Deep Structured Neural Network for Event Temporal Relation Extraction

Rujun Han[*]**, I-Hung Hsu**[*]**, Mu Yang**
{rujunhan, ihunghsu, yangmu} @isi.edu
Aram Galstyan, Ralph Weischedel, Nanyun Peng
{galstyan, weisched, npeng} @isi.edu
Information Sciences Institute, University of Southern California

Abstract

We propose a novel deep structured learning framework for event temporal relation extraction. The model consists of 1) a recurrent neural network (RNN) to learn scoring functions for pair-wise relations, and 2) a structured support vector machine (SSVM) to make joint predictions. The neural network automatically learns representations that account for long-term contexts to provide robust features for the structured model, while the SSVM incorporates domain knowledge such as transitive closure of temporal relations as constraints to make better globally consistent decisions. By jointly training the two components, our model combines the benefits of both data-driven learning and knowledge exploitation. Experimental results on three high-quality event temporal relation datasets (TCR, MATRES, and TB-Dense) demonstrate that incorporated with pre-trained contextualized embeddings, the proposed model achieves significantly better performances than the state-of-the-art methods on all three datasets. We also provide thorough ablation studies to investigate our model.

1 Introduction

Event temporal relation extraction aims at building a graph where nodes correspond to events within a given text, and edges reflect temporal relations between the events. Figure 1a illustrates an example of such graph for the text shown above. Different types of edges specify different temporal relations: the event **filed** is *SIMULTANEOUS* with **claiming**, **overruled** is *BEFORE* **claiming**, and **overruled** is also *BEFORE* **filed**. Temporal relation extraction is beneficial for many downstream tasks such as question answering, information retrieval, and natural language generation. An event graph can po-

[*] The authors contribute equally, alphabetical order.

*Both U.S. and British officials **filed** objections to the court's jurisdiction in 1995, **claiming** Security Council resolutions **imposed** on Lybia to force the suspects' extradition **overruled** a 1971 Convention which gives Libya the right to try the men.*

(a) Ground-Truth Sub-graph

(b) Local Model Predictions
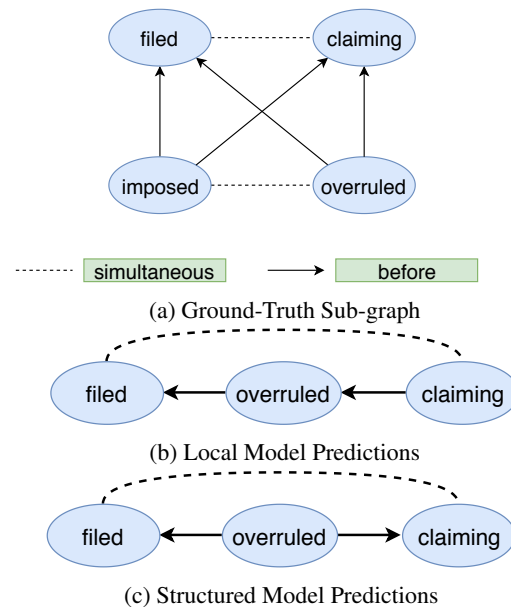

(c) Structured Model Predictions

Figure 1: An illustration of a paragraph with its partial temporal graph. (a) shows the Ground-Truth temporal graph, in which case temporal relation *SIMULTANEOUS* and *BEFORE* are presented. (b) and (c) are the local and structured predictions, respectively, for three of the event pairs in (a). Local predictions are incompatible with temporal transitivity rule: **overruled** has to be *BEFORE* **claiming** if **overruled** is *BEFORE* **filed** and **filed** is *SIMULTANEOUS* with **claiming**. The structured model achieves coherence by reversing the temporal order between **claiming** and **overruled**.

tentially be leveraged to help time-series forecasting and provide guidances for natural language generation. The CaTeRs dataset (Mostafazadeh et al., 2016) which annotates temporal and causal relations is constructed for this purpose.

A major challenge in temporal relation extraction stems from its nature of being a *structured*

prediction problem. Although a relation graph can be decomposed into individual relations on each event pair, any *local* model that is not informed by the whole event graph will usually fail to make globally consistent predictions, thus degrading the overall performance. Figure 1b gives an example where the *local model* classifies the relation between **overruled** and **claiming** incorrectly as it only considers pairwise predictions: graph temporal transitivity constraint is violated given the relation between **filed** and **claiming** is *SIMULTANE-OUS*. In Figure 1c, the *structured model* changes the prediction of relation between **overruled** and **claiming** from *AFTER* to *BEFORE* to ensure compatibility of all predicted edge types.

Prior works on event temporal relation extraction mostly formulate it as a pairwise classification problem (Bethard, 2013; Laokulrat et al., 2013; Chambers, 2013; Chambers et al., 2014) disregarding the global structures. Bramsen et al. (2006a); Chambers and Jurafsky (2008); Do et al. (2012); Ning et al. (2018b,a) explore leveraging global *inference* to ensure consistency for all pairwise predictions. There are a few prior works that directly model global structure in the training process (Yoshikawa et al., 2009; Ning et al., 2017; Leeuwenberg and Moens, 2017). However, these structured models rely on hand-crafted features using linguistic rules and local-context, which cannot adequately capture potential long-term dependencies between events. In the example shown in Figure 1, **filed** occurs in much earlier context than **overruled**. Thus, incorporating long-term contextual information can be critical for correctly predicting temporal relations.

In this paper, we propose a novel deep structured learning model to address the shortcomings of the previous methods. Specifically, we adapt the structured support vector machine (SSVM) (Finley and Joachims, 2008) to incorporate linguistic constraints and domain knowledge for making joint predictions on events temporal relations. Furthermore, we augment this framework with recurrent neural networks (RNNs) to learn long-term contexts. Despite the recent success of employing neural network models for event temporal relation extraction (Tourille et al., 2017a; Cheng and Miyao, 2017; Meng et al., 2017; Meng and Rumshisky, 2018), these systems make pairwise predictions, and do not take advantage of problem structures.

We develop a joint end-to-end *training* scheme that enables the feedback from global structure to directly guide neural networks to learn representations, and hence allows our deep structured model to combine the benefits of both data-driven learning and knowledge exploitation. In the ablation study, we further demonstrate the importance of each global constraints, the influence of linguistic features, as well as the usage of contextualized word representations in the local model.

To summarize, our main contributions are:

- We propose a deep SSVM model for event temporal relation extraction.

- We show strong empirical results and establish new state-of-the-art for three event relation benchmark datasets.

- Extensive ablation studies and thorough error analysis are conducted to understand the capacity and limitations of the proposed model, which provide insights for future research on temporal relation extraction.

2 Related Work

Temporal Relation Data. Temporal relation corpora such as TimeBank (Pustejovsky et al., 2003) and RED (O'Gorman et al., 2016) facilitate the research in temporal relation extraction. The common issue in these corpora is missing annotation. Collecting densely annotated temporal relation corpora with all event pairs fully annotated has been reported to be a challenging task as annotators could easily overlook some pairs (Cassidy et al., 2014; Bethard et al., 2007; Chambers et al., 2014). TB-Dense dataset mitigates this issue by forcing annotators to examine all pairs of events within the same or neighboring sentences. Recent data construction efforts such as MATRES (Ning et al., 2018a) and TCR (Ning et al., 2018b) further enhance the data quality by using a multi-axis annotation scheme and adopting start-point of events to improve inter-annotator agreements. However, densely annotated datasets are relatively small both in terms of number of documents and event pairs, which restricts the complexity of machine learning models used in previous research.

Event Temporal Relation Extraction The series of TempEval competitions (Verhagen et al., 2007, 2010; UzZaman et al., 2013) attract many research interests in predicting event temporal relations. Early attempts (Mani et al., 2006; Verha-

gen et al., 2007; Chambers et al., 2007; Verhagen and Pustejovsky, 2008) only use local pairwise classification with hand-engineered features. Later efforts, such as ClearTK (Bethard, 2013), UTTime (Laokulrat et al., 2013), and NavyTime (Chambers, 2013) improve earlier work by feature engineering with linguistic and syntactic rules. A noteworthy work, CAEVO (Chambers et al., 2014), builds a pipeline with ordered sieves. Each sieve is either a rule-based classifier or a machine learning model; sieves are sorted by precision, i.e. decisions from a lower precision classifier cannot contradict those from a higher precision model.

More recently, neural network-based methods have been employed for event temporal relation extraction (Tourille et al., 2017a; Cheng and Miyao, 2017; Meng et al., 2017; Han et al., 2019a) which achieved impressive results. However, they all treat the task as a pairwise classification problem. Meng and Rumshisky (2018) considered incorporating global context for pairwise relation predictions, but they do not explicitly model the *output* graph structure for event temporal relation.

There are a few prior works exploring structured learning for temporal relation extraction (Yoshikawa et al., 2009; Ning et al., 2017; Leeuwenberg and Moens, 2017). However, their local models use hand-engineered linguistic features. Despite the effectiveness of hand-crafted features in previous research, the design of features usually fails to capture long-term context in the discourse. Therefore, we propose to enhance the hand-crafted features with contextual representations learned through RNN models and develop an integrated joint training process.

3 Methods

We adapt the notations from Ning et al. (2018a), where $\mathcal{R}$ denotes the set of all possible relations; $\mathcal{E}$ denotes the set of all event entities.

3.1 Deep SSVM
Our deep SSVM model adapts the SSVM loss as

$$\mathcal{L} = \sum_{n=1}^{l} \frac{1}{M^n} \big[\max \big(0, \Delta(\boldsymbol{y}^n, \hat{\boldsymbol{y}}^n) + \quad (1)$$
$$S(\hat{\boldsymbol{y}}^n; \boldsymbol{x}^n) - S(\boldsymbol{y}^n; \boldsymbol{x}^n) \big) \big] + C||\Phi||^2,$$

where Φ denotes model parameters, n indexes instances, M^n is the number of event pairs in instance n. $\boldsymbol{y}^n, \hat{\boldsymbol{y}}^n$ denote the gold and predicted global assignments for instance n, each of
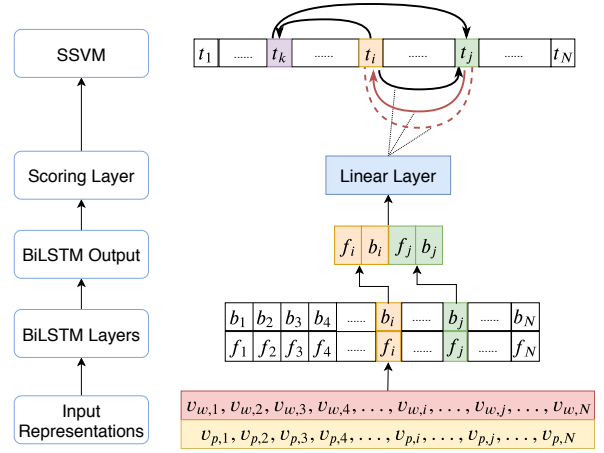

Figure 2: An overview of the proposed deep structured event relation extraction framework. The input representations consist of BERT representations ($v_{w,k}$) and POS tag embeddings ($v_{p,k}$). They are concatenated to pass through BiLSTM layers and classification layers to get pairwise local scores. Incompatible local pairwise prediction (denoted by red lines) is corrected by the SSVM layer. Edge notation follows Figure 1 and $t_1, ...t_N$ denote tokens in the input sentence.

which consists of M^n one hot vectors $\boldsymbol{y}^n_{i,j}, \hat{\boldsymbol{y}}^n_{i,j} \in \{0,1\}^{|\mathcal{R}|}$ representing true and predicted relation labels for event pair i, j respectively. $\Delta(\boldsymbol{y}^n, \hat{\boldsymbol{y}}^n)$ is a distance measurement between the gold and the predicted assignments; we simply use hamming distance. C is a hyper-parameter to balance the loss and the regularizer, and $S(\boldsymbol{y}^n; \boldsymbol{x}^n)$ is a pairwise scoring function to be learned.

The intuition behind the SSVM loss is that it requires the score of gold output structure $\boldsymbol{y}^n$ to be greater than the score of the best output structure under the current model $\hat{\boldsymbol{y}}^n$ with a margin $\Delta(\boldsymbol{y}^n, \hat{\boldsymbol{y}}^n)$[1], or else there will be some loss.

The major difference between our deep SSVM and the traditional SSVM model is the scoring function. Traditional SSVM uses a linear function over hand-crafted features to compute the scores, whereas we propose to use a RNN for estimation.

3.2 RNN-Based Scoring Function

We introduce a RNN-based pair-wise scoring function to learn features in a data-driven way and capture long-term context in the input. The local neural architecture is inspired by prior work in entity relation extraction such as Tourille et al. (2017b). As shown in Figure 2, the input layer consists of word representations and part-of-

---
[1]Note that if the best prediction is the same as the gold structure, the margin is zero.

speech (POS) tag embeddings of each token in the input sentence, denoted as $v_{w,k}$ and $v_{p,k}$ respectively.[2] The word representations are obtained via pre-trained BERT (Devlin et al., 2018)[3] model and are fixed throughout training, while the POS tag embeddings are tuned. The word and POS tag embeddings are concatenated to represent an input token, and then fed into a Bi-LSTM layer to get contextualized representations.

We assume the events are labeled in the text and use indices i and j to denote the tokens associated with an event pair $(i, j) \in \mathcal{EE}$ in the input sentences of length N. For each event pair (i, j), we take the forward and backward hidden vectors corresponding to each event, namely f_i, b_i, f_j, b_j to encode the event tokens. These hidden vectors are then concatenated to form the input to the final linear layer to produce a softmax distribution over all possible pair-wise relations, which we refer to as RNN-based **scoring function**.

3.3 Inference

The inference is needed both during training to obtain $\hat{y}^n$ in the loss function (Equation 1), as well as during the test time to get globally compatible assignments. The inference framework is established by constructing a global objective function using scores from local model and imposing several global constraints: symmetry and transitivity as in Bramsen et al. (2006b); Chambers and Jurafsky (2008); Denis and Muller (2011); Do et al. (2012); Ning et al. (2017); Han et al. (2019b), as well as linguistic rules and temporal-causal constraints proposed by Ning et al. (2018a) to ensure global consistency. In this work, we incorporate the **symmetry**, **transitivity**, and **temporal-causal** constraints.

Objective Function. The objective function of the global inference maximizes the score of global assignments as specified in Equation 2[4].

$$\hat{y} = \arg\max \sum_{(i,j)\in\mathcal{EE}} \sum_{r\in\mathcal{R}} y_{i,j}^r S(y_{i,j}^r; \boldsymbol{x}) \quad (2)$$

$$\text{s.t. } y_{i,j}^r \in \{0, 1\}, \sum_{r\in\mathcal{R}} y_{i,j}^r = 1,$$

---

[2]Following the convention of event relation prediction literature (Chambers et al., 2014; Ning et al., 2018a,b), we only consider event pairs that occur in the same or neighboring sentences, but the architecture can be easily adapted to the case where inputs are longer than two sentences.

[3]We use pre-trained bert-base-uncased model from https://github.com/huggingface/pytorch-transformers.

[4]The objective function is specified on the instance level.

where $y_{i,j}^r$ is a binary indicator specifying if the global prediction is equal to a certain label $r \in \mathcal{R}$ and $S(y_{i,j}^r, \boldsymbol{x}), \forall r \in \mathcal{R}$ is the scoring function obtained from the local model. The output of the global inference $\hat{y}$ is a collection of optimal label assignments for all event pairs in a fixed context. The constraint following immediately from the objective function is that the global inference should only assign one label to each pair of sample inputs.

Symmetry and Transitivity constraint. The symmetry and transitivity constraints are used across all models and experiments in the paper. They can be specified as follows:

$$\forall(i,j), (j,k) \in \mathcal{EE}, y_{i,j}^r = y_{j,i}^{\bar{r}}, \text{ (symmetry)}$$

$$y_{i,j}^{r_1} + y_{j,k}^{r_2} - \sum_{r_3\in Trans(r_1,r_2)} y_{i,k}^{r_3} \leq 1. \text{ (transitivity)}$$

Intuitively, the symmetry constraint forces two pairs with opposite order to have reversed relations. For example, if $r_{i,j}$ = *BEFORE*, then $r_{j,i}$ = *AFTER*. Transitivity constraint rules that if (i, j), (j, k) and (i, k) pairs exist in the graph, the label (relation) prediction of (i, k) pair has to fall into the transitivity set specifying by (i, j) and (j, k) pairs. The full transitivity table can be found in Ning et al. (2018a).

Temporal-causal Constraint. The temporal-causal constraint is used for the TCR dataset which is the only dataset in our experiments that contains causal pairs and it can written as:

$$y_{i,j}^c = y_{j,i}^{\bar{c}} \leq y_{i,j}^b, \forall(i,j) \in \mathcal{EE},$$

where c and $\bar{c}$ correspond to the label *CAUSES* and *CAUSED_BY*, and b represents the label *BEFORE*. This constraint specifies that if event i causes event j, then i has to occur before j. Note that this constraint only has 91.9% accuracy in TCR data (Ning et al., 2018a), but it can help improve model performance based on our experiments.

3.4 Learning

We develop a two-state learning approach to optimize the neural SSVM. We first train the local scoring function without feedback from global constraints. In other words, the local neural network model is optimized using only pair-wise relations in the first stage by minimizing cross-entropy loss. During the second stage, we switch to the global objective function in Equation 1 and

	TB-Dense	MATRES	TCR
# of Documents			
Train	22	22	20
Dev	5	5	0
Test	9	9	5
# of Pairs			
Train	4032	1098	1992
Dev	629	229	0
Test	1427	310	1008

Table 1: Data Overview

	TB-Dense	MATRES	TCR
Local Model			
hid_size	60	40	30
dropout	0.5	0.7	0.5
BiLSTM layers	1	2	1
learning rate	0.002	0.002	0.002
Structured Learning			
learning rate	0.05	0.08	0.08
decay	0.7	0.7	0.9

Table 2: Best hyper-parameters

re-optimize the network to adjust for global properties[5]. We denote the local scoring model in the first stage as local model, and the final model as global model in the following sections.

4 Experimental Setup

In this section, we describe the three datasets that are used in the paper. Then we define the evaluation metrics. Finally, we provide details regarding our model implementation and experiments.

4.1 Data

Experiments are conducted on TB-Dense, MATRES and TCR datasets and an overview of data statistics are shown in Table 1. We focus on event relation, thus, all numbers refer to $\mathcal{EE}$ pairs[6]. Note that in all three datasets, event pairs are always annotated by their appearance order in text, i.e. given a labeled event pair (i, j), event i always appears prior to event j in the text. Following Meng et al. (2017), we augment event pairs with flipped-order pairs. That is, if a pair (i, j) exists, pair (j, i) is also added to our dataset with the opposite label. The augmentation is applied to training and development split, but test set remains unaugmented[7].

TB-Dense (Cassidy et al., 2014) is based on TimeBank Corpus but addresses the sparse-annotation issue in the original data by introducing the *VAGUE* label and requiring annotators to label all pairs of events/times in a given window.

MATRES (Ning et al., 2018b) is based on TB-Dense data, but filters out non-verbal events. The authors project events on multiple axes and only keep those in the main-axis. These two factors explain the large decrease of event pairs in Table 1.

Start-point temporal scheme is adopted when outsourcing the annotation task, which contributes to the performance improvement of machine learning models built on this dataset .

TCR (Ning et al., 2018a) follows the same annotation scheme for temporal pairs in MATRES. It is also annotated with causal pairs. To get causal pairs, the authors select candidates based on EventCausality dataset (Do et al., 2011).

4.2 Evaluation Metrics

To be consistent with the evaluation metrics used in baseline models, we adopt two slightly different calculations of metrics.

Micro-average For all datasets, we compute the micro-average scores. For densely annotated data, the micro-average metric should share the same precision, recall and F1 scores. However, since VAGUE pairs are excluded in the micro-average calculations of TCR and MATRES for fair comparisons with the baseline models, the micro-average for precision, recall and F1 scores are different when reporting results for the two datasets.

Temporal Awareness (TE3) For TB-Dense dataset, TE3 evaluation scheme (UzZaman et al., 2013) is also adopted in previous research (Ning et al., 2017, 2018a). TE3 score not only takes into account of the number of correct pairs but also capture how "useful" a temporal graph is. We report this score for TB-Dense results only. For more details of this metric, please refer to the original paper (UzZaman et al., 2013).

4.3 Implementation Details

Since our work focuses on event relations, we build our models to predict relations between $\mathcal{EE}$ pairs only when conducting experiments. Thus, all micro-average F1 scores only consider $\mathcal{EE}$ pairs. Note that there are also time entities labeled in the TB-Dense denoted as $\mathcal{T}$. $\mathcal{ET}$ and $\mathcal{TT}$ pairs are

---

[5]We experiment with optimizing SSVM loss directly, but model performance degrades significantly. We leave further investigation for future work.

[6]For TCR, we also include causal pairs in the table.

[7]It is noted that if symmetric constraint is applied, scores for testing on augmented or unaugmented set are equal.
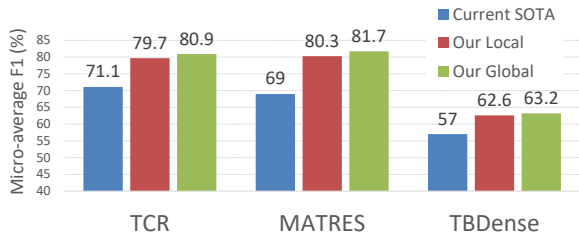
Figure 3: Model Performance (F1 Score) Overview. Our local and global models' performances are averaged over 3 different random seeds for robustness.

generally easier to predict using rule-based classifiers or date normalization technique (Do et al., 2012; Chambers et al., 2014; Mirza and Tonelli, 2016). To be consistent with the baseline models (Ning et al., 2018a,b) for TB-Dense data, we add $\mathcal{ET}$ and $\mathcal{TT}$ pairs for TE3 evaluation metric[8].

In the two-stage learning procedure, the local model is trained by minimizing cross-entropy loss with Adam optimizer. We use pre-trained BERT embedding with 768 dimensions as the input word representations and one-layer MLP as the classification layer. As for the structured learning stage, we observe performance boost by switching from Adam optimizer to SGD optimizer with decay and momentum[9]. To solve ILP in the inference process specified in Section 3.3, we leverage off-the-shelf solver provided by Gurobi optimizer, i.e. the best solutions from the Gurobi optimizer are inputs to the global training.

The hyper-parameters are chosen by the performance on the development set[10], and the best combination of hyper-parameters can be found in Table 2. We run experiments on 3 different random seeds and report the average results.

Note that for TCR data, we need a separate classifier for causal relations. Because of small amount of causal pairs, we simply build an independent final linear layer apart from the original linear layer in Figure 2. In other words, there are two final linear layers: only one of them is active when training temporal or causal pairs.

5   Results and Analysis

Figure 3 shows an overview of our model performance on three different datasets. As the chart

---

[8]We rely on annotated data to distinguish different pair types, i.e. $\mathcal{EE}$, $\mathcal{ET}$ and $\mathcal{TT}$ are assumed to be given.

[9]The weight decay in SGD is exactly the value C in Equation 1. We set the momentum in SGD as 0.9 in all datasets.

[10]We randomly select 4 documents from the training set as development set for TCR.

	Local Model			Global Model		
	P	R	F1	P	R	F1
Before	82.1	86.9	84.3	81.3	90.0	85.4
After	67.1	73.2	69.7	70.9	70.9	70.9
Simultaneous	0.0	0.0	0.0	0.0	0.0	0.0
Vague	0.0	0.0	0.0	0.0	0.0	0.0
Micro-average	77.1	82.5	79.7	78.2	83.9	**80.9****
Ning et al. (2018a)						71.1

Table 3: Model Performance Breakdown for TCR. To make fair comparison, we exclude VAGUE pairs in Micro-average score, which is why P, R and F1 are different. **indicates global model outperforms local model with p-value < 0.01 per McNemar's test.

	Local Model			Global Model		
	P	R	F1	P	R	F1
Before	79.7	88.1	83.6	80.1	89.6	84.6
After	70.5	83.3	76.3	72.3	84.8	78.0
Simultaneous	0.0	0.0	0.0	0.0	0.0	0.0
Vague	0.0	0.0	0.0	0.0	0.0	0.0
Micro-average	76.2	84.9	80.3	77.4	86.4	**81.7***
Ning et al. (2018b)						69

Table 4: Model Performance Breakdown for MATRES. Again, we exclude VAGUE pairs in Micro-average score. * indicates global model outperforms local model with p-value < 0.05 per McNemar's test.

illustrates, our RNN-based local models outperform state-of-the-art (SOTA) results and the global models further improve the performance over local models across all three datasets.

5.1   TCR

Detailed model performances for the TCR dataset are shown in Table 3. We only report model performance on temporal pairs. Both of our local and global models outperform the baseline. Our global model is able to improve overall model performance by more than 1.2% over our local model; per McNemar's test, this improvement is statistically significant (with p-value < 0.01).

5.2   MATRES

Detailed model performances for the MATRES dataset performances can be found in Table 4. Similar to TCR, both our local and structured models outperform this baseline and the global model is able to improve overall model performance by 1.4%; per McNemar's test, this improvement is statistically significant (with p-value < 0.05).

5.3   TB-Dense

Table 5 shows the breakdown performance for all labels as well as the improvement from local to

671

	Local Model			Global Model		
	P	R	F1	P	R	F1
Before	73.5	52.7	61.3	71.1	58.9	64.4
After	71.6	60.8	65.3	75.0	55.6	63.5
Includes	17.5	4.8	7.4	24.6	4.2	6.9
Is_Include	69.1	4.4	8.0	57.9	5.7	10.2
Simultaneous	0.0	0.0	0.0	0.0	0.0	0.0
Vague	57.9	81.5	67.7	58.3	81.2	67.8
Micro-average			62.6			**63.2**
Chambers et al. (2014)						49.4
Cheng and Miyao (2017)						52.9
Meng and Rumshisky (2018)						**57.0**
TE3 Metrics						
$\mathcal{EE}$ only	62.1	61.9	62.2	62.7	58.9	**62.5**
$+ \mathcal{ET}, \mathcal{TT}$	58.6	63.6	61.0	59.0	64.0	61.4
Ning et al. (2018a)						52.1

Table 5: Model Performance Breakdown for TB-Dense (all values are percentage). Upper Table: for event pairs only, we adopt standard Micro-average score. Lower Table: TE3 refers to the temporal awareness score adopted by TE-3 Workshop. To make fair comparison with Ning et al. (2018a), we add CEAVO predictions on $\mathcal{ET}$ and $\mathcal{TT}$ pairs back into the calculation.

global model by adopting the two-stage structured learning method in TB-Dense dataset. Both our local and global models are able to outperform previous SOTA in micro-average metric (reported by Meng and Rumshisky (2018)) or in TE3 metric (results from Ning et al. (2018a)).

Per McNemar's test, the improvements from local to global model only has p-value = 0.126, so we are not able to conclude that the improvement is statistically significant. We think one of the reasons is the large share of VAGUE pairs (42.6%). VAGUE pairs make our transitivity rules less conclusive. For example, if $\mathcal{R}(e1, e2) = $ BEFORE and $\mathcal{R}(e2, e3) = $ VAGUE, $\mathcal{R}(e1, e3)$ can be any relation types. Moreover, this impact is magnified by our local model's prediction bias towards VAGUE pairs. As we can see in Table 5, the recall score for VAGUE pairs are much higher than other relation types, whereas precision score is moderate. Our global model leverages local output structure to enforce global prediction consistency, but when local predictions contain many VAGUE pairs, it introduces lots of noise too.

To make fair comparison between our model and the best reported TE3 F1 score from Ning et al. (2018a), we follow their strategy and add CAEVO system's predictions on $\mathcal{TT}$ and $\mathcal{ET}$ pairs in the evaluation. The scores are shown in Table 5. Our overall system outperforms the baseline over 10% for both micro-average and TE3 F1 scores.

5.4 Error Analysis

To understand why both the local and structured models make mistakes, we randomly sample 50 pairs from 345 cases where both models' predictions are incorrect among all 3 random seeds. We analyze these pairs qualitatively and categorize them into four cases as shown in Table 6, with each case (except other) paired with an example.

The first case illustrates that correct prediction requires broader contextual knowledge. For example, the gold label for **transition** and **discuss** is *BEFORE*, where the nominal event **transition** refers to a specific era in history that ends before **discuss** in the second sentence. Human annotators can easily infer this relation based on their knowledge in history, but it is difficult for machines without prior knowledge. We observe this as a very common mistake especially for pairs with nominal events. As for the second case shows that negation can completely change the temporal order. The gold label for the event pair **planned** and **plans** is *AFTER* because the negation token **no** postpones the event **planned** indefinitely. Our models do not pick up this signal and hence predict the relation as *VAGUE*.

Finally, "intention" events could make temporal relation prediction difficult (Ning et al., 2018b). Case 3 demonstrates that our models could ignore the "intention" tokens such as **aimed at** in the example and hence make an incorrect prediction *VAGUE* between **doubling** and **signed**, whereas the true label is *AFTER* because **doubling** is an intention that has not occurred.

6 Ablation Studies

Although we have presented strong empirical results, the isolated contribution of each component of our model has not been investigated. In this section, we perform a though ablation study to understand the importance of structured constraints, linguistic features, and the BERT representations.

6.1 Effect of the structured constraints

One of our core claims is that our learning benefits from modeling the structural constraints of event temporal graph. To study the contribution of structured constraints, we provide an ablation study on two constraints that are applied to all three datasets: Symmetry and Transitivity.

A straightforward ablation study on symmetric constraint is to remove it from our global inference

	Case 1 (32%): Connection with broader context

Case 1 (32%): Connection with broader context

The program also calls for coordination of economic reforms and joint improvement of social programs in the two countries, where many people have become impoverished during the chaotic post - Soviet **transition** to capitalism. Kuchma also planned to visit Russian gas giant Gazprom, most likely to **discuss** Ukraine's DLRS 1.2 billion debt to the company.

Case 2 (20%): Negation

Annan has **no** trip **planned** so far. Meanwhile, Secretary of State Madeleine Albright, Berger and Defense Secretary William Cohen announced **plans** to travel to an unnamed city in the us heartland next week, to explain to the American people just why military force will be necessary if diplomacy fails.

Case 3 (14%): Intention Axis

A major goal of Kuchma's four - day state visit was the signing of a 10-year economic program **aimed at doubling** the two nations' trade turnover, which fell to DLRS 14 billion last year, down DLRS 2.5 billion from 1996. The two presidents on Friday **signed** the plan, which calls for cooperation in the metallurgy, fuel, energy, aircraft building, missile, space and chemical industries.

Case 4: (34%) Other

Table 6: Error Categories and Examples in TB-Dense

	TB-Dense		Matres		TCR	
	$\overrightarrow{Test}$	$\overleftrightarrow{Test}$	$\overrightarrow{Test}$	$\overleftrightarrow{Test}$	$\overrightarrow{Test}$	$\overleftrightarrow{Test}$
$M_1 : \overrightarrow{L}$	62.9	61.9	80.4	74.7	80.5	75.7
$M_2 : \overrightarrow{L} + T$	**63.2**	62.0	**81.7**	75.7	**81.0**	76.3
$M_3 : \overleftrightarrow{L}$	62.6	62.7	80.3	80.4	79.7	79.6
$M_4 : \overleftrightarrow{L} + T$	63.1	63.0	81.4	81.4	80.3	80.2
$\overleftrightarrow{L} + S + T$ (Proposed)	**63.2**	**63.2**	**81.7**	**81.7**	80.9	**80.9**

Table 7: Ablation over global constraints: Symmetry and Transitivity. Test is conducted on forward test set and both-way test set, which are denoted as $\overrightarrow{Test}$ and $\overleftrightarrow{Test}$ respectively. The local models trained on forward data and both-way data are denoted as $\overrightarrow{L}$ and $\overleftrightarrow{L}$. Symmetry and Transitivity constraints are denoted as S and T. The results demonstrate that symmetry and transitivity constraints both improve model's performance.

step. However, even though we eliminate symmetric constraint explicitly in global inference, it is utilized implicitly in our data augmentation steps (Section 4.1). To better understand the benefits of the symmetry constraints, we study both the contribution of *explicitly* applying symmetry constraint in our SSVM as well as its *implicit* impact in data augmentation.

Hence, in this section, we view a pair with original order and flipped order as different instances for learning and evaluation. We denote the pairs with original order as "**forward**" data, their flipped-order counterparts as "**backward**" data, and their combinations as "**both-way**" data.

We train four additional models to study the impacts of symmetry and transitivity constraints: 1) local model trained on forward data; 2) global model with transitivity constraint trained on forward data; 3) local model trained on both-way data; 4) global model with transitivity constraint trained on both-way data, denoted as M_1, M_2, M_3, M_4 respectively. M_1 and M_2 are models that do not apply any symmetric property; M_3 and M_4 are models that utilize symmetric property implicitly.

Additionally, evaluation setup should be re-scrutinized if we remove the symmetry constraints. In the standard evaluation setup of prior works, evaluation is only performed on the pairs with their original order (forward data) in text. This evaluation assumes a model will work equally well for both forward and backward

data, which certainly holds when we explicitly impose symmetry constraints. However, as we can observe in the later analysis, this assumption fails when we remove symmetry constraints. To demonstrate the improvement of model robustness over backward data, we propose to test the model on both forward and both-way data. If a model is robust, it should perform well on both scenarios.

We summarize our analysis of the results in Table 7 (F1 scores) as follows:

- Impact of Transitivity: By comparing M_1 with M_2 and M_3 with M_4, the consistent improvements across all three datasets demonstrate the effectiveness of global transitivity constraints.

- Impact of Implicit Symmetry (data augmentation): Examining the contrast between M_1 and M_3 as well as M_2 and M_4, we can see significant improvements in both-way evaluation despite slight performance drops in forward evaluation. These comparisons imply that data augmentation can help improve model robustness. Note that Meng and Rumshisky (2018) leveraged this data augmentation trick in their model.

- Impact of Explicit Symmetry: By comparing the proposed model with M_4, the consistent improvements across all datasets demonstrate the benefit of using explicit symmetry property.

- Model Robustness: Although M_1 and M_2 show competitive results when evaluated on forward test data, their performance degrade

	local- w/ feat.	global- w/ feat.	local- w/o feat.	global- w/o feat.
TB-Dense	62.5	63.0	62.6	**63.2**
MATRES	81.4	**81.7**	80.3	**81.7**
TCR	79.5	80.7	79.7	**80.9**

Table 8: Ablation study on linguistic feature usage. Additional linguistic features do not lead to significant improvement and even hurt performance in 2 out of 3 datasets. The results show that our proposed framework is semantic-rich and capable of avoiding the usage of additional linguistic feature.

significantly in the both-way evaluation. In contrast, the proposed model achieves strong performances in both test scenarios (best F1 scores except for one), and hence proves the robustness of our proposed method.

6.2 Effect of linguistic features

Previous research establish the success of leveraging linguistic features in event relation prediction. One advantage of leveraging contextualized word embedding is to provide rich semantic representation and could potentially avoid the usage of extra linguistic features. Here, we study the impact of incorporating linguistic features to our model by using simple features provided in the original datasets: token distance, tense and polarity of event entities. These features are concatenated with the Bi-LSTM hidden states before the linear layer (i.e. f_i, b_i, f_j, b_j in Figure 2). Table 8 shows the F1 scores of our local and global model using or not using linguistic features respectively. These additional features likely cause over-fitting and hence do not improve model performance across all three datasets we test. This set of experiments show that linguistic features do not improve the predicting power of our current framework.

6.3 Effect of BERT representations

In this section, we explore the impact of contextualized BERT representations under our deep SSVM framework. We replace BERT representations with the GloVe (Pennington et al., 2014) word embeddings. Table 9 shows the F1 scores of our local model and global model using BERT and GloVe[11] respectively. BERT improves the performance with a significant margin. Besides, even without BERT representations, our RNN-based local model and the deep structured global model

---

[11]For GloVe model, additional linguistic features are used.

	previous SOTA	local- Glove	global- Glove	local- BERT	global- BERT
TB-Dense	57.0	56.6	57.0	62.6	**63.2**
MATRES	69.0	71.8	75.6	80.3	**81.7**
TCR	71.1	73.5	76.5	79.7	**80.9**

Table 9: Ablation over word representation: BERT vs GloVe. Although BERT representation largely contributes to the performance boost, our proposed framework remains strong and outperforms current SOTA approaches when GloVe is used.

still outperform (MATRES and TCR) or are comparable with (TB-Dense) current SOTA. These results confirm the improvements of our method.

7 Conclusion

In this paper, we propose a novel deep structured model based on SSVM that combines the benefits of structured models' ability to encode structure knowledge, and data-driven deep neural architectures' ability to learn long-range features. Our experimental results exhibit the effectiveness of this approach for event temporal relation extraction.

One interesting future direction is further leveraging commonsense knowledge, domain knowledge in temporal relation, and linguistics information to create more robust and comprehensive global constraints for structured learning. Another direction is to improve feature representations by designing novel neural architectures that better capture negation and hypothetical phrases as discussed in error analysis. We plan to leverage large amount of unannotated corpora to help event temporal relation extraction as well.

Acknowledgments

References

Steven Bethard. 2013. Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 10–14. Association for Computational Linguistics.

Steven Bethard, James H. Martin, and Sara Klingenstein. 2007. Timelines from text: Identification of syntactic temporal relations. In *Proceedings of the International Conference on Semantic Computing*, ICSC '07, pages 11–18, Washington, DC, USA. IEEE Computer Society.

P. Bramsen, P. Deshpande, Y. K. Lee, and R. Barzilay. 2006a. Inducing temporal graphs. In *EMNLP*, Sydney, Australia.

Philip Bramsen, Pawan Deshpande, Yoong Keok Lee, and Regina Barzilay. 2006b. Inducing temporal graphs. In *EMNLP*, Sydney, Australia.

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. Association for Computational Linguistics.

Nate Chambers. 2013. Navytime: Event and time ordering from raw text. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 73–77, Atlanta, Georgia, USA. Association for Computational Linguistics.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. In *ACL*.

Nathanael Chambers and Dan Jurafsky. 2008. Jointly combining implicit constraints improves temporal ordering. In *EMNLP*, Honolulu, United States.

Nathanael Chambers, Shan Wang, and Dan Jurafsky. 2007. Classifying temporal relations between events. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 173–176, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Pascal Denis and Philippe Muller. 2011. Predicting globally-coherent temporal structures from texts via endpoint inference and graph decomposition. In *IJCAI*, Barcelone, Spain.

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

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

Quang Xuan Do, Wei Lu, and Dan Roth. 2012. Joint inference for event timeline construction. In *EMNLP*, Jeju, Korea.

Thomas Finley and Thorsten Joachims. 2008. Training structural svms when exact inference is intractable. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 304–311, New York, NY, USA. ACM.

Rujun Han, Mengyue Liang, Bashar Alhafni, and Nanyun Peng. 2019a. Contextualized word embeddings enhanced event temporal relation extraction for story understanding. *arXiv preprint arXiv:1904.11942*.

Rujun Han, Qiang Ning, and Nanyun Peng. 2019b. Joint event and temporal relation extraction with shared representations and structured prediction.

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, Atlanta, Georgia, USA. Association for Computational Linguistics.

Artuur Leeuwenberg and Marie-Francine Moens. 2017. Structured learning for temporal relation extraction from clinical records. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 1150–1158.

Inderjeet Mani, Marc Verhagen, Ben Wellner, Chong Min Lee, and James Pustejovsky. 2006. Machine learning of temporal relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 753–760, Stroudsburg, PA, USA. Association for Computational Linguistics.

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)*.

Yuanliang Meng, Anna Rumshisky, and Alexey Romanov. 2017. Temporal information extraction for question answering using syntactic dependencies in an lstm-based architecture. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 887–896.

Paramita Mirza and Sara Tonelli. 2016. Catena: Causal and temporal relation extraction from natural language texts. In *ACL*, Osaka, Japan.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories. In *NAACL*, San Diego, USA.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *EMNLP*, Copenhagen, Denmark.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018a. 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. Association for Computational Linguistics.

Qiang Ning, Hao Wu, and Dan Roth. 2018b. A multi-axis annotation scheme for event temporal relations. In *ACL*. Association for Computational Linguistics.

Tim O'Gorman, Kristin Wright-Bettner, and Martha Palmer. 2016. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of 2nd Workshop on Computing News Storylines*, pages 47–56. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *In EMNLP*.

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, and Lisa Ferro. 2003. The timebank corpus. In *Corpus linguistics*, pages 647–656.

Julien Tourille, Olivier Ferret, Aurelie Neveol, and Xavier Tannier. 2017a. Neural architecture for temporal relation extraction: a bi-lstm approach for detecting narrative containers. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 224–230.

Julien Tourille, Olivier Ferret, Xavier Tannier, and Aurélie Névéol. 2017b. Limsi-cot at semeval-2017 task 12: Neural architecture for temporal information extraction from clinical narratives. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 597–602. Association for Computational Linguistics.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9. Association for Computational Linguistics.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 75–80, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marc Verhagen and James Pustejovsky. 2008. Temporal processing with the tarsqi toolkit. In *22Nd International Conference on on Computational Linguistics: Demonstration Papers*, COLING '08, pages 189–192, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marc Verhagen, Roser Saurí, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 57–62, Stroudsburg, PA, USA. Association for Computational Linguistics.

Katsumasa Yoshikawa, Sebastian Riedel, Masayuki Asahara, and Yuji Matsumoto. 2009. Jointly identifying temporal relations with markov logic. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 405–413. Association for Computational Linguistics.