An Improved Neural Baseline for Temporal Relation Extraction

Qiang Ning,[1] **Sanjay Subramanian,**[2] **and Dan Roth**[1,2]
[1]University of Illinois at Urbana-Champaign
[2]University of Pennsylvania
qning2@illinois.edu, {subs,danroth}@seas.upenn.edu

Abstract

Determining temporal relations (e.g., *before* or *after*) between events has been a challenging natural language understanding task, partly due to the difficulty to generate large amounts of high-quality training data. Consequently, neural approaches have not been widely used on it, or showed only moderate improvements. This paper proposes a new neural system that achieves about 10% absolute improvement in accuracy over the previous best system (25% error reduction) on two benchmark datasets. The proposed system is trained on the state-of-the-art MATRES dataset and applies contextualized word embeddings, a Siamese encoder of a temporal common sense knowledge base, and global inference via integer linear programming (ILP). We suggest that the new approach could serve as a strong baseline for future research in this area.

1 Introduction

Temporal relation (TempRel) extraction has been considered as a major component of understanding time in natural language (Do et al., 2012; UzZaman et al., 2013; Minard et al., 2015; Llorens et al., 2015; Ning et al., 2018a). However, the annotation process for TempRels is known to be time consuming and difficult even for humans, and existing datasets are usually small and/or have low inter-annotator agreements (IAA); e.g., UzZaman et al. (2013); Chambers et al. (2014); O'Gorman et al. (2016) reported Kohen's κ and F_1 in the 60's. Albeit the significant progress in deep learning nowadays, neural approaches have not been used extensively for this task, or showed only moderate improvements (Dligach et al., 2017; Lin et al., 2017; Meng and Rumshisky, 2018). We think it is important for to understand: is it because we missed a "magic" neural architecture, because the training dataset is small, or because the quality of the dataset should be improved?

Recently, Ning et al. (2018c) introduced a new dataset called *Multi-Axis Temporal RElations for Start-points* (MATRES). MATRES is still relatively small in its size (15K TempRels), but has a higher annotation quality from its improved task definition and annotation guideline. This paper uses MATRES to show that a long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) system can readily outperform the previous state-of-the-art system, CogCompTime (Ning et al., 2018d), by a large margin. The fact that a standard LSTM system can significantly improve over a feature-based system on MATRES indicates that neural approaches have been mainly dwarfed by the quality of annotation, instead of specific neural architectures or the small size of data.

To gain a better understanding of the standard LSTM method, we extensively compare the usage of various word embedding techniques, including word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), FastText (Bojanowski et al., 2016), ELMo (Peters et al., 2018), and BERT (Devlin et al., 2018), and show their impact on TempRel extraction. Moreover, we further improve the LSTM system by injecting knowledge from an updated version of TEMPROB, an automatically induced temporal common sense knowledge base that provides *typical* TempRels between events[1] (Ning et al., 2018b). Altogether, these components improve over CogCompTime by about 10% in F_1 and accuracy. The proposed system is public[2] and can serve as a strong baseline for future research.

2 Related Work

Early computational attempts to TempRel extraction include Mani et al. (2006); Chambers et al.

[1]For example, "explode" typically happens *before* "die".
[2]https://cogcomp.org/page/publication_view/879

(2007); Bethard et al. (2007); Verhagen and Pustejovsky (2008), which aimed at building classic learning algorithms (e.g., perceptron, SVM, and logistic regression) using *hand-engineered features* extracted for each pair of events. The frontier was later pushed forward through continuous efforts in a series of SemEval workshops (Verhagen et al., 2007, 2010; UzZaman et al., 2013; Bethard et al., 2015, 2016, 2017), and significant progresses were made in terms of data annotation (Styler IV et al., 2014; Cassidy et al., 2014; Mostafazadeh et al., 2016; O'Gorman et al., 2016), structured inference (Chambers and Jurafsky, 2008a; Do et al., 2012; Chambers et al., 2014; Ning et al., 2018a), and structured machine learning (Yoshikawa et al., 2009; Leeuwenberg and Moens, 2017; Ning et al., 2017).

Since TempRel is a specific relation type, it is natural to borrow recent neural relation extraction approaches (Zeng et al., 2014; Zhang et al., 2015; Zhang and Wang, 2015; Xu et al., 2016). There have indeed been such attempts, e.g., in clinical narratives (Dligach et al., 2017; Lin et al., 2017; Tourille et al., 2017) and in newswire (Cheng and Miyao, 2017; Meng and Rumshisky, 2018; Leeuwenberg and Moens, 2018). However, their improvements over feature-based methods were moderate (Lin et al. (2017) even showed negative results). Given the low IAAs in those datasets, it was unclear whether it was simply due to the low data quality or neural methods inherently do not work well for this task.

A recent annotation scheme, Ning et al. (2018c), introduced the notion of multi-axis to represent the temporal structure of text, and identified that one of the sources of confusions in human annotation is asking annotators for TempRels across different axes. When annotating only same-axis TempRels, along with some other improvements to the annotation guidelines, MATRES was able to achieve much higher IAAs.[3] This dataset opens up opportunities to study neural methods for this problem. In Sec. 3, we will explain our proposed LSTM system, and also highlight the major differences from previous neural attempts.

3 Neural TempRel Extraction

One major disadvantage of feature-based systems is that errors occurred in feature extraction propagate to subsequent modules. Here we study the usage of LSTM networks[4] on the TempRel extraction problem as an end-to-end approach that only takes a sequence of word embeddings as input (assuming that the position of events are known). Conceptually, we need to feed those word embeddings to LSTMs and obtain a vector representation for a particular pair of events, which is followed by a fully-connected, feed-forward neural network (FFNN) to generate confidence scores for each output label. Based on the confidence scores, global inference is performed via integer linear programming (ILP), which is a standard procedure used in many existing works to enforce the transitivity property of time (Chambers and Jurafsky, 2008b; Do et al., 2012; Ning et al., 2017). An overview of the proposed network structure and corresponding parameters can be found in Fig. 1. Below we also explain the main components.

3.1 Handling Event Positions

Each TempRel is associated with two events, and for the same text, different pairs of events possess different relations, so it is critical to indicate the positions of those events when we train LSTMs for the task. The most straightforward way is to concatenate the hidden states from both time steps that correspond to the location of those events (Fig. 1b). Dligach et al. (2017) handled this issue differently, by adding XML tags immediately before and after each event (Fig. 1a). For example, in the sentence, *After **eating** dinner, he **slept** comfortably*, where the two events are bold-faced, they will convert the sequence into *After <e1> eating </e1> dinner, he <e2> slept </e2> comfortably.* The XML markups, which was initially proposed under the name of *position indicators* for relation extraction (Zhang and Wang, 2015), uniquely indicate the event positions to LSTM, such that the final output of LSTM can be used as a representation of those events and their context. We compare both methods in this paper, and as we show later, the straightforward concatenation method is already as good as XML tags for this task.

3.2 Common Sense Encoder (CSE)

In naturally occurring text that expresses TempRels, connective words such as *since*,

[3] Between experts: Kohen's $\kappa \approx 0.84$. Among crowdsourcers: accuracy 88%. More details in Ning et al. (2018c).

[4] We also tried convolutional neural networks but did not observe that CNNs improved performance significantly compared to the LSTMs. Comparison between LSTM and CNN is also not the focus of this paper.

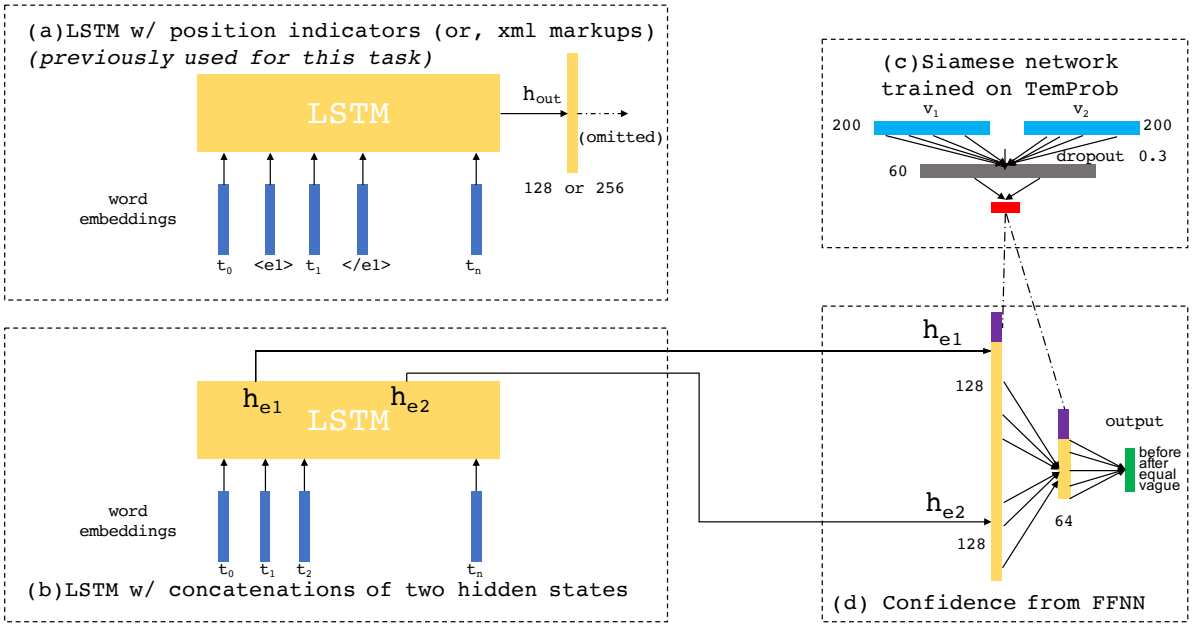

Figure 1: Overview of the neural network structures studied in this paper. Networks (a) and (b) are two ways to handle event positions in LSTMs (Sec. 3.1). (c) The Siamese network used to fit TemProb (Sec. 3.2). Once trained on TemProb, the Siamese network is fixed when training other parts of the system. (d) The FFNN that generates confidence scores for each label. Sizes of hidden layers are already noted. Embeddings of the same color share the same matrix.

when, or *until* are often not explicit; nevertheless, humans can still infer the TempRels using common sense with respect to the events. For example, even without context, we know that *die* is typically after *explode* and *schedule* typically before *attend*. Ning et al. (2018b) was an initial attempt to acquire such knowledge, by aggregating automatically extracted TempRels from a large corpus. The resulting knowledge base, TEMPROB, contains observed frequencies of tuples $(v1, v2, r)$ representing the probability of verb 1 and verb 2 having relation r and it was shown a useful resource for TempRel extraction.

However, TEMPROB is a simple counting model and fails (or is unreliable) for unseen (or rare) tuples. For example, we may see (ambush, die) less frequently than (attack, die) in a corpus, and the observed frequency of (ambush, die) being *before* or *after* is thus less reliable. However, since "ambush" is semantically similar to "attack", the statistics of (attack, die) can actually serve as an auxiliary signal to (ambush, die). Motivated by this idea, we introduce common sense encoder (CSE): We fit an updated version of TEMPROB via a Siamese network (Bromley et al., 1994) that generalizes to unseen tuples through the resulting embeddings for each verb (Fig. 1c). Note that the TEMPROB we use is reconstructed using the same method described in Ning et al. (2018b) with the base method changed to CogCompTime. Once trained, CSE will remain fixed when training the LSTM part (Fig. 1a or b) and the feedforward neural network part (Fig. 1d). We only use CSE for its output. In the beginning, we tried to directly use the output (i.e., a scalar) and the influence on performance was negligible. Therefore, here we discretize the CSE output, change it to categorical embeddings, concatenate them with the LSTM output, and then produce the confidence scores (Fig. 1d).

4 Experiments

4.1 Data

The MATRES dataset[5] contains 275 news articles from the TempEval3 workshop (UzZaman et al., 2013) with newly annotated events and TempRels. It has 3 sections: TimeBank (TB), AQUAINT (AQ), and Platinum (PT). We followed the official split (i.e., TB+AQ for training and PT for testing), and further set aside 20% of the training data as the development set to tune learning rates and epochs. We also show our performance on another dataset,

[5] http://cogcomp.org/page/publication_view/834

TCR[6] (Ning et al., 2018a), which contains both temporal and causal relations and we only need the temporal part. The label set for both datasets are *before*, *after*, *equal*, and *vague*.

	Purpose	#Doc	#Events	#TempRels
TB+AQ	Train	255	8K	13K
PT	Test	20	537	837
TCR	Test	25	1.3K	2.6K

Table 1: TimeBank (TB), AQUAINT (AQ), and Platinum (PT) are from MATRES (Ning et al., 2018c) and TCR from Ning et al. (2018a).

4.2 Results and Discussion

We compare with the most recent version of CogCompTime, the state-of-the-art on MATRES.[7] Note that in Table 2, CogCompTime performed slightly different to Ning et al. (2018d): CogCompTime reportedly had F_1=65.9 (Table 2 Line 3 therein) and here we obtained F_1=66.6. In addition, Ning et al. (2018d) only reported F_1 scores, while we also use another two metrics for a more thorough comparison: classification accuracy (acc.) and temporal awareness F_{aware}, where the awareness score is for the graphs represented by a group of related TempRels (more details in the appendix). We also report the average of those three metrics in our experiments.

Table 2 compares the two different ways to handle event positions discussed in Sec. 3.1: position indicators (P.I.) and simple concatenation (Concat), both of which are followed by network (d) in Fig. 1 (i.e., without using Siamese yet). We extensively studied the usage of various pretrained word embeddings, including conventional embeddings (i.e., the medium versions of word2vec, GloVe, and FastText provided in the Magnitude package (Patel et al., 2018)) and contextualized embeddings (i.e., the original ELMo and large uncased BERT, respectively); except for the input embeddings, we kept all other parameters the same. We used cross-entropy loss and the StepLR optimizer in PyTorch that decays the learning rate by 0.5 every 10 epochs (performance not sensitive to it).

Comparing to the previously used P.I. (Dligach et al., 2017), we find that, with only two exceptions (underlined in Table 2), the Concat system saw consistent gains under various embeddings and metrics. In addition, contextualized embeddings (ELMo and BERT) expectedly improved over the conventional ones significantly, although no statistical significance were observed between using ELMo or BERT (more significance tests in Appendix).

System	*Emb.*	*Acc.*	F_1	F_{aware}	Avg.
P.I.	word2vec	63.2	67.6	60.5	63.8
	GloVe	64.5	69.0	61.1	64.9
	FastText	60.5	64.7	59.5	61.6
	ELMo	67.5	73.9	63.0	**68.1**
	BERT	68.8	73.6	61.7	68.0
Concat	word2vec	65.0	69.5	59.4	64.6
	GloVe	64.9	69.5	60.9	65.1
	FastText	64.0	68.6	60.1	64.2
	ELMo	67.7	74.0	63.3	68.3
	BERT	69.1	74.4	63.7	**69.1**
Concat+CSE	ELMo	71.7	76.7	66.0	**71.5**
	BERT	71.3	76.3	66.5	71.4
CogCompTime	-	61.6	66.6	60.8	**63.0**

Table 2: Performances on the MATRES test set (i.e., the PT section). CogCompTime (Ning et al., 2018d) is the previous state-of-the-art feature-based system. Position indicator (P.I.) and concatenation (Concat) are two ways to handle event positions in LSTMs (Sec. 3.1). Concat+CSE achieves significant improvement over CogCompTime on MATRES.

Given the above two observations, we further incorporated our common sense encoder (CSE) into "Concat" with ELMo and BERT in Table 2. We split TEMPROB into train (80%) and validation (20%). The proposed Siamese network (Fig. 1c) was trained by minimizing the cross-entropy loss using Adam (Kingma and Ba, 2014) (learning rate 1e-4, 20 epochs, and batch size 500). We first see that CSE improved on top of Concat for both ELMo and BERT under all metrics, confirming the benefit of TEMPROB; second, as compared to CogCompTime, the proposed Concat+CSE achieved about 10% absolute gains in accuracy and F_1, 5% in awareness score F_{aware}, and 8% in the three-metric-average metric, with $p < 0.001$ per the McNemar's test. Roughly speaking, the 8% gain is contributed by LSTMs for 2%, contextualized embeddings for 4%, and CSE for 2%. Again, no statistical significance were observed between using ELMo and BERT. Table 3 furthermore applies CogCompTime and the proposed Concat+CSE system on a different test set called TCR (Ning et al., 2018a). Both systems achieved better scores (suggesting that TCR is easier than MATRES), while the proposed sys-

[6] http://cogcomp.org/page/publication_view/835

[7] http://cogcomp.org/page/publication_view/844

tem still outperformed CogCompTime by roughly 8% under the three-metric-average metric, consistent with our improvement on MATRES.

System	*Emb.*	*Acc.*	F_1	F_{aware}	Avg.
CogCompTime	-	68.1	70.7	61.6	**66.8**
Concat+CSE	ELMo	80.8	78.6	69.9	**76.4**
	BERT	78.4	77.0	69.0	74.9

Table 3: Further evaluation of the proposed system, i.e., Concat (Table 3.1) plus CSE (Sec. 3.2), on the TCR dataset (Ning et al., 2018a).

5 Conclusion

Temporal relation extraction has long been an important yet challenging task in natural language processing. Lack of high-quality data and difficulty in the learning problem defined by previous annotation schemes inhibited performance of neural-based approaches. The discoveries that LSTMs readily improve the feature-based state-of-the-art CogCompTime on the MATRES and TCR datasets by a large margin not only give the community a strong baseline, but also indicate that the learning problem is probably better defined by MATRES and TCR. Therefore, we should move along that direction to collect more high-quality data, which can facilitate more advanced learning algorithms in the future.

Acknowledgements

This research is supported by a grant from the Allen Institute for Artificial Intelligence (allenai.org), the IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR) - a research collaboration as part of the IBM AI Horizons Network, and contract HR0011-18-2-0052 with the US Defense Advanced Research Projects Agency (DARPA). Approved for Public Release, Distribution Unlimited. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

References

Steven Bethard, Leon Derczynski, Guergana Savova, James Pustejovsky, and Marc Verhagen. 2015. SemEval-2015 Task 6: Clinical TempEval. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 806–814.

Steven Bethard, James H Martin, and Sara Klingenstein. 2007. Timelines from text: Identification of syntactic temporal relations. In *Semantic Computing, 2007. ICSC 2007. International Conference on*, pages 11–18. IEEE.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. SemEval-2016 Task 12: Clinical TempEval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1052–1062.

Steven Bethard, Guergana Savova, Martha Palmer, and James Pustejovsky. 2017. Semeval-2017 task 12: Clinical tempeval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1994. Signature verification using a "siamese" time delay neural network. In *The Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 737–744.

Taylor Cassidy, Bill McDowell, Nathanel Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 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 Dan Jurafsky. 2008a. Jointly combining implicit constraints improves temporal ordering. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Nathanael Chambers and Daniel Jurafsky. 2008b. Unsupervised Learning of Narrative Event Chains. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, ACL 2008*, pages 789–797.

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*, pages 173–176. Association for Computational Linguistics.

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional LSTM over dependency paths. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*, volume 2, pages 1–6.

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

Thomas G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*.

Dmitriy Dligach, Timothy Miller, Chen Lin, Steven Bethard, and Guergana Savova. 2017. Neural temporal relation extraction. volume 2, pages 746–751.

Quang Do, Wei Lu, and Dan Roth. 2012. Joint inference for event timeline construction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Brian S Everitt. 1992. *The analysis of contingency tables*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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*.

Artuur Leeuwenberg and Marie-Francine Moens. 2018. Temporal information extraction by predicting relative time-lines. *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Chen Lin, Timothy Miller, Dmitriy Dligach, Steven Bethard, and Guergana Savova. 2017. Representations of time expressions for temporal relation extraction with convolutional neural networks. *BioNLP 2017*, pages 322–327.

Hector Llorens, Nathanael Chambers, Naushad UzZaman, Nasrin Mostafazadeh, James Allen, and James Pustejovsky. 2015. SemEval-2015 Task 5: QA TEMPEVAL - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800.

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*, pages 753–760. Association for Computational Linguistics.

Yuanliang Meng and Anna Rumshisky. 2018. Context-aware neural model for temporal information extraction. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*, volume 1, pages 527–536.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of NIPS 2013*, pages 1–9.

Anne-Lyse Minard, Manuela Speranza, Eneko Agirre, Itziar Aldabe, Marieke van Erp, Bernardo Magnini, German Rigau, Ruben Urizar, and Fondazione Bruno Kessler. 2015. SemEval-2015 Task 4: TimeLine: Cross-document event ordering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 778–786.

Nasrin Mostafazadeh, Alyson Grealish, Nathanael Chambers, James Allen, and Lucy Vanderwende. 2016. CaTeRS: Causal and temporal relation scheme for semantic annotation of event structures. In *Proceedings of the 4th Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 51–61.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, pages 1038–1048, Copenhagen, Denmark.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018a. Joint reasoning for temporal and causal relations. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 2278–2288.

Qiang Ning, Hao Wu, Haoruo Peng, and Dan Roth. 2018b. Improving temporal relation extraction with a globally acquired statistical resource. In *Proceedings of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*, pages 841–851.

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

Qiang Ning, Ben Zhou, Zhili Feng, Haoruo Peng, and Dan Roth. 2018d. Cogcomptime: A tool for understanding time in natural language. In *Proceedings of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

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 the 2nd Workshop on Computing News Storylines (CNS 2016)*,

pages 47–56, Austin, Texas. Association for Computational Linguistics.

Ajay Patel, Alexander Sands, Chris Callison-Burch, and Marianna Apidianaki. 2018. Magnitude: A fast, efficient universal vector embedding utility package. *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*.

William F Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. 2014. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143.

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

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 Task 1: TEMPEVAL-3: Evaluating time expressions, events, and temporal relations. **SEM*, 2:1–9.

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*, pages 75–80. 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*, pages 189–192. Association for Computational Linguistics.

Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62. Association for Computational Linguistics.

Zhenqi Xu, Jiani Hu, and Weihong Deng. 2016. Recurrent convolutional neural network for video classification. In *Multimedia and Expo (ICME), 2016 IEEE International Conference on*, pages 1–6. IEEE.

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.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proc. the International Conference on Computational Linguistics (COLING)*, pages 2335–2344.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006*.

Shu Zhang, Dequan Zheng, Xinchen Hu, and Ming Yang. 2015. Bidirectional long short-term memory networks for relation classification. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, pages 73–78.

A Illustration of the metrics used in this paper

There are three widely-used evaluation metrics for the TempRel extraction task. The first standard metric is classification *accuracy* (Acc). The second metric is to view this task as a general relation extraction task, treat the label of *vague* for TempRel as *no relation*, and then compute the precision and recall. The third metric came into use since the TempEval3 workshop (UzZaman et al., 2013), which involves graph closure and reduction on top of the second metric, hoping to better capture how useful a TempRel system is.. Take the confusion matrix in Fig. 2 for example. The three metrics used in this paper are

1. Accuracy. $Acc = (C_{b,b} + C_{a,a} + C_{e,e} + C_{v,v})/S$.
2. Precision, recall, and F_1. $P = (C_{b,b} + C_{a,a} + C_{e,e})/S_1$, $R = (C_{b,b} + C_{a,a} + C_{e,e})/S_2$, and $F_1 = 2PR/(P + R)$.
3. Awareness score F_{aware}. Before calculating precision, perform a graph closure on the gold temporal graph and a graph reduction on the predicted temporal graph. Similarly, before calculating recall, perform a graph reduction on the gold temporal graph and a graph closure on the predicted temporal graph. Finally, compute the F_1 score based on this revised precision and recall. Since graph reduction and closure are involved in computing this metric, the temporal graphs all need to satisfy the global transitivity constraints of temporal relations (e.g., if A happened *before* B, and B happened *before* C, then C cannot be *before* A).

In this paper, we also report the average of the three metrics above, which we call *three-metric-average*.

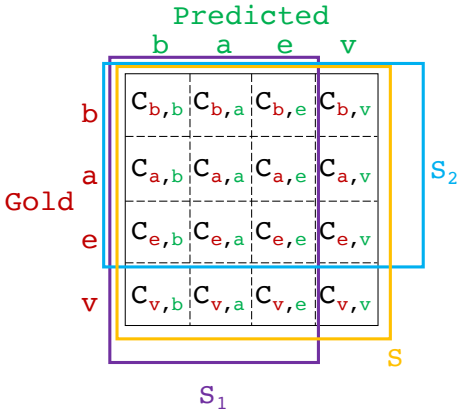

Figure 2: An example confusion matrix, where the four labels are *before* (b), *after* (a), *equal* (e), and *vague* (v), respectively. The variables, S, S_1, and S_2, are the summation of all the numbers in the corresponding area. (This figure is better viewed in color)

B Significance test for Tables 2-3

In Table 2, we mainly compared the performance of position indicator (P.I.) and simple concatenation (Concat), using 5 different word embeddings and 3 metrics, so there were 15 performances for both P.I. and Concat. Under the paired t-test, Concat is significantly better than P.I. with $p < 0.01$.

Another observation we had in Table 2 was that contextualized embeddings, i.e., ELMo and BERT, were much better than conventional ones, i.e., word2vec, GloVe and FastText. For both P.I. and Concat, we found that the difference between contextualized embeddings and conventional embeddings was significant with $p < 0.001$ under the McNemar's test (Everitt, 1992; Dietterich, 1998); however, between the two contextualized embeddings, ELMo and BERT, we did not see a significant difference, although it has been reported that in many *other* tasks, that BERT is better than ELMo.

In Table 3, we further improved Concat using the proposed common sense encoder (CSE). Under the McNemar's test, Concat+CSE was significantly better than Concat with $p < 0.001$, no matter either ELMo or BERT was used. Again, no significant difference was observed between ELMo and BERT.

Finally, since Concat+CSE improved over CogCompTime by a large margin either on MATRES or on TCR, it was not surprising to see that the proposed Concat+CSE is significantly better than CogCompTime with $p < 0.001$ as well.