Temporal expression normalisation in natural language texts

Michele Filannino
School of Computer Science
The University of Manchester, UK
filannim@cs.man.ac.uk

Abstract

Automatic annotation of temporal expressions is a research challenge of great interest in the field of information extraction. In this report, I describe a novel rule-based architecture, built on top of a pre-existing system, which is able to normalise temporal expressions detected in English texts. Gold standard temporally-annotated resources are limited in size and this makes research difficult. The proposed system outperforms the state-of-the-art systems with respect to TempEval-2 Shared Task (VALUE attribute) and achieves substantially better results with respect to the pre-existing system on top of which it has been developed. I will also introduce a new free corpus consisting of 2822 unique annotated temporal expressions. Both the corpus and the system are freely available on-line[1].

Keywords: information extraction, temporal expression, text mining, natural language processing

1 Introduction

In many domains, the possibility of using and interpreting temporal aspects and events is important in order to organise information. Temporal knowledge allows people to filter information and even infer temporal flows of events. Furthermore, it permits an improving of intelligence for question answering, information retrieval and information filtering systems.

A temporal expression [7], also called *timex*, refers to every natural language phrase that denotes a temporal entity such as an interval or an instant. For example, in a sentence like *"Italian prime minister Mario Monti said yesterday that the reform has been very successful."* the phrase *"yesterday"* is actually a temporal expression. Timexes elicit a binding between the natural language domain and the time domain because it is always possible to represent such expressions as a time point, interval or set using ISO 8601 standard[2]. Temporal expressions could be of three different types [1]: *fully-qualified*, *deictic* and *anaphoric*.

Fully-qualified A temporal expression is fully-qualified with respect to the binding when all the information required to infer a point in the time domain are fully included inside the expression. In this category the following expressions falls: *March 15 2001*, *21st July 1985* or *31/04/2011*. Fully-qualified expressions are the easiest to detect because of their rigid lexical form.

Deictic In this case, inferring the binding with the time domain necessarily requires to take into account the time of utterance (when the document has been written or when the speech has been given). Deictic expressions could not be properly associated to a precise time without that information. Typical deictic temporal expressions are: *today*, *yesterday*, *last Sunday* and *two months ago*.

Anaphoric These expressions can be mapped to a precise point in the time domain only taking into account temporal expressions previously mentioned in the text or during the speech. Examples of this category are: *March 15*, *the next week*, *Saturday*. The only difference between deictic and anaphoric expressions is the location of the temporal reference: for deictic expressions it is the time of utterance or publication, for anaphoric expressions it is a time previously evoked in the text or speech. Anaphoric expressions constitute a future challenge for the scientific research in this field.

For the sake of completeness, another kind of categorisation [13] is also adopted in the field. It identifies

[1] http://www.cs.man.ac.uk/~filannim/

[2] http://www.w3.org/TR/NOTE-datetime

the possible shapes of timexes with respect to their semantics and admits the following types: *time or date references*, *time references that anchor on another time*, *durations*, *recurring times*, *context-dependent times*, *vague references* and *times indicated by an event*.

In my taster project I focussed on the normalisation of fully-qualified and deictic temporal expressions. I will not use the last categorisation because of the fuzziness of boundaries among types.

2 Background

The idea of annotating temporal expressions automatically from texts appeared for the first time in 1998 [3]. This topic aroused an increasing interest with the proposal of a proper temporal annotation scheme [12]. The original aim was to make the annotation phase easier with respect to the previous scheme in order to collect annotated data and use the temporal information to enhance performances of question answering systems [12]. All the most recent systems [19, 20] proposed for the temporal expressions extraction task go through two different steps: identification and normalisation. This dichotomy has become universally accepted by the research community because it makes the extraction phase easier to approach [1].

In the identification phase the effort is concentrated on how to detect properly the sub-expressions that are real temporal expressions in natural language texts. This step is usually done by using machine learning techniques. Ahn et al. [1] firstly used Conditional Random Fields [10] showing better performances with respect to a previous work [2] in which they used Support Vector Machines [4]. Poveda et al. [13] introduced a sophisticated Bootstrapping technique enhancing the recognition of temporal expressions while Mani et al. [12] used rules learned by a decision tree classifier, C4.5 [15], and Ling and Weld [11] tried Markov Logic Network [6].

The second step is the normalisation. In this phase the main goal is to interpret the expression, extract the temporal information and represent it in a proper pre-defined format. The universally accepted standard for temporal expressions annotation is TimeML [14]. It provides a specification for the representation of temporal expressions and also events. In this work the normalisation task aims at producing the proper TimeML code that correctly represents the temporal information (see Figure 1). This step is usually accomplished using rule-based approaches. Grover et al. [9] used a rule-based approach on top of a pre-existing information extraction system, whereas Strötgen and Gertz [16] produces a small set of hand-crafted rules with an ad-hoc selection algorithm. UzZaman and Allen [17] produced a rule-based normaliser focussing just on TYPE and VALUE attributes of TIMEX3 tag (the one used to represent timexes).

3 Method

The contribution of my short taster project is twofold. Firstly, I will illustrate a temporal expression corpus explicitly designed for the normalisation phase. Then I will describe the software architecture of a new normaliser built on top of a pre-existing one.

3.1 Temporal expressions corpus

Gold-standard temporally-annotated resources are very limited in general domain [5], and even less in specific ones like medical, clinical and biological [8]. In the last decade, different sources of annotated temporal expressions have been developed. Because of the rapid evolution of this research field, usually the sources differ even with respect to the annotation guidelines. This leads to the existence of different corpora not entirely compatible to each other.

The main difference among them consists in the tag used to annotate temporal expressions: TIMEX2 against TIMEX3. These two tags reflect totally different way of annotating the same temporal expressions leading to the impossibility of using both corpora at the same time.

I created a corpus of temporal expressions collecting all TIMEX3 tags in four different corpora: AQUAINT[3], TimeBank 1.2[4], WikiWars[5] and TRIOS TimeBank v0.1[6]. I extracted from each document all the possible temporal expressions and for each one I also saved the related document creation time, the type (*DATE*, *TIME*, *SET* or *DURATION*) and the normalisation provided by the human annotators. Then I compacted the corpus removing possible duplicates. With the expression *duplicates* I refer to completely identical tuples, i.e. same text, same normalisation, same utterance time and same type.

I obtained a corpus of 2822 unique annotated temporal expressions. The Table 2 shows an excerpt of the corpus. Further information about the distribution of temporal expression types in it is provided in Table 1.

The corpus is freely available [7] in CSV format using a tabulation character as delimiter.

[3] http://www.ldc.upenn.edu/Catalog/docs/LDC2002T31/

[4] http://www.timexportal.info/corpora-timebank12

[5] http://www.timexportal.info/wikiwars

[6] http://www.cs.rochester.edu/u/naushad/trios-timebank-corpus

[7] http://www.cs.man.ac.uk/~filannim/timex3s_corpus.csv

```
<?xml version="1.0" ?>
<TimeML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:noNamespaceSchemaLocation="http://timeml.org/timeMLdocs/TimeML_1.2.1.xsd">
    <DOCID>Example_document</DOCID>
    <DCT>2012, Manchester, Apr 17, 2012</DCT>
    <TITLE>Example document</TITLE>
    <TEXT>
        Italian prime minister Mario Monti
        <EVENT eid="e1" class="OCCURRENCE">said</EVENT>
        <MAKEINSTANCE eiid="ei1" eventID="e1" pos="VERB" tense="PAST" aspect="NONE" />
        <TIMEX3 tid="t1" type="DATE" value="2012-04-16">yesterday</TIMEX3>
        that the reform has been very successful.
        <TLINK eventInstanceID="ei1" relatedToTime="t1" relType="DURING" />
    </TEXT>
</TimeML>
```

Figure 1: Example of TimeML code. In the sentence there is a deictic temporal expression; "yesterday" can be correctly annotated only taking into account the document creation time (DCT).

Temporal expression	**Type**	**Value**	**Utterance**
...			
more than two years	DURATION	P2Y	20110926
much of 2010	DATE	FUTURE_REF	20110926
nearly a month	DATE	P1M	20110926
nearly an hour	DURATION	PT1H	19910225
nearly forty years	DURATION	P40Y	1919980120
nearly four years ago	DATE	1994	19980227:081300
nearly three years	DURATION	P3Y	19891030
nearly two months	DURATION	P2M	19980306:131900
nearly two months afterwards	DATE	FUTURE_REF	20110926
nearly two weeks ago	DATE	1989-WXX	19891030
nearly two years	DURATION	P2Y	19980301:141100
next day	DATE	2011-09-27	20110926
...			

Table 2: Brief excerpt of the corpus.

Timex type	Frequency
DATE	2307
DURATION	416
TIME	71
SET	28
TOTAL	2822

Table 1: Distribution of TIMEX3 tags in the corpus.

3.2 Temporal expressions normaliser

I built a new normaliser on top of the one freely available from University of Rochester[8]: TRIOS. It is a rule-based normaliser and it has been proved to provide the second best performance in TempEval-2 Shared Task [17]. All the rules are in the form of regular expressions in a switch architecture: the activation of one of them excludes the activation of all the others.

I introduced a top layer with three new kinds of rules: extension, manipulation and post-manipulation rules.

The extension rules are just new rules that cover non-expected cases and are checked immediately before the pre-existing rules. If a temporal expression do not activate any of the extension rule, it goes into TRIOS. For example, some of these rules are used to normalise expressions of festivities dates such as *"Thanksgiving day"* or *"Saint Patrick's day"*.

The manipulation rules have been introduced to turn particular well-known expressions into an easier form before TRIOS processes them. Once one of these rules is activated, the original temporal expression is transformed into a reduced one that is easier to normalise properly for the pre-existing set of rules. After the transformation, the new temporal expression is taken in input by TRIOS for the normalisation task.

Lastly, I used the post-manipulation rules to solve some deficiencies in the normaliser by adding further information lost by TRIOS and finally improving the performance. In this case the temporal expression is evaluated through the extension rules or the original set. At the end of the normalisation process the result is enriched with further information. For example, I used these rules to add information about seasons which are not considered in TRIOS at all.

In the end, I introduced 32 new regular expression patterns: 16 extension rules, 12 manipulation rules and 4 post-manipulation rules. The entire system is freely available online[9] under GNU licence[10].

[8] http://www.cs.rochester.edu/u/naushad/temporal

[9] http://www.cs.man.ac.uk/~filannim/timex_normaliser.zip

[10] http://www.gnu.org/licenses/gpl.html

4 Evaluation

I evaluated the normalisation system using the new corpus previously described as a training set and then I measured the performances with respect to the TempEval-2 Shared Task test set. This offered me the possibility of comparing my normaliser with all the others evaluated in that challenge.

In order to measure the difference between TRIOS and my extension I also tested both of them by using the new corpus. It is important to notice that TRIOS has been trained on the same data provided in the new corpus. For this reason a comparison between these systems is legitimate.

In both cases, the evaluation procedure is based on counting. Because the normalisation task is aimed at providing the right TYPE attribute and the right VALUE attribute, the evaluation is carried out by counting how many times the system provides the same value with respect to the human ones. It is important to emphasise that every value provided by the system that differs form the human one for at least one character is considered error.

If this method is quite reasonable for TYPE attribute, it might be too restrictive for VALUE attribute. Some practical examples could be of help to explain the problem.

- The human annotation of a certain timex is {*type: "DATE", value: "FUTURE_REF"*} whereas the system provides a the more specific annotation {*type: "DATE", value: "2013-09-XX"*}.

- The system provides an annotation that is less specific than that provided by humans. For example, it happens when the human-annotation is {*type: "DATE", value: "2011-04-18"*} and the system provide {*type: "DATE", value: "2011-04-XX"*}.

In all these cases the annotations are considered completely wrong. Even when the system provides a partially wrong annotation, e.g. {*type: "DATE", value: "2011-04-23"*} for a human annotation of {*type: "DATE", value: "2011-04-18"*}, considering it a complete wrong result may be too strict because year and month are correct however. This fact has justified the investigation of other measurement metrics [18].

4.1 Results

The normalisation results with respect to TempEval-2 Shared Task are shown in Table 3. The new TRIOS

	type	value
Edinburgh	0.84	0.63
HeidelTime	0.96	0.85
KUL	0.91	0.55
TERSEO	**0.98**	0.65
TipSem	0.92	0.65
TRIOS	0.94	0.76
TRIOS extension	0.95	**0.86**

Table 3: Results obtained from TempEval-2 test set.

	type	value
TRIOS	0.8572	0.6257
TRIOS extension	**0.8853**	**0.7170**

Table 4: Results obtained from the corpus.

extension outperforms each system in the normalisation of VALUE attributes and performs competitively in the normalisation of TYPE attributes.

The table already shows that the normalisation of value attributes is slightly harder than that of type attributes. The extension of TRIOS outperformed the original system of 2.81% for TYPE attribute and 9.13% for VALUE attribute.

I randomly sub-sampled (400 temporal expressions) the original corpus 10 times and I measured the performances with TRIOS and my extension. I conducted a statistical analysis on the results and I proved that the difference is statistically significant (Willcoxon test), respectively $\rho = 0.00586$ and $\rho = 0.0001621$.

The normalisation results with respect to the new corpus are shown in Table 4.

4.2 Error analysis

The original TRIOS normaliser made 1023 value mistakes and 402 type mistakes while its extension respectively made 779 and 323. Through an accurate analysis of the errors, I found plenty of human annotations that seemed to be wrong at first impression. Once I analysed the same annotations taking into account the entire sentence from which each expression had been extracted, I found that the human annotations were actually right. Some examples are shown in Table 5.

This leads to the conclusion that further improvements are possible only if I consider also the resolution of anaphoric expressions. To do this, it will be necessary to consider a wider window for each temporal expression that takes into account at least the entire sentence in which each temporal expression is located.

5 Conclusions

I introduced a new rule-based normaliser of temporal expressions and I showed that it resulted in better performances than the current state-of-the-art system with respect to TempEval-2 Shared Task. I also illustrated the corpus of temporal expressions for normalisation and its purpose. I made both, the normaliser and the corpus, freely available on-line (GNU public licence apply).

5.1 Future work

The work presented in this report is the product of a preliminary study in the field of information extraction. The results presented in this report clearly show the necessity of coping with anaphoric temporal expression to substantially enhance the performances of normalisation phase. Currently, the normalisation task takes into account only the temporal expressions, without considering a wider window, such as the entire sentence or a pre-defined number of words after and before the expression. This is required in order to cope with anaphoric expressions.

My long-term goal is to develop novel temporal expressions extraction techniques and use them in clinical domain. Because of the lack of pre-annotated clinical data, I will explore the use of semi-supervised machine learning approaches for the identification phase.

5.2 Acknowledgements

I would like to thank Naushad UzZaman from the University of Rochester to have shared his normaliser with the scientific community. I would also like to acknowledge the support of UK Engineering and Physical Science Research Council in the form of doctoral training grant.

References

[1] D. Ahn, S. F. Adafre, and M. de Rijke. Towards task-based temporal extraction and recognition. In G. Katz, J. Pustejovsky, and F. Schilder, editors, *Annotating, Extracting and Reasoning about Time and Events*, number 05151 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2005. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

	human	**system**
25	1999-04-25	n/a
last year	1988-Q2	1988
three years before	FUTURE_REF	PAST_REF
the summer of 1862	FUTURE_REF	1862-SU
the weekend	P2D	PRESENT_REF

Table 5: Some errors made by the normaliser.

[2] D. Ahn, J. van Rantwijk, and M. de Rijke. A cascaded machine learning approach to interpreting temporal expressions. In *HLT-NAACL*, pages 420–427, 2007.

[3] G. Becher, F. Clerin-Debart, and P. Enjalbert. A model for time granularity in natural language. In *Proceedings of the Fifth International Workshop on Temporal Representation and Reasoning*, pages 29–, Washington, DC, USA, 1998. IEEE Computer Society.

[4] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Computational Learning Theory*, pages 144–152, 1992.

[5] L. Derczynski, H. Llorens, and E. Saquete. Massively Increasing TIMEX3 Resources: A Transduction Approach. *ArXiv e-prints*, Mar. 2012.

[6] P. Domingos. Real-world learning with markov logic networks. In *PKDD*, page 17, 2004.

[7] L. Ferro, I. Mani, B. Sundheim, and G. Wilson. TIDES Temporal Annotation Guidelines - Version 1.0.2. Technical report, The MITRE Corporation, McLean, Virginia, June 2001.

[8] L. Galescu and N. Blaylock. A corpus of clinical narratives annotated with temporal information. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, IHI '12, pages 715–720, New York, NY, USA, 2012. ACM.

[9] C. Grover, R. Tobin, B. Alex, and K. Byrne. Edinburgh-ltg: Tempeval-2 system description. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 333–336, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[10] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289, 2001.

[11] X. Ling and D. S. Weld. Temporal information extraction. In *Proceedings of the AAAI 2010 Conference*, pages 1385 – 1390, Atlanta, Georgia, USA, July 11-15 2010. Association for the Advancement of Artificial Intelligence.

[12] I. Mani and G. Wilson. Temporal granularity and temporal tagging of text. In *Proceedings of the AAAI-2000 Workshop on Spatial and Temporal Granularity*, 2000.

[13] J. Poveda, M. Surdeanu, and J. Turmo. An analysis of bootstrapping for the recognition of temporal expressions. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*, SemiSupLearn '09, pages 49–57, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[14] J. Pustejovsky, J. Castaño, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer, and G. Katz. Timeml: Robust specification of event and temporal expressions in text. In *in Fifth International Workshop on Computational Semantics (IWCS-5*, 2003.

[15] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[16] J. Strötgen and M. Gertz. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 321–324, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[17] N. UzZaman and J. F. Allen. Event and temporal expression extraction from raw text: First step towards a temporally aware system. *Int. J. Semantic Computing*, 4(4):487–508, 2010.

[18] N. UzZaman and J. F. Allen. Temporal evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:*

Human Language Technologies: short papers - Volume 2, HLT '11, pages 351–356, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[19] M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, G. Katz, and J. Pustejovsky. Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80, Prague, 2007.

[20] M. Verhagen, R. Saurí, T. Caselli, and J. Pustejovsky. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 57–62, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.