Acquiring Structured Temporal Representation via Crowdsourcing: A Feasibility Study

Yuchen Zhang
Brandeis University
yuchenz@brandeis.edu

Nianwen Xue
Brandeis University
xuen@brandeis.edu

Abstract

Temporal Dependency Trees are a structured temporal representation that represents temporal relations among time expressions and events in a text as a dependency tree structure. Compared to traditional pair-wise temporal relation representations, temporal dependency trees facilitate efficient annotations, higher inter-annotator agreement, and efficient computations. However, annotations on temporal dependency trees so far have only been done by expert annotators, which is costly and time-consuming. In this paper, we introduce a method to crowdsource temporal dependency tree annotations, and show that this representation is intuitive and can be collected with high accuracy and agreement through crowdsourcing. We produce a corpus of temporal dependency trees, and present a baseline temporal dependency parser, trained and evaluated on this new corpus.

1 Introduction

Temporal relation extraction is an important NLP task for a range of downstream applications, such as question answering, summarization, and storyline generation. This task has attracted a significant amount of research interest (Pustejovsky et al., 2003a; Verhagen et al., 2007, 2010; Uz-Zaman et al., 2012; Bethard et al., 2016, 2017; Dligach et al., 2017; Leeuwenberg and Moens, 2017; Ning et al., 2017, 2018a,b; Zhang and Xue, 2018a,b). One practical challenge in temporal relation extraction is to represent the temporal relations in a text in a way that is feasible for manual annotation and producing training data for machine learning models. Given a text of n events and time expressions, there are $\binom{n}{2}$ possible relations if the temporal relation between all pairs of events and time expressions is annotated. This quickly becomes infeasible even for a text of modest length. One way to address this problem is to represent the temporal relations in a text as a Temporal Dependency Tree (TDT) structure (Zhang and Xue, 2018b). TDT models all time expressions and events in a text as "nodes" in a dependency tree, and temporal relations between each time/event and its parent time/event as "edges" in the tree. Figure 1 gives an example text and its TDT. Each (parent, child) pair in Figure 1 is annotated with a temporal relation. The number of temporal relations that need to be annotated in a text is therefore linear to the number of events and time expressions in a text, making the annotation task feasible. At the same time, additional temporal relations can be inferred as needed based on the TDT structure. For example, in Figure 1 since "1918" *includes* the "born" event and "1929" *includes* the "won" event, it can be inferred that the "born" event occurred *before* the "won" event.

By providing annotators with detailed guidelines and training them in multiple iterations, Zhang and Xue (2018b) have shown that the TDT representation can be annotated with high inter-annotator agreement. Zhang and Xue (2018a) further show that a neural ranking model can be successfully trained on the corpus. However, this "traditional" approach to annotation is time-consuming and expensive. The question we want to answer in this paper is whether TDT can be performed with crowdsourcing, a method that has gained popularity as a means to acquire linguistically annotated data quickly and cost-effectively for NLP research.

Crowdsourcing has been used to annotate data for a wide range of NLP tasks that include question answering, word similarity, text entailment, word sense disambiguation, machine translation, information extraction, summarization, and semantic role labeling (Snow et al., 2008; Finin et al., 2010; Zaidan and Callison-Burch, 2011; Lloret et al., 2013; Rajpurkar et al., 2018). The

*Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 178–185
Minneapolis, June 6–7, 2019.

key to acquiring high quality data via crowdsourcing is to make sure that the tasks are intuitive or can be decomposed into intuitive subtasks. In this paper, we show that it is possible to acquire high quality temporal dependency structures through crowdsourcing, and that a temporal dependency parser can be successfully trained on crowdsourced TDTs.

The rest of the paper is organized as follows. We first explain in detail how we set up this dependency tree crowdsourcing annotation task (§2). In (§3) we present experimental results that show that if temporal dependency structures are broken into smaller subtasks, high inter-annotator agreement can be achieved. In (§4), we show that crowdsource data can be used to successfully train temporal dependency parsers, including an attention-based neural model (§4). We discuss related work (§5) and conclude with future work (§6).

The main contributions of this paper are: (1) we introduce an effective approach to crowdsource structured temporal annotations, a relatively complex annotation task; (2) we build an English temporal dependency tree corpus through crowdsourcing that we plan to make publicly available; and (3) we experiment with automatic temporal dependency parsers on this new corpus and report competitive results.

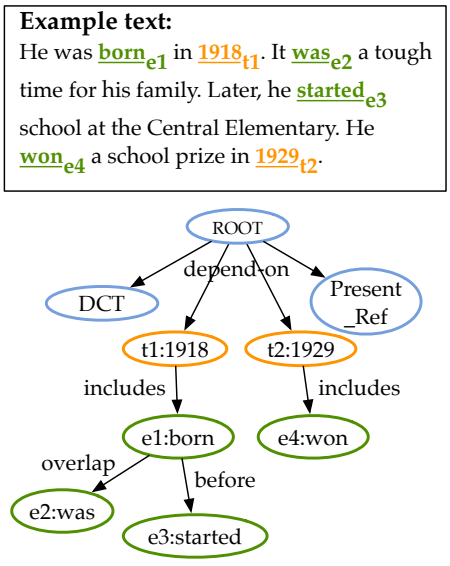

Figure 1: Example text and its temporal dependency tree. The nodes in blue are meta nodes (e.g., document creation time "DCT", present reference time "Present_Ref", etc.), the nodes in orange are time expressions, and the nodes in green are events.

2 Crowdsourcing Tasks Setup

2.1 Data Setup

Our TDT annotations are performed on top of the TimeBank corpus (Pustejovsky et al., 2003b), with time expressions and events already extracted. Following (Zhang and Xue, 2018b), we focus only on events that are matrix verbs (i.e. main verbs) in a sentence. In order to extract matrix verbs, we use the gold constituent trees for the part of TimeBank that overlaps with the Penn Treebank, and parse the rest of TimeBank with the Berkeley Neural Parser (Kitaev and Klein, 2018). All time expressions in TimeBank are kept.

To facilitate quality control in crowdsourcing and agreement evaluation, we distinguish two subsets of the TimeBank dataset: (1) TB-small is a small subset of 10 short Wall Street Journal news documents with 59 matrix verbs. (2) TB-dense consists of the same 36 documents as in the TimeBank-Dense corpus (Cassidy et al., 2014). It contains 654 matrix verbs. TB-small and TB-dense are annotated by both crowd workers and experts.

2.2 Annotation Tasks

We set up two annotation tasks. The first is full temporal dependency tree annotation, where crowd workers need to annotate both the dependency tree structure and the temporal relations between each parent and child. The second is relation-only annotation, where crowd workers are given the gold temporal dependency trees and their job is just to label the temporal relation for each parent-child pair.

2.3 Crowdsourcing Design

For the full temporal dependency tree annotation, in order to simplify the questions/instructions to crowd workers, we split the task of annotating a full dependency tree into (1) finding the "parent" for each individual event, and then (2) deciding the temporal relation between the "parent" and the event. A crowd worker is given a text with a highlighted target event and a list of candidate parent time expressions and events. The job of the crowd worker is to select one parent from the given list of candidates, and label the temporal relation between the parent and the target event. For relation-only annotation, a crowd worker is presented a text with the target event and its parent highlighted. The job of the worker is to decide the temporal

relation between the two. See Appendix A for example crowdsourcing instructions and questions.

Following standard crowdsourcing quality control, we perform a qualifying test on both annotation tasks. Any crowd worker who wants to work on these tasks needs to complete annotations on TB-small and reach at least 70% accuracy against the expert gold annotation. We also perform a surviving test on the relation-only annotation task. Crowd workers have to maintain at least a cumulative accuracy of 70% for their annotation. Workers with a lower accuracy will get blocked from the task and all of their annotations will be discarded. Every annotation is completed by at least 3 annotators and the majority vote is the final annotation.

3 Annotation Experiments

Crowdsourcing annotations on the full TimeBank corpus was performed. We report Inter-Annotator Agreement (IAA) scores in Table 1.

	UAA	LOA	LAA
Crowd v.s. Expert	.82	.83	.53
Crowd IAA	.81	.85	.52

Table 1: Inter-Annotator Agreement scores between crowdsourced and expert annotations, and IAAs among crowd worker annotations.

First, crowdsourced majority annotations on TB-dense are evaluated against expert annotations, representing the quality of the crowdsourced data. For this comparison, the standard dependency parsing evaluation metrics (Kübler et al., 2009) are used as our IAA scores: structure-only annotation subtask is evaluated with the Unlabeled Attachment Agreement (UAA) score, relation-only annotation subtask is evaluated with the Label Only Agreement (LOA) score, and full pipeline annotation is evaluated with the Labeled Attachment Agreement (LAA) score.

Second, crowd worker annotations are compared against each other, indicating the difficulty, consistency, and confidence of the crowdsourced data. Since crowd workers annotate isolated events/times instead of full dependency structures, the standard dependency parsing metrics are not applicable for this comparison[1]. Therefore, we adopt the Worker Agreements With Aggregate (WAWA) metric (Ning et al., 2018a) as our IAA scores. WAWA indicates the average number of crowd worker responses agreed with the aggregate answer (i.e. majority aggregation for each annotation instance), representing the agreements among crowd workers and how consistent their annotations are with each other.

As shown in the table, high accuracies and agreements are achieved for both the subtasks of structure annotation and relation-only annotation (above 80%).

Statistics on our corpus and other similar TimeBank-based temporal relation corpora are presented in Table 2. As the number of temporal relations is linear to the number of events and time expressions in a text, fewer temporal relations need to be annotated in our corpus. In comparison, the recently crowdsourced temporal structure corpus MATRES (Ning et al. (2018a), see Section 5 for more details) only annotates verb events in a document while TB-dense annotates a larger number of time expressions and events in a much smaller number of documents. Our corpus retains the full set of TimeBank time expressions and covers comparable number of events as MATRES. We pay $0.01 for each individual annotation and the entire TimeBank TDT annotation cost about $300 in total.

	Docs	Timex	Events	Rels
TimeBank	183	1,414	7,935	6,418
TB-Dense	36	289	1,729	12,715
MATRES	275	-	1,790	13,577
This work	183	1,414	2,691	4,105

Table 2: Documents, timex, events, and temporal relation statistics in various temporal corpora.

4 System Experiments

We experiment with a state-of-the-art attention-based neural temporal dependency parser (Zhang and Xue, 2018a)[2] on our newly annotated data. Our training data consists of two parts. The first part is the crowdsourced temporal dependency annotations over the TimeBank documents (excluding documents that are in the dev and test sets in the TimeBank-Dense corpus[3]). The second part is our expert-annotated TDTs on the TimeBank-Dense training set documents. The parser is tuned

[1] And for the same reason, Cohen's kappa and Fleiss' kappa scores are not applicable here either.

[2] https://github.com/yuchenz/tdp_ranking

[3] Standard TimeBank-Dense train/dev/test split can be found in Cassidy et al. (2014).

and evaluated on our expert TDT annotations on the TimeBank-Dense dev and test sets, respectively. This neural model represents words with bi-LSTM vectors and uses an attention-based mechanism to represent multi-word time expressions and events.

We also experiment with two baseline parsers from Zhang and Xue (2018a): (1) a simple baseline that takes an event's immediate previous time expression or event as its parent and assigns the majority "overlap" as the temporal relation between them; and (2) a logistic regression model that represents time expressions and events with their time/event type features, lexical features, and distance features. Table 3 shows the performance of these systems on our data.

Model	Structure -only F		Structure + Relation F	
	dev	test	dev	test
Simple Baseline	.43	.42	.15	.18
LogReg Baseline	.64	.70	.26	.29
Neural Model	**.75**	**.79**	**.53**	**.60**

Table 3: Parsing results of the simple baseline, logistic regression baseline, and the neural temporal dependency model.

Improved performance over the simple baseline with both the LogReg system and the Neural system show that temporal dependency information can be learned from this crowdsourced corpus. Comparisons between the LogReg baseline and the Neural model show that the Neural model adapts better to new data sets than the LogReg model with manually-crafted language-specific features.

5 Related Work

Although crowdsourcing is widely used in other NLP tasks, there have been only a few temporal relation annotation tasks via crowdsourcing. The first attempt on crowdsourcing temporal relation annotations is described in Snow et al. (2008). They selected a restricted subset of verb events from TimeBank and performed strict before/after temporal relation annotation through crowdsourcing. They reported high agreements showing that simple temporal relations are crowdsourceable. Ng and Kan (2012) adopts the TimeML representation from the TimeBank, and crowdsourced temporal annotations on news articles crawled from news websites. Their experiments show that the large crowdsourced data improved classifier performance significantly. However, both of these works focused on pair-wise temporal relations and didn't experiment with crowdsourcing more complex temporal structures. Vempala and Blanco (2018) uses a crowdsourcing approach to collect temporal and spatial knowledge. However, they first automatically generated such knowledge and then used crowdsourcing to either validate or discard these automatically generated information, and crowdsourcing was not utilized to do annotation from scratch.

Ning et al. (2018a) proposed a "multi-axis" representation of temporal relations in a text, and published the MATRES corpus by annotating "multi-axis" temporal structures on top of the TempEval-3 data through crowdsourcing. In this representation, events are annotated on different "axes" according to their eventuality types, and for events on the same axis, pair-wise temporal relations are annotated. Their annotation task is broken down to two smaller subtasks too. In the first subtask, crowd workers annotate whether an event is on a given axis. In the second subtask, crowd workers annotate the temporal relations between pairs of events on the same axis. The main differences between their work and ours are as follows. First, they only model events, excluding time expressions which are important temporal components in text too. Second, our temporal dependency tree representation is very different from their multi-axis temporal representation, which requires different crowdsourcing task designs. In their first subtask, crowd workers need to distinguish different eventuality types, while our annotation experiments show that crowd workers can also consistently recognize "parents" as defined in Zhang and Xue (2018b) for given events.

6 Conclusion and Future Work

In this paper, we introduce a crowdsourcing approach for acquiring annotations on a relatively complex NLP concept – temporal dependency structures. We build the first English temporal dependency tree corpus through high quality crowdsourcing. Our system experiments show that competitive temporal dependency parsers can be trained on our newly collected data. In future work, we plan to crowdsource more TDT data across different domains.

References

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.

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)*, volume 2, pages 501–506.

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

Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. 2010. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia. Association for Computational Linguistics.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127.

Tuur 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*, pages 1150–1158.

Elena Lloret, Laura Plaza, and Ahmet Aker. 2013. Analyzing the capabilities of crowdsourcing services for text summarization. *Language resources and evaluation*, 47(2):337–369.

Jun-Ping Ng and Min-Yen Kan. 2012. Improved temporal relation classification using dependency parses and selective crowdsourced annotations. *Proceedings of COLING 2012*, pages 2109–2124.

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

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

Qiang Ning, Zhongzhi Yu, Chuchu Fan, and Dan Roth. 2018b. Exploiting partially annotated data in temporal relation extraction. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 148–153.

James Pustejovsky, José M Castano, Robert Ingria, Roser Sauri, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R Radev. 2003a. Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

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

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics.

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. Tempeval-3: Evaluating events, time expressions, and temporal relations. *arXiv preprint arXiv:1206.5333*.

Alakananda Vempala and Eduardo Blanco. 2018. Annotating temporally-anchored spatial knowledge by leveraging syntactic dependencies. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

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, 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.

Omar F Zaidan and Chris Callison-Burch. 2011. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1220–1229. Association for Computational Linguistics.

Yuchen Zhang and Nianwen Xue. 2018a. Neural ranking models for temporal dependency structure parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3339–3349.

Yuchen Zhang and Nianwen Xue. 2018b. Structured interpretation of temporal relations. In *Proceedings of 11th edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

A Appendix: Example Crowdsourcing Instructions and Questions

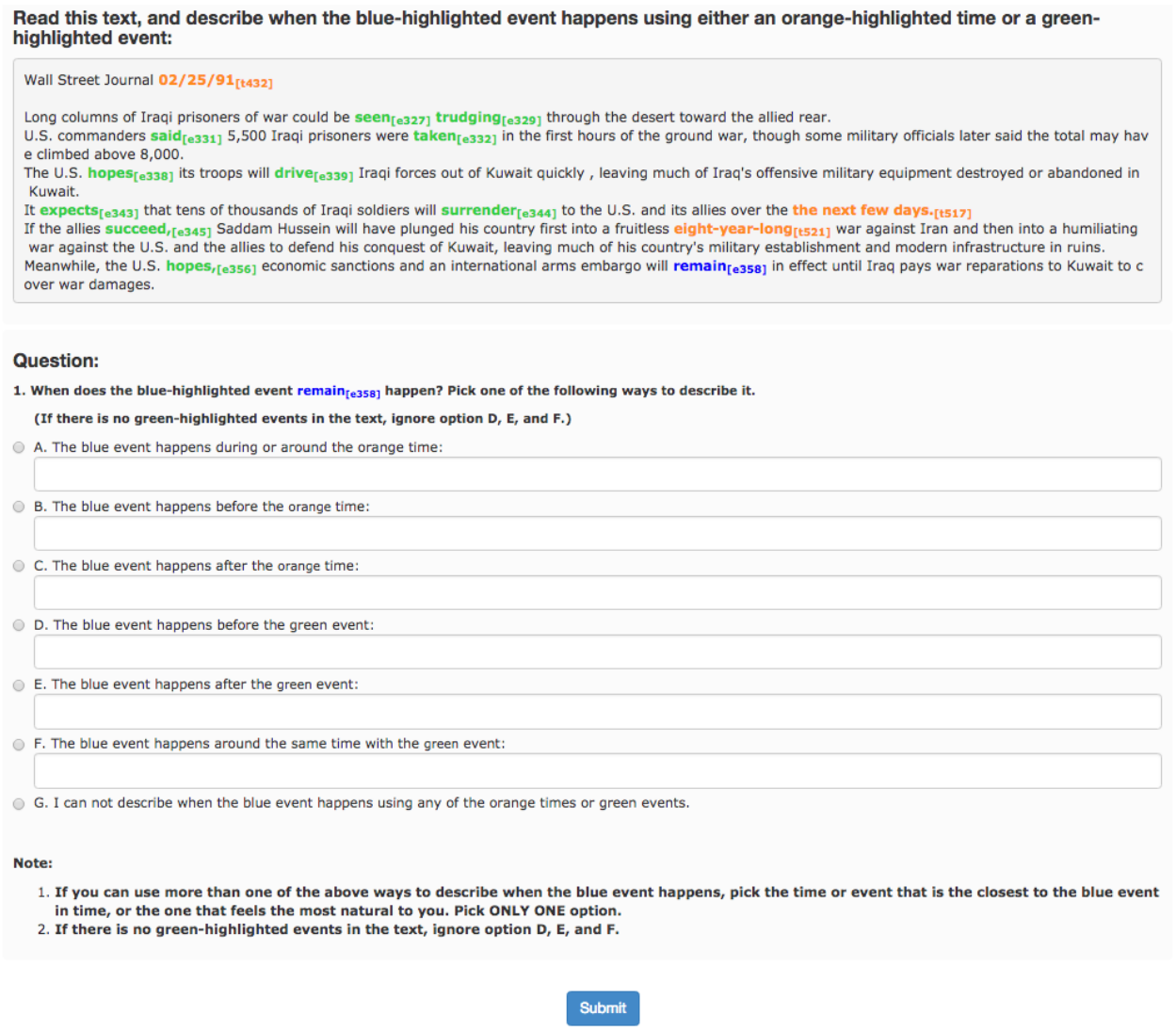

Figure 2: Example crowdsourcing instructions and questions for full structure and relation annotation.

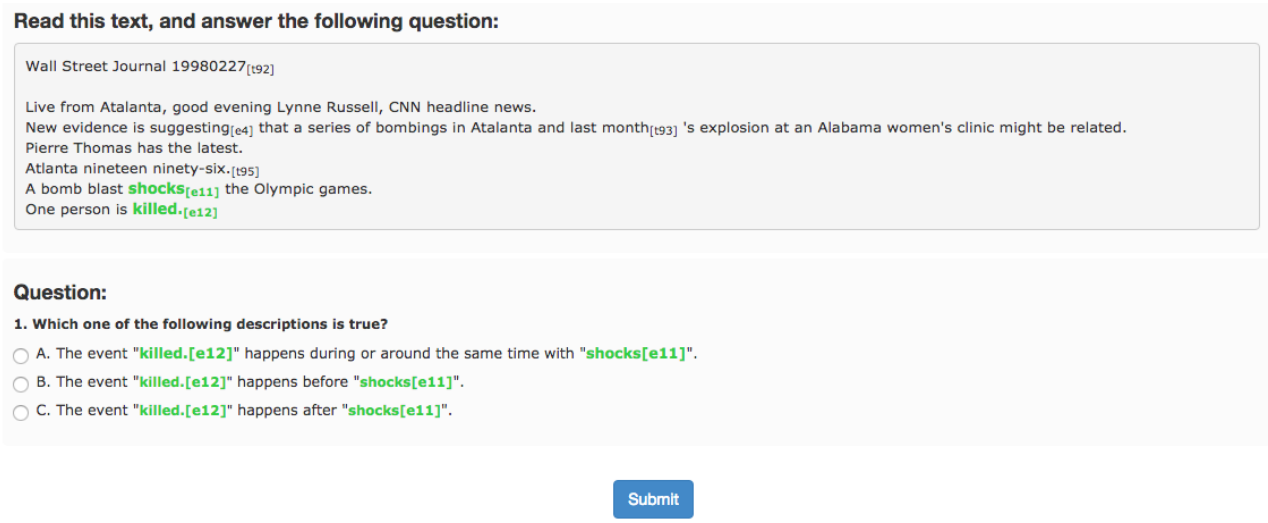

Figure 3: Example crowdsourcing instructions and questions for relation only annotation.