Association of Absolute Times with Events: Roadmap for Development

Date: 20 July 2002
Author: Beth Sundheim

Introduction

This document is concerned with the current status of TimeML in relation to the handling of TIMEX3 expressions, and where it could go in the future. The document lists TIMEX3-related tasks in an order that approximately reflects increasing amounts of context and increasing degrees of difficulty. The list represents a detailed sort of roadmap for developing a broad capability to automatically associate absolute times with TimeML events. This particular roadmap is not concerned with the relative ordering of events, but with the "timestamping" of events.

The overall goal is to enable placement of each event at a certain point (or pair of points, if the event has obvious duration) on a timeline. This capability will enable effective question answering about event times within and across documents.

Level of annotation in Timebank

The annotation in Timebank reflects the sorts of analysis covered by the current version of the TimeML specification and guidelines. Items 1, 2 and 3 on the list below are included in the current TimeML and are therefore represented in Timebank. The remaining items may be considered in the development of future versions of TimeML and Timebank.

Some of the remaining items will require only elaborations of the current guidelines concerning usage of the current schema, while others will require new ideas for representation in the annotation. A judgment about the difficulty of coming up with new representation solutions is shown for each item on the list. A judgment of the difficulty of implementation of each item is also made. Some items present challenges that are well beyond the state of the art.

Evolution from TIMEX2 to TIMEX3 and beyond

TIMEX2 and TIMEX3 treat essentially the same types of time expressions, but from different perspectives. TIMEX2 views them as the sole focus of attention -- everything that needs to be represented about a time, including its relation to another time, is treated as an attribute of the time expression itself. TIMEX3 views them as part of TimeML, where the focus of attention is on events. TimeML offers several advantages over TIMEX2:

- TimeML permits more connections to be made between time expressions than was possible with TIMEX2. Some relations between time expressions that were implicit in TIMEX2 (e.g., the relation that causes "8 p.m." in the phrase "8 p.m. on Friday" to be normalized to include the value that comes from the totally separate TIMEX2, "Friday") are now captured in a more explicit way via TLINK tags between an EVENT and a TIMEX3 (e.g., as "They are meeting at 8 p.m. on Friday").
- TimeML defines a SIGNAL tag, which is used in part to identify the temporal prepositions that introduce a time expression. These prepositions, which are outside the extent of the TIMEX2/TIMEX3 tag, are referenced implicitly in TIMEX2 in order to distinguish durations from points, a distinction which is reflected in the form of the ISO notation used. TimeML makes the presence of the signal explicit.
- TimeML sometimes offers a more natural way of representing some of the same information that was covered by TIMEX2. Given TimeML, attributes that capture some of the meaning of time expressions can now be defined for EVENT or MAKEINSTANCE when that is reasonable (e.g., the magnitude attribute on EVENT and the cardinality attribute on MAKEINSTANCE).
- TimeML has "more to say" about the meaning of some time expressions, especially those that are anchored in events. With TIMEX2, an expression such as "20 minutes before the meeting" would be bracketed as a time expression but would have no assigned attributes. With TimeML, the annotation captures the fact that the interval is prior to the meeting and that its duration has the value of 20 minutes.

TIMEX3 is consistent with TIMEX2 not only in the range of expressions that they are concerned with, but also in the commitment to an extended ISO 8601 normalization. However, there are some differences in the way in which the two tags are defined; a few of the most obvious ones are outlined below:

- TIMEX3 does not use TIMEX2's SET, PERIODICITY and GRANULARITY attributes. Instead, these set-denoting attributes are expected to be handled principally through the MAKEINSTANCE tag.
- TIMEX3 does not use TIMEX2's NON_SPECIFIC attribute. This attribute, which identifies generic, indefinite and other nonreferential types of expressions, is not defined clearly enough in the TIMEX2 guidelines to result in good interannotator agreement, and its meaning and application in TimeML will need to be considered in future discussions.
- TIMEX3 has a defined "type" attribute, whose possible values are DATE, TIME and DURATION. Although the value of this attribute is predictable from the form of the ISO value (and therefore there is no equivalent attribute defined for TIMEX2), it may be used in the future to permit an expression such as "2 months from today" to have an overall DATE type, although the matrix portion of the expression by itself expresses a DURATION.

- In contrast to TIMEX2, TIMEX3 does not permit the nesting of one TIMEX3 inside another. Since TimeML does not currently permit TLINKs to connect one time expression directly to another, the consequence of disallowing nesting is that the embedded expression is not explicitly represented in the annotation. Thus, in "two weeks from next Tuesday" and "three years ago today", the embedded expressions, "Tuesday" and "today", respectively, are not tagged as distinct time expressions.
- With the exception of the types of expressions covered in the previous bullet, the extent of an expression in TIMEX3 excludes right modifiers. Although TIMEX2 and TIMEX3 differ in this regard, the difference is only superficial.
- TIMEX3 requires an anchorTimeID attribute for all indexical expressions. TimeML permits a variety of reference times to be identified in a document; thus, the annotation must specify which of the available reference times is to be used in normalizing any particular expression. The available reference times are annotated using the TIMEX3 functionInDocument attribute. In contrast, TIMEX2 made the simpifying assumption that the reference time for indexicals was to be found ultimately in the only time defined for the document, namely the "story reference time" (which was merely implicit in the TIMEX2 schema, but was made explicit via a separate tag, STORY_REF_TIME, for the ACE RDC task).

Ordered list of tasks for associating absolute times with events

1. Goal: Generate ISO values for the unanchored durations and complete dates/times.

Analysis tasks: Recognize markable TIMEX3s, determine correct extent, identify TIMEX3s that convey unanchored durations and "complete", non-indexical dates/times.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
Y	Y	N/A	Easy-Med

The analysis tasks entailed by this item can be readily handled by current technology.

Durations can have 'X' placeholder in the numeric field, e.g., "PXM". Interpretation of the placeholder character is left for a later task (see item 5).

Example expressions[1]: "[July 20, 2002]", "[04/02/1998 22:52:00]" (from a document header), "for [6 months]", "for [two and a half years]", "for [a few hours]"

[1] In the examples given in the list, the extent of the time expression, as defined by TIMEX3 guidelines, is indicated by the use of square brackets.

2. Goal: Compute and generate the ISO value of simple indexicals.
Analysis tasks: Identify "simple" TIMEX3s that are indexicals, determine correct reference time.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
Y	Y	N/A	Easy-Med

"Simple" expressions are those that do not contain a nested time expression. The handling of "complex" TIMEX3s is addressed in other steps in the roadmap.

For TIMEX2, these tasks are handled by programs such as TEMPEX, with the simplifying assumption that the reference time is always the "document time".

For TIMEX3, these tasks are handled in part by temporal associated with TIMEX3s and in part by algorithms associated with TLINKs, as indicated below. The latter algorithms are only partially implemented at this time.

Example expressions (these are completely handled by TIMEX3 temporal functions): "[yesterday]", "on [Thursday]", "[last week]", "[eight o'clock]".

Example phrases containing more than one simple indexical expression (TIMEX3 temporal functions handle only part of the analysis; incorporation of the value of the value of the more general expression into the value of the more specific expression is handled in TimeML by the TLINK algorithms): "at [eight o'clock] on [Thursday]", "[last week] on [Wednesday]"

3. Goal: Compute and generate the ISO value of complex indexicals.
Analysis tasks: Identify complex indexicals in which the granularity of the "magnitude" (duration) expression is no finer than the granularity of the anchoring time expression. The anchor may be expressed directly or indirectly.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
Y	Y/N	Med	Easy-Med

Cases where the granularity of the magnitude expression is *coarser* than that of the anchoring expression are addressed later in this document (see item 10).

TIMEX2 adequately handles type (a) expressions, below, but not type (b). Type (b) expressions are tagged as TIMEX2s, but are not given any value.

TimeML covers both types of expression. Type (a) expressions are handled by temporal functions, while type (b) expressions are (or will be) handled by TLINK

generation algorithms. The guidelines need to be reviewed to see if all expected types of examples are handled in a natural way, without redundancy between the temporal functions and the TLINK algorithms.

Example expressions:

(a) Anchor is expressed directly: "[two days before April 15]", "[three years ago today]". These are treated as a single TIMEX3.

(b) Anchor is expressed indirectly: "He arrived [20 minutes] before the [8:00] meeting", "He arrived [the day] before the meeting on [Tuesday]". These are treated using TLINKs between each event and its associated time expression.

4. Goal: Compute the ISO value of the missing endpoint of a semi-bounded duration; generate a TLINK that expresses that value (?)

Analysis tasks: Recognize cases where an event instance (or multiple instances TLINKed in an IDENTITY relation) has a TLINK that expresses a HOLDS relation *and* a TLINK that expresses a BEGUN_BY or ENDED_BY relation, e.g., "Jan spoke for *four hours* starting at *8:30*"). Determine appropriate granularity for value of the missing endpoint.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	Easy	Easy-Med

Given one endpoint and a duration, it should be possible to supply the other endpoint in order to anchor an event at both ends on the timeline, but TimeML currently does not permit TLINK to have an ISO value attribute (it has to obtain that value indirectly through a TLINK to a TIMEX3). A simple solution might be to create not only a TLINK, but also an empty TIMEX3 tag.

5. Goal: Anchor unbounded duration expression from general context, by either (a) computing and generating ISO value for the endpoint(s) or (b) adopting an existing ISO value from another time expression that refers to a time that's within the time range denoted by the duration expression.

Analysis tasks: Find non-local evidence (via network of SLINKs and TLINKs) that identifies (a) one or both endpoints of the duration, (b) an appropriate reference time that's somewhere between the endpoints of the duration, or (c) an appropriate reference time that the duration is "contained in".

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	Easy	Hard

This task is focused on durations that have either a known or unknown length. The former type is indicated by a numeric in the value (e.g., "P3M") and the latter by a placeholder in the value (e.g., "PXM").

Given a network of SLINKs and TLINKs, it may be possible to track down an absolute time to associate with an unbounded duration expression. Preferably, the analysis would permit the endpoints of the duration to be identified, but if not, it should permit some intermediate point in the duration to be identified in many cases. Presumably, the results of analysis would be represented in the form of new TLINKs. However, TimeML currently does not permit a TLINK to have an ISO value attribute (it has to obtain that value indirectly through a TLINK to a TIMEX3). A simple solution might be to create not only a TLINK, but also an empty TIMEX3 tag.

Example of identifying endpoints: " Two Russians and a Frenchman left the Mir and endured a rough landing on the snow-covered plains of Central Asia on [Thursday]. ... The two Russians arrived on the Mir [last August] ... Solovyov ... celebrated his 50th birthday during his [six-month] space voyage."

Example of identifying a containment time: "Two Russians and a Frenchman left the Mir and endured a rough landing on the snow-covered plains of Central Asia on [Thursday]. ... It took the crew [three hours] to descend from the Mir, ..."

6. Goal: Overlay fuzzy time unit values with precise values

Analysis tasks: Find evidence in general context for intended interpretation of one or both endpoints of interval denoted by expressions of fuzzy time units (including elements such as "SU" within an otherwise precise ISO value, and also values such as "PRESENT_REF").

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	Med	Hard

This "overlay" could be in the form of a document-level annotation defining the interval of the fuzzy unit.

Example in which one or more points (as in item 5, analysis tasks (a) and (b)) pertaining to a fuzzy unit can be identified: "She went to Germany [last summer]. She arrived in [mid-June]; by [mid-July], she was ready to go home."

7. Goal: Overlay "mod" attribute values with precise values

Analysis tasks: Find evidence in general context for a more precise interpretation of modifier expressions that are represented using "mod".

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	Med	Hard

This "overlay" could be in the form of one or more new TIMEX3 attributes that permit expression of the precise meaning of the modifier, or in the form of an IDENTITY type of TLINK between TIMEX3s that have the same intended value, given precise interpretation of the modifier, and/or in some other form. Since the mod attribute conveys a variety of fuzzy things, the solution to this task could require significant extensions to TIMEX3/TimeML.

Examples of adding precision to modifier terms (modifier terms are in italics): "He left work in the [*early* afternoon]... He sneaked out of the office at [1:00] to go to the ball game." "The meeting ended at [*around* 11:00]... They were adjourned [15 minutes] before the cafeteria opened at [11:30] for lunch."

8. Goal: Given a set of event instances for a single event and a set of associated times, select the most precise one for use in applications such as timeline display

Analysis tasks: Recognize event instances that are TLINKed via an IDENTICAL relation and that have different but "compatible" event-TIMEX3 TLINK types and TIMEX3 values, and determine the nature of the compatibility.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	N/A?	Easy-Med

This may be an application-dependent task, and may not require any explicit representation in the TimeML output.

9. Goal: Overlay vague TLINK type values with precise values.

Analysis tasks: In the case where an event that has no time associated with it is linked via a TLINK that has a fuzzy type value such as I_AFTER or IS_INCLUDED, use general context to determine a precise interpretation of the type value.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	N	Med	Hard

10. Goal: Generate vague value for complex indexicals with vague anchor.

Analysis tasks: Recognize granularity of "magnitude" (duration) portion of complex indexicals that are anchored directly or indirectly to a time expression that has coarser (or vague) granularity.

Covered by TIMEX2 guidelines	Covered by TimeML/TIMEX3 guidelines	Difficulty of TimeML/TIMEX3 extension	Difficulty for algorithm development
N	Y/N	Med-Hard	N/A

Under TIMEX2 guidelines, both types of expression are tagged as TIMEX2s, but are not given any value.

If we assume that other tasks on this roadmap have failed to identify an exact date value for "the end of April" and "the August meeting", then these complex expressions cannot be resolved to a particular date either. TimeML needs to develop a way to represent the vague endpoint of a duration.

Example expressions:

(a) Anchor is expressed directly, e.g. "[two days before the end of April]". This type is treated as a single TIMEX3.

(b) Anchor is expressed indirectly, e.g., "He arrived [two days] before the [August] meeting". These are treated using TLINKs between each event and its associated time expression.

11. Characterize elements of a set

12. Do something with non-specific times