

MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY



Stanford Temporal Tagger: SUTime

PA164 – MACHINE LEARNING AND NATURAL LANGUAGE

Jakub Holotík
325140

Brno, 2012

Contents

1	Temporal Tagging	2
1.1	<i>Types of temporal expressions</i>	2
1.2	<i>TimeML</i>	2
2	SUTime	3
2.1	<i>The tagging process</i>	3
2.2	<i>Annotation and its limits</i>	3
2.3	<i>Other systems</i>	4
	Bibliography	5

1 Temporal Tagging

In the field of natural language processing temporal tagging is a process of marking up temporal expressions such as times, dates, durations etc. in documents. Temporal tagging identifies such temporal expressions, anchors them in time relative to the document in question and their context and marks them up using some markup language.

1.1 Types of temporal expressions

Time: A particular point in time. Time expressions can be absolute – such as *December 16, 20012*, or relative – such as *now* or *next Wednesday*. They can also be partially specified – e.g. *the twenties*.

Duration: The amount of time between the two points on the time scale. Duration is identified by a combination of quantity and a time unit e.g. *2 days, a few weeks, 6 to 8 years*.

Interval: A range of time defined by start and end time points such as *monday to friday*.

Combination: A combination of aforementioned temporal expressions e.g. *a weekend in December 2012*.

1.2 TimeML

One of the languages designed for temporal expression markup is TimeML. TimeML deals with several problems encountered in natural language processing and provides features like:

- Explicit temporal expression tagging
- Ordering events in time with respect to one another
- Setting underspecified temporal expressions in context (e.g. 'last week')

One of this markup language tags is `<TIMEX3>`. It is based on the specifications of TIMEX (2001) and TIMEX2 (2002) tags, but introduces new attributes and usage. [1]

2 SUTime

Stanford Temporal Tagger – SUTime is a Java library and a part of Stanford CoreNLP pipeline. It's main purpose is rule-based temporal tagging of English documents. It's approach is rule-based and relies on regular expression patterns over tokens. Using regular expressions over tokens instead of whole strings gives the tagger ability to recognize several natural language features, that are otherwise unavailable or hard to implement such as parts-of-speech recognition.

2.1 The tagging process

The tagger works in three phases. At first, it maps simple regular expressions over tokens to inner representation of temporal objects. Then it iteratively applies several compositional rules combining simple temporal objects into more complex ones. If such option is selected when running the tagger, nested time expressions are removed and new temporal objects are created. In the final stage, ambiguous expressions are removed so as to prevent the tagger from false recognition. All temporal objects are placed in time with respect to their context, relative times are transformed to absolute time points and `<TIMEX3>` tags are produced. The final annotated text is outputted or passed to the next tool in the pipeline. [2]

2.2 Annotation and its limits

The SUTime makes advantage of TimeML markup language mentioned before, specifically an extension of the `<TIMEX3>` tag to annotate temporal expressions in English text. It supports most of the types of temporal expressions like time, duration, interval and the combination thereof, although it has some limitations:

- Ambiguity of named entities - e.g. *'fall'* can serve as a time expression indicating a season as well as a form of the verb *to fall*. Such ambiguity is not handled by SUTime and would have to be resolved using probabilistic methods or a named entity recognizer.

- Resolution of relative expressions is not fully supported. See for example an instance of *Wednesday* – it is unclear whether the reference is made to the Wednesday before or after the reference date of the document. This is a language specific problem and a tight cooperation of several tools would have to be used in order to resolve such unclarity.
- The problem of recognizing temporal ranges is also a language specific problem and as such is poorly supported by SUTime.

2.3 Other systems

There are several other time expression tagging systems available:

- GUTime - A temporal tagger provided by Georgetown University as a Perl application.
- HeidelTime - Another rule-based temporal tagging system.
- TRIPS/TRIOS - A system using a combination of rule-based and conditional random field approaches.

Bibliography

- [1] *TimeML 1.2.1* [online]. TimeML Working Group, October 2005 [cit. 16.12.2012].
Available at:
<http://timeml.org/site/publications/timeMLdocs/timeml_1.2.1.html>

- [2] *Stanford Temporal Tagger: SUTime* [online]. The Stanford Natural Language Processing Group, 9.7.2012 [cit. 16.12.2012].
Available at:
<<http://nlp.stanford.edu/software/sutime.shtml>>