RC22974 (W0311-044) November 5, 2003
Computer Science

IBM Research Report

Extraction of Temporal Information from Text Documents

Branimir K. Boguraev
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Extraction of Temporal Information from Text Documents

Branimir K. Boguraev
IBM T.J. Watson Research Center
19 Skyline Drive, Hawthorne, NY 10532, USA
E-mail: bran@us.ibm.com

Abstract

Detailed analysis of time information in documents is a complex problem; the payoffs, however, for advanced applications capable of temporal reasoning are huge. This brief note argues that the graph-like representation typically maintained by temporal reasoners is derivable from what is an emerging standard for rich and robust annotation of temporal information in text.

We highlight some of the main features of TimeML, a temporal annotation language, and outline a mapping process which derives, from a TimeML-compliant representation, an isomorphic set of time-points and intervals. The problem of automatically analysing a document into TimeML is still too complex to tackle fully; however, a non-trivial fragment of TimeML analysis can be carried out by a finite-state based temporal expressions recogniser, running concurrently with a syntactic shallow parser. Broadly, we focus on strategies for identification and temporally anchoring of events. We also present an evaluation of some of the recognition capabilities as they apply to identification of temporal information fragments. The results are encouraging, as an independent evaluation shows that a temporal parser can be grounded into high accuracy recognition of key TimeML components. This, in its own turn, points at the viability of practical end-to-end natural language analysis and reasoning systems for advanced information management applications.

1 Temporal Analysis of Documents

It has been generally accepted that, short of full and deep understanding of contents of documents, a variety of 'gisting' approaches offer surrogate views into what a document is about. Consequently, (practical) content analysis has largely focused on identifying high information quotient-bearing text fragments: typically, mentions of named entities, and broader semantic categories of concepts: in isolation, chained, or linked in relational structures. These trends can be observed in the definition of community-wide efforts like the Message Understanding Conferences (MUC)[1] and the Automatic Content Extraction (ACE) evaluations[2] .

Clearly, documents are not so much about entities and concepts alone; rather they focus on events, which define a broad range of relationships among entities. Indeed, the evolution of content analysis tasks, from MUC to ACE, to include some relation identification, reflects this. However, one of the characteristic properties—if not the defining property—of events is that they take place in time. Still, the only extent to which some of the MUC or ACE tasks address the time analysis issue is is to look at a relatively narrow range of time expressions.

Recent efforts to broaden document analysis thus are beginning to focus on the temporal aspects of document content. In particular, work in automatic document summarization has addressed questions like identification and normalisation of time stamps [Mani and Wilson, 2000], time stamping of event clauses [Filatova and Hovy, 2001], and temporal ordering of events in news [Mani et al., 2003]. In the context of question answering (QA), operational systems can now produce literal answers to *e.g.* 'when' or 'how long' questions (assuming there has been, in a document, a factual statement with an explicit label of TIME, DATE, or DURATION).

Lately, additional constraints are emerging in the face of projects which require some form of temporal reasoning. Any form of advanced question answering, for instance, would need to concern itself with more than utilising, and manipulating, just information derived from 'bare' temporal markers (as those illustrated above). What is needed is a framework for making a temporal reasoner aware of the events described in a text, as well as of the ways in which these events are anchored in time, and relate to each other. This, in its own right, raises the complementary questions of representation rich and flexible enough to accommodate components of a temporal structure, and a text analysis process capable of yielding such a structure.

This brief note will highlight the main features of TimeML, an emerging standard for the annotation of temporal information in documents. As part of the definition of a sequence of temporal analysis operations, we will outline a mapping process which derives, from a TimeML-compliant representation, an isomorphic set of time-points and intervals: the grist to a temporal reasoner's

[1] See http://www.itl.nist.gov/iaui/894.02/related_projects/muc/main.html.

[2] See http://www.nist.gov/speech/tests/ace/index.htm.

mill. We will also outline a strategy for temporal analysis of document text, which uses a combination of finite-state based temporal expression patterns with a more syntactically oriented shallow parser; the argument is that linguistic structure informs a temporal analysis process which seeks to go beyond temporal expression identification alone.

2 TimeML: a Scheme for Temporal Annotation

The community (see *e.g.* [Ferro, 2001], [Gaizauskas and Setzer, 2002]) is still in relatively early stages of establishing uniform methods for representing temporal information; if for no other reason, this is largely due to the fact that no realistic application to date has connected the results of a document analysis process with a temporal reasoning component which, as mentioned above, typically traffics in more than just temporal markers like DATE or DURATION. Following largely Allen's pioneering work on the representation and maintenance of time intervals [Allen, 1983], temporal resoning systems need to maintain knowledge about time points, intervals, and temporal relations both among time expressions and events.

A broad community effort, TERQAS (Temporal and Event Recognition for QA Systems)[3], over the last 18 months has undertaken the design of a special purpose representation language for events and temporal expressions. The language, TimeML, aims at being able to capture the richness of temporal information in documents. In particular, TimeML goes beyond specification of a tagging scheme for temporal expressions only, and focuses, among other things, on ways of systematically anchoring event predicates to a broad range of temporally denotating expressions, and on ordering such event expressions (relative to each other). The language provides for delayed evaluation of contextually underspecified, or partially determined, temporal expressions (such as *last year* and *two months before*). What follows is a brief sketch of TimeML's characteristic features; [Pustejovsky et al., 2003] offer more details.

TimeML derives larger expressive power by means of explicitly separating the representation of temporal expressions from that of events; additionally, it allows for anchoring, or ordering, dependencies that may exist in text. The reresentation makes use of four component structures: TIMEX3, SIGNAL, EVENT, and LINK.

TIMEX3 extends the TIDES TIMEX2 [Ferro, 2001] annotation attributes; it is taken to denote temporal expressions (subsuming common notions like DATE, TIME, DURATION), as well as intensionally specified expressions like the examples above, handled by the definition of temporal functions. SIGNAL is a tag for annotating (typically) function words which indicate how temporal objects are to be related to one another; examples here include temporal prepositions (like *for*, *during*, *at*) or temporal connectives (*vefore*, *after*, *while*). EVENT is a cover term for situations that happen or occur; these can be punctual, or last for a

[3] See `http://www.timeml.org/terqas/index.html`.

period of time.

TimeML introduces a refined ontology of events [Pustejovsky et al., 2003]. All classes of event expressions: tensed verbs, stative adjectives and other modifiers, event nominals, are marked up with suitable properties on the EVENT tag. The LINK tag is used to encode a variety of relations that exist between the temporal elements in a document, as well as to establish an explicit ordering of events. Three subtypes to the LINK tag are used to represent strict temporal relationships between events, or between an event and a time (TLINK), subordination between two events or an event and a signal (SLINK), and aspectual relationship between an aspectual event and its argument (ALINK).

Without going into specific detail, the flavour of a TimeML representation can be conveyed by showing the analysis, and tagging, of *"The terrorists convened two days before the attack"*.

```
The terrorists
<EVENT eid="e1" class="OCCURRENCE" tense="PAST" aspect="PERFECTIVE">
convened
</EVENT>
<TIMEX3 tid="t1" type="DURATION" value="P2D" temporalFunction="false">
two days
</TIMEX3>
<SIGNAL sid="s1">before</SIGNAL>
the
<EVENT eid="e2" class="OCCURRENCE" tense="NONE" aspect="NONE">
attack
</EVENT>
<MAKEINSTANCE eiid="ei1" eventID="e1"/>
<MAKEINSTANCE eiid="ei2" eventID="e2"/>
<TLINK eventInstance="ei1" signalId="s1" relatedToEvent="ei2"
       relType="BEFORE" magnitude="t1"/>
```

3 TimeML and Temporal Reasoning

TimeML is sufficiently rich in expressive power; in particular, it is capable of deconstructing Allen's relations on time intervals. This suggests that any temporal analysis scheme which is consistent with TimeML representational principles could be harnessed, with relatively low 'translation' effort, for driving an existing knowledge-based reasoner.

For the KANI[4] project in the NIMD program, Stanford's Knowledge Systems Laboratory is developing a hybrid reasoner to be deployed in intelligence analysis scenarios [Fikes et al., 2003]. The reasoner maintains a directed graph of time points, which is based on temporal relations such as BEFORE, AFTER, and EQUAL_POINT; it also represents intervals using their strating and ending

[4] Knowledge Associates for Novel Intelligence.

points. Temporal relations are operationalised, and temporal algebra facilitates evaluation over instances, draws inference over instances of goals, and broadens a base of inferred assertions on the basis of relational axioms. An example of an operation within the reasoner's inferential capability would be find instances of `?int` such that `(during ?int 2003)`.

The figure below cites a sample text, for which the reasoner would assume a graph with relations (among others) such as *during* (associating an event with a time point), *costarts* (associating two events), *etc.*

On 9 August Iran accuses the Taliban of taking 9 diplomats and 35 truck drivers hostage in Mazar-e-Sharif. The crisis began with that accusation. On 2 November Iran concludes the Zolfaghar-2 military exercise peacefully, ending the crisis between the two sides. On 5 September Iran states that it has the right under international law to strike the Taliban after Iranian media sources report that the Taliban have killed 5 Iranian diplomats.

On the basis of predicates like:

(during Iran-accuses-Taliban-of-taking-hostages August-9-1998)

(costarts Iran-accuses-Taliban-of-taking-hostages Iranian-Taliban-Crisis)

the reasoner would, for instance, infer that the answer for the question *"When did the Iranian-Taliban crisis begin?"* is *"August 9, 1998"*.

The details of this inferential process need not concern us here. What is more central to the argument in this note is that the particular representation assumed by the reasoner is derivable from a TimeML analysis of the same text. The fragment below is indicative of such an analysis.

```
<signal sid="s1"> On </signal>
<timex3 tid="t1" type="DATE" temporalFunction="true" value="........">
9 August
</timex3>
Iran
<event eid="e1" class="I_ACTION"> accuses </event>
the Taliban
of taking 9 diplomats and 35 truck drivers hostage in Mazar-e-Sharif.
The
<event eid="e8" class="OCCURRENCE"> crisis </event>
<event eid="e12" class="ASPECTUAL"> began </event>
<signal sid="s2" type="DATE" mod="START"> with </signal>
that
<event eid="e16" class="I_ACTION"> accusation </event>
.
<makeinstance eiid="ei1" eventId="e1"/>
<makeinstance eiid="ei2" eventId="e8"/>
<makeinstance eiid="ei3" eventId="e12"/>
<makeinstance eiid="ei4" eventId="e16"/>
```

```
<tlink eventInstanceId="ei1" relatedToTime="t1"
       relType="IS\_INCLUDED"/>
<tlink eventInstanceId="ei4" relatedToEventInstance="ei1"
       relType="IDENTITY"/>
<alink eventInstanceId="ei2" relatedToEventInstance="ei4"
       relType="INITIATES"/>
```

The event instance identifiers, ei1, ei2, and ei4 refer to, respectively, the accusation in the first sentence, the reference to it (*"that accusation"*) in the second sentence, and the crisis. Notice the relType attributes on the event instance definitions. It is the combination of event descriptors, their anchoring to time points (*e.g.* t1, namely *"9 August"*), and the semantics of relational links, which makes it possible to derive *during* and *costarts* associations that the reasoner understands from the particular combination of IS_INCLUDED, IDENTITY and INITIATES relational labels in the TimeML analysis.

4 TimeML and Temporal Analysis

The problem of automatically analysing a document into TimeML is too complex to tackle fully. In particular, deriving, reliably, the LINK information crucial for the completeness of TimeML representation—and for the mapping outlined in the previous section—is beyond the capability of present day automatic language analysis. Even event identification is far from a solved problem, given the complexity of the linguistic notion of 'event', which TimeML's representation relies upon.

The TERQAS workshop is committed to applying the annotation standard to a reference TimeBank corpus. The intent, then, is to use that corpus (when complete) as a language resource from which an analysis device could be trained to do at least some LINK typing. In the mean time, the question remains of how much of an analysis can be carried out by automatic means.

Broadly, any temporal parser needs to address the following issues during the process of temporal anchoring:

- Find a temporal expression;
- Analyse that, in structural terms;
- Find an associated event;
- Represent that, in structural terms;
- Associate a temporal expression (or another event, as appropriate) with the event;
- Resolve references to other times / events;
- Order events temporally.

These operations are intrinsic to the process of temporal analysis, rather than specifically mediated by *e.g.* TimeML requirements. Viewed from such a perspective, it is possible both to assess the feasibility / complexity of individual steps, as well as to plan for an optimal combination of accurate analysis and a rich representation.

A full paper will analyse, in more detail, each of the phases above. The remainder of this section outlines the approach we adopt for temporal analysis.

A core component of the KANI project within the NIMD program is an information extraction associate which assumes a capability for identifying text fragments referring to a broad range of ontologically relevant semantic categories. Within the set of categories that this extractor focuses on, we have developed finite-state grammars for abstract temporal units, such as UNIT, POINT, SPAN, PERIOD, RELATION, and so forth. The syntax of temporal expressions is such that, given an expressive formalism for writing patterns over linguistic annotations, it is possible to cover a broad range of open-ended expression types. We use a flexible FST system within an annotations-based pipelined architecture for document processing and analysis [Boguraev and Neff, 2003], [Neff et al., 2003].

The point of specifying patterns over *linguistic* units, as opposed to simply lexical cues (as most temporal taggers to date do) cannot be over-emphasised. One of the big issues in temporal analysis, as discussed at length already, is that of event identification. A temporal tagger, if narrowly focused on time expressions only (see, for instance [Schilder and Habel, 2003]), offers no clues as to what events there are in the text. A temporal parser, on the other hand, capable of a temporal expression like *"during the long and ultimately unsuccessful war in Afghanistan"* is very close to knowing—by virtue of interpreting the syntactic constraints underlying a prepositional expression—that the head of the noun phrase which is the argument of the temporal preposition (*i.e. "war"*) is an event nominal.

The temporal grammars, therefore, run in coordination with a shallow syntactic parser; this is also realised [Boguraev, 2000] as a cascade of finite-state devices, which makes for smooth integration of temporal and syntactic analysis. Using, in addition, a mechanism for accessing external resources, it is also possible to query an authority file about the event-denoting status of certain lexical items in key syntactic positions. This facilitates the implementation of a temporal parser which, in effect, deposits three types of TimeML tags into the document stream: TIMEX3, SIGNAL, and EVENT.

It is a matter of additional study to evaluate the breadth of EVENT tag coverage, as well as the correctness of attributes derived by the parser. Also, it is still an open research question what would be an effective strategy for identifying and typing the LINK's between events, other events, and times.

However, initial experiments suggest that an existing, finite-state based, temporal expressions tagger developed for independent purposes can be adapted for partial instantiation of a TimeML representation. [Boguraev, 2003] describes an experiment in hybrid named entity tagging, which uses a mix of finite-state grammars and statistical learning methods concurrently. As part of that experiment, an evaluation was performed of the coverage of the FS-based grammars alone, over a range of categories. Focusing on temporal categories alone, the results—over an independently developed and manually annotated test data—are as follows.

```
     AGO: precision: 100.00%; recall:  83.33%; FB1:  90.91
    DATE: precision:  96.48%; recall:  95.31%; FB1:  95.89
DURATION: precision:  85.10%; recall:  80.82%; FB1:  82.90
    TIME: precision:  97.06%; recall:  95.65%; FB1:  96.35
    YEAR: precision:  89.13%; recall:  97.04%; FB1:  92.92
```

What makes these figures interesting, and relevant to this discussion, is that the temporal tagger evaluated was derived, by minimal re-organisation, from the temporal parser discussed earlier in this section. The changes to it, introduced in the process of adapting it to the set of pre-defined temporal categories are naturally, and equally easily, transferrable back to the original TimeML parser component, making it demonstrably a high precision recogniser.

5 Conclusion

The results reported in the previous section are encouraging, not only because they are indicative of high accuracy within a class of grammars comprising a subset of the larger solution to building a TimeML parser, but because they also suggest that reliable seed analyses can be derived, over which larger syntactic fragments can be constructed and their internal structure exploited. Our intent is to develop enough of a parsing function, to be able to take full advantage of the TimeBank corpus, when it becomes available.[5]

Even if a TimeML parser would be incomplete in terms of coverage, it will serve an important function as a bridge connecting state-of-the-art, scalable, natural language processing techniques for content analysis and information extraction with temporal reasoning logic, which is crucially required by advanced information management applications like question answering, summarisation, scenario analysis, and hypothesis generation.

[5] Current expectations are for a pre-release version towards the middle of 2004.

References

[Allen, 1983] Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11).

[Boguraev, 2000] Boguraev, B. (2000). Towards finite-state analysis of lexical cohesion. In *Proceedings of the 3rd International Conference on Finite-State Methods for NLP*, INTEX-3, Liege, Belgium.

[Boguraev, 2003] Boguraev, B. (2003). The contribution of finite-state technology to named entity recognition and typing. Technical Report KSL-03-01, IBM T.J. Watson Research Center, Yorktown Heights, New York.

[Boguraev and Neff, 2003] Boguraev, B. and Neff, M. (2003). *The Talent 5.1 TFst system: user documentation and grammar writing manual*. IBM T.J. Watson Research Center, Yorktown Heights, New York.

[Ferro, 2001] Ferro, L. (2001). TIDES: Instruction manual for the annotation of temporal expressions. Technical Report MTR 01W0000046V01, The MITRE Corporation.

[Fikes et al., 2003] Fikes, R., Jenkins, J., and Frank, G. (2003). JTP: A system architecture and component library for hybrid reasoning. Technical report, Knowledge Systems Laboratory, Stanford University.

[Filatova and Hovy, 2001] Filatova, E. and Hovy, E. (2001). Assigning timestamps to event-clauses. In *Proceedings of the 10th Conference of the European Chapter of the ACL*, Toulouse, France.

[Gaizauskas and Setzer, 2002] Gaizauskas, R. and Setzer, A., editors (2002). *Annotation Standards for Temporal Information in NL*, Las Palmas, Spain.

[Mani et al., 2003] Mani, I., Schiffman, B., and Zhang, J. (2003). Inferring temporal ordering of events in news. In *Human Language Technology/North American Chapter of the ACL (HLT-NAACL*, Edmonton, Canada.

[Mani and Wilson, 2000] Mani, I. and Wilson, G. (2000). Robust temporal processing of news. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hong Kong.

[Neff et al., 2003] Neff, M., Byrd, R., and Boguraev, B. (2003). The Talent system: TEXTRACT architecture and data model. Technical report, IBM T.J. Watson Research Center. Originally presented at HLT-NAACL Workshop on Software Engineering and Architectures of Language Technology Systems, Edmonton, Canada.

[Pustejovsky et al., 2003] Pustejovsky, J., Mani, I., Gaizauskas, R., Ferro, L., Katz, G., and Radev, D. (2003). TimeML: Robust specification of event and temporal expressions in text. In *AAAI Spring Symposium on Intelligent Question-Answering Systems*, Stanford, CA.

[Schilder and Habel, 2003] Schilder, F. and Habel, C. (2003). Temporal information extraction for temporal QA. In *AAAI Spring Symposium on Intelligent Question-Answering Systems*, Stanford, CA.