Identifying Temporal Relations by Sentence and Document Optimizations

Katsumasa Yoshikawa[1] *Masayuki Asahara*[2] *Ryu Iida*[3]
(1) IBM Research, Tokyo, Japan
(2) National Institute for Japanese Language and Linguistics, Japan
(3) Tokyo Institute of Technology, Japan
katsumasay@gmail.com, masayu-a@ninjal.ac.jp, ryu-i@cl.cs.titech.ac.jp

Abstract

This paper presents a temporal relation identification method optimizing relations at sentence and document levels. Temporal relation identification is to identify temporal orders between events and time expressions. Various approaches of this task have been studied through the shared tasks TempEval (Verhagen et al., 2007, 2010). Not only identifying each temporal relation independently, some works also try to find multiple temporal relations jointly by logical constraints in Integer Linear Programming (Chambers and Jurafsky, 2008; Do et al., 2012) or Markov Logic Networks (Yoshikawa et al., 2009; Ling and Weld, 2010; Ha et al., 2010).

Though previous joint approaches optimize temporal relations in an entire document, we first optimize our model at sentence level and then extend it to document level. We consider that different types of temporal relations require different types of optimizations. By evaluating our sentence and document optimized model on the TempEval-2 data, we show that our approaches can achieve competitive performance in comparison to other state-of-the-art systems. We find that the sentence and document optimized model has strong tasks in TempEval-2, respectively.

Keywords: temporal relation identification, time, markov logic, semantic role.

Keywords in L_2: .

Proceedings of COLING 2012: Posters, pages 1371–1380,
COLING 2012, Mumbai, December 2012.

1 Introduction

Recent work on temporal analysis has focused on several sub-tasks, such as event (time) recognition, event (time) classification, time normalization, and temporal relation identification. Temporal relation identification (or temporal ordering) has especially been given much attention among studies in recent years. Since temporal orders often effect causal relations (*cause and effect*), identifying them is an essential task for deep language understanding.

Various approaches to temporal ordering have been proposed through shared tasks called *TempEval* (Verhagen et al., 2007, 2010). TempEval-2 involved four temporal ordering tasks corresponding to four types of temporal relations: between events and time expressions in a sentence (Task C), [1] between events of a document and the document creation time (DCT) (Task D), between two main events in two consecutive sentences (Task E), and between two events where one event syntactically dominates the other event (Task F).

Figure 1 shows an example of temporal relations. This example has five events and one time expression and includes the four types of relations corresponding to Tasks C, D, E, and F of TempEval-2. The temporal relations (TLINKs) are annotated as shown in Table 1 and we have to estimate these TLINK labels such as BEFORE, OVERLAP, and AFTER.

task	relation
Task C	e53 (change) OVERLAP t10 (a couple of years)
Task D	e50 (think) OVERLAP t0 (DCT)
Task D	e52 (think) OVERLAP t0 (DCT)
Task D	e53 (change) AFTER t0 (DCT)
Task E	e50 (think) OVERLAP e57 (reposition)
Task F	e50 (think) OVERLAP e51 (gloomy)
Task F	e52 (think) BEFORE e53 (change)

Table 1: Temporal Relations (TLINKs) in Figure 1

While the first studies handled this task as local classification problems (Boguraev and Ando, 2005; Mani et al., 2006), some recent works regard temporal relation identification as a global optimization problem in an entire document. Global optimization approaches take into account several relations and jointly identify all relations within a document. In order to ensure the consistencies among relations, previous work exploited global approaches with *transitivity* constraints in Integer Linear Programming (Chambers and Jurafsky, 2008; Do et al., 2012) or Markov Logic (Yoshikawa et al., 2009; Ling and Weld, 2010).

In this paper, we propose a new approach to temporal relation identification by optimizing temporal relations at sentence or document levels. We have two motivations to improve conventional global approaches. First, we consider that identifying each type of temporal relations requires different type of optimization. Optimizing at sentence level are suitable for some types of TLINKs rather than optimizing at document level. In addition, optimizing at sentence level allows us to effectively utilize rich syntactic and semantic features.

Secondly, it is difficult to construct global model by controlling many global constraints simultaneously. It is well-known that overly strong constraints hurt the performance of

[1] Note, Task C of TempEval-2 is further restricted by requiring that either the event syntactically dominates the time expression or the event and time expression occur in the same noun phrase.

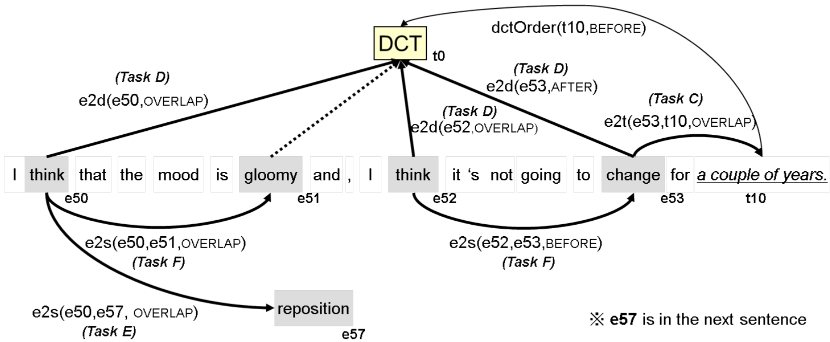

Figure 1: Temporal Relations

identification. Ha et al. adopted a Markov Logic for identifying temporal relations in TempEval-2 (Ha et al., 2010), but their global model at document level could not improve upon their local model in the majority of tasks. Especially in Tasks D and F, the results of their global model were much worse than those of their local model. According to their analysis, utilizing *hard* constraints caused many errors. Though Markov Logic (Richardson and Domingos, 2006) makes it possible to utilize both *hard* and *soft* constraints, both types of constraints are very sensitive and hard to be controlled well. It is possible for even a *soft* constraint to drastically improve (or hurt) the performance of temporal relation identification. Thus, finding effective constraints sometimes becomes more difficult task than feature selection for general machine learning.

To effectively control the sensitivity of constraints, we need first to reduce our large problem down to a smaller one. We construct our models optimized at two levels, sentence and document. First, we create a model which optimizes temporal relations at sentence level and then extend it to a model for document optimization. The *sentence-optimized* model focuses on only the temporal relations within the same sentence. The *document-optimized* model covers all relations in a document. For both models, we employ Markov Logic and control sensitive soft constraints. Each optimized model has respective advantages. The sentence-optimized model is good at handling TLINKs inside sentences (Tasks C and F) by exploiting rich syntactic and semantic features. The document-optimized model is strong at solving TLINKs beyond sentence boundaries (Task E). We evaluate our models on TempEval-2 data. As a result of advantages above, our sentence and document-optimized models outperform TempEval-2 participants on Tasks C and E, respectively.

2 Proposed Method

In this section we introduce the Markov Logic Network designed for our global models. Marlov Logic is a combination of first-order logic and Markov Networks (Richardson and Domingos, 2006). It can be understood as a formalism that extends first-order logic to allow formulae that can be violated with some penalties. From an alternative point of view, it is an expressive template language that uses first order logic formulae to instantiate Markov Networks of repetitive structure. Unfortunately, we do not have enough space to explain the details about Markov Logic. Since we can refer various previous works with Markov Logic (Singla and Domingos, 2006; Poon and Domingos, 2007, 2008), this section focuses on model constructions by Markov Logic Network.

First, we define four *hidden* predicates, corresponding to Tasks C, D, E, and F listed in Table 2. We do not know their extensions at test time. Our *observed* predicates reflect observed information extracted from the corpus (such as words, POS, etc.). Note that the TempEval data also contains temporal relations that were not supposed to be predicted. These relations are represented using an *observed* predicate: $dctOrder(t, R)$ for the relation R between a time expression t and a fixed DCT. An illustration of all "temporal" predicates can be seen in Figure 1.

In the following parts, we describe our three models: (1) *local* model which solves each task independently, (2) *sentence-optimized* model which targets Tasks C, D and F by sentence level optimization, and (3) *document-optimized* model which solves Tasks C-F by document level optimization. The sentence-optimized model also includes local features same as local model. The document-optimized model utilizes both features of local and sentence-optimized. So, the document-optimized model is a full version which contains all the local and global features.

2.1 Local Model

Our local model utilizes only local features and solves each task independently as a local classification problem. In the Markov Logic framework, local features are represented as *local formulae*. We say that a formula is local if it only considers the *hidden* temporal relation of a single event-event, event-time or event-DCT pair. The formulae in the second class are *global*: they involve two or more temporal relations at the same time.

The local features are based on features employed in previous work (UzZaman and Allen, 2010; Llorens et al., 2010) and are listed in Table 3. In order to illustrate how we implement each feature as a formula, we show a simple example. Consider the tense-feature for Task F. For this feature we introduce a predicate $tense(e, te)$ that denotes the tense te for an event e. In Table 3, this feature corresponds to the second row "EVENT-tense". For Task F, we employ the tense combinations of two events (e1 x e2). Then we add a formula such as

$$tense(e1, +te1) \wedge tense(e2, +te2) \Rightarrow e2s(e1, e2, +R) \tag{1}$$

which represents the properties of combinations between tense and event-event relations. Note, "+" sign means that the ground formulae derived from this formula have different weights for each label. Formula (1) are grounded for all possible combinations of tenses and temporal relations such as

$$tense(e1, \mathrm{PRESENT}) \wedge tense(e2, \mathrm{FUTURE}) \Rightarrow e2s(e1, e2, \mathrm{BEFORE}) \tag{2a}$$

$$tense(e1, \mathrm{PRESENT}) \wedge tense(e2, \mathrm{PRESENT}) \Rightarrow e2s(e1, e2, \mathrm{BEFORE}). \tag{2b}$$

This type of "template-based" formulae generation can be performed automatically by the Markov Logic Engine. Markov Logic Engine assigns different weights to Formulae (2a) and (2b). For example, Formula (2a) possibly obtains higher weight than Formula (2b). Actually, Formula (2a) matches the example in Figure 1 (Consider it replacing $e1$ with e52 and $e2$ with e53, respectively).

For Tasks E and F, there is no time expression directly related to the targeted events. So, we employ the time expressions which are syntactically dominated by the events or which are identified as arguments of them by a semantic role labeler. We also apply semantic role features (SR-Label) as a rich semantic feature introduced in (Llorens et al., 2010).

Task	predicate	description
Task C	$e2t(e,t,R)$	temporal relation between an event e and a time expression t is R
Task D	$e2d(e,R)$	temporal relation between an event e and DCT is R
Task E	$e2e(e1,e2,R)$	temporal relation between two main events of the adjacent sentences, $e1$ and $e2$ is R
Task F	$e2s(e1,e2,R)$	temporal relation between two events where one event $e1$ syntactically dominates the other event $e2$ is R

Table 2: Hidden Predicates and Targeted Temporal Relations

Feature	MLN predicate	C	D	E	F
EVENT-class	$class(e,c)$	Y	Y	e1 x e2	e1 x e2
EVENT-tense	$tense(e,te)$	Y	Y	e1 x e2	e1 x e2
EVENT-aspect	$aspect(e,a)$	Y	Y	e1 x e2	e1 x e2
EVENT-tense-aspect	$tense(e,te)\&aspect(e,a)$	Y	Y	e1 x e2	e1 x e2
EVENT-polarity	$polarity(e,p)$	Y	Y	e1 x e2	e1 x e2
EVENT-stem	$stem(e,s)$	Y	Y	e1 x e2	e1 x e2
EVENT-word	$wordEvent(e,w)$	Y	Y	Y	Y
EVENT-POS	$eventPos(e,p)$	Y	Y	e1 x e2	e1 x e2
TIME-type	$type(t,ty)$	Y	Y	Y	Y
TIME-value	$value(t,ty)$	Y	Y	Y	Y
TIME-word	$wordTime(t,w)$	Y			
TIME-POS	$posTime(t,p)$	Y			
TIME-DCT order	$dctOrder(t,r)$	Y	Y	Y	Y
Dependency-Word	$depWord(e(\text{or t}),w)$	Y	Y	Y	
Dependency-POS	$depPos(e(\text{or t}),p)$	Y	Y	Y	
Dependency-Label	$dep(e1,e2(\text{or t}),l)$	Y			Y
SR-Word	$srlWord(e(\text{or t}),w)$	Y	Y	Y	Y
SR-POS	$srlPOS(e,(\text{or t}),p)$	Y	Y	Y	Y
SR-Label	$srl(e1,e2(\text{or t}),l)$	Y			Y

Table 3: Local Features

2.2 Sentence-optimized Model

The original Markov Logic approach to temporal relation identification solves problems in a document-by-document manner (Yoshikawa et al., 2009). On the other hand, our sentence-optimized model is a global model optimized at sentence level and solves problems in a sentence-by-sentence manner. Optimizing at sentence level gives us at least two advantages: (1) it allows us to keep a problem simple and control sensitive constraints well, (2) we can exploit rich syntactic and semantic features and constraints. The first advantage is an original motivation of sentence-optimized model. Even though our models have only several global formulae, they are very sensitive and it is difficult for us to control them well. In addition, since the optimization at document level is sometimes computationally hard, we cannot employ large number of features. The sentence-optimized model provides us with a solution to overcome these difficulties.

Though we need to solve four types of relations in TempEval-2, the sentence-optimized model focuses on only three of them corresponding to Tasks C, D, and F. Our global formulae are designed to enforce consistency between the three *hidden* predicates $e2t, e2d,$

and $e2s$. In the following parts, we show the set of formula templates we use to generate the global formulae. Here each template produces several instantiations, one for each assignment of temporal relation classes to the variables R1, R2, etc.

Our global formulae mainly employ DCT as a reference time. First, the global formulae between Tasks C and D are,

$$dctOrder(t, +R1) \wedge e2t(e, t, +R2) \Rightarrow e2d(e, +R3) \tag{3}$$

$$dctOrder(t, +R1) \wedge e2d(e, +R2) \Rightarrow e2t(e, t, +R3) \tag{4}$$

which ensure the consistency between $e2t$ and $e2d$. We implement these formulae as *soft* constraints. If a possible world violates some soft constraints, they give it some penalties with corresponding weights. In contrast to hard constraints, a possible world which causes some violation of soft constraints is *less* probable (not prohibited). Soft constraints are good way to control ambiguous transition rules. For example, Formula (4) can be instantiated as,

$$dctOrder(t, \mathrm{BEFORE}) \wedge e2d(e, \mathrm{AFTER}) \Rightarrow e2t(e, t, \mathrm{AFTER}), \tag{5a}$$

$$dctOrder(t, \mathrm{BEFORE}) \wedge e2d(e, \mathrm{AFTER}) \Rightarrow e2t(e, t, \mathrm{OVERLAP}) \tag{5b}$$

which possibly hold but not always do. [2] Fortunately, this type of soft rule poses no problem for Markov Logic: after training, Formula (5b) will have a lower weight than Formula (5a).

The global formulae for Tasks D and F are,

$$e2d(e1, +R1) \wedge e2d(e2, +R2) \Rightarrow e2s(e1, e2, +R3), \tag{6}$$

$$e2d(e1, +R1) \wedge e2s(e1, e2, +R2) \Rightarrow e2d(e2, +R3) \tag{7}$$

which enforce the consistency between $e2d$ and $e2s$. Formula (6) is especially effective because the results of $e2d$ (Task D) are much higher than $e2s$ (Task F).

Since some events share the same time expression, we add the following global formulae,

$$e2t(e1, t, +R1) \wedge e2t(e2, t, +R2) \Rightarrow e2s(e1, e2, +R3), \tag{8}$$

$$e2t(e1, t, +R1) \wedge e2s(e1, e2, +R2) \Rightarrow e2t(e2, t, +R3) \tag{9}$$

which ensure the consistency between Tasks C and F.

With event-argument relations (semantic roles), we construct some more global formulae. For $e2d$, we assume that the relations sharing the same time expression have the same relations. Such properties can be expressed as,

$$srl(e1, t, \mathrm{AM\text{-}TMP}) \wedge srl(e2, t, \mathrm{AM\text{-}TMP}) \Rightarrow e2d(e1, R1) \wedge e2d(e2, R2) \wedge R1 = R2. \tag{10}$$

Likewise, for $e2t$, we assume that the relations sharing the same time expression affect each other:

$$srl(e1, t, \mathrm{AM\text{-}TMP}) \wedge srl(e2, t, \mathrm{AM\text{-}TMP}) \wedge e2t(e1, t, +R1) \Rightarrow e2t(e2, t, +R2). \tag{11}$$

It is easy for the sentence-optimized model to implement much more features and constraints as in other tasks (Meza-Ruiz and Riedel, 2009; Yoshikawa et al., 2011).

2.3 Document-optimized Model

The last model is the method which optimizes problems at document level. We add another hidden predicate $e2e$ which handles Task E of TempEval-2. Note, in order to pursue computational efficiency, we should deal with $e2t, e2d$, and $e2s$ as *observed* predicates and solve only $e2e$ in this phase. However, we only add a few global formulae and can construct

[2] Formula (5b) is instantiated by the relations in Figure 1

a global model which jointly optimizes four tasks. We no longer change the formulae we constructed for the sentence-optimized model. So, what we have to make is constructing global formulae for only $e2e$. Transition rules also apply to $e2e$ in a similar way to $e2s$.

$$e2d(e1, +R1) \wedge e2d(e2, +R2) \Rightarrow e2e(e1, e2, +R3) \quad (12)$$

$$e2d(e1, +R1) \wedge e2e(e1, e2, +R2) \Rightarrow e2d(e2, +R3) \quad (13)$$

which represent the transitive relations between $e2e$ and $e2d$. We can add more constraints such as relations between $e2e$ and $e2t$ or $e2s$. However, these constraints sometimes cause error propagations because $e2t$ and $e2s$ are difficult to solve and possibly include many errors. Thus, we add only the two formulae above for document-optimized model.

3 Experiments and Results

With our experiments we want to answer two questions: (1) do optimizations at sentence and document levels help to increase the overall accuracy of temporal relation identification? (2) How does our approach compare to the state-of-the-art results? In the following we will first present the experimental set-up we chose to answer these questions.

In our experiments we used the test and training sets provided by the TempEval-2 shared task. The language we target is only English. We further split the original training data into a training and a development set, used for optimizing parameters and formulae. We employ 147 documents for training, 15 for development, and 20 for testing.

For feature generation we use the following tools. POS tagging is performed with the stanford-POS-tagger; [3] as parser and semantic role labeler for our syntactic and semantic features we employ LTH semantic parser. [4] As a Markov Logic Engine, we employ *Markov thebeast*, which is tailored for NLP applications. For evaluation of temporal relation identification, we employ an accuracy-based scoring of TempEval-2. It is a simple metric: the number of correct answers divided by the number of answers.

3.1 Impact of Sentence and Document Optimizations

Here we present our comparison of three models. Let us show the results on TEST set in Table 4. We can find four columns corresponding to Tasks C–F, for our models of "Local", "Sentence-optimized" and "Document-optimized".

Both optimized models outperform the local model (Local). The scores with bold characters are the best scores of the tasks. The sentence-optimized model got the best position in Task C and the document-optimized model won the other tasks D–F. The sentence-optimized model also outputs competitive results to the document-optimized model in Task F. Unfortunately, our improvements are not statistically significant. But can our joint modelling help to reach or improve state-of-the-art results? We will try to answer this question in the next section.

3.2 Comparison to State-of-the-art

In order to put our results into context, Table 5 shows them alongside those of other TempEval-2 participants. We show only five teams: the winners of Tasks C–F in TempEval-2 and NCSU-joint which a global model with Markov Logic. The best result of each task is

[3] `http://nlp.stanford.edu/software/tagger.shtml`
[4] `http://nlp.cs.lth.se/`

	C	D	E	F
Local	0.652	0.745	0.553	0.520
Sentence-optimized	**0.674**	0.742	-	0.546
Document-optimized	0.652	**0.759**	**0.569**	**0.556**

Table 4: Results of the All Models

team	C	D	E	F
TRIOS*	**0.65**	0.79	0.56	0.60
TIPSem	0.55	**0.82**	0.55	0.59
TRIPS*	0.63	0.76	**0.58**	0.59
NCSU-indi	0.63	0.68	0.48	**0.66**
NCSU-joint	0.62	0.21	0.51	0.25
Sentence-optimized	**0.67**	0.74	-	0.55
Document-optimized	0.65	0.76	**0.57**	0.56

Table 5: Results with Other Systems (Systems with * have recall errors)

shown with bold characters. As shown in the last two rows, our Sentence and Document-optimized won Tasks C and E, respectively. Note, for Task E TRIPS (UzZaman and Allen, 2010) got 0.58 on precision but 0.50 on recall. Hence our Document-optimized outperforms TRIPS system on F-measure (0.57 vs 0.54). These results fit our intuitions that Task C requires rich linguistic knowledge inside sentences and Task E requires global knowledge such as inter-sentential logical constraints or ontological features. In TempEval-2's final report, it is not clear why the results on Task C (event-time) have not improved compared with the corresponding task in TempEval-1, notwithstanding TempEval-2 is added restriction that the event and time expression had to be syntactically adjacent. However, our system achieved over 0.67 pt and was better than TempEval-1's participants.

For Tasks D and F our results cannot reach the best TempEval-2 scores. But our results have some interesting points compared with the best results. TIPSem (Llorens et al., 2010), the best team of Task D, also employed semantic roles as features. Their learning classifiers are CRF with local features. So, a global joint approach is not always advantageous for event-DCT classifications. NCSU-indi (Ha et al., 2010), the best system for Task F, outperformed other participants (at more than 6 pt margins for all). This point suggests that ontological features NCSU-indi applied are more effective than global optimization. Compared with NCSU-joint which is a global model applied hard constraints in Markov Logic, we can find that our Markov Logic approach successfully controls soft constraints.

4 Conclusion

In this paper we presented a novel global approach to temporal relation identification. Our approach first optimized our model at sentence level and then extended it at document level. We revealed that the sentence-optimized model is better than the document-optimized model at least for identifying event-time relations (Task C) in TempEval-2 data. The document-optimized model is also strong at identifying relations between two events (Task E). As future work we are planning to use external or untagged data along with methods for unsupervised learning in Markov Logic (Poon and Domingos, 2008). We would also like to investigate the utility of our models for multilingual temporal ordering.

References

Boguraev, B. and Ando, R. K. (2005). Timeml-compliant text analysis for temporal reasoning. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, pages 997–1003.

Chambers, N. and Jurafsky, D. (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.

Do, Q., Lu, W., and Roth, D. (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.

Ha, E., Baikadi, A., Licata, C., and Lester, J. (2010). Ncsu: Modeling temporal relations with markov logic and lexical ontology. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 341–344, Uppsala, Sweden. Association for Computational Linguistics.

Ling, X. and Weld, D. S. (2010). Temporal information extraction. In *Proceedings of the Twenty Fifth National Conference on Artificial Intelligence*.

Llorens, H., Saquete, E., and Navarro, B. (2010). Tipsem (english and spanish): Evaluating crfs and semantic roles in tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 284–291, Uppsala, Sweden. Association for Computational Linguistics.

Mani, I., Verhagen, M., Wellner, B., Lee, C. M., and Pustejovsky, J. (2006). Machine learning of temporal relations. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 753–760, Morristown, NJ, USA. Association for Computational Linguistics.

Meza-Ruiz, I. and Riedel, S. (2009). Jointly identifying predicates, arguments and senses using markov logic. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 155–163, Boulder, CO, USA. Association for Computational Linguistics.

Poon, H. and Domingos, P. (2007). Joint inference in information extraction. In *Proceedings of the Twenty-Second National Conference on Artificial Intelligence*, pages 913–918, Vancouver, Canada. AAAI Press.

Poon, H. and Domingos, P. (2008). Joint unsupervised coreference resolution with Markov Logic. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 650–659, Honolulu, Hawaii. Association for Computational Linguistics.

Richardson, M. and Domingos, P. (2006). Markov logic networks. *Machine Learning*, 62(1-2):107–136.

Singla, P. and Domingos, P. (2006). Entity resolution with markov logic. In *Proceedings of the Sixth International Conference on Data Mining (ICDM)*, pages 572–582, Washington, DC, USA. IEEE Computer Society.

UzZaman, N. and Allen, J. (2010). Trips and trios system for tempeval-2: Extracting temporal information from text. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 276–283, Uppsala, Sweden. Association for Computational Linguistics.

Verhagen, M., Gaizaukas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on SemEval-2007.*, pages 75–80.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010). Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, Uppsala, Sweden. Association for Computational Linguistics.

Yoshikawa, K., Asahara, M., and Matsumoto, Y. (2011). Jointly extracting japanese predicate-argument relation with markov logic. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1125–1133, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Yoshikawa, K., Riedel, S., Asahara, M., and Matsumoto, Y. (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.