RoBERTa) and the results on TBD are shown in Table 7. It shows that the E-T and E-D tasks play an important role in our DTRE framework, which contributes the performance gains of 1.5 and 3.9 for E-E. Moreover, although the sample size of E-T is larger than that of E-D (2001

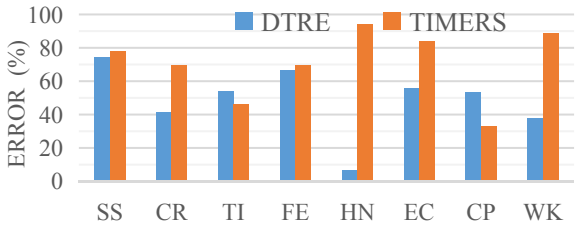

Figure 3: Error analysis on manually annotated phenomena in the test set of TDD-Man. SS: SingleSent, CR: Chain Reasoning, TI: Tense Indicator, FE: Future Events, HN: Hypothetical/ Negated, EC: Event Coreference, CP: Causal/ Prereq, WK: World Knowledge (detail definitions please refer to Naik et al. (2019))

vs 1494), E-D is better than E-T as the auxiliary task of E-E. This result indicates that E-D is more effective than E-T for this multi-task framework and verifies the core role of DCT in temporal relation extraction. Besides, with the help of E-E, E-T can significantly improve the F1-score by 12.0 and the reason is that the number of E-E instances are larger than that of E-T.

5.4 Error Analysis

To analyze the errors in our DTRE, we use the annotated cues (Naik et al., 2019) between events in TDD-Man and compare them with the SOTA model TIMERS (Mathur et al., 2021). Figure 3 shows the error percentages of eight cue types on TDD-Man. We can find out that our DTRE deals well with HN (Hypothetical/Negated), WK (World Knowledge) and CR (Chain Reasoning), while it suffers from SS (SingleSent) and FE (Future Events). The reason behind this is that SS and FE need more event-level semantics to predict temporal relation while our DTRE only focuses on using the novel DCT representation and the intrinsic relations among event, timex and DCT to boost all three temporal relation extraction tasks.

TIMERS suffers from TLINK pairs which depend on CR, HN, EC (Event Coreference) and WK, while our DTRE achieves significant progress on them. This result indicates that the document-level knowledge (e.g., DCT) is a core clue for temporal relation extraction and our DCT-anchoring multi-task framework regards the whole document as the input and can incorporate the document-level knowledge. However, TIMERS is better to deal with TI (Tense Indicator) and CP (Causal / Prereq), because it focuses on mining more semantic information inside E-E relations.

As for the errors in E-T and E-D, most of them come from two aspects: 1) there are no explicit temporal words or clues in events, and 2) some timexes do not express a specific duration or time point (e.g., "recently" and "a few years ago").

6 Conclusion

In this paper, we proposed a unified DCT-centered temporal relation extraction model DTRE to discover the relations among events, timexes and DCT. Specifically, we first introduce sentence-style DCT to unify the expressions of event, timex and DCT. Then, we apply a DCT-aware graph to obtain their contextual structural representations. Furthermore, we propose a DCT-anchoring multi-task framework to jointly predict three tasks of temporal relation extraction in a batch. Finally, we provide a DCT-guided global inference to further enhance the global consistency among different relations. Experimental results on three popular datasets show that our DTRE outperforms several SOTA baselines on E-E, E-T and E-D significantly. Our future work will focus on discovering effective graph structure and inference mechanism for temporal relation extraction.

Acknowledgements

The authors would like to thank the three anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (Nos. 61836007, 62276177 and 62006167), and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

Miguel Ballesteros, Rishita Anubhai, Shuai Wang, Nima Pourdamghani, Yogarshi Vyas, Jie Ma, Parminder Bhatia, Kathleen McKeown, and Yaser Al-Onaizan. 2020. Severing the edge between before and after: Neural architectures for temporal ordering of events. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5412–5417.

Taylor Cassidy, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506.

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.

Nathanael Chambers and Daniel Jurafsky. 2008. Jointly combining implicit constraints improves temporal ordering. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 698–706.

Nathanael Chambers, Shan Wang, and Dan Jurafsky. 2007. Classifying temporal relations between events. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 173–176.

Fei Cheng, Masayuki Asahara, Ichiro Kobayashi, and Sadao Kurohashi. 2020. Dynamically updating event representations for temporal relation classification with multi-category learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1352–1357.

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional LSTM over dependency paths. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–6.

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.

Quang Do, Wei Lu, and Dan Roth. 2012. Joint inference for event timeline construction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 677–687.

Jennifer D'Souza and Vincent Ng. 2013. Classifying temporal relations with rich linguistic knowledge. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 918–927.

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)*, pages 434–444.

Rujun Han, Xiang Ren, and Nanyun Peng. 2021. ECONET: Effective continual pretraining of language models for event temporal reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5367–5380.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197.

Manling Li, Tengfei Ma, Mo Yu, Lingfei Wu, Tian Gao, Heng Ji, and Kathleen McKeown. 2021. Timeline summarization based on event graph compression via time-aware optimal transport. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6443–6456.

Jian Liu, Jinan Xu, Yufeng Chen, and Yujie Zhang. 2021. Discourse-level event temporal ordering with uncertainty-guided graph completion. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 3871–3877.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv*, abs/1907.11692.

Puneet Mathur, Rajiv Jain, Franck Dernoncourt, Vlad Morariu, Quan Hung Tran, and Dinesh Manocha. 2021. TIMERS: Document-level temporal relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 524–533.

Yuanliang Meng and Anna Rumshisky. 2018. Context-aware neural model for temporal information extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 527–536.

Paramita Mirza and Sara Tonelli. 2016. CATENA: CAusal and TEmporal relation extraction from NAtural language texts. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 64–75.

Aakanksha Naik, Luke Breitfeller, and Carolyn Rose. 2019. TDDiscourse: A dataset for discourse-level temporal ordering of events. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 239–249.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037.

Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020. TORQUE: A reading comprehension dataset of temporal ordering questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1158–1172.

Qiang Ning, Hao Wu, and Dan Roth. 2018. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1318–1328.

Yunseok Noh, Yongmin Shin, Junmo Park, A-Yeong Kim, Su Jeong Choi, Hyun-Je Song, Seong-Bae Park, and Seyoung Park. 2020. Wire: An automated report generation system using topical and temporal summarization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2169–2172.

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, et al. 2003. The timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

Xinyu Zhao, Shih-Ting Lin, and Greg Durrett. 2021. Effective distant supervision for temporal relation extraction. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 195–203.