Jointly Combining Implicit Constraints Improves Temporal Ordering

Nathanael Chambers and Dan Jurafsky
Department of Computer Science
Stanford University
Stanford, CA 94305
{natec,jurafsky}@stanford.edu

Abstract

Previous work on ordering events in text has typically focused on local pairwise decisions, ignoring globally inconsistent labels. However, temporal ordering is the type of domain in which global constraints should be relatively easy to represent and reason over. This paper presents a framework that informs local decisions with two types of implicit global constraints: transitivity (*A before B* and *B before C* implies *A before C*) and time expression normalization (e.g. *last month* is before *yesterday*). We show how these constraints can be used to create a more densely-connected network of events, and how global consistency can be enforced by incorporating these constraints into an integer linear programming framework. We present results on two event ordering tasks, showing a 3.6% absolute increase in the accuracy of *before/after* classification over a pairwise model.

1 Introduction

Being able to temporally order events is a necessary component for complete document understanding. Interest in machine learning approaches for this task has recently been encouraged through the creation of the Timebank Corpus (Pustejovsky et al., 2003). However, most work on event-event ordering has focused on improving classifiers for pairwise decisions, ignoring obvious contradictions in the global space of events when misclassifications occur. A global framework to repair these event ordering mistakes has not yet been explored.

This paper addresses three main factors involved in a global framework: the global optimization algorithm, the constraints that are relevant to the task, and the level of connectedness across pairwise decisions. We employ Integer Linear Programming to address the first factor, drawing from related work in paragraph ordering (Bramsen et al., 2006). After finding minimal gain with the initial model, we explore reasons for and solutions to the remaining two factors through temporal reasoning and transitivity rule expansion.

We analyze the connectivity of the Timebank Corpus and show how textual events can be indirectly connected through a time normalization algorithm that automatically creates new relations between time expressions. We show how this increased connectivity is essential for a global model to improve performance.

We present three progressive evaluations of our global model on the Timebank Corpus, showing a 3.6% gain in accuracy over its original set of relations, and an 81% increase in training data size from previous work. In addition, we present the first results on Timebank that include an *unknown* relation, establishing a benchmark for performance on the full task of document ordering.

2 Previous Work

Recent work on classifying temporal relations within the Timebank Corpus built 6-way relation classifiers over 6 of the corpus' 13 relations (Mani et al., 2006; Mani et al., 2007; Chambers et al., 2007). A wide range of features are used, ranging from surface indicators to semantic classes. Classifiers make

local pairwise decisions and do not consider global implications between the relations.

The TempEval-07 (Verhagen et al., 2007) contest recently used two relations, *before* and *after*, in a semi-complete textual classification task with a new third relation to distinguish relations that can be labeled with high confidence from those that are uncertain, called *vague*. The task was a simplified classification task from Timebank in that only one verb, the main verb, of each sentence was used. Thus, the task can be viewed as ordering the main events in pairwise sentences rather than the entire document.

This paper uses the core relations of TempEval (*before*,*after*,*vague*) and applies them to a full document ordering task that includes every labeled event in Timebank. In addition, we extend the previous work by including a temporal reasoning component and embedding it within a global constraint model.

3 The Timebank Corpus

The Timebank Corpus (Pustejovsky et al., 2003) is a corpus of 186 newswire articles that are tagged for events, time expressions, and relations between the events and times. The individual events are further tagged for temporal information such as tense, modality and grammatical aspect. Time expressions use the TimeML (Ingria and Pustejovsky, 2002) markup language. There are 6 main relations and their inverses in Timebank: *before*, *ibefore*, *includes*, *begins*, *ends* and *simultaneous*.

This paper describes work that classifies the relations between events, making use of relations between events and times, and between the times themselves to help inform the decisions.

4 The Global Model

Our initial model has two components: (1) a pairwise classifier between events, and (2) a global constraint satisfaction layer that maximizes the confidence scores from the classifier. The first is based on previous work (Mani et al., 2006; Chambers et al., 2007) and the second is a novel contribution to event-event classification.

4.1 Pairwise Classification

Classifying the relation between two events is the basis of our model. A soft classification with confidence scores is important for the global maximization step that is described in the next section. As in Chambers et al. (2007), we build support vector machine (SVM) classifiers and use the probabilities from pairwise SVM decisions as our confidence scores. These scores are then used to choose an optimal global ordering.

Following our previous work, we use the set of features summarized in figure 1. They vary from POS tags and lexical features surrounding the event, to syntactic dominance, to whether or not the events share the same tense, grammatical aspect, or aspectual class. These features are the highest performing set on the basic 6-way classification of Timebank.

Feature	Description
Word*	The text of the event
Lemma*	The lemmatized head word
Synset*	The WordNet synset of head word
POS*	4 POS tags, 3 before, and 1 event
POS bigram*	The POS bigram of the event and its preceding tag
Prep*	Preposition lexeme, if in a prepositional phrase
Tense*	The event's tense
Aspect*	The event's grammatical aspect
Modal*	The modality of the event
Polarity*	Positive or negative
Class*	The aspecual class of the event
Tense Pair	The two concatenated tenses
Aspect Pair	The two concatenated aspects
Class Pair	The two concatenated classes
POS Pair	The two concatenated POS tags
Tense Match	true if the events have the same tense
Aspect Match	true if the events have the same aspect
Class Match	true if the events have the same class
Dominates	true if the first event syntactically dominates the second
Text Order	true if the first event occurs first in the document
Entity Match	true if they share an entity as an argument
Same Sent	true if both events are in the same sentence

Figure 1: The features to learn temporal relations between two events. Asterisks (*) indicate features that are duplicated, one for each of the two events.

We use Timebank's hand tagged attributes in the feature values for the purposes of this comparative

	before	after	unknown
A r1 B	.5	.3	.2
B r2 C	.4	.3	.3
A r3 C	.4	.5	.1
total	**1.3**	1.1	.6
A r1 B	.5	.3	.2
B r2 C	.4	.3	.3
A r3 C	.2	.7	.1
total	1.1	**1.3**	.6

Figure 2: Two sets of confidence scores. The first set chooses *before* for all three labels, and the second chooses *after*. Other lower-scoring valid relation sets also exist, such as *before*, *unknown*, and *before*.

study of global constraints, described next.

4.2 Global Constraints

Pairwise classifiers can make contradictory classifications due to their inability to consider other decisions. For instance, the following three decisions are in conflict:

```
A before B
B before C
A after C
```

Transitivity is not taken into account. In fact, there are several ways to resolve the conflict in this example. Given confidence scores (or probabilities) for each possible relation between the three pairs, we can compute an optimal label assignment. Different scores can lead to different conflict resolutions. Figure 2 shows two resolutions given different sets of scores. The first chooses *before* for all three relations, while the second chooses *after*.

Bramsen et al. (2006) presented a variety of approaches to using transitivity constraints to help inform pairwise decisions. They found that Integer Linear Programming (ILP) performed the best on a paragraph ordering task, consistent with its property of being able to find the optimal solution for a set of constraints. Other approaches are variations on a *greedy* strategy of adding pairs of events one at a time, ordered by their confidence. These can lead to suboptimal configurations, although they are guaranteed to find a solution. Mani et al. (2007) subsequently proposed one of these greedy strategies as well, but published results are not available. We also implemented a greedy best-first strategy, but found ILP outperformed it.

Our Integer Linear Programming framework uses the following objective function:

$$max \sum_i \sum_j p_{ij} x_{ij} \quad (1)$$

with added constraints:

$$\forall i \forall j \;\; x_{ij} \in \{0, 1\} \quad (2)$$

$$\forall i \;\; x_{i1} + x_{i2} + ... + x_{im} = 1 \quad (3)$$

where x_{ij} represents the ith pair of events classified as the jth relation of m relations. Thus, each pair of events generates m variables. Given n pairs of events, there are $n * m$ variables. p_{ij} is the probability of classifying pair i with relation j. Equation 2 (the first constraint) simply says that each variable must be 0 or 1. Equation 3 contains m variables for a single pair of events i representing its m possible relations. It states that one relation must be set to 1 and the rest to 0. In other words, a pair of events cannot have two relations at the same time. Finally, a transitivity constraint is added for all connected pairs i, j, k, for each transitivity condition that infers relation c given a and b:

$$x_{ia} + x_{jb} - x_{kc} <= 1 \quad (4)$$

We generated the set of constraints for each document and used lpsolve[1] to solve the ILP constraint problem.

The transitivity constraints are only effective if the available pairwise decisions constitute a connected graph. If pairs of events are disconnected, then transitivity makes little to no contribution because these constraints are only applicable to connected chains of events.

4.3 Transitive Closure

In order to connect the event graph, we draw on work from (Mani et al., 2006) and apply *transitive closure* to our documents. Transitive closure was first proposed not to address the problem of connected event graphs, but rather to expand the size of training data for relations such as *before*. Timebank is a relatively small corpus with few examples

[1] http://sourceforge.net/projects/lpsolve

Total Event-Event Relations After Closure

	before	*after*
Timebank	592	656
+ closure	3919	3405

Figure 3: The number of event-event relations after transitive closure.

Comparative Results

Training Set	Accuracy
Timebank Pairwise	66.8%
Global Model	66.8%

Figure 4: Using the base Timebank annotated tags for testing, accuracy on before/after tags in the two models.

of each relation. One way of expand the training set is through transitive rules. A few rules are given here:

$A\ simultaneous\ B \wedge A\ before\ C \rightarrow B\ before\ C$
$A\ includes\ B \wedge A\ ibefore\ C \rightarrow B\ before\ C$
$A\ before\ B \wedge A\ ends\ C \rightarrow B\ after\ C$

While the original motivation was to expand the training size of tagged relations, this approach also creates new connections in the graph, replacing previously unlabeled event pairs with their true relations. We adopted this approach and closed the original set of 12 relations to help connect the global constraint model.

4.4 Initial Experiment

The first evaluation of our global temporal model is on the Timebank Corpus over the labeled relations *before* and *after*. We merged *ibefore* and *iafter* into these two relations as well, ignoring all others. We use this task as a reduced evaluation to study the specific contribution of global constraints. We also chose this strict ordering task because it is well defined from a human understanding perspective. Snow et al. (2008) shows that average internet users can make *before/after* decisions with very high confidence, although the distinction with an *unknown* relation is not as clear. An evaluation including *unknown* (or *vague* as in TempEval) is presented later.

We expanded the corpus (prior to selecting the before/after relations) using transitive closure over all 12 relations as described above. Figure 3 shows the increase in data size. The number of *before* and *after* relations increase by a factor of six.

We trained and tested the system with 10-fold cross validation and micro-averaged accuracies. The folds were randomly generated to separate the 186 files into 10 folds (18 or 19 files per fold). The same 10-way split is used for all the evaluations. We used libsvm[2] to implement our SVM classifiers.

Figure 4 shows the results from our ILP model with transitivity constraints. The first row is the baseline pairwise classification trained and tested on the original Timebank relations. The second row gives performance with ILP. The model shows no improvement. The global ILP constraints did affect local decisions, changing 175 of them (out of 7324), but the changes cancelled out and had no affect on overall accuracy.

4.5 Loosely Connected Graph

Why didn't a global model help? The problem lies in the graph structure of Timebank's annotated relations. The Timebank annotators were not required to annotate relations between any particular pair of events. Instead, they were instructed to annotate what seemed appropriate due to the almost insurmountable task of annotating all pairs of events. A modest-sized document of 30 events, for example, would contain $\binom{30}{2} = 435$ possible pairs. Annotators thus marked relations where they deemed fit, most likely between obvious and critical relations to the understanding of the article. The vast majority of possible relations are untagged, thus leaving a large set of unlabeled (and disconnected) *unknown* relations.

Figure 5 graphically shows all relations that are annotated between events and time expressions in one of the shorter Timebank documents. Nodes represent events and times (event nodes start with the letter 'e', times with 't'), and edges represent temporal relations. Solid lines indicate hand annotations, and dotted lines indicate new rules from transitive closure (only one, from event *e4* to time *t14*). As can be seen, the graph is largely disconnected and a global model contributes little information since transitivity constraints cannot apply.

[2]http://www.csie.ntu.edu.tw/~cjlin/libsvm

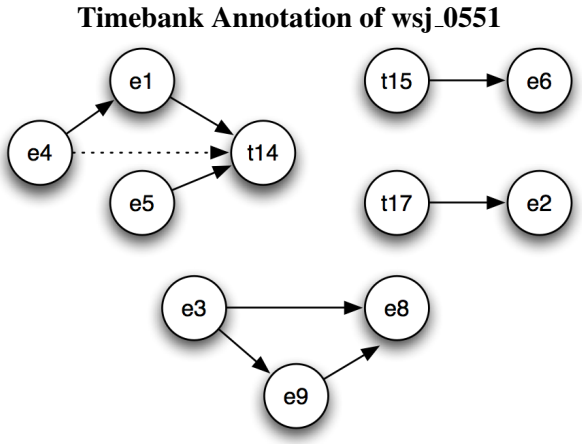

Figure 5: Annotated relations in document wsj_0551.

The large amount of unlabeled relations in the corpus presents several problems. First, building a classifier for these *unknown* relations is easily overwhelmed by the huge training set. Second, many of the untagged pairs have non-*unknown* ordering relations between them, but were missed by the annotators. This point is critical because one cannot filter this noise when training an *unknown* classifier. The noise problem will appear later and will be discussed in our final experiment. Finally, the space of annotated events is very loosely connected and global constraints cannot assist local decisions if the graph is not connected. The results of this first experiment illustrate this latter problem.

Bethard et al. (2007) strengthen the claim that many of Timebank's untagged relations should not be left unlabeled. They performed an independent annotation of 129 of Timebank's 186 documents, tagging all events in verb-clause relationships. They found over 600 valid *before/after* relations that are untagged in Timebank, on average three per document. One must assume that if these nearby verb-clause event pairs were missed by the annotators, the much larger number of pairs that cross sentence boundaries were also missed.

The next model thus attempts to fill in some of the gaps and further connect the event graph by using two types of knowledge. The first is by integrating Bethard's data, and the second is to perform temporal reasoning over the document's time expressions (e.g. *yesterday* or *january 1999*).

5 A Global Model With Time

Our initial model contained two components: (1) a pairwise classifier between events, and (2) a global constraint satisfaction layer. However, due to the sparseness in the event graph, we now introduce a third component addressing connectivity: (3) a temporal reasoning component to inter-connect the global graph and assist in training data expansion.

One important aspect of transitive closure includes the event-time and time-time relations during closure, not just the event-event links. Starting with 5,947 different types of relations, transitive rules increase the dataset to approximately 12,000. However, this increase wasn't enough to be effective in global reasoning. To illustrate the sparsity that still remains, if each document was a fully connected graph of events, Timebank would contain close to 160,000 relations[3], more than a 13-fold increase.

More data is needed to enrich the Timebank event graph. Two types of information can help: (1) more event-event relations, and (2) a separate type of information to indirectly connect the events: event-X-event. We incorporate the new annotations from Bethard et al. (2007) to address (1) and introduce a new temporal reasoning procedure to address (2). The following section describes this novel approach to adding time expression information to further connect the graph.

5.1 Time-Time Information

As described above, we use event-time relations to produce the transitive closure, as well as annotated time-time relations. It is unclear if Mani et al. (2006) used these latter relations in their work.

However, we also add new time-time links that are deduced from the logical time intervals that they describe. Time expressions can be resolved to time intervals with some accuracy through simple rules. New time-time relations can then be added to our space of events through time stamp comparisons. Take this newswire example:

The Financial Times 100-share index shed 47.3 points to close at 2082.1, down 4.5% from the ***previous Friday****, and 6.8% from* ***Oct. 13****, when Wall Street's plunge helped spark the* ***current*** *weakness in London.*

[3] Sum over the # of events n_d in each document d, $\binom{n_d}{2}$

The first two expressions (‘*previous Friday*’ and ‘*Oct. 13*’) are in a clear *before* relationship that Timebank annotators captured. The ‘current’ expression, is correctly tagged with the *PRESENT_REF* attribute to refer to the document’s timestamp. Both ‘*previous Friday*’ and ‘*Oct. 13*’ should thus be tagged as being *before* this expression. However, the annotators did not tag either of these two *before* relations, and so our timestamp resolution procedure fills in these gaps. This is a common example of two expressions that were not tagged by the annotators, yet are in a clear temporal relationship.

We use Timebank’s gold standard TimeML annotations to extract the dates and times from the time expressions. In addition, those marked as *PRESENT_REF* are resolved to the document timestamp. Time intervals that are strictly before or after each other are thus labeled and added to our space of events. We create new *before* relations based on the following procedure:

```
if event1.year < event2.year
  return true
if event1.year == event2.year
  if event1.month < event2.month
    return true
  if event1.month == event2.month
    if event1.day < event2.day
      return true
  end
end
return false
```

All other time-time orderings not including the *before* relation are ignored (i.e. *includes* is not created, although could be with minor changes).

This new time-time knowledge is used in two separate stages of our model. The first is just prior to transitive closure, enabling a larger expansion of our tagged relations set and reduce the noise in the *unknown* set. The second is in the constraint satisfaction stage where we add our automatically computed time-time relations (with the gold event-time relations) to the global graph to help correct local event-event mistakes.

Total Event-Event Relations After Closure

	before	*after*
Timebank	3919	3405
+ time-time	5604	5118
+ time/bethard	7111	6170

Figure 6: The number of event-event *before* and *after* relations after transitive closure on each dataset.

Comparative Results with Closure

Training Set	Accuracy
Timebank Pairwise	66.8%
Global Model	66.8%
Global + time/bethard	70.4%

Figure 7: Using the base Timebank annotated tags for testing, the increase in accuracy on before/after tags.

5.2 Temporal Reasoning Experiment

Our second evaluation continues the use of the two-way classification task with *before* and *after* to explore the contribution of closure, time normalization, and global constraints.

We augmented the corpus with the labeled relations from Bethard et al. (2007) and added the automatically created time-time relations as described in section 5.1. We then expanded the corpus using transitive closure. Figure 6 shows the progressive data size increase as we incrementally add each to the closure algorithm.

The time-time generation component automatically added 2459 new *before* and *after* time-time relations into the 186 Timebank documents. This is in comparison to only 157 relations that the human annotators tagged, less than 1 per document on average. The second row of figure 6 shows the drastic effect that these time-time relations have on the number of available event-event relations for training and testing. Adding both Bethard’s data and the time-time data increases our training set by 81% over closure without it.

We again performed 10-fold cross validation with micro-averaged accuracies, but each fold tested only on the transitively closed Timebank data (the first row of figure 6). The training set used all available data (the third row of figure 6) including the Bethard data as well as our new time-time links.

Figure 7 shows the results from the new model. The first row is the baseline pairwise classification trained and tested on the original relations only. Our model improves by 3.6% absolute. This improvement is statistically significant ($p < 0.000001$, McNemar's test, 2-tailed).

5.3 Discussion

To further illustrate why our model now improves local decisions, we continue our previous graph example. The actual text for the graph in figure 5 is shown here:

docstamp: 10/30/89 ***(t14)***
*Trustcorp Inc. will become****(e1)*** *Society Bank & Trust when its merger****(e3)*** *is completed****(e4)*** *with Society Corp. of Cleveland, the bank said****(e5)****. Society Corp., which is also a bank, agreed****(e6)*** *in June****(t15)*** *to buy****(e8)*** *Trustcorp for 12.4 million shares of stock with a market value of about $450 million. The transaction****(e9)*** *is expected****(e10)*** *to close****(e2)*** *around year end****(t17)****.*

The automatic time normalizer computes and adds three *new* time-time relations, two connecting t15 and t17 with the document timestamp, and one connecting t15 and t17 together. These are not otherwise tagged in the corpus.

Time-Time + Closure

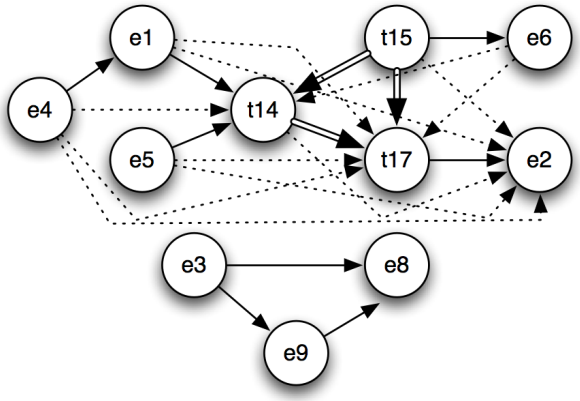

Figure 8: Before and after time-time links with closure.

Figure 8 shows the augmented document. The double-line arrows indicate the three new time-time relations and the dotted edges are the new relations added by our transitive closure procedure. Most critical to this paper, three of the new edges are event-event relations that help to expand our training data. If this document was used in testing (rather than training), these new edges would help inform our transitive rules during classification.

Even with this added information, disconnected segments of the graph are still apparent. However, the 3.6% performance gain encourages us to move to the final full task.

6 Final Experiment with Unknowns

Our final evaluation expands the set of relations to include unlabeled relations and tests on the entire dataset available to us. The following is now a classification task between the three relations: *before*, *after*, and *unknown*.

We duplicated the previous evaluation by adding the labeled relations from Bethard et al. (2007) and our automatically created time-time relations. We then expanded this dataset using transitive closure. Unlike the previous evaluation, we also use this entire dataset for testing, not just for training. Thus, all event-event relations in Bethard as well as Timebank are used to expand the dataset with transitive closure and are used in training and testing. We wanted to fully evaluate document performance on every possible event-event relation that logically follows from the data.

As before, we converted *IBefore* and *IAfter* into *before* and *after* respectively, while all other relations are reduced to *unknown*. This relation set coincides with TempEval-07's core three relations (although they use *vague* instead of *unknown*).

Rather than include all unlabeled pairs in our *unknown* set, we only include the unlabeled pairs that span at most one sentence boundary. In other words, events in adjacent sentences are included in the *unknown* set if they were not tagged by the Timebank annotators. The intuition is that annotators are more likely to label nearby events, and so events in adjacent sentences are more likely to be actual *unknown* relations if they are unlabeled. It is more likely that distant events in the text were overlooked by convenience, not because they truly constituted an *unknown* relationship.

The set of possible sentence-adjacent *unknown* relations is very large (approximately 50000 *unknown* compared to 7000 *before*), and so we randomly select a percentage of these relations for each evalu-

Classification Accuracy

% unk	base	global	global+time
0	72.0%	72.2%	74.0%
1	69.4%	69.5%	71.3%
3	65.5%	65.6%	67.1%
5	63.7%	63.8%	65.3%
7	61.2%	61.6%	62.8%
9	59.3%	59.5%	60.6%
11	58.1%	58.4%	59.4%
13	57.1%	57.1%	58.1%

Figure 9: Overall accuracy when training with different percentages of *unknown* relations included. 13% of *unknowns* is about equal to the number of *befores*.

Base Pairwise Classification

	precision	recall	f1-score
before	61.4	55.4	58.2
after	57.6	53.1	55.3
unk	53.0	62.8	57.5

Global+Time Classification

	precision	recall	f1-score
before	63.7 (+2.3)	57.1 (+2.2)	60.2 (+2.0)
after	60.3 (+2.7)	54.3 (+2.9)	57.1 (+1.8)
unk	52.0 (-1.0)	62.9 (+0.1)	56.9 (-0.6)

Figure 10: Precision and Recall for the base pairwise decisions and the global constraints with integrated time information.

ation. We used the same SVM approach with the features described in section 4.1.

6.1 Results

Results are presented in figure 9. The rows in the table are different training/testing runs on varying sizes of *unknown* training data. There are three columns with accuracy results of increasing complexity. The first, **base**, are results from pairwise classification decisions over Timebank and Bethard with no global model. The second, **global**, are results from the Integer Linear Programming global constraints, using the pairwise confidence scores from the **base** evaluation. Finally, the **global+time** column shows the ILP results when all event-time, time-time, and automatically induced time-time relations are included in the global graph.

The ILP approach does not alone improve performance on the event-event tagging task, but adding the time expression relations greatly increases the global constraint results. This is consistent with the results from out first two experiments. The evaluation with 1% of the *unknown* tags shows an almost 2% improvement in accuracy. The gain becomes smaller as the *unknown* set increases in size (1.0% gain with 13% *unknown*). *Unknown* relations will tend to be chosen as more weight is given to *unknowns*. When there is a constraint conflict in the global model, *unknown* tends to be chosen because it has no transitive implications. All improvements from base to global+time are statistically significant ($p < 0.000001$, McNemar's test, 2-tailed).

The first row of figure 9 corresponds to the results in our second experiment in figure 7, but shows higher accuracy. The reason is due to our different test sets. This final experiment includes Bethard's event-event relations in testing. The improved performance suggests that the clausal event-event relations are easier to classify, agreeing with the higher accuracies originally found by Bethard et al. (2007).

Figure 10 shows the precision, recall, and f-score for the evaluation with 13% *unknowns*. This set was chosen for comparison because it has a similar number of *unknown* labels as *before* labels. We see an increase in precision in both the *before* and *after* decisions by up to 2.7%, an increase in recall up to 2.9%, and an fscore by as much as 2.0%. The *unknown* relation shows mixed results, possibly due to its noisy behavior as discussed throughout this paper.

6.2 Discussion

Our results on the two-way (before/after) task show that adding additional implicit temporal constraints and then performing global reasoning results in significant improvements in temporal ordering of events (3.6% absolute over simple pairwise decisions).

Both *before* and *after* also showed increases in precision and recall in the three-way evaluation. However, *unknown* did not parallel this improvement, nor are the increases as dramatic as in the two-way evaluation. We believe this is consistent with the noise that exists in the Timebank corpus for unlabeled relations. Evidence from Bethard's indepen-

dent annotations directly point to missing relations, but the dramatic increase in the size of our closure data (81%) from adding a small amount of time-time relations suggests that the problem is widespread. This noise in the *unknown* relation may be dampening the gains that the two way task illustrates.

This work is also related to the task of event-time classification. While not directly addressed in this paper, the global methods described within clearly apply to pairwise models of event-time ordering as well.

Further progress in improving global constraints will require new methods to more accurately identify *unknown* events, as well as new approaches to create implicit constraints over the ordering. We expect such an improved ordering classifier to be used to improve the performance of tasks such as summarization and question answering about the temporal nature of events.

Acknowledgments

This work is funded in part by DARPA through IBM and by the DTO Phase III Program for AQUAINT. We also thank our anonymous reviewers for many helpful suggestions.

References

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

Philip Bramsen, Pawan Deshpande, Yoong Keok Lee, and Regina Barzilay. 2006. Inducing temporal graphs. In *Proceedings of EMNLP-06*.

Nathanael Chambers, Shan Wang, and Dan Jurafsky. 2007. Classifying temporal relations between events. In *Proceedings of ACL-07*, Prague, Czech Republic.

R Ingria and James Pustejovsky. 2002. TimeML specification 1.0. In *http://www.time2002.org*.

Inderjeet Mani, Marc Verhagen, Ben Wellner, Chong Min Lee, and James Pustejovsky. 2006. Machine learning of temporal relations. In *Proceedings of ACL-06*, July.

Inderjeet Mani, Ben Wellner, Marc Verhagen, and James Pustejovsky. 2007. Three approaches to learning tlinks in timeml. Technical Report CS-07-268, Brandeis University.

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

Rion Snow, Brendan O'Connor, Dan Jurafsky, and Andrew Ng. 2008. Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP-08*, Waikiki, Hawaii, USA.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. Semeval-2007 task 15: Tempeval temporal relation identification. In *Workshop on Semantic Evaluations*.