Utilizing Relative Event Time to Enhance Event-Event Temporal Relation Extraction

Haoyang Wen and **Heng Ji**
Department of Computer Science
University of Illinois Urbana-Champaign
`{wen17, hengji}@illinois.edu`

Abstract

Event time is one of the most important features for event-event temporal relation extraction. However, explicit event time information in text is sparse. For example, only about 20% of event mentions in TimeBank-Dense have event-time links. In this paper, we propose a joint model for event-event temporal relation classification and an auxiliary task, relative event time prediction, which predicts the event time as real numbers. We adopt a Stack-Propagation framework to incorporate predicted relative event time for temporal relation classification and keep the differentiability. Our experiments on MA-TRES dataset show that our model can significantly improve the RoBERTa-based baseline and achieve state-of-the-art performance. [1]

1 Introduction

Event temporal ordering is an important task to understand the evolution of events. Event-event temporal relation extraction aims to automatically extract the temporal order given a pair of events and further build a temporal graph. Figure 1 illustrates an example sentence and its temporal graph. There are three events in the sentence, **said**, **identified** and **run**. The temporal relation between **said** and **identified** is AFTER, and the temporal relations between **said** and **run** and between **identified** and **run** are BEFORE.

Neural network based methods have achieved promising improvement for temporal relation extraction (Meng et al., 2017; Meng and Rumshisky, 2018; Cheng et al., 2020; Ballesteros et al., 2020; Wen et al., 2021a). They mostly consider the task as pairwise classification. There are also efforts focusing on the global structures, including Markov logical networks and Integer Linear Programming
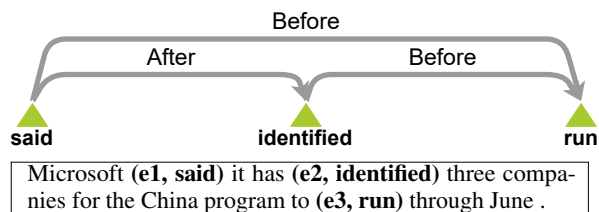

Figure 1: An example sentence and their temporal relations. In this example, there three different events, and the final extracted graph shows the pair-wise temporal relation extraction results.

(ILP) based methods (Bramsen et al., 2006; Chambers and Jurafsky, 2008; Yoshikawa et al., 2009; Do et al., 2012; Ning et al., 2017, 2018a, 2019; Han et al., 2019). Though achieving great success, event time, an important feature, is often overlooked by previous work. Conceptually, if we know the exact time information for all events, their temporal relations can be naturally derived. For example, if we know that event A happened on Monday while event B happened on Tuesday in the same week, it is obvious that A happened BEFORE B. However, explicit time arguments can be rarely found in text, especially in news articles (Wen et al., 2021b). Leeuwenberg and Moens (2018) propose to predict the relative timeline and directly compare the relative timestamps of events to derive their temporal relations. Although showing promising performance, those predicted timestamps do not consider information from event pairs and cannot handle the uncertain temporal boundary of an event expressed in text to predict relations such as VAGUE.

In this paper, we follow the idea of Leeuwenberg and Moens (2018) to predict the relative event time for temporal relation extraction, where the relative time is a real number indicating the relative position of the event in the timeline. Instead of directly comparing relative event time as Leeuwenberg and Moens (2018), we consider them as additional features and incorporate them into training

---

the temporal relation classifier. We propose a joint model with Stack-Propagation framework (Zhang and Weiss, 2016) to connect relative event time prediction and temporal relation extraction, which can exploit the explicit features from the former task for the latter task (Qin et al., 2019). Our model directly uses the output of relative event time prediction as the input for temporal relation classification so that the classification can benefit from both representations of event pairs and the predicted relative event time. Because we do not break the differentiability between two tasks, the training objective for one task can also promote another task. Similarly to Leeuwenberg and Moens (2018), we adapt margin-based optimization to constrain the distance between two relative event time values, given their temporal relation.

Our experiments show that the relative event time prediction can significantly help learn better temporal relation extraction, compared to vanilla RoBERTa-based (Liu et al., 2019) baseline. We have also achieved similar performance compared to the state-of-the-art temporal relation extraction approach that uses additional data (Ballesteros et al., 2020).
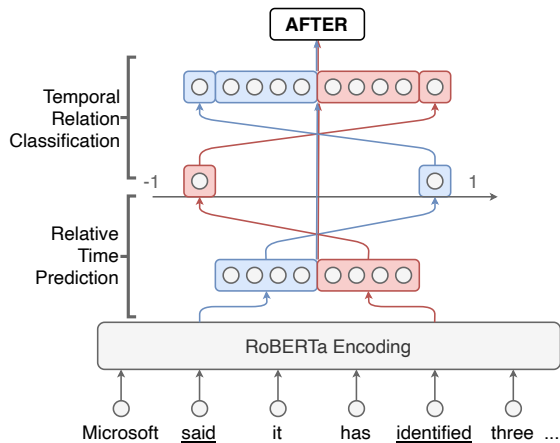
2 Approach

Figure 2: The overall architecture.

In this section, we will discuss the relative event time prediction for temporal relation extraction. The overall approach is illustrated in Figure 2.

2.1 Our Baseline Model

Our baseline is a pretrained language model based pairwise classification model. It takes a sequence of tokens X with length n as input after preprocessing required by pretrained language models,

such as subword tokenization (Devlin et al., 2019) or byte pair encoding (BPE, Liu et al., 2019). The input also includes the positions of two event mentions (e_i, e_j) in the text. For simplicity, we only use the start positions from the corresponding event mention spans. We denote the start position of an event mention e_i as p_i. The baseline model is used to predict the temporal relations between the given two event mentions.

The model first computes the contextualized representation for each input token using a pretrained language model (Devlin et al., 2019; Liu et al., 2019). We denote the contextualized representation as $\boldsymbol{H}$, where $\boldsymbol{h}_i$ is the contextualized representation for the token at position i.

Then we concatenate the representations of any given two event mentions using the representation at their corresponding start positions,

$$\boldsymbol{c}_{i,j} = \left[\boldsymbol{h}_{p_i}; \boldsymbol{h}_{p_j}\right]. \qquad (1)$$

We use a two-layer feed-forward neural network (FFN) with a tanh activation function and a softmax layer to convert the representation into a probability distribution,

$$P(r \mid e_i, e_j) = \text{softmax}\left(\boldsymbol{a}_{i,j}\right), \qquad (2)$$
$$\boldsymbol{a}_{i,j} = \text{FFN}_2(\tanh(\text{FFN}_1(\boldsymbol{c}_{i,j}))), \quad (3)$$

where $\text{FFN}_i(\boldsymbol{x}) = \boldsymbol{W}_i\boldsymbol{x} + \boldsymbol{b}_i$.

2.2 Relative Timestamp Prediction

To better utilize the contextual information for events, we use an auxiliary task, relative event time prediction, to predict event time as a real number for all event mentions given its context, similar to Leeuwenberg and Moens (2018). Contrary to the above baseline method that takes a pair of representations and predict the pair-wise relation, event time information is only related to the event itself. Therefore, we predict the relative event time information by mapping the representation of an event mention e_i from the pretrained language model to a real number, $t_i \in (-1, 1)$, where we use a two-layer feed-forward network with $\tanh$ activation functions,

$$t_i = \tanh(\text{FFN}_4(\tanh(\text{FFN}_3(\boldsymbol{h}_{p_i})))). \quad (4)$$

Although we may not have explicit time information in the given context, the gold-standard pairwise temporal relations can be considered as incidental supervision to constrain the predicted time.

For example, given two event mentions e_i and e_j, and their temporal relation e_i BEFORE e_j, then their predicted time t_i and t_j should follow $t_i < t_j$. Similarly, if their relation is EQUAL, then the distance between their predicted time should be as close as possible.

Therefore, we use a margin-based optimization method to constrain our predicted relative event time. We use different margins based on different temporal relations,

$$
\begin{aligned}
\mathcal{L}_t = {} & \mathbb{1}[r_{(e_i,e_j)} = \text{BEFORE}] \max\left(0, 1 - (t_j - t_i)\right) \\
& + \mathbb{1}[r_{(e_i,e_j)} = \text{AFTER}] \max\left(0, 1 - (t_i - t_j)\right) \\
& + \mathbb{1}[r_{(e_i,e_j)} = \text{EQUAL}]|t_i - t_j|. \quad (5)
\end{aligned}
$$

If e_i is BEFORE e_j, the above optimization will maximize the distance $(t_j - t_i)$ unless it is equal or larger than 1, which follows the constraint $t_i < t_j$. On the contrary, If e_i is AFTER e_j, it will maximize the distance $(t_i - t_j)$, which follows the constraint $t_i > t_j$. If e_i is EQUAL e_j, then it instead minimizes the distance $|t_i - t_j|$.

2.3 Stack-Propagation on Relative Timestamp

After obtaining the relative time for each event, we further incorporate this predicted feature into event-event temporal relation extraction. Since both relative time prediction and temporal relation extraction are based on contextualized representations from the pretrained language model, we adopt Stack-Propagation framework to connect these two tasks while preserving the differentiability.

Specifically, besides the event-pair contextualized representation that the baseline method uses for pair-wise temporal relation classification, we further incorporate their predicted relative event time into classification,

$$
\boldsymbol{a}_{i,j} = \text{FFN}_2(\tanh(\text{FFN}_1([\boldsymbol{c}_{i,j}; t_i; t_j]))). \quad (6)
$$

During training, we use cross-entropy objective for temporal relation classification,

$$
\mathcal{L}_r = -\log P(r = r_{(e_i,e_j)} \mid e_i, e_j). \quad (7)
$$

The final training objective is the interpolation of the classification task and time prediction task,

$$
\mathcal{L} = \alpha \mathcal{L}_t + \mathcal{L}_r. \quad (8)
$$

Since we keep the differentiability for classification, the gradient from cross-entropy can be propagated to timestamps and jointly train relative time prediction.

3 Experiments

	Train	Development	Test
Docs	260	21	20
Relations	10,888	1,852	840

Table 1: Data splits and statistics on MATRES.

3.1 Dataset

We conduct our experiments on MATRES (Ning et al., 2018b). MATRES contains refined annotations on TimeBank (Pustejovsky et al., 2003; Cassidy et al., 2014) and TempEval (UzZaman et al., 2013) (containing AQUAINT and Platinum subsets) documents. We follow the previous work that uses TimeBank and AQUAINT for training and we use Platinum as the test set. We randomly select 21 documents as development set. The detailed statistics can be found in Table 1.

3.2 Experimental Setup

We use F_1 to evaluate our system performance, following (Ning et al., 2019), where we consider VAGUE as "no relation". We compare our model with existing systems including 1) BiLSTM+MAP: A BiLSTM based joint event and temporal relation extraction model with MAP inference (Han et al., 2019). 2) LSTM+TEMPROB+ILP: LSTM-based method incorporating pretrained language model embedding, commensense prior (TEMPROB) and ILP (Ning et al., 2019). 3) Joint Constrained Learning: A constrained learning based optimization for joint event temporal and hierarchy relation extraction (Wang et al., 2020). 4) Self-Training: Multi-task self-training on temporal relation extraction using additional time annotation from ACE2005 and the original TimeBank (Ballesteros et al., 2020).

We use RoBERTa-large as our pretrained language model. Our best model is optimized using AdamW for 30 epochs with learning rate between {1e-5, 2e-5} for both pretrained model and other parameters. We use linear scheduler with warmup proportion at 0.1. We set weight decay to 0.01 and dropout rate to 0.1 for all parameters. The training batch size is 16. We use 5 different random seeds for our experiments, and choose the learning rate and model with the best averaged performance on development set for comparison on test set. The hidden size of $\text{FFN}_1(\cdot)$ is 1024. The hidden size of $\text{FFN}_3(\cdot)$ is 1026 (adding two predicted time as input). α is set to 1.

Model	Precision	Recall	F_1
BiLSTM+MAP	-	-	75.5
LSTM+TEMPROB+ILP	71.3	82.1	76.3
Joint Constrained Learning*	73.4	85.0	78.8
Self-Training*	-	-	81.6
Our Model	**78.4**	**85.2**	**81.7**

Table 2: Temporal relation extraction results on MA-TRES. Precision and recall are not reported by (Han et al., 2019; Ballesteros et al., 2020). We report our averaged test performance on all metrics over 5 random seeds. Models marked * use additional training resources.

3.3 Overall Performance

Our overall performance is shown in Table 2. Among these baseline systems, the multi-task self-training method (Ballesteros et al., 2020) has achieved the best performance. Our proposed method can achieve slightly better performance against their system, without introducing additional annotated and raw data, which demonstrates the effectiveness of the relative timestamp prediction objective, and the stack-propagation based method to incorporate predicted timestamps.

3.4 Ablation Study

Model	Precision	Recall	F_1
Vanilla Classifier	78.1	82.5	80.2
Multi-Task	76.5	85.2	80.6
Stack-Propagation	**78.4**	85.2	**81.7**
Time Comparison	75.5	**86.8**	80.7

Table 3: Ablation study results (%) on our proposed method. We report our averaged test performance on all metrics. $p < 0.05$ for the two-sided heteroscedastic independent t-test between Vanilla RoBERTa and our model.

We conduct ablation study for relative event time prediction and Stack-Propagation and the results are shown in Table 3. We can find that adding relative time prediction as an auxiliary task (Multi-Task) helps improve the performance, and incorporating relative event time as features (Stack-Propagation) further boosts the performance. We also compare with training relative time prediction and directly using relative timestamps to derive the temporal relation labels, similar to Leeuwenberg and Moens (2018). The model has the highest recall performance because it aggressively classifies relations as BEFORE or AFTER labels. However, because it cannot handle VAGUE, it's low in pre-
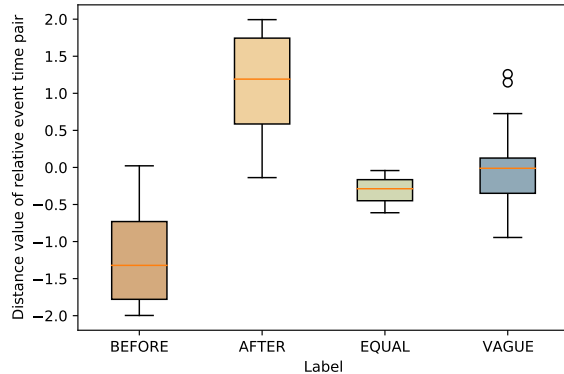

Figure 3: Visualization of distances of the relative timestamps and their predicted labels.

He (**said, e1**) he (**discussed, e2**) the issue with Mr. Netanyahu during his visit to Israel this week, and that they (**agreed, e3**) the timing was good for a discussion with the Turkish leader.
$t_{e1} = 0.9937, \quad t_{e2} = -0.7451, \quad t_{e3} = -0.0770$
$r_{(e1,e2)} = $ AFTER, $r_{(e1,e3)} = $ AFTER, $r_{(e2,e3)} = $ BEFORE
Traditionally , the (intentionally) funny lines by our presidents have (**had, e1**) one thing in common: They were self-deprecating. Sure, some presidents have (**used, e2**) jokes to take jabs at their opponents, but not to the extent of Obama.
$t_{e1} = -0.5162, \quad t_{e2} = -0.5209$
$r_{(e1,e2)} = $ VAGUE

Table 4: Examples of temporal relation extraction and relative even time prediction results. The first example shows the correlation between predicted relative event time and temporal relations. The second example shows that the model can use event-pair information to predict VAGUE labels.

cision even compared to vanilla RoBERTa-based classifier (Vanilla Classifier).

3.5 Qualitative Analysis

We visualize the distance values of the relative event time for given event pairs and their predicted labels in Figure 3, where a negative value, for example, naturally indicates that the time of the former event is earlier than the latter event. We can find that BEFORE and AFTER predictions almost correlate with the distance values of the relative event time pairs. We can also find that VAGUE and EQUAL predictions are centered near 0, which shows that our model can take some event pairs that are hard to compare their predicted relative timestamps and classify them as VAGUE or EQUAL.

Table 4 demonstrates two example system outputs. The first case shows that the temporal relation extraction correlates with relative event time prediction, while the second case shows that our model can utilize the event-pair representation and

classify relation as VAGUE rather than completely depending on predicted timestamps.

4 Related Work

Earlier efforts on temporal relation extraction focus on global inference using methods such as Integer Linear Programming (Bramsen et al., 2006; Chambers and Jurafsky, 2008; Yoshikawa et al., 2009; Do et al., 2012; Chambers et al., 2014). Recently, neural network-based methods have also achieved promising improvement (Meng et al., 2017; Meng and Rumshisky, 2018; Ning et al., 2018a, 2019; Han et al., 2019; Wang et al., 2020; Cheng et al., 2020). Especially, Goyal and Durrett (2019) use LSTM to encode Timex for temporal relation extraction, Ballesteros et al. (2020) jointly train event time arguments extraction from ACE2005. The most related work is (Leeuwenberg and Moens, 2018), which proposes to predict relative event time and uses the comparison of relative timestamps as temporal relations. Our work focuses on jointly training relative time prediction and temporal relation extraction and utilizes relative timestamps as features via the Stack-Propagation framework.

5 Conclusions and Future Work

In this paper, we leverage relative event time prediction that can ground events onto a relative timeline to help event-event temporal relation extraction. We use Stack-Propagation framework to further incorporate predicted timestamp explicit for relation classification. Our experiment results demonstrate the effectiveness of our proposed method. In the future, we will focus on extending the relative time prediction to cross-document setting and support cross-document temporal relation extraction.

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, Online. Association for Computational Linguistics.

Philip Bramsen, Pawan Deshpande, Yoong Keok Lee, and Regina Barzilay. 2006. Inducing temporal graphs. In *EMNLP 2006, Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, 22-23 July 2006, Sydney, Australia*, pages 189–198. ACL.

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, Baltimore, Maryland. Association for Computational Linguistics.

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, Honolulu, Hawaii. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Jeju Island, Korea. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2019. Embedding time expressions for deep temporal ordering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4400–4406, Florence, Italy. Association for Computational Linguistics.

Rujun Han, Qiang Ning, and Nanyun Peng. 2019. Joint event and temporal relation extraction with shared representations and structured prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 434–444. Association for Computational Linguistics.

Artuur Leeuwenberg and Marie-Francine Moens. 2018. Temporal information extraction by predicting relative time-lines. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1237–1246, Brussels, Belgium. Association for Computational Linguistics.

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

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, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 527–536. Association for Computational Linguistics.

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, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 887–896. Association for Computational Linguistics.

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, Copenhagen, Denmark. Association for Computational Linguistics.

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, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2278–2288. Association for Computational Linguistics.

Qiang Ning, Sanjay Subramanian, and Dan Roth. 2019. An improved neural baseline for temporal relation extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6202–6208. Association for Computational Linguistics.

Qiang Ning, Hao Wu, and Dan Roth. 2018b. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1318–1328. Association for Computational Linguistics.

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.

Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. 2019. A stack-propagation framework with token-level intent detection for spoken language understanding. 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 2078–2087, Hong Kong, China. 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, Atlanta, Georgia, USA. Association for Computational Linguistics.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020. Joint constrained learning for event-event relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 696–706, Online. Association for Computational Linguistics.

Haoyang Wen, Ying Lin, Tuan Lai, Xiaoman Pan, Sha Li, Xudong Lin, Ben Zhou, Manling Li, Haoyu Wang, Hongming Zhang, Xiaodong Yu, Alexander Dong, Zhenhailong Wang, Yi Fung, Piyush Mishra, Qing Lyu, Dídac Surís, Brian Chen, Susan Windisch Brown, Martha Palmer, Chris Callison-Burch, Carl Vondrick, Jiawei Han, Dan Roth, Shih-Fu Chang, and Heng Ji. 2021a. RESIN: A dockerized schema-guided cross-document cross-lingual cross-media information extraction and event tracking system. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 133–143, Online. Association for Computational Linguistics.

Haoyang Wen, Yanru Qu, Heng Ji, Qiang Ning, Jiawei Han, Avi Sil, Hanghang Tong, and Dan Roth. 2021b. Event time extraction and propagation via graph attention networks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 62–73, Online. 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*, pages 405–413, Suntec, Singapore. Association for Computational Linguistics.

Yuan Zhang and David Weiss. 2016. Stack-propagation: Improved representation learning for syntax. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1557–1566, Berlin, Germany. Association for Computational Linguistics.