10.12871/clicit201427

HeidelTime at EVENTI: Tuning Italian Resources and Addressing TimeML's Empty Tags

Giulio Manfredi and **Jannik Strötgen** and **Julian Zell** and **Michael Gertz**
Institute of Computer Science, Heidelberg University, 69120 Heidelberg, Germany
manfredi@stud.uni-heidelberg.de,
{stroetgen,zell,gertz}@informatik.uni-heidelberg.de

Abstract

English. In this paper, we describe our participation in the EVENTI task. We addressed subtask A, the extraction and normalization of temporal expressions in Italian texts, by adapting our existing multilingual temporal tagger HeidelTime. In addition to improving its ability to handle Italian texts, we added further functionality to support empty tags. Based on the main evaluation criterion, HeidelTime ranked first among the participating systems. The new HeidelTime version is publicly available.[1]

Italiano. *In questo articolo descriviamo la nostra partecipazione al task EVENTI. Ci siamo dedicati al sottotask A, cioè l'estrazione e normalizzazione di espressioni temporali all'interno di testi in lingua italiana, e a questo scopo abbiamo adattato il nostro temporal tagger multilingue, HeidelTime. Oltre a migliorare le sue capacità di elaborare testi in italiano, abbiamo aggiunto nuove funzionalità per supportare i tag vuoti. In base al principale criterio di valutazione, HeidelTime è risultato primo rispetto agli altri sistemi che hanno partecipato al task. La nuova versione di HeidelTime è disponibile pubblicamente.*[1]

1 Introduction

EVENTI (EValuation of Events aNd Temporal Information) is a task of EVALITA 2014, an initiative aimed at the evaluation of NLP tools for Italian.[2] It comprises four subtasks: the extraction and normalization of temporal expressions, i.e., temporal tagging (A), the extraction of events (B), and the annotation of temporal relations (C, D).

Together, they form the task of temporal annotation, which is helpful in many natural language processing and understanding applications such as question answering and summarization. But even the temporal tagging subtask itself is valuable for many applications, e.g., in information retrieval (Alonso et al., 2011; Campos et al., 2014).

In this paper, we describe our efforts to address the temporal tagging subtask of EVENTI, for which we extended and improved our temporal tagger HeidelTime (Strötgen and Gertz, 2013). In addition to earlier approaches to Italian temporal tagging (e.g., Negri 2007) and to manually annotated Italian corpora (Magnini et al., 2006), Italian was also one of six languages offered at TempEval-2 (Verhagen et al., 2010). However, participants only addressed English and Spanish, and we also added Italian to HeidelTime only more recently (Strötgen et al., 2014). While Italian had thus already been implemented in HeidelTime, there was room for improvement in the context of the EVENTI challenge as will be detailed in this paper. As reference point for our work, the EVENTI task guidelines (Caselli et al., 2014) and the Ita-TimeBank corpus (Caselli et al., 2011) – newly released as training data – were used.

The rest of the paper is structured as follows. After an overview of HeidelTime's architecture and challenges that needed to be addressed, our adaptations to HeidelTime are detailed in Section 3. In Section 4, evaluation results are reported and compared to those of HeidelTime's previous version and the systems of the other participants.

2 Starting Point & Challenges

In this section, we first describe HeidelTime's architecture and then explain the challenges that had to be addressed although HeidelTime already supported Italian temporal tagging.

[1] http://code.google.com/p/heideltime/

[2] http://www.evalita.it/2014

2.1 HeidelTime's Architecture

HeidelTime is a rule-based, multilingual, and cross-domain temporal tagger initially developed for English in the context of TempEval-2 (Strötgen and Gertz, 2010). It is based on the Unstructured Information Management Architecture[3] (UIMA), which allows to easily combine different modules because all rely on the same data structure, called *Common Analysis Structure* (CAS).

In a UIMA pipeline for temporal tagging with HeidelTime, input documents are first read by a *collection reader*, which initializes a CAS object for each document. The subsequent tasks are sentence splitting, tokenization, and part-of-speech tagging before HeidelTime itself is called. The TreeTagger for Italian linguistic preprocessing (Schmid, 1994), and HeidelTime are employed as *analysis engines*. Eventually, the output is created by a *CAS consumer*, which writes the text and its annotations to a database or file.

An important characteristic of HeidelTime's architecture is the strict separation of source code and language dependent resources. This allows adding new languages and improving already implemented ones without affecting the functionality of the system itself and without requiring a deep knowledge of its mechanisms. Several languages were thus integrated by different research groups: German (Strötgen and Gertz, 2011), Dutch (van de Camp and Christiansen, 2012), Spanish (Strötgen et al., 2013), French (Moriceau and Tannier, 2014), Italian, Arabic, Vietnamese (Strötgen et al., 2014), Chinese (Li et al., 2014), Russian, and Croatian (Skukan et al., 2014).

HeidelTime's language resources are of three types: patterns, normalizations, and rules. There is one rule file for each possible value of the TIMEX3 *type* attribute (`DATE`, `TIME`, `DURATION` and `SET`), and each rule has three mandatory fields: `RULENAME`, `EXTRACTION` and `NORM_VALUE`. The `EXTRACTION` field is a regular expression that also contains references to the patterns, which are themselves sets of regular expressions. The field `NORM_VALUE` uses the normalization resources to translate the patterns into a standard format and to normalize extracted temporal expressions according to the TimeML specifications (Pustejovsky et al., 2003).[4]

2.2 Challenges for EVENTI Participation

HeidelTime's initial resources for Italian were developed on the Italian TempEval-2 training data (Strötgen et al., 2014), although the TempEval-2 corpus developers stated that the non-English "annotations are a bit experimental" (Verhagen, 2011). Thus, using now more sophisticated guidelines and training data, several adaptations were required. With regard to language-dependent resources, most work consisted of extending patterns, adding rules, and improving existing ones.

Furthermore, a main challenge was that in the EVENTI data, empty TIMEX3 tags – which represent implicit temporal information – are considered. Although such empty tags are also defined in the original TimeML annotation specifications,[5] they have hardly been considered so far, neither in manually annotated corpora nor in research competitions nor by temporal taggers. They were also not created by HeidelTime so far, and were thus a feature that needed to be implemented.

Finally, the particular format of the EVENTI training and test data required specific tools to read the documents and output HeidelTime's annotations in the required format as described below.

3 HeidelTime Adaptations

Our efforts can be split into three parts: developing UIMA components, extending HeidelTime, and improving HeidelTime's Italian resources.

3.1 UIMA Components for EVENTI Data

The EVENTI training and test data consist of Ita-TimeBank documents (news articles). Each document is provided as an XML file containing sentence and token annotations. In the training data, TIMEX3 tags are additionally provided.

To handle this format at the input and output stages, we wrote a collection reader and a CAS consumer. These are also part of the new HeidelTime-kit, which allows to easily reproduce our evaluation results on the EVENTI data.

3.2 Empty TIMEX3 Tags

The main feature we needed to add, though, was the creation of empty tags. These are part of the It-TimeML specifications but were not present in previous temporal tagging corpora and competitions. Empty tags are TIMEX3 tags that do not

[3] `http://uima.apache.org/`

[4] For further details about HeidelTime's rule syntax, we refer to (Strötgen and Gertz, 2013).

[5] `http://timeml.org/site/publications/timeMLdocs/timeml_1.2.1.html`.

contain any tokens and should be created whenever a temporal expression can be inferred from already existing text-consuming TIMEX3 tags. Two cases are implicit begin and end points of temporal expressions of type `DURATION`, e.g., *un mese fa* (a month ago), and implicit durations which can be deduced from two TIMEX3 tags of type `DATE`, e.g., *dal 2010 al 2014* (from 2010 to 2014). We refer to the former as *anchored durations* and to the latter as *range expressions*.[6]

To handle anchored durations, we modified HeidelTime's rule syntax by adding an additional field, called `EMPTY_VALUE`. It is syntactically similar to `NORM_VALUE` and contains an offset to a reference time. This offset, combined with the value returned by `NORM_VALUE`, is then used by HeidelTime to compute a normalized date. Note that this `EMPTY_VALUE` extension is language-independent and had to be realized by modifying HeidelTime's source code.

To extract range expressions, the UIMA HeidelTime kit already contained an analysis engine called Interval Tagger, which creates TIMEX3 independent temporal annotations. So far, however, only English interval rules were available, and not TIMEX3 duration values but start and end time points of range expressions were determined. In addition to writing Italian rules, we thus added the ability to calculate the difference between the two `DATE` expressions, i.e., duration values for range expressions, as defined in the specifications.

In both cases, the computed values are included as additional, HeidelTime-internal attributes to text-consuming TIMEX3 annotations. Our EVENTI CAS consumer reads out these attributes to print empty TIMEX3 tags with the respective *value* information. Furthermore, it adds to each empty tag a reference to the TIMEX3 tag(s) that triggered it.

3.3 Tuning Italian Resources

Despite the efforts required to implement the empty tag feature, most time was spent on extending the existing Italian resources. This was done by carefully applying the guidelines provided by the EVENTI task organizers. While modifying normalization information of existing patterns was rather simple, quite a lot of work was needed to improve the performance in the extraction phase.

Since HeidelTime is a rule-based system that makes use of regular expressions, new patterns were added to extract expressions which had not been considered before and, as a consequence, to improve the recall of the system. While doing this, we tried to write the rules as general as possible without producing many false positives. In Italian, however, there are several expressions that can be ambiguous and therefore require context knowledge to be correctly interpreted. Obviously, this is somewhat limited by the abilities of a rule-based system and thus particularly challenging.

An example is the adverb *allora*, which, depending on the context, can mean "at that time" or "therefore". Our system only identifies the temporal meaning if it can be inferred from neighboring words, as in *già allora* (already at that time).

Some of the patterns that were added are those representing sets of months or years, e.g., *bimestre* (two months) and *lustro* (five years), and specific post-modifiers that affect the normalization of an expression, e.g., *esaminato*, *in discussione* and *di che trattasi*, all referring to the period of time that is being dealt with.

4 EVENTI Evaluation

The extraction quality of all participating systems and of all runs of each system is evaluated using precision, recall, and F1-score for strict and relaxed matches. To evaluate normalization abilities, the accuracy of the *type* and *value* attributes are multiplied by the F1-score for strict matches in order to normalize it. The resulting *value F1* measure is used as main evaluation criterion.

Table 1 shows official results of all participating teams. We submitted three runs: HeidelTime 1.7 (publicly available before EVENTI), HeidelTime 1.8 (comprising all adaptations described above), and version 1.8 without the empty tags feature. With regard to this aspect, the measures show only small differences, mainly because empty tags are rare compared to other tags. Although precision is slightly higher when ignoring empty tags, recall, F1-score, and normalization quality increase significantly when taking them into account.

Most important, however, is the massive improvement of HeidelTime 1.8 over 1.7 with respect to extraction and normalization quality.

The extraction quality of the system of team B is similar to HeidelTime 1.8. Its F1-score is slightly higher for strict but lower for relaxed matches.

[6] A third empty tag type is described as further challenge in Section 4 since we have not yet addressed it.

	relaxed match			strict match			normalization	
	P	R	F1	P	R	F1	type F1	**value F1**
HT 1.7	92.1	67.6	78.0	78.2	57.4	66.2	64.3	57.1
HT 1.8 (no ET)	94.0	82.4	87.8	86.1	75.5	80.4	77.5	69.0
HT 1.8	93.5	85.4	*89.3*	86.0	78.5	82.1	79.2	**70.9**
Team B-1	93.6	84.1	88.6	87.3	78.5	*82.7*	80.0	66.5
Team C-1	92.9	65.4	76.8	80.2	56.4	66.2	64.3	56.6
Team C-2	92.2	66.2	77.1	78.8	56.6	65.9	64.0	56.3

Table 1: EVENTI evaluation results on test data.

With respect to the normalization quality, HeidelTime outperforms team B by 4.4 and team C by 14.3 percentage points (value F1).

Finally, comparing HeidelTime's performance on the test set and the FBK and ILC training sets reveals some differences. While value F1 is only slightly higher on the FBK set (73.5), it is much higher on the ILC set (84.2) – mainly due to many rather difficult expressions in the FBK set.

4.1 Error Analysis

In general, four error types can be distinguished: false positives, false negatives, partial matches, and incorrect normalizations. Although the main evaluation criterion combines correct value normalization with strict matching, in our opinion, value F1 with relaxed matching is even more meaningful (HeidelTime 1.8: 74.7). Expressions that are only partial matches but correctly normalized are often equally valuable as correctly normalized strict matches for any NLP or IR tasks relying on temporal taggers.

Considering relaxed matching, only 37 false positives are extracted by HeidelTime, and of 624 gold expressions, 533 are retrieved with either strict or relaxed matching. 446 of them are additionally normalized correctly.

Simple examples of partial matches with correct value normalization are expressions such as *un lasso di tempo di 14 giorni* (a lapse of time of 14 days), where HeidelTime extracts only *14 giorni*, but the normalization is correct.

A further issue occurs if two tags are created instead of one. Instead of *ieri verso le 11* (yesterday around 11), HeidelTime extracts *ieri* and *verso le 11* separately. Nonetheless, the value of *verso le 11* is the same as the gold annotation. Considering strict matching, such mistakes generate two false positives and one false negative.

A reason for incorrect normalizations is that several `DATE` expressions have a value of `XXXX-XX-XX` in the gold standard. HeidelTime, however, tries to resolve extracted `DATE` expressions instead of leaving them unspecified. Another reason is the occurrence of `TIME` values that contain a time without date in the gold standard. However, it is often difficult to decide if a `TIME` expression refers to a specific day or if it is used generically. HeidelTime usually assigns values to `TIME` expressions with specified day information. Furthermore, its strategy to select the previously mentioned day as reference day is sometimes incorrect. Often, however, this strategy works fine as in the example above where *ieri* is selected as reference time for the expression *verso le 11*.

4.2 Open Challenges

What needs to be addressed in the future is a third category of expressions that generate empty tags, namely *framed durations*. These are durations located in a specific time frame and for which a begin and an end point can be inferred. An example is *i primi 6 mesi dell'anno* (the first 6 months of the year), where, in addition to a `DURATION` (*i primi 6 mesi*) and a `DATE` (*anno*), two additional `DATE` expressions can be deduced, referring to the first and sixth month of the year in question. Thus, two empty tags with values pointing to January and June of the respective year should be created.

A further example of an ambiguity issue in addition to the ones described in Section 3.3, are expressions referring to ages which are often ambiguous in Italian. For instance, the Italian expression *26 anni* can mean "26 year old" or "26 years" – but only in the latter case it should be annotated.

Finally, the creation of empty tags has been developed specifically for the EVENTI task, so that it is currently only available for Italian. However, the expansion to the other languages supported by HeidelTime should not be time consuming because it merely requires an adaptation of the rules.

5 Summary

In this paper, we described our participation in the temporal tagging task of EVENTI 2014. By extending HeidelTime to cover TimeML's empty TIMEX3 tags and by tuning HeidelTime's Italian resources based on high quality specifications and training data, we significantly improved HeidelTime's tagging quality for Italian. We outperformed the other participants' systems by at least 4.4 percentage points for correct extraction and normalization (value F1).

References

Omar Alonso, Jannik Strötgen, Ricardo Baeza-Yates, and Michael Gertz. 2011. Temporal Information Retrieval: Challenges and Opportunities. In *Proceedings of the 1st International Temporal Web Analytics Workshop (TWAW 2011)*, pages 1–8.

Ricardo Campos, Gaël Dias, Alípio M. Jorge, and Adam Jatowt. 2014. Survey of Temporal Information Retrieval and Related Applications. *ACM Computing Surveys*, 47(2):15:1–15:41.

Tommaso Caselli, Valentina Bartalesi Lenzi, Rachele Sprugnoli, Emanuele Pianta, and Irina Prodanof. 2011. Annotating Events, Temporal Expressions and Relations in Italian: the It-TimeML Experience for the Ita-TimeBank. In *Proceedings of the 5th Linguistic Annotation Workshop (LAW 2011)*, pages 143–151.

Tomasso Caselli, Rachele Sprugnoli, Manuela Speranza, and Monica Monachini. 2014. EVENTI: EValuation of Events aNd Temporal Information Task Guidelines for Participants v 1.0. Technical report, TrentoRISE, FBK, University of Trento, and ILC-CNR.

Hui Li, Jannik Strötgen, Julian Zell, and Michael Gertz. 2014. Chinese Temporal Tagging with HeidelTime. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 133–137.

Bernardo Magnini, Emanuele Pianta, Christian Girardi, Matteo Negri, Lorenza Romano, Manuela Speranza, Valentina Bartalesi Lenzi, and Rachele Sprugnoli. 2006. I-CAB: the Italian Content Annotation Bank. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 963–968.

Véronique Moriceau and Xavier Tannier. 2014. French Resources for Extraction and Normalization of Temporal Expressions with HeidelTime. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 3239–3243.

Matteo Negri. 2007. Dealing with Italian Temporal Expressions: The ITA-CHRONOS System. In *Proceedings of Evalita 2007*.

James Pustejovsky, José M. Castaño, Robert Ingria, Roser Sauri, Robert J. Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R. Radev. 2003. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *New Directions in Question Answering*, pages 28–34.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Luka Skukan, Goran Glavaš, and Jan Šnajder. 2014. HEIDELTIME.HR: Extracting and Normalizing Temporal Expressions in Croatian. In *Proceedings of the 9th Slovenian Language Technologies Conferences (IS-LT 2014)*, pages 99–103.

Jannik Strötgen and Michael Gertz. 2010. HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 321–324.

Jannik Strötgen and Michael Gertz. 2011. WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions. In *Proceedings of the Conference of the German Society for Computational Linguistics and Language Technology (GSCL 2011)*, pages 129–134.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298.

Jannik Strötgen, Julian Zell, and Michael Gertz. 2013. HeidelTime: Tuning English and Developing Spanish Resources for TempEval-3. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*, pages 15–19.

Jannik Strötgen, Ayser Armiti, Tran Van Canh, Julian Zell, and Michael Gertz. 2014. Time for More Languages: Temporal Tagging of Arabic, Italian, Spanish, and Vietnamese. *ACM Transactions on Asian Language Information Processing (TALIP)*, 13(1):1–21.

Matje van de Camp and Henning Christiansen. 2012. Resolving Relative Time Expressions in Dutch Text with Constraint Handling Rules. In *Constraint Solving and Language Processing – 7th International Workshop (CSLP 2012)*, pages 166–177.

Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 57–62.

Marc Verhagen. 2011. TempEval2 Data – Release Notes. Technical report, Brandeis University.