UFPRSheffield: Contrasting Rule-based and Support Vector Machine Approaches to Time Expression Identification in Clinical TempEval

Hegler Tissot
Federal University of Parana
Cel. Franc. H. dos Santos, 100
Curitiba, PR 81531-980, Brazil
hctissot@inf.ufpr.br

Genevieve Gorrell
The University of Sheffield
211 Portobello
Sheffield, S1 4DP, UK
g.gorrell@shef.ac.uk

Angus Roberts
The University of Sheffield
211 Portobello
Sheffield, S1 4DP, UK
angus.roberts@shef.ac.uk

Leon Derczynski
The University of Sheffield
211 Portobello
Sheffield, S1 4DP, UK
leon.derczynski@shef.ac.uk

Marcos Didonet Del Fabro
Federal University of Parana
Cel. Franc. H. dos Santos, 100
Curitiba, PR 81531-980, Brazil
marcos.ddf@inf.ufpr.br

Abstract

We present two approaches to time expression identification, as entered in to SemEval-2015 Task 6, Clinical TempEval. The first is a comprehensive rule-based approach that favoured recall, and which achieved the best recall for time expression identification in Clinical TempEval. The second is an SVM-based system built using readily available components, which was able to achieve a competitive F1 in a short development time. We discuss how the two approaches perform relative to each other, and how characteristics of the corpus affect the suitability of different approaches and their outcomes.

1 Introduction

SemEval-2015 Task 6, Clinical TempEval (Bethard et al., 2015), was a temporal information extraction task over the clinical domain. The combined University of Sheffield/Federal University of Parana team focused on identification of spans and features for time expressions (TIMEX3), based on specific annotation guidelines (TS and TA subtasks).

For time expressions, participants identified expression spans within the text and their corresponding classes: DATE, TIME, DURATION, QUANTIFIER, PREPOSTEXP or SET.[1] Participating systems had to annotate timexes according to the guidelines for the annotation of times, events and temporal relations in clinical notes – THYME Annotation Guidelines (Styler et al., 2014) – which is an extension of ISO TimeML (Pustejovsky et al., 2010) developed by the THYME project.[2] Further, ISO TimeML extends two other guidelines: a) TimeML Annotation Guidelines (Sauri et al., 2006), and b) TIDES 2005 Standard for the Annotation of Temporal Expressions (Ferro et al., 2005). Clinical TempEval temporal expression results[3] were given in terms of Precision, Recall and F1-score for identifying spans and classes of temporal expressions.

For Clinical TempEval two datasets were provided. The first was a training dataset comprising 293 documents with a total number 3818 annotated time expressions. The second dataset comprised 150 documents with 2078 timexes. This was used for evaluation and was then made available to participants, after evaluations were completed. Annotations identified the span and class of each timex. Table 1 shows the number of annotated timex by class in each dataset.

We present a rule-based and a SVM-based approach to time expression identification, and we discuss how they perform relative to each other, and how characteristics of the corpus affect outcomes and the suitability of the two approaches.

[1] There was no time normalisation task in Clinical TempEval

[2] http://thyme.healthnlp.org/ (accessed 27 Mar. 2015)

[3] http://alt.qcri.org/semeval2015/task6/index.php?id=results (accessed 27 Mar. 2015)

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 835–839,
Denver, Colorado, June 4-5, 2015.

Class	Training	Evaluation
DATE	2583	1422
TIME	117	59
DURATION	433	200
SET	218	116
QUANTIFIER	162	109
PREPOSTEXP	305	172
Total	**3818**	**2078**

Table 1: Time expressions per dataset.

2 HINX: A Rule-Based Approach

HINX is a rule-based system developed using GATE[4] (Cunningham et al., 2011). It executes a hierarchical set of rules and scripts in an information extraction pipeline that can be split into the 3 modules: 1) text pre-processing; 2) timex identification; and 3) timex normalisation, which are described below. These modules identify and normalise temporal concepts, starting from finding basic tokens, then grouping such tokens into more complex expressions, and finally normalising their features. An additional step was included to produce the output files in the desired format.

2.1 Text Pre-processing

This module is used to pre-process the documents and identify the document creation time (DCT).

HINX used GATE's ANNIE (Cunningham et al., 2011) – a rule-based system that was not specifically adapted to clinical domain – to provide tokenization, sentence splitting and part of speech (POS) tagging. We used the Unicode Alternate Tokenizer provided by GATE to split the text into very simple tokens such as numbers, punctuation and words. The Sentence Splitter identifies sentence boundaries, making it possible to avoid creating a timex that connects tokens from different sentences or paragraphs. POS Tagging produces a part-of-speech tag as an annotation on each word or symbol, which is useful in cases such as identifying whether the word "may" is being used as a verb or as a noun (the month).

We use rules written in JAPE, GATE's pattern matching language, to identify the DCT annotation reference within the "[meta]" tag at the beginning of each document. The DCT value was split into different features to be stored at the document level – year, month, day, hour, minute, and second.

2.2 Timex Identification

This module uses a set of hierarchical JAPE rules to combine 15 kinds of basic temporal tokens into more complex expressions, as described in the sequence of steps given below:

- **Numbers**: A set of rules is used to identify numbers that are written in a numeric or a non-numeric format, as numbers as words (e.g. "two and a half").
- **Temporal tokens**: Every word that can be used to identify temporal concepts is annotated as a basic temporal token - e.g. temporal granularities; periods of the day; names of months; days of the week; season names; words that represent past, present and future references; and words that can give an imprecise sense to a temporal expression (e.g. the word "few" in "the last few days"). Additionally, as a requirement for Clinical TempEval, we included specific rules to identify those words that corresponded to a timex of class PREPOSTEXP (e.g. "postoperative" and "pre-surgical").
- **Basic expressions**: A set of rules identifies the basic temporal expressions, including explicit dates and times in different formats (e.g. "2014", "15th of November", "12:30"), durations (e.g. "24 hours", "the last 3 months"), quantifiers, and isolated temporal tokens that can be normalised.
- **Complex expressions**: Complex expressions are formed by connecting two basic expressions or a basic expression with a temporal token. These represent information corresponding to ranges of values (e.g. "from July to August this year"), full timestamps (e.g. "Mar-03-2010 09:54:31"), referenced points in time (e.g. "last month"), and precise pre/post-operative periods (e.g. "two days postoperative").
- **SETs**: Temporal expressions denoting a SET (number of times and frequency, or just frequency) are identified by this specific set of rules (e.g. "twice-a-day", "three times every month", "99/minute", "every morning").
- **Imprecise expressions**: These expressions comprise language-specific structures used to refer to imprecise periods of time, including im-

[4] `http://gate.ac.uk` (accessed 27 Mar. 2015)

precise expressions defined with boundaries (e.g. "around 9-11 pm yesterday"), imprecise values of a given temporal granularity (e.g. "a few days ago", "the coming months"), precise and imprecise references (e.g. "that same month", "the end of last year", "the following days"), imprecise sets (e.g. "2 to 4 times a day"), and vague expressions (e.g. "some time earlier", "a long time ago").

2.3 Timex Normalisation

As the above identification process is run, the basic temporal tokens are combined to produce more complex annotations. Annotation features on these complex annotations are used to store specific time values, for use by the normalisation process. Such features comprise explicit values like "year=2004", references to the document creation time/DCT (e.g. "month=(DCT.month)+1" for the expression "in the following month", and "day=(DCT.day)-3" in "three days ago"), and a direct reference to the last mentioned timex in the previous sentences (e.g. "year=LAST.year" for the timex "April" in "In February 2002,... Then, in April,...").

The normalisation process uses these features to calculate corresponding final values. It also captures a set of other characteristics, including the precision of an expression, and whether or not it represents a boundary period of time. This last one is used to split the DURATION timexes into two different DATE expressions, as explicitly defined in the THYME Annotation Guidelines (e.g. "between November/2012 and March/2013").

3 Using an SVM-Based Approach

GATE provides an integration of LibSVM (Chang and Lin, 2011) technology with some modifications enabling effective rapid prototyping for the task of locating and classifying named entities. This was used to quickly achieve competitive results. An initial system was created in a few hours, and although a couple of days were spent trying parameter and feature variants, the initial results could not be improved. No development effort was required, the system being used as "off the shelf" software.

State of the art machine learning approaches to timex recognition often use sequence labeling (e.g. CRF) to find timex bounds (UzZaman et al., 2013), then a use separate instance-based classification step (e.g. with SVM) to classify them (Sun et al., 2013). Our approach uses SVM to implement separate named entity recognizers for each class, then makes a final selection for each span based on probability. GATE's LibSVM integration incorporates the uneven margins parameter (UM) (Li et al., 2009), which has been shown to improve results on imbalanced datasets especially for smaller corpora. In positioning the hyperplane further from the (smaller) positive set, we compensate for a tendency in smaller corpora for the larger (negative) class to push away the separator in a way that it doesn't tend to do when sufficient positive examples exist for them to populate their space more thoroughly, as this default behaviour can result in poor generalization and a conservative model. Since NLP tasks such as NER often do involve imbalanced datasets, this inclusion, as well as robust default implementation choices for NLP tasks, make it easy to get a respectable result quickly using GATE's SVM/UM, as our entry demonstrates. The feature set used is:

- String and part of speech of the current token plus the preceding and ensuing five.
- If a date has been detected for this span using the Date Normalizer rule-based date detection and normalization resource in GATE, then the type of date in this location is included as a feature. The mere presence of such a date annotation may be the most important aspect of this feature. Note that this Date Normalizer was not used in HINX, which used a custom solution.
- As above, but using the "complete" feature on the date, to indicate whether the date present in this location is a fully qualified date. This may be of value as an indicator of the quality of the rule-based date annotation.

A probabilistic polynomial SVM is used, of order 3. Probabilistic SVMs allow us to apply confidence thresholds later, so we may: 1) tune to the imbalanced dataset and task constraints, 2) use the "one vs. rest" method for transforming the multiclass problem to a set of binary problems, and 3) select the final class for the time expression. In the "one vs. rest" approach, one classifier is created for each class, allowing it to be separated from all others, and the class with the

SVM	Threshold	P	R	F1
Linear	0.2	0.68	0.59	0.63
Linear	0.4	0.76	0.55	0.64
Poly (3)	0.2	0.64	0.61	0.63
Poly (3)	0.25	0.69	0.61	**0.65**
Inc. hinx feats	0.25	0.72	0.54	0.62

Table 2: SVM tuning results.

	Span			Class		
Submission	P	R	F1	P	R	F1
HINX-1	0.479	0.747	0.584	0.455	0.709	0.555
HINX-2	0.494	0.770	0.602	0.470	0.733	0.573
HINX-3	0.311	0.794	0.447	0.296	**0.756**	0.425
HINX-4	0.311	**0.795**	0.447	0.296	**0.756**	0.425
HINX-5	0.411	**0.795**	0.542	0.391	**0.756**	0.516
SVM-1	0.732	0.661	0.695	0.712	0.643	0.676
SVM-2	0.741	0.655	0.695	0.723	0.640	0.679

Table 3: Final Clinical TempEval results.

highest confidence score is chosen. A UM of 0.4 is selected based on previous work (Li et al., 2005).

Two classifiers are trained for each class; one to identify the start of the entity and another to identify the end. This information is then post-processed into entity spans first by removing orphaned start or end tags and secondly by filtering out entities with lengths (in number of words) that did not appear in the training data. Finally, where multiple annotations overlap, a confidence score is used to select the strongest candidate. A separate confidence score is also used to remove weak entities.

Table 2 shows negligible difference between a linear and polynomial SVM (degree 3). A confidence threshold of 0.25 was selected empirically. Task training data was split 50:50 to form training and test sets to produce these figures. An additional experiment involved including the output from the HINX rule-based system as features for the SVM. This did not improve the outcome.

4 Results and Discussion

We submitted 5 runs using the HINX system and 2 runs using our SVM approach to Clinical TempEval. Results of both systems are shown in Table 3. For completeness, both SVM runs submitted are included. However the only difference between the two is that SVM-2 included the full training set, whereas SVM-1 included only the half reserved for testing at development time, and submitted as a backup for its quality of being a tested model. As expected, including more training data leads to a slightly superior result, and the fact that the improvement is small suggests the training set is adequate in size.

The HINX runs shown in Table 3 correspond to the following variants: 1) using preposition "at" as part of the timex span; 2) disregarding timexes of class QUANTIFIER; 3) using full measures span for QUANTIFIERs (e.g. "20 mg"); 4) considering measure tokens as non-markable expressions; and 5) disregarding QUANTIFIERs that represent measures. The timex type QUANTIFIER was targeted in different submitted runs as it was not clear how these expressions were annotated when comparing the training corpus to the annotation guidelines.

The HINX system had the best Recall over all Clinical TempEval systems in both subtasks. The low precision of the rule-based system was, however, a surprise, and led us to examine the training and test corpora in detail. While we would expect to see inconsistencies in any manually created corpus, we found a surprising number of repeated inconsistencies between the guidelines and the corpora for certain very regular and unambiguous temporal language constructs. These included: a) timex span and class inconsistencies, b) non-markable expressions that were annotated as timexes, c) many occurrences of SET expressions that were not manually annotated in the corpus, and d) inconsistencies in the set of manually annotated QUANTIFIERs. Had these inconsistencies not been present in the gold standard, HINX would have attained a precision between 0.85 and 0.90 (Tissot et al., 2015).

We suggest that inconsistent data such as this will tend to lower the precision of rule-based systems. To illustrate this, we ran HeidelTime (Strötgen et al., 2013) on this year's dataset and found that precision and recall were low $(0.44; 0.49)$ despite this being a demonstrably successful system in TempEval-3. Similarly low results can be observed from ClearTK-TimeML $(0.593; 0.428)$, used to evaluate the THYME Corpus (Styler et al., 2014). Systems were run "as-is", unadapted to the clinical domain. Styler et al. (2014) suggest that clinical narratives introduce new challenges for temporal information extraction systems, and performance degrades when moving to this domain. However, they do not con-

sider how far performance can be impaired by inconsistencies in the annotated corpus.

The appearance of a superior result by our machine learning system, which is agnostic about what information it uses to replicate the annotators' assertions, is therefore not to be taken at face value. A machine learning system may have learned regularities in an annotation style, rather than having learned to accurately find time expressions. This is an example of data bias (Hovy et al., 2014). Machine learning systems have a flexibility and power in finding non-obvious cues to more subtle patterns, which makes them successful in linguistically complex tasks, but also gives them a deceptive appearance of success where the irregularity in a task comes not from its inherent complexity but from flaws in the dataset.

Acknowledgments

We would like to thank the Mayo Clinic for permission to use the THYME corpus, and CAPES,[5] which is partially financing this work. This work also received funding from the European Union's Seventh Framework Programme (grant No. 611233,PHEME). AR, GG and LD are part funded by the National Institute for Health Research (NIHR) Biomedical Research Centre and Dementia Biomedical Research Unit at South London and Maudsley NHS Foundation Trust and King's College London.

References

Steven Bethard, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2015. SemEval-2015 Task 6: Clinical TempEval. In *Proceedings of the 9th International Workshop on Semantic Evaluation*.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3).

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. 2011. *Text Processing with GATE (Version 6)*.

Lisa Ferro, Laurie Gerber, Inderjeet Mani, Beth Sundheim, and George Wilson. 2005. TIDES 2005 Standard for the Annotation of Temporal Expressions. Technical report, MITRE Corp.

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. When POS data sets don't add up: Combatting sample bias. In *Proc. LREC*, LREC.

Yaoyong Li, Kalina Bontcheva, and Hamish Cunningham. 2005. SVM Based Learning System For Information Extraction. In Joab Winkler, Mahesan Niranjan, and Neil Lawerence, editors, *Deterministic and Statistical Methods in Machine Learning: First International Workshop, 7–10 September, 2004*, volume 3635 of *Lecture Notes in Computer Science*, pages 319–339, Sheffield, UK.

Yaoyong Li, Kalina Bontcheva, and Hamish Cunningham. 2009. Adapting SVM for Data Sparseness and Imbalance: A Case Study on Information Extraction. *Natural Language Engineering*, 15(2):241–271.

James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. ISO-TimeML: An international standard for semantic annotation. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC'10)*.

Roser Sauri, Jessica Littman, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2006. TimeML Annotation Guidelines, v1.2.1.

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

William Styler, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and James Pustejovsky. 2014. Temporal Annotation in the Clinical Domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Hegler Tissot, Angus Roberts, Leon Derczynski, Genevieve Gorrell, and Marcos Didonet Del Fabro. 2015. Analysis of temporal expressions annotated in clinical notes. In *Proceedings of 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, pages 93–102, London, UK.

Naushad UzZaman, Hector Llorens, Leon Derczynski, Marc Verhagen, James F. Allen, and James Pustejovsky. 2013. SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of the 7th International Workshop on Semantic Evaluations*.

[5] `http://www.iie.org/en/programs/capes` (accessed 27 Mar. 2015)