Timelines from Text: Identification of Syntactic Temporal Relations

Conference Paper · October 2007

DOI: 10.1109/ICSC.2007.77 · Source: IEEE Xplore

3 authors, including:

James H. Martin
University of Colorado Boulder

Timelines from Text: Identification of Syntactic Temporal Relations

Steven Bethard and James H. Martin
Department of Computer Science
University of Colorado at Boulder
430 UCB, Boulder, CO 80309, USA
{bethard,martin}@colorado.edu

Sara Klingenstein
Department of Linguistics
University of Colorado at Boulder
295 UCB, Boulder CO 80309, USA
klingesa@colorado.edu

Abstract

We propose and evaluate a linguistically motivated approach to extracting temporal structure necessary to build a timeline. We considered pairs of events in a verb-clause construction, where the first event is a verb and the second event is the head of a clausal argument to that verb. We selected all pairs of events in the TimeBank that participated in verb-clause constructions and annotated them with the labels BEFORE, OVERLAP *and* AFTER. *The resulting corpus of 895 event-event temporal relations was then used to train a machine learning model. Using a combination of event-level features like tense and aspect with syntax-level features like the paths through the syntactic tree, we were able to train a support vector machine (SVM) model which could identify new temporal relations with 89.2% accuracy. High accuracy models like these are a first step towards automatic extraction of timeline structures from text.*

1. Introduction

Recent developments in natural language processing have allowed a variety of fine-grained semantic components to be extracted automatically from text. Machine learning systems have shown good performance on a variety of tasks, including the detection of people, organizations and locations [5, 6], as well as the semantic roles these entities play [11, 15]. The goal of our research is to robustly produce knowledge representations from text by identifying semantic components and integrating them into graph structures. This means identifying the important events and entities, determining what roles the entities play in the events, and identifying what temporal and causal relations tie the events together [3]. For example, consider a sentence like:

(1) The top commander of a Cambodian resistance force said Thursday he has sent a team to recover the remains of a British mine removal expert kidnapped and presumed killed by Khmer Rouge guerrillas almost two years ago.

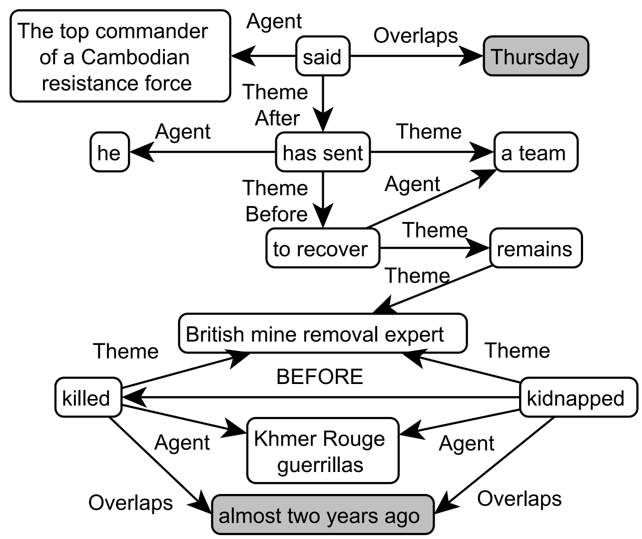

Figure 1. A semantic graph incorporating events, entities, argument roles and temporal relations.

Our goal is to extract from this text a semantic graph like Figure 1, showing important relations like that the *Khmer Rouge* was the *Agent* of the the *kidnapping*, and that the *kidnapping* occurred before the *killing*. Such knowledge allows deeper reasoning about the meaning of the text.

In the current paper we look at a sub-task of this research: the automatic extraction of timeline information. Timelines structure knowledge about important events and the temporal relations between them. A simple timeline might look like:

9:59 AM	WTC South Tower collapses
10:03 AM	UA 93 crashes in Shanksville, Pennsylvania
10:28 AM	WTC North Tower collapses

0-7695-2997-6/07 $25.00 © 2007 IEEE
DOI 10.1109/ICSC.2007.77

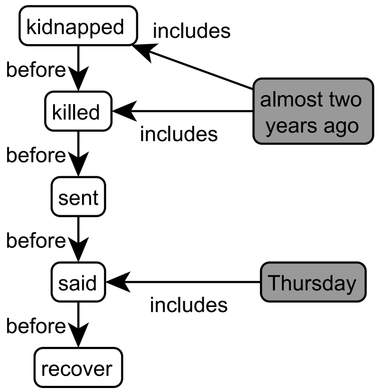

Figure 2. Temporal graph for Example 1.

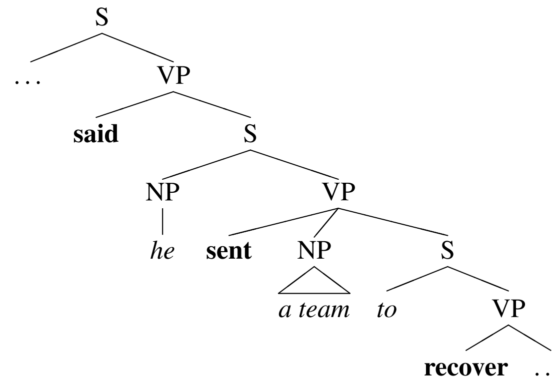

Figure 3. Syntactic tree containing the events *said***,** *sent* **and** *recover*

This timeline summarizes all the information necessary to answer questions like:

- Which tower collapsed first? *The WTC South Tower.*
- Was either tower standing when the plane crashed in Pennsylvania? *Yes, the WTC North Tower.*

Of course, textual resources are not generally available in timeline form, much less a machine-readable timeline representation. So for this task, we are interested in automatically extracting from a text the important events and temporal relations between them, and producing temporal graphs like Figure 2. These graphs are a subset of the full semantic graphs, focusing on the important events, e.g. *sent* and *recover* in the example, and the important temporal relations, e.g. (*kidnapped* BEFORE *killed*).

In this paper, we evaluate machine learning approaches to extracting such temporal structure. We propose and evaluate a linguistically motivated approach to temporal relation classification: classification by syntactic construction. In this approach, event pairs to be analyzed are selected and classified based on the syntactic constructions in which they occur. For example, a simple verb-clause construction would select pairs of events where the first event is a verb and the second is the head of a clausal argument to that verb. This verb-clause construction would select the event pairs *said-sent* and *sent-recover* from the syntactic tree in Figure 3. As we will show in this paper, selecting event pairs in this way gives a more principled approach for annotation, and results in better system performance.

The rest of the paper is structured as follows. Section 2 explains some of the prior work that has motivated this research. Section 3 describes how a corpus of verb-clause syntactic constructions was collected and annotated. Section 4 explains what features and algorithms were used to train machine learning models on this data. Section 5 reports and analyzes the performance of these models. Finally, Section 6 describes some prospects for future work.

2 Prior Work

The natural language processing (NLP) community has recently seen increased interest in extracting important events and temporal structure from text. Researchers have developed a markup language, TimeML, for expressing the structure of events and temporal relations in text [12]. This markup language has allowed human annotators to encode parts of the temporal structure of the 186 document TimeBank corpus to serve as a testbed for automatic techniques [13].

Using the TimeBank corpus, researchers trained models for extracting events and temporal structure. Models trained to find the important events in a document typically combined machine learning techniques with low level features like word stems and parts of speech, and found that important events could be identified with precision and recall in the 70s and 80s [1, 4, 14]. For certain classes of events, like reporting events or verbal events, precision and recall could reach as high as the 90s [1]. On the other hand, identifying the temporal relations between these events proved to be much more difficult. Systems reported some problems with the consistency of the temporal relation annotations in the TimeBank, and found that even in simplified tasks, performance remained in the 50s and 60s [4, 9].

Incorporating such feedback, the creators of the TimeBank organized the 2007 TempEval competition, providing a new set of data annotated for temporal relations [16]. A number of precautions were taken in the TempEval data to encourage higher inter-annotator agreement. First, TimeML relation types that were hard for annotators to distinguish, like IS_INCLUDED and DURING, were merged into simpler relation types, like OVERLAP. Second, rather than having annotators scan an entire document looking for temporal relations, the annotators were shown one event pair at a time and asked to assign a temporal relation to each. Performing annotation this way meant that annotators would no longer accidentally overlook an important temporal relation.

Of course, most documents contain many events, and annotating all pairs of events with temporal relations would be intractable – trying to assign a temporal relation to two unrelated events many sentences apart would be too difficult for even the best of annotators. Instead, the TempEval competition selected three different types of pairs for annotation:

Task A Events[1] paired with all times in the same sentence. For example, consider the sentence:

(2) Qantas [EVENT *plans*] daily flights between Sydney and Bombay to [EVENT *boost*] business and tourism ties with India, the airline [EVENT *announced*] [TIME *Friday*].

In this task, the events *plans*, *boost* and *announced* would each be paired with the time *Friday*.

Task B Events[1] paired with the document creation time. For example, consider the document fragment:

(3) [TIME *11/02/89*]
Hadson Corp. [EVENT *said*] it [EVENT *expects*] to [EVENT *report*] a third-quarter net [EVENT *loss*] of $17 million to $19 million.

In this task, the events *said*, *expects*, *report* and *loss* would each be paired with the document time *11/02/89*.

Task C The matrix verb events of two adjacent sentences. For example, consider the text:

(4) The action [EVENT *came*] in response to a petition filed by Timex Inc.
Previously, watch imports were [EVENT *denied*] such duty-free treatment.

In this task, the event *came* would be paired with the event *denied*.

A variety of research groups developed systems to compete in these tasks, typically combining machine learning techniques with lexical, syntactic and semantic features [2]. Though performance on Tasks A and C was in the 50s and low 60s, systems trained for Task B reached accuracies as high as the low 80s.

These results suggest that when extracting temporal structure from text, certain types of temporal relations form better starting points than others. Systems performed best in the TempEval Task B, where events were related to the the document creation time. This task characterizes the same sorts of relations as the linguistic notion of *tense* (which relates *utterance time* to *event time*). In contrast, on Tasks A and C, whose pairings of events and times were not motivated by linguistic structures, system performance was much poorer.

We take these results to indicate that research on temporal relations has a greater chance of success when guided by linguistic knowledge. In particular, we explore guiding temporal relation annotation using syntactic constructions. That is, we select pairs of events that are related to each other by a particular syntactic pattern.

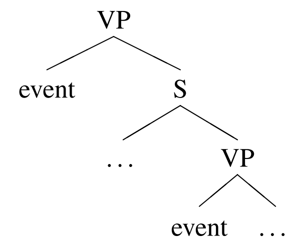

Figure 4. The verb-clause syntactic construction

3 Data

For this research, we looked at temporal relations expressed through the verb-clause construction, depicted in Figure 4. In the verb-clause construction, the first event is a verb and the second event is the head of a clausal argument to that verb. While this syntactic pattern is fairly specific, it occurs quite frequently – in the TimeBank, around 20% of events participate in such verb-clause constructions, and almost 50% of adjacent pairs of verbal events participate in in exactly a verb-clause construction.

We manually annotated a small corpus of these constructions so that we could train and evaluate automated systems for extracting temporal relations. In order to make our corpus as compatible with previous work as possible, we selected the Wall Street Journal section of the TimeBank for annotation. The 132 newswire documents in this collection had already been manually annotated with both gold-standard events from the TimeBank effort, and gold-standard syntactic trees from the TreeBank effort [10]. After aligning the TimeBank annotations with the TreeBank annotations, we were able to extract 895 such pairs of events from the corpus[2].

[1] To lower the amount of annotation necessary, only events appearing at least 20 times in the TimeBank were annotated.

[2] Originally, 901 verb-clause pairs had been found in the data. Four were removed because an error in the gold-standard syntactic parse had attached a clause to the wrong verb. The other two were removed because

These verb-clause event pairs formed our basic corpus for annotation. Event pairs were viewed by the annotator one at a time, using a modified version of the TANGO tool[3] used to annotate the TimeBank, and were assigned a label of BEFORE, AFTER or OVERLAP. The annotation guidelines followed as closely as possible those of the TimeBank and TempEval annotations, with the following additional guidelines:

- Modal or conditional events should be annotated using a possible worlds analysis. So in the example:

  (6) They [EVENT *expect*] him to [EVENT *cut*] costs throughout the organization.

  the event pair would be assigned the relation (*expect* BEFORE *cut*) because, in the possible world where costs are *cut*, the *cutting* will have occurred after the *expecting* happening right now.

- Events that perform aspectual-like functions, e.g. describing the manner in which another event is performed, should be annotated as OVERLAP. So in the example:

  (7) The move may [EVENT *help*] [EVENT *prevent*] Martin Ackerman from making a run at the computer-services concern.

  the event pair would be assigned the relation (*help* OVERLAP *prevent*) because the *help* event is not really meaningful on its own. It describes how much of the *preventing* the *move* accounts for. (Note that in the TimeBank the event *help* has the INTENTIONAL_ACTION class, not the ASPECTUAL class, but it is still considered to be covered by this guideline.)

- Verbs like *allow*, *permit* and *require*, which describe states of affairs that come into being at a single point and continue indefinitely into the future, should be annotated as including only the point at which the state comes into effect. So in the example:

  (8) The provision would [EVENT *require*] BellSouth to [EVENT *pay*] a price equivalent to what an outside party might have to pay.

  the event pair would be assigned the relation (*require* BEFORE *pay*) because the point at which the *requiring* comes into being precedes the point at which the *paying* begins.

they used the connective *unless*, as in:

(5) ...the best defense lawyers will be [EVENT *unwilling*] to take criminal cases unless they are [EVENT *assured*] of being paid.

Since the *unwilling* and *assured* events occur in different, mutually-exclusive possible worlds, attempting to assign a temporal order is extremely difficult (if it even makes sense at all).

[3] http://timeml.org/site/tango/tool.html

	All	Train	Test
Documents	132	94	38
Event pairs	895	581	314
BEFORE relations	368	229	139
OVERLAP relations	344	221	123
AFTER relations	183	131	52

Table 1. Number of documents, event pairs and types of temporal relations in the annotated corpus.

To get accustomed to the guidelines above, the annotator first annotated 13 documents (around 50 event pairs) from the middle of the corpus. These warm-up annotations were then discarded and the annotator started from the beginning of the corpus, working event-pair by event-pair until all 132 documents (895 event pairs) were annotated[4]. Table 1 shows the overall corpus statistics, including the distribution of relation types. Similar statistics are shown for the training and testing parts of the corpus, used below to evaluate our machine learning models.

4 Machine Learning

We framed the temporal relation identification task as a three-way classification task, where the first event is classified as being either BEFORE, OVERLAP-ing with or AFTER the second event. We used support vector machine (SVM) classifiers[5] as they have shown good performance on a variety of natural language processing tasks [8, 11].

We then modeled the temporal relation identification task with two sets of features. The first set gives a linguistic description of an isolated event: its tense, aspect, and other basic characteristics of the word. The goal of these features is to be able to handle complex tense-aspect interactions like the differences between Example 9 and Example 10:

(9) Travelers [EVENT *estimated*] that the California earthquake last month will [EVENT *cost*] $ 10 million.

(10) Travelers have [EVENT *estimated*] that the California earthquake last month [EVENT *cost*] $ 10 million.

In the former, we have (*estimated* BEFORE *cost*) because *estimate* appears in the past tense and *cost* appears in the future. In the latter, we have (*estimate* AFTER *cost*) because

[4] The annotated data can be downloaded from http://verbs.colorado.edu/~bethard/timebank-verb-clause.txt. Documents wsj_0006-0778 were used for the training set, and documents wsj_0781-1073 were used for the test set

[5] We used the TinySVM implementation from http://chasen.org/~taku/software/TinySVM/, with the standard one-vs-rest formulation to convert the binary SVM classifiers into multiclass classifiers.

estimate appears in the present perfect and *cost* appears in the past. The following features attempt to provide the information necessary to make these kinds of distinctions, and are used once for the first event and once for the second.

word The text of the event itself.

pos The Penn TreeBank gold-standard part-of-speech label for the event, e.g. NNS (plural noun) or VBD (past tense verb).

stem The morphological stem of the event[6], e.g. the stem of *approved* is *approve*

aux A bag-of-words feature including any auxiliaries and adverbs that modify the event, e.g. in the phrase *hasn't yet been fixed*, the words *has*, *n't*, *yet* and *been* are included.

modal True if any of the auxiliaries are modals, e.g. *will direct* and *couldn't estimate* are MODAL events.

time-class The TimeBank gold-standard class label for the event, e.g. STATE or REPORTING.

time-pos The TimeBank gold-standard part-of-speech label for the event, e.g. NOUN or VERB.

time-tense The TimeBank gold-standard tense label for the event, e.g. PAST or PRESENT.

time-aspect The TimeBank gold-standard aspect label for the event, e.g. PROGRESSIVE or PERFECTIVE.

time-polarity The TimeBank gold-standard polarity label for the event, e.g. POS or NEG.

Our second set of features for temporal relation identification aims not at the event words themselves, but at the words that connect one event to the other. Such features are crucial for identifying relations when there are explicit temporal function words like *before* or *after*.

(11) Ratners Group PLC [$_{\text{EVENT}}$ *raised*] its price after another concern [$_{\text{EVENT}}$ *said*] it would be prepared to outbid Ratners's initial offer.

The temporal connective *after* in the example above gives a clear indicator of the expected relation (*raised* AFTER *said*). The following features attempt to characterize these kinds of relations:

compl-word The text of the complementizer for the clause, e.g. *to*, *that* or *because*.

[6]Stems determined by the stemming table at http://xbean.cs.ccu.edu.tw/∼dan/XTag/morph-1.5/data/

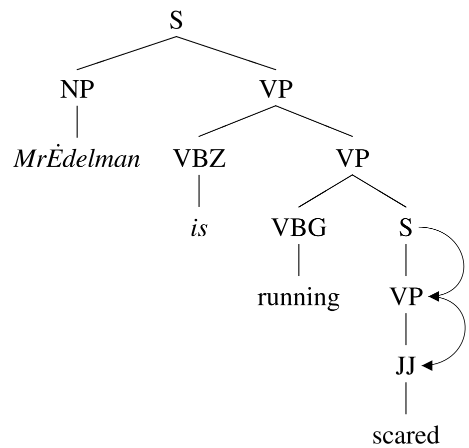

Figure 5. The target-path from *running* to *scared* is `S<VP<JJ`

compl-type The type of the complementizer, determined by a simple set of rules such that, for example, *after*, *because* and *since* are grouped under the type AFTER, and *as* and *while* are grouped under the type OVERLAP.

target-path The path of syntactic tree nodes from the clause to its head, e.g. in Figure 5, the path from *running* to *scared* is `S<VP<JJ`.

inter-words A bag-of-words feature including all words between the first event and the second.

func-words A restricted version of the inter-words feature which only includes auxiliaries, adverbs, prepositions and other function words, e.g. in the phrase *...[$_{\text{EVENT}}$* rose*] 11%, even though claims stemming from Hurricane Hugo [$_{\text{EVENT}}$* reduced*]...*, the function words are *even*, *though* and *from*.

Almost all of the features discussed above relied on either TreeBank or TimeBank annotations. While methods exist for automatically extracting syntactic trees and TimeBank events, we derived our features from the gold standard annotations in this research for a few reasons. First, some of the extraction methods are not currently at the levels of accuracy required for robust processing. For example, state-of-the-art systems for identifying events along with their TimeBank event classes have F-measures only in the 60s [1, 4]. Second, most existing systems which can produce the necessary annotations were trained on the TreeBank or TimeBank. Thus the performance of these systems on our data, which is a subset of the TreeBank and the TimeBank, would be artificially inflated. Finally, using features derived from gold-standard annotations allows other researchers to more easily reproduce and verify our results.

Model	Accuracy
Majority Class	44.3%
Tense & Aspect	50.0%
SVM	89.2%

Table 2. Model accuracies on the test data.

5 Results

The features extracted above were used to train an SVM model on the training data. To set the model's free parameters, five-fold cross-validations were performed on the training data at a number of different parameter settings, and the best of these were selected as the settings for the final model[7]. The performance of this model is compared against two baselines:

majority-class This model classifies all relations as BEFORE, the class which occurred most frequently in the training data.

tense-aspect This model classifies event pairs by looking only at the tense and aspect, and using traditional linguistic analysis to order the events. So for example, events in past tense are BEFORE events in future tense, and events in the present perfect are BEFORE events in the present.

These two baseline models are compared with the SVM model in Table 2. Both baselines perform poorly on the task (50% or less), while the SVM-based model reaches an accuracy of 89.2%.

Though the resulting model performance was quite good, we were interested in seeing whether these results could be improved by increasing the size of the corpus. We trained SVM models on increasing fractions of the training data to produce the training curve in Figure 6. The curve levels off to about 89% model accuracy once about 70% of the training data has been seen, suggesting that even our small corpus of only 895 event pairs was sufficient data for the task. Future progress on verb-clause temporal relations will therefore need to focus on identifying useful features, not building larger corpora.

So, to get a better idea of what kinds of features would be most helpful in this task, we performed a basic analysis of our feature set, training a single-feature model for each feature presented in Section 4. Table 3 shows the ten features whose models resulted in the highest cross-validation accuracies[8]. For example, using just the syntactic path or the tense of the clausal event, models were able to achieve 75% accuracy. Accuracies around 70% were achieved using the part of speech or auxiliaries of the clausal event, or using the words between the two events.

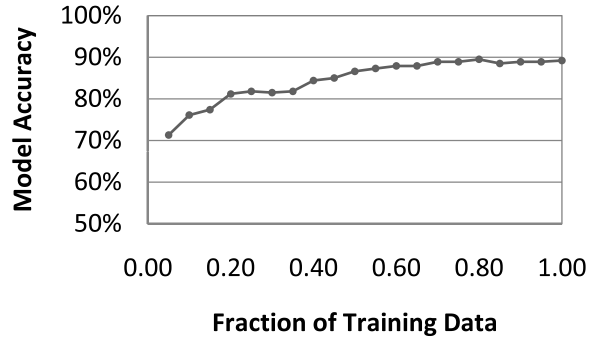

Figure 6. Training curve for the SVM model.

Features	Resulting Accuracy
target-path	75.2%
time-tense (2nd event)	75.0%
pos (2nd event)	71.2%
inter-words	69.7%
aux (2nd event)	69.3%
func-words	65.3%
word (2nd event)	58.8%
compl-word	56.9%
stem (2nd event)	51.7%
stem (1st event)	49.3%
top 10 features above	86.7%
all features	86.9%

Table 3. Cross-validation accuracies for various feature sets. The first ten rows are for single-feature models.

Table 3 also shows the five-fold cross-validation accuracies of the model trained with only the top ten features, and the model trained with all the features. The model trained with just the top ten features achieved an accuracy of 86.7%, while using all the features gained only another 0.2%. This suggests that the ten features identified here do a good job of characterizing the problem. Interestingly, of the top ten features, four characterized the words connecting the events, five characterized the second (clausal) event, but only one characterized the first (verbal) event. This suggests that for the verb-clause construction, the clausal event plays a more important role than the verbal event in determining the temporal relation.

As a final analysis of our models, we took a look at what kinds of errors they were making. We first discovered that our models performed substantially worse when one or the

[7] We only experimented with polynomial kernels. The cross-validations set both the cost of misclassification and the degree of the polynomial to 1.0

[8] We performed our feature analysis using five-fold cross validations rather using the test data so as to maintain the validity of the test set for future research.

other of the events was not a finite verb, like the *plunging* and *development* events in the examples below:

(12) The projection [$_{\text{EVENT}}$ *sent*] Anheuser shares [$_{\text{EVENT}}$ *plunging*] $4.375 in New York Stock Exchange composite trading yesterday.

(13) I [$_{\text{EVENT}}$ *think*] it's a pretty positive [$_{\text{EVENT}}$ *development*].

In general, on relations where the first event had a TimeBank tense label of NONE, our accuracy was only 77.3%, and similarly, when the second event had a tense of NONE our accuracy was only 67.6%. This 10-20% drop from the average model performance suggests that our models are still relying very heavily on the tense of the events to identify the relations between them, and features that can characterize other aspects of the events are still needed.

Our error analysis also showed that the models had more difficulty when the distance between the two events was greater. While on the average in our data there were about 4.7 words between the events in a pair, in the part of our data that was misclassified by our models, there were about 7.2 words between the events. This often corresponded to the presence of an additional embedded clause or other verbal object intervening between the two events, as in the following examples:

(14) Mr. Fournier [$_{\text{EVENT}}$ *said*] the large institutions *that hold nearly 50% of Navigation Mixte's capital* all strongly [$_{\text{EVENT}}$ *support*] him

(15) GM's North American vehicle production [$_{\text{EVENT}}$ *fell*] *8.4% from a year ago*, which [$_{\text{EVENT}}$ *hurt*] Delco Electronic's earnings

This finding suggests that though our features characterize some of the syntactic relations between the two events, better features that focus in on only the most important bits of syntax may still be necessary.

6 Conclusions

In this paper, we presented a new approach to automatically extracting timeline structure from text. We used syntactic information to guide our search for temporal relations, and focused on the verb-clause construction, which relates an verb to the head of its clausal argument. Using an extension of the TimeBank and TempEval annotation guidelines, we selected all pairs of events in a verb-clause construction in the Wall Street Journal section of the TimeBank corpus and manually annotated them with the labels BEFORE, OVERLAP and AFTER. The resulting corpus of 132 documents and 895 event-event temporal relations was then used to train a machine learning model.

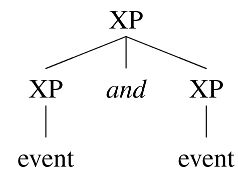

Figure 7. The coordination syntactic construction

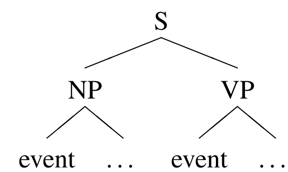

Figure 8. The verb-subject syntactic construction

By combining event-level features like tense and aspect with syntax-level features like the path between events in a syntactic tree, we were able to train a support vector machine (SVM) model which could identify new temporal relations with 89.2% accuracy. Analysis of the models showed that even our small corpus provided sufficient data for learning this task, and that high performance could be achieved with a small set of features characterizing the word level properties of the clausal event, and the lexical and syntactic path between the events.

These results suggest a number of additional avenues of research. We are moving on to annotate other syntactic constructions, in particular the coordination construction in Figure 7, which often indicates either simultaneity or succession, and the verb-subject construction in Figure 8, which often indicates an overlap relation. We also plan to explore using the predicate argument structures annotated in resources like PropBank [7] as a base on which to annotate temporal relations. Working with PropBank in this way would offer us the opportunity to incorporate timeline-style relations into an existing structured resource, thus taking an important step towards deeper and better integrated semantic representations.

7 Acknowledgments

This research was performed under an appointment of the first author to the Department of Homeland Security (DHS) Scholarship and Fellowship Program, administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the U.S. Department of Energy (DOE) and DHS. ORISE is managed by Oak Ridge Associated Universities (ORAU) under DOE contract number DE-AC05-06OR23100. All opinions expressed in this paper are the author's and do not necessarily reflect the policies and views of DHS, DOE, or ORAU/ORISE.

References

[1] S. Bethard and J. H. Martin. Identification of event mentions and their semantic class. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2006.

[2] S. Bethard and J. H. Martin. CU-TMP: Temporal relation classification using syntactic and semantic features. In *SemEval-2007: 4th International Workshop on Semantic Evaluations*, 2007.

[3] S. Bethard, R. Nielsen, J. H. Martin, W. Ward, and M. Palmer. Semantic integration in learning from text. In *AAAI Spring Symposium on Machine Reading*, 2007.

[4] B. Boguraev and R. K. Ando. Timebank-driven timeml analysis. In G. Katz, J. Pustejovsky, and F. Schilder, editors, *Annotating, Extracting and Reasoning about Time and Events*, Dagstuhl Seminars. German Research Foundation, 2005.

[5] R. Florian, A. Ittycheriah, H. Jing, and T. Zhang. Named entity recognition through classifier combination. In *Proceedings of CoNLL-2003*, pages 168–171, 2003.

[6] K. Hacioglu, B. Douglas, and Y. Chen. Detection of entity mentions occurring in english and chinese text. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 379–386, 2005.

[7] P. Kingsbury and M. Palmer. From treebank to propbank. In *Language Resources and Evaluation*, 2002.

[8] T. Kudo and Y. Matsumoto. Chunking with support vector machines. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2001.

[9] I. Mani, M. Verhagen, B. Wellner, C. M. Lee, and J. Pustejovsky. Machine learning of temporal relations. In *International Conference on Computational Linguistics and Association for Computational Linguistics (COLING/ACL)*, 2006.

[10] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330, 1994.

[11] S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J. H. Martin, and D. Jurafsky. Support vector learning for semantic argument classification. *Machine Learning*, 60(1):11–39, 2005.

[12] J. Pustejovsky, J. Castaño, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer, and G. Katz. Timeml: Robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*, 2003.

[13] J. Pustejovsky, P. Hanks, R. Saurí, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro, and M. Lazo. The timebank corpus. In *Corpus Linguistics*, pages 647–656, 2003.

[14] R. Saurí, R. Knippen, M. Verhagen, and J. Pustejovsky. Evita: A robust event recognizer for qa systems. In *Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP)*, 2005.

[15] K. Toutanova, A. Haghighi, and C. D. Manning. Joint learning improves semantic role labeling. In *ACL 2005*, 2005.

[16] M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, and J. Pustejovsky. Semeval-2007 task 15: Tempeval temporal relation identification. In *SemEval-2007: 4th International Workshop on Semantic Evaluations*, 2007.