TimeML in a Nutshell

Roser Saurí, James Pustejovsky

May 7, 2009

1 Introduction

The current document provides a very general overview of the TimeML language, a specification language for events and temporal expressions, and the relations held between them. It is conceived as the first reading for new TimeML annotators, who will then sequentially deepen into the specifics of each annotation task.

TimeML (Pustejovsky et al., 2005) is a specification language for events and temporal expressions. It was first developed in 2002 in an extended workshop called TERQAS (Time and Event Recognition for Question Answering Systems),[1] which focussed on the issue of answering temporally based questions regarding events and entities in news articles. In 2003, TimeML was further developed in the context of the TANGO workshop (TimeML Annotation Graphical Organizer).[2] In addition, TimeML has been consolidated as an international cross-language ISO standard (ISO WD 24617-1:2007), and has been approved as the annotation language for TempEval, one of the tasks in the SemEval International Workshop on Semantic Evaluations (Verhagen et al., 2007, 2009).

Events in articles are naturally anchored in time within the narrative of a text. For this reason, temporally grounded events are the very foundation from which we reason about how the world changes. Without a robust ability to identify and extract events and their temporal anchoring from a text, the real "aboutness" of the article can be missed. Moreover, since entities and their properties change over time, a database of assertions about entities will be incomplete or incorrect if it does not capture how these properties are updated throughout the timeline.

For example, currently questions such as those shown below are not supported by question answering systems.

(1) a. Is Gates currently CEO of Microsoft?

b. When did Iraq finally pull out of Kuwait during the war in the 1990s?

c. Did the Enron merger with Dynegy take place?

[1] http://www.timeml.org/site/terqas/index.html
[2] http://www.timeml.org/site/tango/index.html

What characterizes these questions as beyond the scope of current systems is the following: they refer, respectively, to the temporal aspects of the properties of the entities being questioned, the relative ordering of events in the world, and events that are mentioned in news articles, but which have never occurred.

Specifically, event-temporal identification involve these four basic problems:

(a) Time stamping of events, that is, identifying an event and anchoring it in time;

(b) Ordering events with respect to one another;

(c) Reasoning with contextually underspecified temporal expressions (temporal functions such as *last week* and *two weeks before*);

(d) Reasoning about the persistence of events (how long does an event or the outcome of an event last).

The specification language, TimeML, is designed to address these issues, in addition to handling relevant grammatical features such as tense and aspect.

2 TimeML

There are four major data structures that are specified in TimeML: `EVENT`, `TIMEX3`, `SIGNAL`, and `LINK`. Each of these are described in some detail in the following sections.

2.1 Events

TimeML considers "event" a cover term for situations that happen or occur. Events can be punctual or last for a period of time. We also consider as events those predicates describing states or circumstances in which something obtains or holds true.

Events may be expressed by means of tensed or untensed verbs (2a-2b), nominalizations (2c), adjectives (2d), predicative clauses (2e), or prepositional phrases (2f):

(2) a. A fresh flow of lava, gas and debris **erupted** there Saturday.

b. Prime Minister Benjamin Netanyahu called the prime minister of the Netherlands **to thank** him for thousands of gas masks his country has already contributed.

c. Israel will ask the United States to delay a military **strike** against Iraq until the Jewish state is fully prepared for a possible Iraqi **attack**.

d. A Philippine volcano, **dormant** for six centuries, began exploding with searing gases, thick ash and deadly debris.

e. "There is no reason why we would not **be prepared**," Mordechai told the Yediot Ahronot daily.

f. All 75 people **on board** the Aeroflot Airbus died.

Furthermore, events can be of different kind, as illustrated below:

- **Reporting:** Events describing the action of a person or organization declaring or stating something, narrating an event, etc. Some examples are: *say, report, announce.*

- **Perception:** Events involving the physical perceptioon of another event. Typical predicates in this class are: *see, hear, watch, feel.*

- **Aspectual:** This class includes predicates which select an event as their argument and points to some structural aspect of this (for instance, its initiation, ongoing phase, or termination). Examples of predicates expressing aspectual events are: *begin, finish, stop, continue.*

- **I_Action:** The class `I_Action` compares with classes `Reporting`, `Perception`, `Aspectual`, and `I_State` (this last one described next) in that all five refer to events that select a second event as a complement. Nevertheless, classes `Reporting`, `Perception`, and `Aspectual` differ from `I_Action` due to their very specific semantics. And `I_Actions` and `I_States` can be distinguished by the fact that the former are dynamic events, whereas the second are stative. `I_Actions` are expressed by verbs or nouns like: *Attempt, try, promise, offer.*

- **I_State:** Like `I_Actions`, `I_State` events select an argument that expresses any sort of event. Unlike `I_Actions`, however, the I_State class is used for events which are states. Some examples: *Believe, intend, want.*

- **State:** States describe *circumstances* in which something obtains or holds true. The class `State` does **not** contain states that have been tagged as I_States. For example: *(be) on board, kidnapped, like.*

- **Occurrence:** This class includes all of the many other kinds of events that describe something that happens or occurs in the world. Some examples: *die, crash, build, merge, sell.*

2.2 Temporal expressions

Temporal expressions in TimeML are marked up with the tag `timex3`. These can refer to: dates, durations, sets, etc. There are three major types of TIMEX3 expressions:

Fully specified temporal expressions, which provide all the information necessary in order to identify the point or period of time they are referring to; e.g., *June 11, 1989*, or *the Summer of 2002*.

Underspecified temporal expressions, which require the use of some contextual information in order to interpret the point in time they are referring to; e.g., *early in the morning, Monday, in recent days, few days ago, two weeks from next Tuesday, next September, the current month, last year, a decade ago.*

Durations, such as: *three months* and *two years*.

Below are some sentences containing expressions that in TimeML are annotated as `TIMEX3` (in bold face):

(3) a. They are leaving on vacation **two weeks from next Tuesday**.

b. A major earthquake struck Los Angeles **three years ago today**.

c. **This year's summer** was unusually hot.

d. Bacon tutored an English student **some Thursdays in 1998**.

e. She spent the **following twelve years** in various health care positions around Minnesota.

2.3 Signals

The tag `SIGNAL` is used to annotate sections of text, typically function words, that indicate how temporal objects (events and temporal expressions) are to be related to each other. The material marked by `SIGNAL` constitutes several types of linguistic elements: indicators of temporal relations such as temporal prepositions (e.g., *on, during*) and other temporal connectives (e.g., *when*), and subordinators (e.g., *if*). For example:

(4) a. All passengers died **when** the plane crashed into the mountains.

b. They will investigate the role of the US **before**, **during** and **after** the genocide.

2.4 Links

One of the major innovations introduced in TimeML is the `LINK` tag. There are three types of `LINK` tags in TimeML, which encode the various relations that exist between the temporal elements of a document, namely, events and temporal expressions.

1. `TLINK:` a Temporal Link representing the temporal relationship holding between two events, or between an event and a time;
2. `SLINK:` a Subordination Link used for contexts introducing relations between two events;
3. `ALINK:` an Aspectual Link representing the relationship between an aspectual event and its argument event.

2.4.1 TLINKs

`TLINK` represents the temporal relationship holding between two events, or between an event and a time. It establishes a link between the involved entities, making explicit if they are:

- Simultaneous.
- Identical. That is, referring to the same event, as is the case with the two underlined event expreesions below.
  e.g., *John <u>drove</u> to Boston. During his <u>drive</u> he ate a donut.*

- One before the other.
  e.g., *John left before Mary arrived.*
- One after the other –corresponding to the reverse of the previous relation.
- One immediately before the other.
  e.g., *All passengers died when the plane crashed into the mountain.*
- One immediately after the other –reverse of the previous relation.
- One including the other. For example, the temporal reference *last Thursday* below includes the event referred to by *arrived*.
  e.g., *John arrived in Boston last Thursday.*
- One being included in the other –reverse of the previous relation.
- One holding during the duration of the other.
  e.g., *John and Mary talked during the whole ride.*
- One being the beginning of the other.
  e.g., *John has lived in Boston since 1998.*
- One being begun by the other –reverse of the previous relation.
- One being the ending of the other:
  e.g., *John stayed in Boston till 1999.*
- One being ended by the other –reverse of the previous relation.

2.4.2 SLINKs

`SLINK` (or Subordination Link) is used for contexts introducing relations between two events, of the following sort:

Factive: Certain predicates introduce an entailment (or presupposition) of the argument's veracity. They include *regret, manage*, *forget (that)*, etc., indicated in bold face below.

(5) a. John **forgot** that he was in Boston last year.
  b. Mary **regrets** that she didn't marry John.
  c. John **managed** to leave the party.

The `SLINK` expresses the relation between these predicates (in bold face above) and their event arguments (underlined).

Counterfactive: The event introduces a presupposition about the non-veracity of its argument. This type of `SLINK` is typically triggered by predicates such as: *forget (to), unable to* (in past tense),*prevent, cancel, avoid, decline*, etc.

(6) a. John **forgot** to buy some wine.
b. Mary was **unable** to marry John.
c. John **prevented** the divorce.

Evidential: Evidential relations are introduced by `REPORTING` or `PERCEPTION` predicates:

(7) a. John **said** he bought some wine.
b. Mary **saw** John carrying only beer.

Negative evidential: Introduced by `REPORTING` and some `PERCEPTION` events conveying negative polarity:

(8) John **denied** he bought only beer.

Modal: Relation generally triggered by events that introduce a reference to a possible world; these are mainly `I_STATEs` (see above), marked in bold face below.

(9) a. Mary **wanted** John to buy some wine.
b. John **promise** to buy some wine.

2.4.3 ALINKs

The `ALINK` (or Aspectual Link) represent the relationship between an aspectual event and its argument event. Examples of the possible aspectual relations that are encoded are shown below, where the aspectual expression is in bold face, and the embedded event underlined.

Initiation. For example, as in: *John* **started** *to read.*

Culmination. As in: *John* **finished** *assembling the table.*

Termination. As in: *John* **stopped** *talking.*

Continuation. For instance: *John* **kept** *talking.*

References

Pustejovsky, J., Knippen, B., Littman, J., & Saurí, R. (2005). Temporal and event information in natural language text. *Language Resources and Evaluation*, *39*(2), 123–164.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., & Pustejovsky, J. (2007). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, (pp. 75–80)., Prague, Czech Republic. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Moszkowicz, J., & Pustejovsky, J. (2009). The tempeval challenge: identifying temporal relations in text. *Language Resources and Evaluation*.