Machine Learning of Temporal Relations

Inderjeet Mani[¥§], **Marc Verhagen**[¶], **Ben Wellner**[¥¶]
Chong Min Lee[§] **and James Pustejovsky**[¶]
[¥]The MITRE Corporation
202 Burlington Road, Bedford, MA 01730, USA
[§]Department of Linguistics, Georgetown University
37th and O Streets, Washington, DC 20036, USA
[¶]Department of Computer Science, Brandeis University
415 South St., Waltham, MA 02254, USA
{imani, wellner}@mitre.org, {marc, jamesp}@cs.brandeis.edu, cml54@georgetown.edu

Abstract

This paper investigates a machine learning approach for temporally ordering and anchoring events in natural language texts. To address data sparseness, we used temporal reasoning as an over-sampling method to dramatically expand the amount of training data, resulting in predictive accuracy on link labeling as high as 93% using a Maximum Entropy classifier on human annotated data. This method compared favorably against a series of increasingly sophisticated baselines involving expansion of rules derived from human intuitions.

1 Introduction

The growing interest in practical NLP applications such as question-answering and text summarization places increasing demands on the processing of temporal information. In multi-document summarization of news articles, it can be useful to know the relative order of events so as to merge and present information from multiple news sources correctly. In question-answering, one would like to be able to ask when an event occurs, or what events occurred prior to a particular event.

A wealth of prior research by (Passoneau 1988), (Webber 1988), (Hwang and Schubert 1992), (Kamp and Reyle 1993), (Lascarides and Asher 1993), (Hitzeman et al. 1995), (Kehler 2000) and others, has explored the different knowledge sources used in inferring the temporal ordering of events, including temporal adverbials, tense, aspect, rhetorical relations, pragmatic conventions, and background knowledge. For example, the narrative convention of events being described in the order in which they occur is followed in (1), but overridden by means of a discourse relation, Explanation in (2).

(1) Max *stood up*. John *greeted* him.

(2) Max *fell*. John *pushed* him.

In addition to discourse relations, which often require inferences based on world knowledge, the ordering decisions humans carry out appear to involve a variety of knowledge sources, including tense and grammatical aspect (3a), lexical aspect (3b), and temporal adverbials (3c):

(3a) Max *entered* the room. He *had drunk* a lot of wine.

(3b) Max *entered* the room. Mary *was seated* behind the desk.

(3c) The company *announced* Tuesday that third-quarter sales *had fallen*.

Clearly, substantial linguistic processing may be required for a system to make these inferences, and world knowledge is hard to make available to a domain-independent program. An important strategy in this area is of course the development of annotated corpora than can facilitate the machine learning of such ordering inferences.

This paper[1] investigates a machine learning approach for temporally ordering events in natural language texts. In Section 2, we describe the annotation scheme and annotated corpora, and the challenges posed by them. A basic learning approach is described in Section 3. To address data sparseness, we used temporal reasoning as an over-sampling method to dramatically expand the amount of training data.

As we will discuss in Section 5, there are no standard algorithms for making these inferences that we can compare against. We believe strongly that in such situations, it's worthwhile for computational linguists to devote consider-

[1] Research at Georgetown and Brandeis on this problem was funded in part by a grant from the ARDA AQUAINT Program, Phase II.

Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 753–760,
Sydney, July 2006. ©2006 Association for Computational Linguistics

able effort to developing insightful baselines. Our work is, accordingly, evaluated in comparison against four baselines: (i) the usual majority class statistical baseline, shown along with each result, (ii) a more sophisticated baseline that uses hand-coded rules (Section 4.1), (iii) a hybrid baseline based on hand-coded rules expanded with Google-induced rules (Section 4.2), and (iv) a machine learning version that learns from imperfect annotation produced by (ii) (Section 4.3).

2 Annotation Scheme and Corpora

2.1 TimeML

TimeML (Pustejovsky et al. 2005) (www.timeml.org) is an annotation scheme for markup of events, times, and their temporal relations in news articles. The TimeML scheme flags tensed verbs, adjectives, and nominals with EVENT tags with various attributes, including the class of event, tense, grammatical aspect, polarity (negative or positive), any modal operators which govern the event being tagged, and cardinality of the event if it's mentioned more than once. Likewise, time expressions are flagged and their values normalized, based on TIMEX3, an extension of the ACE (2004) (tern.mitre.org) TIMEX2 annotation scheme.

For temporal relations, TimeML defines a TLINK tag that links tagged events to other events and/or times. For example, given (3a), a TLINK tag **orders** an instance of the event of entering to an instance of the drinking with the relation type AFTER. Likewise, given the sentence (3c), a TLINK tag will **anchor** the event instance of announcing to the time expression *Tuesday* (whose normalized value will be inferred from context), with the relation IS_INCLUDED. These inferences are shown (in slightly abbreviated form) in the annotations in (4) and (5).

```
  (4) Max <EVENT eventID="e1"
class="occurrence" tense="past" aspect="none">entered</EVENT> the room.
He <EVENT eventID="e2"
class="occurrence" tense="past" aspect="perfect">had drunk</EVENT>a
lot of wine.
  <TLINK eventID="e1" relatedToEventID="e2" relType="AFTER"/>
   (5) The company <EVENT eventID="e1" class="reporting"
tense="past" aspect="none">announced</EVENT>
<TIMEX3 tid="t2" type="DATE" temporalFunction="false" value="1998-01-08">Tuesday </TIMEX3> that third-quarter sales <EVENT eventID="e2"
class="occurrence" tense="past" aspect="perfect"> had fallen</EVENT>.
  <TLINK eventID="e1" relatedToEventID="e2" relType="AFTER"/>
  <TLINK eventID="e1" relatedToTimeID="t2" relType="IS_INCLUDED"/>
```

The anchor relation is an Event-Time TLINK, and the order relation is an Event-Event TLINK. TimeML uses 14 temporal relations in the TLINK *RelTypes*, which reduce to a disjunctive classification of 6 temporal relations *RelTypes* = {SIMULTANEOUS, IBEFORE, BEFORE, BEGINS, ENDS, INCLUDES}. An event or time is SIMULTANEOUS with another event or time if they occupy the same time interval. An event or time INCLUDES another event or time if the latter occupies a proper subinterval of the former. These 6 relations and their inverses map one-to-one to 12 of Allen's 13 basic relations (Allen 1984)[2]. There has been a considerable amount of activity related to this scheme; we focus here on some of the challenges posed by the TLINK annotation, the part that is directly relevant to the temporal ordering and anchoring problems.

2.2 Challenges

The annotation of TimeML information is on a par with other challenging semantic annotation schemes, like PropBank, RST annotation, etc., where high inter-annotator reliability is crucial but not always achievable without massive preprocessing to reduce the user's workload. In TimeML, inter-annotator agreement for time expressions and events is 0.83 and 0.78 (average of Precision and Recall) respectively, but on TLINKs it is 0.55 (P&R average), due to the large number of event pairs that can be selected for comparison. The time complexity of the human TLINK annotation task is quadratic in the number of events and times in the document.

Two corpora have been released based on TimeML: the TimeBank (Pustejovsky et al. 2003) (we use version 1.2.a) with 186 documents and

[2] Of the 14 TLINK relations, the 6 inverse relations are redundant. In order to have a disjunctive classification, SIMULTANEOUS and IDENTITY are collapsed, since IDENTITY is a subtype of SIMULTANEOUS. (Specifically, X and Y are identical if they are simultaneous *and* coreferential.) DURING and IS_INCLUDED are collapsed since DURING is a subtype of IS_INCLUDED that anchors events to times that are durations. IBEFORE (immediately before) corresponds to Allen's MEETS. Allen's OVERLAPS relation is not represented in TimeML. More details can be found at timeml.org.

64,077 words of text, and the Opinion Corpus (www.timeml.org), with 73 documents and 38,709 words. The TimeBank was developed in the early stages of TimeML development, and was partitioned across five annotators with different levels of expertise. The Opinion Corpus was developed very recently, and was partitioned across just two highly trained annotators, and could therefore be expected to be less noisy. In our experiments, we merged the two datasets to produce a single corpus, called OTC.

Table 1 shows the distribution of EVENTs and TIMES, and TLINK *RelTypes*[3] in the OTC. The majority class percentages are shown in parentheses. It can be seen that BEFORE and SIMULTANEOUS together form a majority of event-ordering (Event-Event) links, whereas most of the event anchoring (Event-Time) links are INCLUDES.

12750 Events, 2114 Times		
Relation	**Event-Event**	**Event-Time**
IBEFORE	131	15
BEGINS	160	112
ENDS	208	159
SIMULTANEOUS	1528	77
INCLUDES	950	3001 (65.3%)
BEFORE	3170 (51.6%)	1229
TOTAL	6147	4593

Table 1. TLINK Class Distributions in OTC Corpus

The lack of TLINK coverage in human annotation could be helped by preprocessing, provided it meets some threshold of accuracy. Given the availability of a corpus like OTC, it is natural to try a machine learning approach to see if it can be used to provide that preprocessing. However, the noise in the corpus and the sparseness of links present challenges to a learning approach.

3 Machine Learning Approach

3.1 Initial Learner

There are several sub-problems related to inferring event anchoring and event ordering. Once a tagger has tagged the events and times, the first task (A) is to link events and/or times, and the second task (B) is to label the links. Task A is hard to evaluate since, in the absence of massive preprocessing, many links are ignored by the human in creating the annotated corpora. In addition, a program, as a baseline, can trivially link all tagged events and times, getting 100% recall on Task A. We focus here on Task B, the labeling task. In the case of humans, in fact, when a TLINK is posited by both annotators between the same pairs of events or times, the inter-annotator agreement on the labels is a .77 average of P&R. To ensure replicability of results, we assume perfect (i.e., OTC-supplied) events, times, and links.

Thus, we can consider TLINK inference as the following classification problem: *given an ordered pair of elements X and Y, where X and Y are events or times which the human has related temporally via a TLINK, the classifier has to assign a label in RelTypes*. Using *RelTypes* instead of *RelTypes* ∪ {NONE} also avoids the problem of heavily skewing the data towards the NONE class.

To construct feature vectors for machine learning, we took each TLINK in the corpus and used the given TimeML features, with the TLINK class being the vector's class feature. For replicability by other users of these corpora, and to be able to isolate the effect of components, we used 'perfect' features; no feature engineering was attempted. The features were, for each event in an event-ordering pair, the *event-class, aspect, modality, tense* and *negation* (all nominal features); *event string*, and *signal* (a preposition/adverb, e.g., *reported on Tuesday*), which are string features, and contextual features indicating whether the *same tense* and *same aspect* are true of both elements in the event pair. For event-time links, we used the above event and signal features along with TIMEX3 time features.

For learning, we used an off-the-shelf Maximum Entropy (ME) classifier (from Carafe, available at sourceforge.net/projects/carafe). As shown in the **UNCLOSED (ME)** column in Table 2[4], accuracy of the unclosed ME classifier does not go above 77%, though it's always better than the majority class (in parentheses). We also tried a variety of other classifiers, including the SMO support-vector machine and the naïve Bayes tools in WEKA (www.weka.net.nz). SMO performance (but not naïve Bayes) was comparable with ME, with SMO trailing it in a few cases (to save space, we report just ME performance). It's possible that feature engineering could improve performance, but since this is "perfect" data, the result is not encouraging.

[3] The number of TLINKs shown is based on the number of TLINK vectors extracted from the OTC.

[4] All machine learning results, except for ME-C in Table 4, use 10-fold cross-validation. 'Accuracy' in tables is Predictive Accuracy.

	UNCLOSED (ME)						**CLOSED (ME-C)**					
	Event-Event			**Event-Time**			**Event-Event**			**Event-Time**		
Accuracy:	62.5 (51.6)			76.13 (65.3)			93.1 (75.2)			88.25 (62.3)		
Relation	**Prec**	**Rec**	**F**	**Prec**	**Rec**	**F**	**Prec**	**Rec**	**F**	**Prec**	**Rec**	**F**
IBEFORE	50.00	27.27	35.39	0	0	0	77.78	60.86	68.29	0	0	0
BEGINS	50.00	41.18	45.16	60.00	50.00	54.54	85.25	82.54	83.87	76.47	74.28	75.36
ENDS	94.74	66.67	78.26	41.67	27.78	33.33	87.83	94.20	90.90	79.31	77.97	78.62
SIMULTANEOUS	50.35	50.00	50.17	33.33	20.00	25.00	62.50	38.60	47.72	73.68	56.00	63.63
INCLUDES	47.88	34.34	40.00	80.92	62.72	84.29	90.41	88.23	89.30	86.07	80.78	83.34
BEFORE	68.85	79.24	73.68	70.47	62.72	66.37	94.95	97.26	96.09	90.16	93.56	91.83

Table 2. Machine learning results using unclosed and closed data

3.2 Expanding Training Data using Temporal Reasoning

To expand our training set, we use a temporal closure component SputLink (Verhagen 2004), that takes known temporal relations in a text and derives new implied relations from them, in effect making explicit what was implicit. SputLink was inspired by (Setzer and Gaizauskas 2000) and is based on Allen's interval algebra, taking into account the limitations on that algebra that were pointed out by (Vilain et al. 1990). It is basically a constraint propagation algorithm that uses a transitivity table to model the compositional behavior of all pairs of relations in a document. SputLink's transitivity table is represented by 745 axioms. An example axiom:

```
If relation(A, B) = BEFORE &&
   relation(B, C) = INCLUDES
then infer relation(A, C) = BEFORE
```

Once the TLINKs in each document in the corpus are closed using SputLink, the same vector generation procedure and feature representation described in Section 3.1 are used. The effect of closing the TLINKs on the corpus has a dramatic impact on learning. Table 2, in the **CLOSED (ME-C)** column shows that accuracies for this method (called ME-C, for Maximum Entropy learning with closure) are now in the high 80's and low 90's, and still outperform the closed majority class (shown in parentheses).

What is the reason for the improvement?[5] One reason is the dramatic increase in the amount of training data. The more connected the initial unclosed graph for a document is in TLINKs, the greater the impact in terms of closure. When the OTC is closed, the number of TLINKs goes up by more than 11 times, from 6147 Event-Event and 4593 Event-Time TLINKs to 91,157 Event-Event and 29,963 Event-Time TLINKs. The number of BEFORE links goes up from 3170 (51.6%) Event-Event and 1229 Event-Time TLINKs (26.75%) to 68585 (75.2%) Event-Event and 18665 (62.3%) Event-Time TLINKs, making BEFORE the majority class in the closed data for both Event-Event and Event-Time TLINKs. There are only an average of 0.84 TLINKs per event before closure, but after closure it shoots up to 9.49 TLINKs per event. (Note that as a result, the majority class percentages for the closed data have changed from the unclosed data.)

Being able to bootstrap more training data is of course very useful. However, we need to dig deeper to investigate how the increase in data affected the machine learning. The improvement provided by temporal closure can be explained by three factors: (1) closure effectively creates a new classification problem with many more instances, providing more data to train on; (2) the class distribution is further skewed which results in a higher majority class baseline (3) closure produces additional data in such a way as to increase the frequencies and statistical power of existing features in the unclosed data, as opposed to adding new features. For example, with unclosed data, given A BEFORE B and B BEFORE C, closure generates A BEFORE C which provides more significance for the features related to A and C appearing as first and second arguments, respectively, in a BEFORE relation.

In order to help determine the effects of the above factors, we carried out two experiments in which we sampled 6145 vectors from the *closed*

[5] Interestingly, performance does not improve for SIMULTANEOUS. The reason for this might be due to the relatively modest increase in SIMULTANEOUS relations from applying closure (roughly factor of 2).

data – i.e. approximately the number of Event-Event vectors in the unclosed data. This effectively removed the contribution of factor (1) above. The first experiment (Closed Class Distribution) simply sampled 6145 instances uniformly from the closed instances, while the second experiment (Unclosed Class Distribution) sampled instances according to the same distribution as the unclosed data. Table 3 shows these results. The greater class distribution skew in the closed data clearly contributes to improved accuracy. However, when using the same class distribution as the unclosed data (removing factor (2) from above), the accuracy, 76%, *is higher than using the full unclosed data*. This indicates that closure does indeed help according to factor (3).

4 Comparison against Baselines

4.1 Hand-Coded Rules

Humans have strong intuitions about rules for temporal ordering, as we indicated in discussing sentences (1) to (3). Such intuitions led to the development of pattern matching rules incorporated in a TLINK tagger called GTag. GTag takes a document with TimeML tags, along with syntactic information from part-of-speech tagging and chunking from Carafe, and then uses 187 syntactic and lexical rules to infer and label TLINKs between tagged events and other tagged events or times. The tagger takes pairs of TLINKable items (event and/or time) and searches for the single most-confident rule to apply to it, if any, to produce a labeled TLINK between those items. Each (if-then) rule has a left-hand side which consists of a conjunction of tests based on TimeML-related feature combinations (TimeML features along with part-of-speech and chunk-related features), and a right-hand side which is an assignment to one of the TimeML TLINK classes.

The rule patterns are grouped into several different classes: (i) the event is anchored with or without a signal to a time expression within the same clause, e.g., (3c), (ii) the event is anchored without a signal to the document date (as is often the case for reporting verbs in news), (iii) an event is linked to another event in the same sentence, e.g., (3c), and (iv) the event in a main clause of one sentence is anchored with a signal or tense/aspect cue to an event in the main clause of the previous sentence, e.g., (1-2), (3a-b).

The performance of this baseline is shown in Table 4 (line GTag). The top most accurate rule (87% accuracy) was GTag Rule 6.6, which links a past-tense event verb joined by a conjunction to another past-tense event verb as being BEFORE the latter (e.g., *they traveled and slept the night* ..):

```
If sameSentence=YES &&
    sentenceType=ANY &&
    conjBetweenEvents=YES &&
    arg1.class=EVENT &&
    arg2.class=EVENT &&
    arg1.tense=PAST &&
    arg2.tense=PAST &&
    arg1.aspect=NONE &&
    arg2.aspect=NONE &&
    arg1.pos=VB &&
    arg2.pos=VB &&
    arg1.firstVbEvent=ANY &&
    arg2.firstVbEvent=ANY
then infer relation=BEFORE
```

The vast majority of the intuition-bred rules have very low accuracy compared to ME-C, with intuitions failing for various feature combinations and relations (for relations, for example, GTag lacks rules for IBEFORE, STARTS, and ENDS). The bottom-line here is that even when heuristic preferences are intuited, those preferences need to be guided by empirical data, whereas hand-coded rules are relatively ignorant of the distributions that are found in data.

4.2 Adding Google-Induced Lexical Rules

One might argue that the above baseline is too weak, since it doesn’t allow for a rich set of lexical relations. For example, pushing can result in falling, killing always results in death, and so forth. These kinds of defeasible rules have been investigated in the semantics literature, including the work of Lascarides and Asher cited in Section 1.

However, rather than hand-creating lexical rules and running into the same limitations as with GTag’s rules, we used an empirically-derived resource called VerbOcean (Chklovski and Pantel 2004), available at http://semantics.isi.edu/ocean. This resource consists of lexical relations mined from Google searches. The mining uses a set of lexical and syntactic patterns to test for pairs of verb strongly associated on the Web in an asymmetric ‘happens-before’ relation. For example, the system discovers that *marriage* happens-before *divorce*, and that *tie* happens-before *untie*.

We automatically extracted all the ‘happens-before’ relations from the VerbOcean resource at the above web site, and then automatically converted those relations to GTag format, producing 4,199 rules. Here is one such converted rule:

```
If arg1.class=EVENT &&
   arg2.class=EVENT &&
   arg1.word=learn &&
   arg2.word=forget &&
then infer relation=BEFORE
```

Adding these lexical rules to GTag (with morphological normalization being added for rule matching on word features) amounts to a considerable augmentation of the rule-set, by a factor of 22. GTag with this augmented rule-set might be a useful baseline to consider, since one would expect the gigantic size of the Google 'corpus' to yield fairly robust, broad-coverage rules.

What if both a core GTag rule and a VerbOcean-derived rule could both apply? We assume the one with the higher confidence is chosen. However, we don't have enough data to reliably estimate rule confidences for the original GTag rules; so, for the purposes of VerbOcean rule integration, we assigned either the original VerbOcean rules as having greater confidence than the original GTag rules in case of a conflict (i.e., a preference for the more specific rule), or vice-versa.

The results are shown in Table 4 (lines GTag+VerbOcean). The combined rule set, under both voting schemes, had no statistically significant difference in accuracy from the original GTag rule set. So, ME-C beat this baseline as well.

The reason VerbOcean didn't help is again one of data sparseness, due to most verbs occurring rarely in the OTC. There were only 19 occasions when a happens-before pair from VerbOcean correctly matched a human BEFORE TLINK, of which 6 involved the same rule being right twice (including *learn* happens-before *forget*, a rule which students are especially familiar with!), with the rest being right just once. There were only 5 occasions when a VerbOcean rule incorrectly matched a human BEFORE TLINK, involving just three rules.

	Closed Class Distribution				**UnClosed Class Distribution**			
Relation	**Prec**	**Rec**	**F**	**Accuracy**	**Prec**	**Rec**	**F**	**Accuracy**
IBEFORE	100.0	100.0	100.0	**87.20** (72.03)	83.33	58.82	68.96	**76.0** (40.95)
BEGINS	0	0	0		72.72	50.0	59.25	
ENDS	66.66	57.14	61.53		62.50	50.0	55.55	
SIMULTANEOUS	14.28	6.66	9.09		60.54	66.41	63.34	
INCLUDES	73.91	77.98	75.89		75.75	77.31	76.53	
BEFORE	90.68	92.60	91.63		84.09	84.61	84.35	

Table 3. Machine Learning from subsamples of the closed data

Baseline	**Accuracy**	
	Event-Event	**Event-Time**
GTag	63.43	72.46
GTag+VerbOcean - GTag overriding VerbOcean	64.80	74.02
GTag+VerbOcean - VerbOcean overriding GTag	64.22	73.37
GTag+closure+ME-C	53.84 (57.00)	67.37 (67.59)

Table 4. Accuracy of 'Intuition' Derived Baselines

4.3 Learning from Hand-Coded Rules Baseline

The previous baseline was a hybrid confidence-based combination of corpus-induced lexical relations with hand-created rules for temporal ordering. One could consider another obvious hybrid, namely learning from annotations created by GTag-annotated corpora. Since the intuitive baseline fares badly, this may not be that attractive. However, the dramatic impact of closure could help offset the limited coverage provided by human intuitions.

Table 4 (line GTag+closure+ME-C) shows the results of closing the TLINKs produced by GTag's annotation and then training ME from the resulting data. The results here are evaluated against a held-out test set. We can see that even after closure, the baseline of learning from unclosed human annotations is much poorer than ME-C, and is in fact substantially worse than the majority class on event ordering.

This means that for preprocessing new data sets to produce noisily annotated data for this classification task, it is far better to use machine-learning from closed human annotations rather than machine-learning from closed annotations produced by an intuitive baseline.

5 Related Work

Our approach of classifying pairs independently during learning does not take into account dependencies between pairs. For example, a classifier may label <X, Y> as BEFORE. Given the pair <X, Z>, such a classifier has no idea if <Y, Z> has been classified as BEFORE, in which case, through closure, <X, Z> should be classified as BEFORE. This can result in the classifier producing an inconsistently annotated text. The machine learning approach of (Cohen et al. 1999) addresses this, but their approach is limited to total orderings involving BEFORE, whereas TLINKs introduce partial orderings involving BEFORE and five other relations. Future research will investigate methods for tighter integration of temporal reasoning and statistical classification.

The only closely comparable machine-learning approach to the problem of TLINK extraction was that of (Boguraev and Ando 2005), who trained a classifier on Timebank 1.1 for event anchoring for events and times within the same sentence, obtaining an F-measure (for tasks A and B together) of 53.1. Other work in machine-learning and hand-coded approaches, while interesting, is harder to compare in terms of accuracy since they do not use common task definitions, annotation standards, and evaluation measures. (Li et al. 2004) obtained 78-88% accuracy on ordering within-sentence temporal relations in Chinese texts. (Mani et al. 2003) obtained 80.2 F-measure training a decision tree on 2069 clauses in anchoring events to reference times that were inferred for each clause. (Berglund et al. 2006) use a document-level evaluation approach pioneered by (Setzer and Gaizauskas 2000), which uses a distinct evaluation metric. Finally, (Lapata and Lascarides 2004) use found data to successfully learn which (possibly ambiguous) temporal markers connect a main and subordinate clause, without inferring underlying temporal relations.

In terms of hand-coded approaches, (Mani and Wilson 2000) used a baseline method of blindly propagating TempEx time values to events based on proximity, obtaining 59.4% on a small sample of 8,505 words of text. (Filatova and Hovy 2001) obtained 82% accuracy on 'timestamping' clauses for a single type of event/topic on a data set of 172 clauses. (Schilder and Habel 2001) report 84% accuracy inferring temporal relations in German data, and (Li et al. 2001) report 93% accuracy on extracting temporal relations in Chinese. Because these accuracies are on different data sets and metrics, they cannot be compared directly with our methods.

Recently, researchers have developed other tools for automatically tagging aspects of TimeML, including EVENT (Sauri et al. 2005) at 0.80 F-measure and TIMEX3[6] tags at 0.82-0.85 F-measure. In addition, the TERN competition (tern.mitre.org) has shown very high (close to .95 F-measures) for TIMEX2 tagging, which is fairly similar to TIMEX3. These results suggest the time is ripe for exploiting 'imperfect' features in our machine learning approach.

6 Conclusion

Our research has uncovered one new finding: semantic reasoning (in this case, logical axioms for temporal closure), can be extremely valuable in addressing data sparseness. Without it, performance on this task of learning temporal relations is poor; with it, it is excellent. We showed that temporal reasoning can be used as an oversampling method to dramatically expand the amount of training data for TLINK labeling, resulting in labeling predictive accuracy as high as 93% using an off-the-shelf Maximum Entropy classifier. Future research will investigate this effect further, as well as examine factors that enhance or mitigate this effect in different corpora.

The paper showed that ME-C performed significantly better than a series of increasingly sophisticated baselines involving expansion of rules derived from human intuitions. Our results in these comparisons confirm the lessons learned from the corpus-based revolution, namely that rules based on intuition alone are prone to incompleteness and are hard to tune without access to the distributions found in empirical data. Clearly, lexical rules have a role to play in semantic and pragmatic reasoning from language, as in the discussion of example (2) in Section 1. Such rules, when mined by robust, large corpus-based methods, as in the Google-derived VerbOcean, are clearly relevant, but too specific to apply more than a few times in the OTC corpus.

It may be possible to acquire confidence weights for at least some of the intuitive rules in GTag from Google searches, so that we have a

[6] http://complingone.georgetown.edu/~linguist/GU_TIME_DOWNLOAD.HTML

level field for integrating confidence weights from the fairly general GTag rules and the fairly specific VerbOcean-like lexical rules. Further, the GTag and VerbOcean rules could be incorporated as features for machine learning, along with features from automatic preprocessing.

We have taken pains to use freely downloadable resources like Carafe, VerbOcean, and WEKA to help others easily replicate and quickly ramp up a system. To further facilitate further research, our tools as well as labeled vectors (unclosed as well as closed) are available for others to experiment with.

References

James Allen. 1984. Towards a General Theory of Action and Time. Artificial Intelligence, 23, 2, 123-154.

Anders Berglund, Richard Johansson and Pierre Nugues. 2006. A Machine Learning Approach to Extract Temporal Information from Texts in Swedish and Generate Animated 3D Scenes. Proceedings of EACL-2006.

Branimir Boguraev and Rie Kubota Ando. 2005. TimeML-Compliant Text Analysis for Temporal Reasoning. Proceedings of IJCAI-05, 997-1003.

Timothy Chklovski and Patrick Pantel. 2004.VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. Proceedings of EMNLP-04. http://semantics.isi.edu/ocean

W. Cohen, R. Schapire, and Y. Singer. 1999. Learning to order things. Journal of Artificial Intelligence Research, 10:243–270, 1999.

Janet Hitzeman, Marc Moens and Clare Grover. 1995. Algorithms for Analyzing the Temporal Structure of Discourse. Proceedings of EACL'95, Dublin, Ireland, 253-260.

C.H. Hwang and L. K. Schubert. 1992. Tense Trees as the fine structure of discourse. Proceedings of ACL'1992, 232-240.

Hans Kamp and Uwe Ryle. 1993. From Discourse to Logic (Part 2). Dordrecht: Kluwer.

Andrew Kehler. 2000. Resolving Temporal Relations using Tense Meaning and Discourse Interpretation, in M. Faller, S. Kaufmann, and M. Pauly, (eds.), *Formalizing the Dynamics of Information*, CSLI Publications, Stanford.

Mirella Lapata and Alex Lascarides. 2004. Inferring Sentence-internal Temporal Relations. In *Proceedings of the North American Chapter of the Assocation of Computational Linguistics*, 153-160.

Alex Lascarides and Nicholas Asher. 1993. Temporal Relations, Discourse Structure, and Commonsense Entailment. Linguistics and Philosophy 16, 437-494.

Wenjie Li, Kam-Fai Wong, Guihong Cao and Chunfa Yuan. 2004. Applying Machine Learning to Chinese Temporal Relation Resolution. Proceedings of ACL'2004, 582-588.

Inderjeet Mani, Barry Schiffman, and Jianping Zhang. 2003. Inferring Temporal Ordering of Events in News. Short Paper. Proceedings of HLT-NAACL'03, 55-57.

Inderjeet Mani and George Wilson. 2000. Robust Temporal Processing of News. Proceedings of ACL'2000.

Rebecca J. Passonneau. A Computational Model of the Semantics of Tense and Aspect. Computational Linguistics, 14, 2, 1988, 44-60.

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, 647-656.

James Pustejovsky, Bob Ingria, Roser Sauri, Jose Castano, Jessica Littman, Rob Gaizauskas, Andrea Setzer, G. Katz, and I. Mani. 2005. The Specification Language TimeML. In I. Mani, J. Pustejovsky, and R. Gaizauskas, (eds.), The Language of Time: A Reader. Oxford University Press.

Roser Saurí, Robert Knippen, Marc Verhagen and James Pustejovsky. 2005. Evita: A Robust Event Recognizer for QA Systems. Short Paper. Proceedings of HLT/EMNLP 2005: 700-707.

Frank Schilder and Christof Habel. 2005. From temporal expressions to temporal information: semantic tagging of news messages. In I. Mani, J. Pustejovsky, and R. Gaizauskas, (eds.), The Language of Time: A Reader. Oxford University Press.

Andrea Setzer and Robert Gaizauskas. 2000. Annotating Events and Temporal Information in Newswire Texts. Proceedings of LREC-2000, 1287-1294.

Marc Verhagen. 2004. Times Between The Lines. Ph.D. Dissertation, Department of Computer Science, Brandeis University.

Marc Vilain, Henry Kautz, and Peter Van Beek. 1989. Constraint propagation algorithms for temporal reasoning: A revised report. In D. S. Weld and J. de Kleer (eds.), Readings in Qualitative Reasoning about Physical Systems, Morgan-Kaufman, 373-381.

Bonnie Webber. 1988. Tense as Discourse Anaphor. Computational Linguistics, 14, 2, 1988, 61-73.