

Automating Temporal Annotation with TARSQI

Marc Verhagen[†], Inderjeet Mani[‡], Roser Sauri[†],
Robert Knippen[†], Seok Bae Jang[‡], Jessica Littman[†],
Anna Rumshisky[†], John Phillips[‡], James Pustejovsky[†]

[†] Department of Computer Science, Brandeis University, Waltham, MA 02254, USA
{marc, roser, knippen, jlittman, arum, jamesp}@cs.brandeis.edu

[‡] Computational Linguistics, Georgetown University, Washington DC, USA
{im5, sbj3, jbp24}@georgetown.edu

Abstract

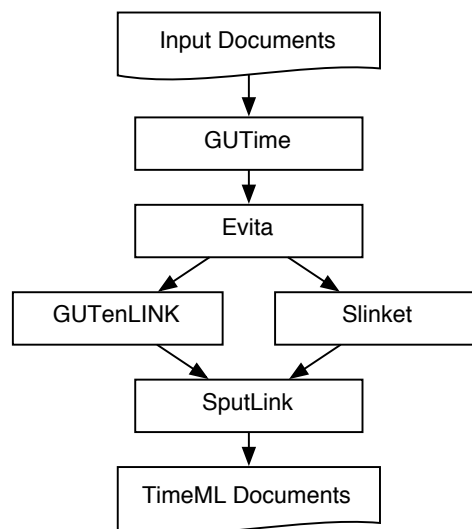
We present an overview of TARSQI, a modular system for automatic temporal annotation that adds time expressions, events and temporal relations to news texts.

1 Introduction

The TARSQI Project (Temporal Awareness and Reasoning Systems for Question Interpretation) aims to enhance natural language question answering systems so that temporally-based questions about the events and entities in news articles can be addressed appropriately. In order to answer those questions we need to know the temporal ordering of events in a text. Ideally, we would have a total ordering of all events in a text. That is, we want an event like *marched* in *ethnic Albanians marched Sunday in downtown Istanbul* to be not only temporally related to the nearby time expression *Sunday* but also ordered with respect to all other events in the text. We use TimeML (Pustejovsky et al., 2003; Sauri et al., 2004) as an annotation language for temporal markup. TimeML marks time expressions with the TIMEX3 tag, events with the EVENT tag, and temporal links with the TLINK tag. In addition, syntactic subordination of events, which often has temporal implications, can be annotated with the SLINK tag.

A complete manual TimeML annotation is not feasible due to the complexity of the task and the sheer amount of news text that awaits processing. The TARSQI system can be used stand-alone

or as a means to alleviate the tasks of human annotators. Parts of it have been intergrated in Tango, a graphical annotation environment for event ordering (Verhagen and Knippen, Forthcoming). The system is set up as a cascade of modules that successively add more and more TimeML annotation to a document. The input is assumed to be part-of-speech tagged and chunked. The overall system architecture is laid out in the diagram below.



In the following sections we describe the five TARSQI modules that add TimeML markup to news texts.

2 GUTime

The GUTime tagger, developed at Georgetown University, extends the capabilities of the TempEx tagger (Mani and Wilson, 2000). TempEx, developed

at MITRE, is aimed at the ACE TIMEX2 standard (timex2.mitre.org) for recognizing the extents and normalized values of time expressions. TempEx handles both absolute times (e.g., *June 2, 2003*) and relative times (e.g., *Thursday*) by means of a number of tests on the local context. Lexical triggers like *today*, *yesterday*, and *tomorrow*, when used in a specific sense, as well as words which indicate a positional offset, like *next month*, *last year*, *this coming Thursday* are resolved based on computing direction and magnitude with respect to a reference time, which is usually the document publication time.

GUTime extends TempEx to handle time expressions based on the TimeML TIMEX3 standard (timeml.org), which allows a functional style of encoding offsets in time expressions. For example, *last week* could be represented not only by the time value but also by an expression that could be evaluated to compute the value, namely, that it is the week preceding the week of the document date. GUTime also handles a variety of ACE TIMEX2 expressions not covered by TempEx, including durations, a variety of temporal modifiers, and European date formats. GUTime has been benchmarked on training data from the Time Expression Recognition and Normalization task (timex2.mitre.org/tern.html) at .85, .78, and .82 F-measure for *timex2*, *text*, and *val* fields respectively.

3 EVITA

Evita (Events in Text Analyzer) is an event recognition tool that performs two main tasks: robust event identification and analysis of grammatical features, such as tense and aspect. Event identification is based on the notion of event as defined in TimeML. Different strategies are used for identifying events within the categories of verb, noun, and adjective. Event identification of verbs is based on a lexical look-up, accompanied by a minimal contextual parsing, in order to exclude weak stative predicates such as *be* or *have*. Identifying events expressed by nouns, on the other hand, involves a disambiguation phase in addition to lexical lookup. Machine learning techniques are used to determine when an ambiguous noun is used with an event sense. Finally, identifying adjectival events takes the conservative approach of tagging as events only those ad-

jectives that have been lexically pre-selected from TimeBank¹, whenever they appear as the head of a predicative complement. For each element identified as denoting an event, a set of linguistic rules is applied in order to obtain its temporally relevant grammatical features, like tense and aspect. Evita relies on preprocessed input with part-of-speech tags and chunks. Current performance of Evita against TimeBank is .75 precision, .87 recall, and .80 F-measure. The low precision is mostly due to Evita's over-generation of generic events, which were not annotated in TimeBank.

4 GUTenLINK

Georgetown's GUTenLINK TLINK tagger uses hand-developed syntactic and lexical rules. It handles three different cases at present: (i) the event is anchored without a signal to a time expression within the same clause, (ii) the event is anchored without a signal to the document date speech time frame (as in the case of reporting verbs in news, which are often at or offset slightly from the speech time), and (iii) the event in a main clause is anchored with a signal or tense/aspect cue to the event in the main clause of the previous sentence. In case (iii), a finite state transducer is used to infer the likely temporal relation between the events based on TimeML tense and aspect features of each event. For example, a past tense non-stative verb followed by a past perfect non-stative verb, with grammatical aspect maintained, suggests that the second event precedes the first.

GUTenLINK uses default rules for ordering events; its handling of successive past tense non-stative verbs in case (iii) will not correctly order sequences like *Max fell. John pushed him*. GUTenLINK is intended as one component in a larger machine-learning based framework for ordering events. Another component which will be developed will leverage document-level inference, as in the machine learning approach of (Mani et al., 2003), which required annotation of a reference time (Reichenbach, 1947; Kamp and Reyle, 1993) for the event in each finite clause.

¹TimeBank is a 200-document news corpus manually annotated with TimeML tags. It contains about 8000 events, 2100 time expressions, 5700 TLINKs and 2600 SLINKs. See (Day et al., 2003) and www.timeml.org for more details.

An early version of GUTenLINK was scored at .75 precision on 10 documents. More formal Precision and Recall scoring is underway, but it compares favorably with an earlier approach developed at Georgetown. That approach converted event-event TLINKs from TimeBank 1.0 into feature vectors where the TLINK relation type was used as the class label (some classes were collapsed). A C5.0 decision rule learner trained on that data obtained an accuracy of .54 F-measure, with the low score being due mainly to data sparseness.

5 Slinket

Slinket (SLINK Events in Text) is an application currently being developed. Its purpose is to automatically introduce SLINKs, which in TimeML specify subordinating relations between pairs of events, and classify them into factive, counterfactive, evidential, negative evidential, and modal, based on the modal force of the subordinating event. Slinket requires chunked input with events.

SLINKs are introduced by a well-delimited subgroup of verbal and nominal predicates (such as *regret*, *say*, *promise* and *attempt*), and in most cases clearly signaled by the context of subordination. Slinket thus relies on a combination of lexical and syntactic knowledge. Lexical information is used to pre-select events that may introduce SLINKs. Predicate classes are taken from (Kiparsky and Kiparsky, 1970; Karttunen, 1971; Hooper, 1975) and subsequent elaborations of that work, as well as induced from the TimeBank corpus. A syntactic module is applied in order to properly identify the subordinated event, if any. This module is built as a cascade of shallow syntactic tasks such as clause boundary recognition and subject and object tagging. Such tasks are informed from both linguistic-based knowledge (Papageorgiou, 1997; Leffa, 1998) and corpora-induced rules (Sang and Déjéan, 2001); they are currently being implemented as sequences of finite-state transducers along the lines of (Aït-Mokhtar and Chanod, 1997). Evaluation results are not yet available.

6 SputLink

SputLink is a temporal closure component that takes known temporal relations in a text and derives new

implied relations from them, in effect making explicit what was implicit. A temporal closure component helps to find those global links that are not necessarily derived by other means. SputLink is based on James Allen's interval algebra (1983) and was inspired by (Setzer, 2001) and (Katz and Arosio, 2001) who both added a closure component to an annotation environment.

Allen reduces all events and time expressions to intervals and identifies 13 basic relations between the intervals. The temporal information in a document is represented as a graph where events and time expressions form the nodes and temporal relations label the edges. The SputLink algorithm, like Allen's, is basically a constraint propagation algorithm that uses a transitivity table to model the compositional behavior of all pairs of relations. For example, if A precedes B and B precedes C, then we can compose the two relations and infer that A precedes C. Allen allowed unlimited disjunctions of temporal relations on the edges and he acknowledged that inconsistency detection is not tractable in his algebra. One of SputLink's aims is to ensure consistency, therefore it uses a restricted version of Allen's algebra proposed by (Vilain et al., 1990). Inconsistency detection is tractable in this restricted algebra.

A SputLink evaluation on TimeBank showed that SputLink more than quadrupled the amount of temporal links in TimeBank, from 4200 to 17500. Moreover, closure adds non-local links that were systematically missed by the human annotators. Experimentation also showed that temporal closure allows one to structure the annotation task in such a way that it becomes possible to create a complete annotation from local temporal links only. See (Verhagen, 2004) for more details.

7 Conclusion and Future Work

The TARSQI system generates temporal information in news texts. The five modules presented here are held together by the TimeML annotation language and add time expressions (GUTime), events (Evita), subordination relations between events (Slinket), local temporal relations between times and events (GUTenLINK), and global temporal relations between times and events (SputLink).

In the nearby future, we will experiment with more strategies to extract temporal relations from texts. One avenue is to exploit temporal regularities in SLINKs, in effect using the output of Slinket as a means to derive even more TLINKs. We are also compiling more annotated data in order to provide more training data for machine learning approaches to TLINK extraction. SputLink currently uses only qualitative temporal information, it will be extended to use quantitative information, allowing it to reason over durations.

References

- Salah Ait-Mokhtar and Jean-Pierre Chanod. 1997. Subject and Object Dependency Extraction Using Finite-State Transducers. In *Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications. ACL/EACL-97 Workshop Proceedings*, pages 71–77, Madrid, Spain. Association for Computational Linguistics.
- James Allen. 1983. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, 26(11):832–843.
- David Day, Lisa Ferro, Robert Gaizauskas, Patrick Hanks, Marcia Lazo, James Pustejovsky, Roser Saurí, Andrew See, Andrea Setzer, and Beth Sundheim. 2003. The TimeBank Corpus. *Corpus Linguistics*.
- Joan Hooper. 1975. On Assertive Predicates. In John Kimball, editor, *Syntax and Semantics*, volume IV, pages 91–124. Academic Press, New York.
- Hans Kamp and Uwe Reyle, 1993. *From Discourse to Logic*, chapter 5, Tense and Aspect, pages 483–546. Kluwer Academic Publishers, Dordrecht, Netherlands.
- Lauri Karttunen. 1971. Some Observations on Factivity. In *Papers in Linguistics*, volume 4, pages 55–69.
- Graham Katz and Fabrizio Arosio. 2001. The Annotation of Temporal Information in Natural Language Sentences. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 104–111, Toulouse, France. Association for Computational Linguistics.
- Paul Kiparsky and Carol Kiparsky. 1970. Fact. In Manfred Bierwisch and Karl Erich Heidolph, editors, *Progress in Linguistics. A collection of Papers*, pages 143–173. Mouton, Paris.
- Vilson Leffa. 1998. Clause Processing in Complex Sentences. In *Proceedings of the First International Conference on Language Resources and Evaluation*, volume 1, pages 937–943, Granada, Spain. ELRA.
- Inderjeet Mani and George Wilson. 2000. Processing of News. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000)*, pages 69–76.
- Inderjeet Mani, Barry Schiffman, and Jianping Zhang. 2003. Inferring Temporal Ordering of Events in News. Short Paper. In *Proceedings of the Human Language Technology Conference (HLT-NAACL’03)*.
- Harris Papageorgiou. 1997. Clause Recognition in the Framework of Allignment. In Ruslan Mitkov and Nicolas Nicolov, editors, *Recent Advances in Natural Language Recognition*. John Benjamins, Amsterdam, The Netherlands.
- James Pustejovsky, José Castaño, Robert Ingria, Roser Saurí, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *IWCS-5 Fifth International Workshop on Computational Semantics*.
- Hans Reichenbach. 1947. *Elements of Symbolic Logic*. MacMillan, London.
- Tjong Kim Sang and Erik Herve Déjéan. 2001. Introduction to the CoNLL-2001 Shared Task: Clause Identification. In *Proceedings of the Fifth Workshop on Computational Language Learning (CoNLL-2001)*, pages 53–57, Toulouse, France. ACL.
- Roser Saurí, Jessica Littman, Robert Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2004. TimeML Annotation Guidelines. <http://www.timeml.org>.
- Andrea Setzer. 2001. *Temporal Information in Newswire Articles: an Annotation Scheme and Corpus Study*. Ph.D. thesis, University of Sheffield, Sheffield, UK.
- Marc Verhagen and Robert Knippen. Forthcoming. TANGO: A Graphical Annotation Environment for Ordering Relations. In James Pustejovsky and Robert Gaizauskas, editors, *Time and Event Recognition in Natural Language*. John Benjamin Publications.
- Marc Verhagen. 2004. *Times Between The Lines*. Ph.D. thesis, Brandeis University, Waltham, Massachusetts, USA.
- Marc Vilain, Henry Kautz, and Peter van Beek. 1990. Constraint propagation algorithms: A revised report. In D. S. Weld and J. de Kleer, editors, *Qualitative Reasoning about Physical Systems*, pages 373–381. Morgan Kaufman, San Mateo, California.