Temporal Ordering of Event Descriptions

Nate Chambers and Shan Wang

This paper describes a machine learning approach to ordering real world events based on their newswire descriptions. Understanding and ordering free form text has been a challenge in natural language processing (NLP) for many years, but only recently have machine learning techniques been utilized. We describe steps taken to advance the state of the art, focusing on learning temporal relations between pairs of events. We propose feature vectors based on linguistic principles, and furthermore, we choose additional features that can be realistically automated instead of dependent on hand-tagged data. We show a 10% increase in accuracy compared to previous work in the area.

1 Introduction

Many areas of language understanding, such as question answering, would benefit from a temporal ordering of events. However, eliciting the true order of world events from a textual description is a very difficult task. Even with a surface understanding of sentences, the temporal order of the individual descriptions is not obvious. Recently, with the creation of hand-tagged newswire corpora, new research in machine learning has taken strides to accomplish this difficult task.

Newspaper articles often describe the most important event first, followed by a series of more descriptive paragraphs of different, related events. The linear order of sentences generally does not follow the linear order of time. This paper describes an approach to temporal ordering that uses linguistic features to train machine learning algorithms. In addition, it describes new features that are automatically extracted from untagged corpora using current NLP tools. We compare these linguistic features to previous work that used hand tagged features.

2 Previous Work

Mani et. al (2006) used time relations between events to build a classifier that marks each pair of events with a temporal relation. Mani builds his classifier off of "perfect" human-tagged features only. He also applies rules of temporal transitivity to expand the small number of trainable relations. He reports 93% accuracy on event-event relations, although 75% is the baseline majority tag.

Lapata and Lascarides (2006) trained an event classifier for inter-sentential events. They built their own corpus by searching for key time words (e.g. "after" and "before") and saved sentences that contained two events, one of which was triggered by the key word. They constructed a learner based on syntax and clausal ordering features. Boguraev and Ando (2005) approached different but related tasks of (1) extracting events and (2) relating events to time periods. They used a chunking approach with part of speech tags

(POS) as features. Both tasks are different than this paper's. We focus on finding time relations between events.

Each of these groups focuses on different subtasks. Our work is most similar to Mani's in that we are learning relations given event pairs. Lapata addresses this task only within sentences, not full document understanding. Our work is unique in that we use imperfect *automatically* extracted linguistic features in addition to the corpus features for learning.

3 Training Data

Our training set is derived from the TimeBank corpus (Pustejovsky 2003), a set of 186 newswire articles tagged with the TimeML schema through a combination of automated tagging and human verification. TimeML is an annotation schema designed to capture and represent temporal information. EVENT tags identify semantic events in the text and classify them with high-level features (e.g. tense, aspect, class). The corpus also contains temporal links between pairs of events, called TLINKs. We condensed the original set of thirteen TLINK relations into six, as many TLINKs are synonymous or inverses of others. Given pairs of events, it is these TLINKs that this paper learns.

We also use the Opinion Corpus (unofficial, unreleased), a set of 73 articles tagged in the same manner as TimeBank, but by a different set of researchers. This corpus is used for comparison purposes to previous work only. The combined TimeBank and Opinion corpus is called the OTC corpus in our evaluation.

3.1 Closure

Since the training data is sparse, we increase its size by performing temporal closure, as suggested in Mani et al. (2006). For example, if an article contains the relations `(A AFTER B)` and `(B INCLUDE C)`, we infer the relation `(A AFTER C)`. We created 86 such rules and performed full closure, roughly increasing corpus size three-fold.

4 Features

Event Features

Every event's EVENT tag contains the attributes: class, tense, aspect, modality, and polarity. An event can belong to one of seven classes ranging from ACTION to SITUATION. Tense, aspect and modality are the verbal features attached if the event is a verb (not a nominal noun), otherwise null. We found seven unique modals in our dataset. The polarity represents if the event is happening or not, in the usual sense. We also added another feature that is the event string itself. Finally, we added pair dependent features that are on or off if the two events share the same tense, aspect, or class. Mani et al. used these *base features* in their work. The rest are unique to this paper for this task.

Part of Speech (POS)

For each identified event, we include the POS tag for the event and the tags for the two tokens preceding and one following. To extract POS tags, we used LingPipe (www.alias-i.com/lingpipe), a suite of Java libraries for linguistic analysis. This tagger uses 93 (Brown) POS tags, but we mapped these tags into a set of 41 (Penn Treebank) POS tags

in order to improve performance in the face of sparse training data. We also created bigram POS features of the event and the token before the event.

Co-referential Entities
We use LingPipe's utilities to identify co-referential named entities. We hypothesize that if two events involve the same entity, they have a higher probability of being related over events involving distinct entities. For example, consider the following pairs of sentences.

(1) Jane fell. Mike pushed her.
(2) Jane fell. Mike pushed Mary.

In the first pair, the events "fell" and "pushed" both involve Jane, so we conclude that the two events are related. In the second pair, the relation no longer exists. We used the Stanford Parser to create the syntactic parse tree for each sentence and aligned the TimeBank EVENT tags with the LingPipe co-referential IDs. When classifying an event pair, we match if both events include the same entity ID among the events' modifiers.

Event-Event Properties
A phrase P is said to *dominate* another phrase Q if Q is a daughter node of P in the syntactic parse tree. We leverage the Stanford Parser syntactic output to create this feature for intra-sentential events. It is either on or off, depending on the two events' syntactic location. Obviously, two events in different sentences are always off. We also include a feature representing the linear ordering of the two events in the text. This applies even when two events appear in separate sentences.

Prepositional Phrase
We created a feature for whether or not the event is part of a prepositional phrase. The feature's values range over 30 English prepositions.

5 Evaluation

We used four corpora: TimeBank, TimeBank with closure, TimeBank/Opinion (OTC), and OTC with closure. The baseline is the relation that occurs most frequently. We also define a test case of *Base Features* that mainly uses the tagged features in the corpus, as in Mani (2006). These consist of the *Event Features* as described above in section 4.

On each corpus, we performed Naïve Bayes classification with 10-fold cross validation. We performed both feature addition and feature removal, reporting performance for the method of feature selection that performed best. We also used LIBSVM (csie.ntu.edu.tw/~cjlin/libsvm) for multi-class SVM classification with linear, polynomial, and radial kernels. Finally, we compared performance of Brown and Penn POS tags, but found they gave comparable performance, thus we only present results using Penn TreeBank tags.

6 Results

Figure 1 shows the Naïve Bayes results on the two corpora and their temporally closed counterparts. Closure raises the baseline, but performance does not increase by the same amount. We saw 55.5% accuracy over the baseline 38% and over previous work's

features 50.4%. It is difficult to directly compare the results as Mani et al. report only OTC scores, however, our implementation of his features performed nearly the same and achieved 59.9% compared to his reported 62.5%. Even with this slight discrepancy, our new features still performed better at 64.2%.

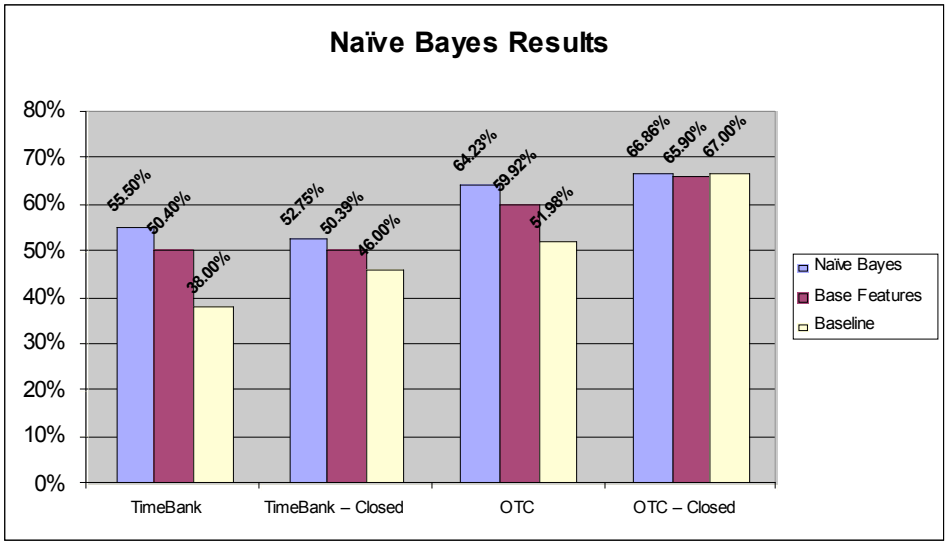

Figure 1: Naïve Bayes results, comparing two baselines to our features.

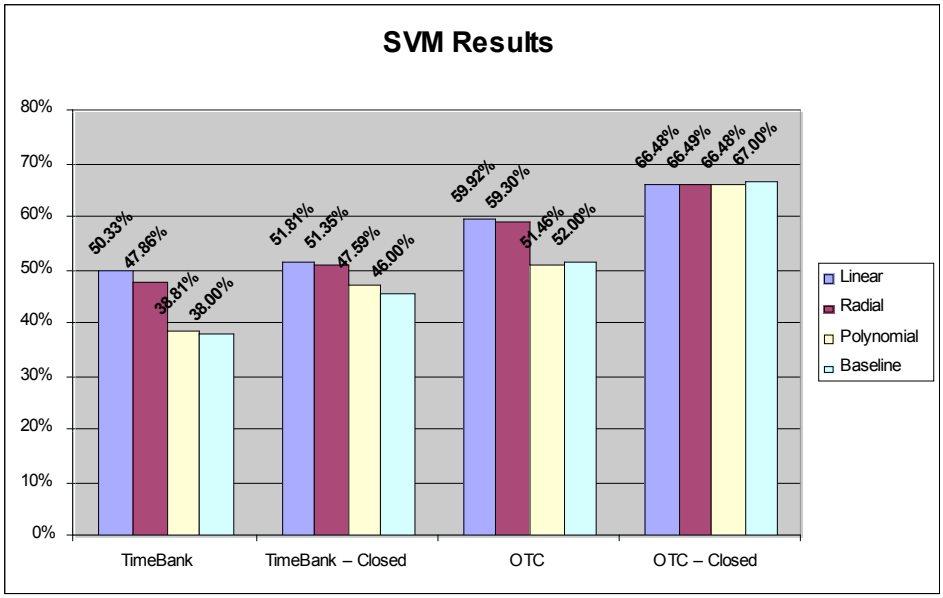

Figure 2: Support Vector Machine results, comparing three different kernels.

Figure 2 shows SVM performance. A linear kernel had the best performance, but was outperformed by Naïve Bayes. As seen with Naïve Bayes, corpus closure surprisingly had no beneficial effect. Our closure algorithm increased TimeBank from 3367 to 8820 relations and increased OTC from 6075 to 18,974 relations. The OTC closed set appears to have introduced significant noise and learning is minimal over the baseline.

7 Discussion

There is a large jump in performance by including new features. Most of the new features included dependencies *between* the two events, rather than new features about each event individually. Phrasal dominance, co-referenced entities, and tense/aspect/class event pairs all capture syntactic and semantic dependencies between the events. This suggests that Naïve Bayes has too many independence assumptions. The 10% increase from Mani's base features, what he calls *perfect* features, rises from 50.4% to 55.5%. This clearly shows that some dependencies need to be captured within the features themselves if not in the statistical model.

The importance of "closing" the data is unclear from our experiments. Our results show that the closed corpus performs *worse* than the unclosed corpus when compared to the baseline. If the closed data is similar to the unclosed, then the large increase in training data should have helped. We suspect that closing the data introduces new event-event relations that have long range dependencies not captured by the original relations. Most event-event pairs in the raw corpora are no farther than one sentence apart. However, closing the data introduces distances that cover the entire article. We believe these new relations require more knowledge beyond sentential syntax. Mani et al. does not show this result, but it is unclear where the difference lies. Our features clearly outperform his, but he reports abnormally high results on an unknown closed dataset.

The results on the OTC corpus show an improvement as with TimeBank alone, but not as large. We evaluated this corpus to compare against Mani's work, but our evaluation performed very differently from theirs. It is important to note the wide difference in event relations in TimeBank and Opinion. The Opinion Corpus has 72.8% BEFORE relations (35.2% in TimeBank) and only 11% SIMUL (38.5% TimeBank). Such a difference is astounding, and it surely affects the baseline as the Opinion corpus is so heavily skewed to the BEFORE relation. We believe the tagging schema needs to be studied, as two divergent corpora should not be compared directly. We also believe this leads to the discrepancy with Mani's paper as all their results are reported from OTC.

8 Conclusion

We have shown how important linguistic features that capture dependencies between two events are to the temporal ordering of events. We showed improvements over previous work from 50.4% to 55.5%, a 10% improvement. We also describe results questioning previous success on a divergent corpus, particularly with the use of temporal closure to expand the training data size.

References

[1] B. Boguraev and R.K. Ando. 2005. TimeML-compliant text analysis for temporal reasoning. In *Proceedings of IJCAI*, UK.
[2] M. Lapata and A. Lascarides. 2006. Learning Sentence-internal Temporal Relations. Journal of AI Research, Volume 27, pages 85-117.
[3] I. Mani et al. Machine Learning of Temporal Relations. 2006. ACL. Australia.
[4] J. Pustejovsky et al. 2003. The TimeBank Corpus. Corpus Linguistics, 647-656.