

Recognising and Interpreting Named Temporal Expressions

Matteo Brucato

University of Bologna

matteo.brucato@studio.unibo.it

Leon Derczynski

University of Sheffield

leon@dcs.shef.ac.uk

Hector Llorens

Nuance Communications

hector.llorens@nuance.com

Kalina Bontcheva

University of Sheffield

kalina@dcs.shef.ac.uk

Christian S. Jensen

Aarhus University

csj@cs.au.dk

Abstract

This paper introduces a new class of temporal expression – named temporal expressions – and methods for recognising and interpreting its members. The commonest temporal expressions typically contain date and time words, like *April* or *hours*. Research into recognising and interpreting these typical expressions is mature in many languages. However, there is a class of expressions that are less typical, very varied, and difficult to automatically interpret. These indicate dates and times, but are harder to detect because they often do not contain time words and are not used frequently enough to appear in conventional temporally-annotated corpora – for example *Michaelmas* or *Vasant Panchami*.

Using Wikipedia and linked data, we automatically construct a resource of English named temporal expressions, and use it to extract training examples from a large corpus. These examples are then used to train and evaluate a named temporal expression recogniser. We also introduce and evaluate rules for automatically interpreting these expressions, and we observe that use of the rules improves temporal annotation performance over existing corpora.

1 Introduction

The ability to express time in language is critical. We require this ability in order to communicate plans, to tell stories, and to describe change in the world around us.

Phrases that explicitly describe certain periods of time, or **temporal expressions**, are particularly useful. They may be calendar dates, mentions of months, relative expressions like “*tomorrow*”, and so on. In-depth accounts of temporal expressions – **timexes** – are given by Ferro et al. (2005) and Llorens et al. (2012a).

In this paper, we discuss a new class of timexes that signify a date or range of dates, but that do not explicitly include information about which dates these are (e.g., *October 31* vs. *Halloween*). Following the description of expressions that clearly identify one entity from a set of others by use of a proper noun as named entities, we call these **named temporal expressions** (or **NTEs**).

As with many linguistic phenomena, the phrases used as timexes have a power law-like frequency distribution in text. A few forms of expression make up for the bulk of occurrences of temporal expressions. However, existing research has been typically evaluated on only a small corpus of hand-annotated temporal expressions. With such resources, it is difficult to build or evaluate tools for recognising or interpreting the less-frequent temporal expressions, and this is reflected in the performance plateau of recent TempEval exercises (Verhagen et al., 2010; UzZaman et al., 2013).

Existing temporal expression recognition tools are typically rule-based (Strötgen and Gertz, 2010). These perform reasonably well on existing datasets, achieving F-scores of around 0.90, and improving them is an active area of research. However, as temporal annotation is expensive, existing datasets are not particularly large, and therefore do not contain as challenging a variety of forms of expression as general, unannotated text. Therefore, evaluations using these resources

are unlikely to indicate the true variety of forms of temporal expression. This leaves us poorly equipped to handle the long tail of temporal expressions, which is likely to be very long (Steedman, 2011), in terms of both tools and resources.

As the most common temporal expressions can be recognised automatically with reasonable accuracy, we propose methods for attacking the long tail of temporal expressions. We address the following questions:

- What share of all temporal expressions is accounted for by existing tools and corpora?
- How can we recognise previously unseen named temporal expressions?
- Having found a named temporal expression, how can we anchor it to a calendar date?

The remainder of this paper discusses the most closely related work, examines variety in temporal expressions in the available corpora, introduces our approach for named timex recognition, briefly examines their role in information seeking, and discusses the problem of interpreting these unusual temporal expressions.

2 Related Work

There is a reasonable amount of prior work on general-purpose timex recognition. The state of the art in temporal expression recognition is extended regularly with TempEval exercises (UzZaman et al., 2013). Currently, HeidelTime (Strötgen and Gertz, 2010) offers strong temporal expression recognition performance, though as it is rule-engineered, it is likely to perform poorly at recognising unseen named timexes. TIPSem (Llorens et al., 2012b) is based on machine learning and, given appropriate training data, has the potential to recognise named timexes. ANNIE (Cunningham et al., 2002) adopts a finite state approach to recognising a commonly-occurring but constrained set of temporal expressions. Han et al. (2006) propose interpreting temporal expressions through iterative constraint satisfaction, which yields some ability to interpret previously unseen timexes. Finally, as opposed to timexes, Shaw et al. (2009) used linked data to aid in event entity recognition. The distinguishing features of our approach are that we concentrate on temporal expressions that do not follow a general, structured format, and that instead of addressing the general timex recognition problem (which has been covered repeatedly in the

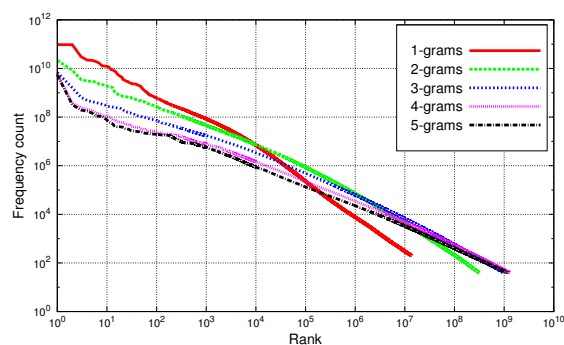


Figure 1: Frequency distribution of general terms

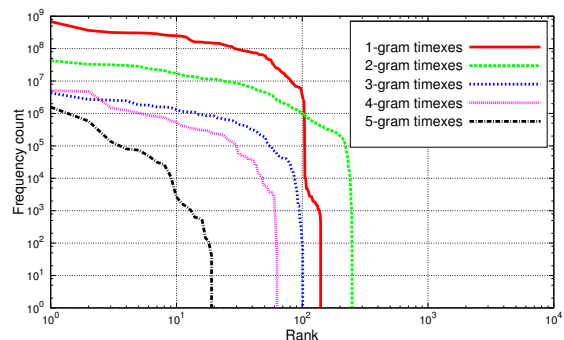


Figure 2: Frequency distribution of terms used as timexes in TimeBank and AQUAINT

literature, often from scratch), we address unusual expressions which are typically ignored by general purpose approaches.

3 Variety in Temporal Expressions

Our goal is to be able to recognise temporal expressions beyond the scope of current temporal annotation systems, thus extending timex recognition. In order to measure the scope of existing systems, we need to estimate the scale of variety in temporal expressions.

Using Google’s Web1T n-gram corpus (Brants and Franz, 2006), we drew the shape of the timex distribution curve. Firstly, we extracted the shape of the general term distribution curve; see Figure 1. Note the characteristic “knee” in the curve, after which terms become rarer than a plain Zipf-Mandelbrot distribution would suggest, as per Montemurro (2001). For timexes, we counted n-grams based on timex strings found in two temporally-annotated corpora; TimeBank (Pustejovsky et al., 2003), and the AQUAINT TimeML corpus. The resulting curve is shown in Figure 2.

The sharp falloff of this timex curve is what

Date	English name	বাংলা নাম
21 February	Language Martyrs' Day	Shoh শহীদ দি
17 March	Father of the Nation's birth anniversary	Jātir জাতির
26 March	Independence Day	Shad স্বাধীন
1 May	May Day	May মে দি
15 August	National Mourning Day	Jātir জাতির
16 December	Victory Day	Bijôe বিজয় দি
25 December	Christmas Day	Bôro বড়দিন

Figure 3: Holidays from a country (Bangladesh), as shown on a Wikipedia page

one might expect to see from a very small corpus. Namely, some of the more common expressions are found, in relatively high frequency (the initial shallow curve). The remaining expressions found in the small sample that this corpus represents are much rarer, as shown by the sharp drop at the low-frequency end of the curve.

This suggests that existing TimeML corpora are so small that they do not include a sufficiently diverse selection of these terms. Indeed, TimeBank has only around 65K tokens. To build and evaluate approaches for recognising NTEs, a new source of data is required.

4 Automatic Named Timex Recognition

Having described named timexes, we build a named timex resource taking a re-usable, low-supervision approach, and then construct a tool for automatic named timex discovery.

4.1 Mining Existing Named Timexes

Current TIMEX3-annotated resources do not account for a representatively broad set of temporal expressions (Figure 2). To supplement these resources, we automatically mined named temporal expressions from Wikipedia.

We started by identifying collections of these terms, for example on pages listing public holidays. The selection criterion was that the page be in English and have a reasonable number of NTE descriptions, marked up in a wiki table (e.g., Figure 3). The pages used are listed in Figure 4. We

```
en.wikipedia.org/wiki/...
Federal_holidays_in_the_United_States
Public_and_Bank_holidays_in_Scotland
Public_holidays_in_Australia
Public_holidays_in_Canada
Public_holidays_in_Denmark
Public_holidays_in_France
Public_holidays_in_Germany
Public_holidays_in_Hong_Kong
Public_holidays_in_India
Public_holidays_in_Italy
Public_holidays_in_Malaysia
Public_holidays_in_South_Africa
Public_holidays_in_the_European_Union
Public_holidays_in_the_United_Kingdom
Public_holidays_in_the_United_States
```

Figure 4: URLs from which source NTE descriptions were extracted

Official name	Date
Columbus Day	Second Monday in October
Veterans Day	November 11
Thanksgiving Day	Fourth Thursday in November
Christmas	December 25

Table 1: Sample Wikipedia events and interpretations

then automatically extracted the terms and their textual descriptions from these collections. An example extract is given in Table 1.

This data was supplemented using the holiday terms given in JollyDay, a Java date-handling library.¹ In total, we found 247 unique terms from 15 manually-selected Wikipedia pages, and 239 from JollyDay (containing an overlap of 54), for a total of 432 named timexes.

The resulting list of candidate named temporal expressions contained two types of anomaly. It contained some conventional temporal expressions (e.g., *August*) which should be removed; these were filtered out using HeidelTime, a rule-engineered timex system. It also contained polysemous named timexes, that were not only used in a temporal sense. For example, *Carnival* is both a specific festival, a tour operator, and a polysemous common noun indicating a period of revelry or an exciting mixture of something.

4.2 Disambiguating NTEs with Linked Data

Following Shaw et al. (2009), we used linked open data to handle ambiguous temporal entities. We discriminated monosemous timexes (e.g., *Reformation Day*) from polysemous ones (e.g., *Easter*,

¹See <http://jollyday.sourceforge.net/>

which may be both a holiday and part of a compound noun referring to e.g. a chocolate egg) via DBpedia (Bizer et al., 2009), looking for entities with matching names.

After discarding URIs of media that were in film and song titles, NTEs that still had more than one remaining corresponding entity URI were identified as polysemous. The final set comprised 424 expressions, of which 342 were monosemous and 82 were polysemous.

4.3 Recognising Named Timexes in Text

Having built a collection of named temporal expressions, we moved on to the task of NTE discovery. Our approach was to first develop a statistical tagger adapted to NTE recognition, and then apply it to new data, to observe what expressions it annotates beyond those in the collection extracted from Wikipedia.

The collection was used to construct a corpus and then a statistical named temporal expression recogniser. The corpus was constructed as follows. Using our list of monosemous named timexes, we searched the Gigaword corpus to retrieve paragraphs containing the timexes. These paragraphs were split into sentences (Kiss and Strunk, 2006), and the sentences matching any NTE were extracted; the sentences were then broken down into lists of tokens. We marked all monosemous named timexes in the sentences as target entities.

Some NTEs are polysemous, having both temporal and non-temporal sense. Observation of a small part of the corpus suggested that these polysemous NTEs generally occurred in a temporal sense when in the same sentence as other temporal phrases. Rather than excluding any sentence containing a polysemous NTE from the corpus on grounds of ambiguity, based on this observation, we adopted a simple heuristic: polysemous NTEs are included if they are collocated with a monosemous NTE. This reduced the considered set of polysemous NTEs by 22 to 60, for a total of 402 unique expressions.

Tokens in each sentence were then labelled according to a simple in-entity/out-of-entity binary format. The sentences were then split into training and evaluation sets, with no named temporal expressions found in both groups, i.e., every NTE is exclusively in either one or the other set.

In total, 3 861 sentences (117 060 tokens) were

System	Recall	Precision	F1
<i>strict</i>			
Gazetteer baseline	5.6%	15.2%	8.2%
TIPSem	56.5%	71.7%	63.2%
TIPSem-B	56.6%	75.5%	64.7%
Stanford NER	56.7%	74.2%	64.3%
<i>lenient</i>			
Gazetteer baseline	6.8%	19.4%	10.1%
TIPSem	75.8%	97.3%	85.9%
TIPSem-B	71.4%	95.0%	81.5%
Stanford NER	73.7%	97.2%	83.8%

Table 2: Sample Wikipedia events and interpretations. Lenient matches includes annotations that at least overlap with the reference.

extracted from English Gigaword v5 (Graff et al., 2003), containing 4 180 named timex annotations. The training split contained 1 053 of these sentences. The entire corpus construction method requires no human intervention aside from supplying source Wikipedia pages.

Regarding the NTE recognisers, we adapted three entity recognition approaches to the task by discarding their default models and rebuilding new models based solely on this NTE corpus. The recognition tools were CRF-based: a multi-purpose system incorporating non-local information, Stanford NER (Finkel et al., 2005); one for temporal entity recognition that uses semantic role information, TIPSem (Llorens et al., 2012b); and TIPSem-B, a baseline temporal entity recognition variant of TIPSem.

Recognisers were learned from the training split and evaluated on the test split. As we are attempting to recognise named timexes only, we do not do comparison against tools designed for standard timex recognition, as these are designed for a different task.

A naïve gazetteer-matching baseline was used, based on timex strings found in existing resources (TimeBank and the AQUAINT TimeML annotations). This behaved exactly as a direct case-insensitive word look-up, matching any whole phrases found within the corpus. Its recall should tell us how broad the range of temporal expressions found in prior TimeML resources is. Evaluation was performed using GATE (Cunningham et al., 2013); results are reported in Table 2.

Precision was generally higher than recall, with both at reasonable levels for a first attempt at this new class of entities. This indicates that while our

Recogniser	% of query texts	% of queries
HeidelTime	2.90	1.97
NTE gazetteer	0.06	0.14

Table 3: Temporal intent indicator prevalence in a web search query log

approaches do not identify too many non-timexes as being timexes, further work is called for at improving the range of named timexes they recognise. In particular, the temporal expressions used in the TimeBank and AQUAINT corpora have a very small overlap with the named temporal expressions we identified.

4.4 Finding New NTEs

With a system that is capable of recognising named temporal expressions in our test data, which contains previously-unseen NTEs, it may be possible to discover new NTEs. Unlabelled text can be labelled using statistical NTE recognisers. One may have concerns over using a system with strict recognition precision in the 70s for this purpose; however, lenient recognition precision is in the mid- to high-90s, which indicates that the negative impact of spurious annotations will be low.

We attempted to find new NTEs by applying the TIPSem model to another portion of the Gigaword text. Sample results include phrases such as:

- *European Cup*
- *Hamlet Cup*
- *bank holiday*
- *Dayton peace agreement*

Although these are difficult to evaluate directly, they can readily be applied in semi-supervised approaches to temporal annotation, e.g., in part of a bootstrapping approach to NTE recognition and general timex recognition.

5 Temporal Intent Queries

This section contains a brief investigation of named temporal expressions (and general temporal expression recognition) in information retrieval query interpretation.

In classical information retrieval with a textual query over a document collection, the query represents the lexicalisation of a searcher’s information need. To identify a *temporal* information need, one must recognise signals in the query that reflect

this (Jones and Diaz, 2007; Metzler et al., 2009). Detecting temporal intent in queries may benefit from linguistic approaches to query understanding and decomposition (Campos et al., 2012).

Beyond common formulations of timexes, this is a challenging problem in two regards. As we have already explained, certain forms of temporal expression are not recognised by existing tools. Also, event-related queries (e.g., “*stock market reaction to michael jackson’s death*”) signify temporal intent but may not contain any temporal expressions at all. While the second class is not covered here, we do address the first.

We are interested in the proportion of temporal intent search queries that can be captured with awareness of named temporal expressions. Our method is to examine existing records of text questions and search engine queries, similar to the approach of Nunes et al. (2008). We used 1 200 000 randomly sampled query strings from the AOL search log (Pass et al., 2006) as a corpus. This corpus comprises 167 794 unique terse query strings.

We ran HeidelTime (Strötgen and Gertz, 2010) over this corpus. We also computed the intersection of query texts with our mined named timexes. Results are given in Table 3.

While temporal expressions in general are notably frequent in the data, it can be seen that only a relatively small proportion of queries contain named temporal expressions (0.14%). Named temporal expressions are not dominant in queries from this corpus. Indeed, while the data suggests that general temporal expressions are in the long tail (as the proportion of timexes recognised in unique queries is greater than that in all queries), the inverse is true for named temporal expressions. Examining the data, only a few variants of NTE occur in the query log.

6 Temporal Expression Interpretation

Once one has recognised that a particular expression is used in a temporal sense, the next step is to interpret the expression. This may entail anchoring it to a calendar or other formal representation.

We consider the task of interpreting timexes to the TimeML/TIMEX3 standard (Ferro et al., 2005). This produces normalised values from timexes, as shown below.

- (1) *January 2nd, 1980* → 1980-01-02
Summer 2012 → 2012-SU
now → PRESENT_REF

id	expression	interpretation
92	Autumn_Holiday	DATE_WEEK_WEEKNUM(DCT, -1, Monday ,TO_MONTH("September"))
178	Liberation_Day	DATE_MONTH_DAY(DCT, TO_MONTH("April"), TO_DAY("25"))
179	Republic_Day	DATE_MONTH_DAY(DCT, TO_MONTH("June"), TO_DAY("2"))
180	Ferragosto	DATE_MONTH_DAY(DCT, TO_MONTH("August"), TO_DAY("15"))

Figure 5: Example named timex rules in TIMEN

Discovering such interpretations is a difficult task. For example, based on text, it is difficult to automatically learn or infer the link between “*New Year’s Day*” and 1st January, or the associations between north/south hemisphere and which months fall in summer, especially given the cost of temporal annotation and resulting scarcity of annotated resources. This often leaves the task of developing such interpretations to human computation (Sabou et al., 2012). The closest computational method for solving this problem uses a more flexible compositional approach to timex interpretation (Angeli et al., 2012), though it is prone to floundering and failing on completely new expressions, such as named timexes.

As the named timexes mined from Wikipedia were generally accompanied by a textual description of the time (e.g., as in Figure 3), we used these descriptions to work out how to interpret the expression. We created a custom parser that worked well with the majority of uncurated, natural language descriptions of named timex dates. Having gathered information from Wikipedia, we then encoded it as rules in a popular timex interpretation system, TIMEN (Llorens et al., 2012a).

TIMEN operates using expression capture rules over a language-specific knowledge base that contains information on temporal primitives such as weekday and month names. Rules chosen for normalisation are those that match the timex’s pattern, in order of priority, highest first. If a rule has conditions, it can only be applied if the timex satisfies them. Matched rules operate on a priority and constraint-satisfaction basis.

The rules in TIMEN allow the linking of contextual temporal information not explicit in the expression (such as document creation year) with time information in the expression. This expression-based information is often qualitative (i.e., text), and so TIMEN also includes rules for rendering it quantitative. For example, there are built-in functions that convert language-specific terms such as *Monday*, *lunes* or *the second* into quantitative offsets that operate over an internal knowledge base provided for that language. The

Corpus	TIMEN	Augmented	ER
TempEval-3	69.6%	69.8%	0.7%
TimenEval	68.0%	69.4%	4.3%

Table 4: Timex interpretation accuracy with and without rules mined from Wikipedia. ER is Error Reduction

result is a numeric representation of the temporal expression. This representation can be under-specified. For example, in the scope of NTEs, often the year is not mentioned, as it is document-dependent. As a result, the TIMEN rules for handling NTEs often do not declare any information about years, leaving this to TIMEN’s management of reference time (Reichenbach, 1947).

Example rules for NTEs are shown in Figure 5. In total, we successfully extracted interpretation rules for 298 of the previously-identified named timexes (70.3% of the NTEs in our inventory).

To evaluate this approach, we did timex interpretation only, using reference annotations of timex bounds. We ran the standard and augmented TIMEN over recent existing corpora (the TempEval-3 corpus and the TIMEN test data); results are in Table 4. The additional rules improved TIMEN’s ability to interpret named timexes. The error reduction figures demonstrate that improvements can be achieved by accounting for these timexes.

Note the small improvement over the small TempEval-3 corpus (0.7%); upon examination, we found that this newswire corpus’ content not only contained few named timexes, but in fact seemed to take pains to avoid mentioning festivals, possibly as part of areligious journalist guidelines.

In any event, the indication is that newswire is a poor genre for the evaluation of timex annotation systems, due to its limited forms of expression. The TimenEval corpus was designed to be difficult to process, and it is over this data that we see the greatest improvement. The real contribution here is increasing the range of expressions that can be recognised and interpreted.

7 Discussion

While recognising and interpreting named timexes is useful in many scenarios, and while it is possible to perform this task automatically, we encountered some interesting problems during our work.

Spatial Variations: Many expressions are interpreted differently depending on the locale. For example, *Labor Day* is May 1 in much of the world, but is the first Monday in May in parts of Australia (Queensland and the Northern Territories) and the first Monday in September in the USA. While TIMEN can process variations in named timex interpretation over time (e.g., *Washington Day* is February 22 until 1971, after which it falls on the third Monday in February), this locale-based information is not always available and is not considered for the interpretation task. This may be possible as a future extension: separate modules can assess the origin or subject locale of the input text (based on, e.g., newswire lead-in, spelling variation, or location mentions, the last of which also requires spatial grounding or entity disambiguation) and pass this region information to rules for normalising, e.g., *Summer*.

Easter: Easter is difficult to interpret.² Its time is based on locale, year, which equinox is to be used (astronomical vs. religious), and many other factors. Also, many other named timexes depend on Easter, such as Pentecost, Lent, and Pancake Day. Being able to use Easter as an offset in date calculus will improve the coverage of named timex interpretation. The liturgical origins of the named timexes associated with the date provide some indication of the frequency of texts associated with named temporal expressions.

Multiple Calendars: Not all named timexes can be calculated with one calendar. When building interpretation rules, demand for, e.g., lunar, astronomical, and Hebrew calendars emerges quickly. Even conventional dates require different calendars when one goes far back enough. A comprehensive timex interpretation tool must account for multiple calendars (Urgun et al., 2007).

Forms of Expression: Finally, diversity of expression may impair named timex recognition. The NTE *Martin Luther King day*, for example, may also be expressed as *MLK day*. In a sufficiently long text, one may use co-reference resolution to link and resolve the two. A statistical

²See <https://en.wikipedia.org/wiki/Computus>

approach like our named timex recogniser (Section 4.3) may help here.

8 Conclusion

In this paper, we have introduced a new class of entities: named temporal expressions. These are hard to deal with because they do not resemble conventional temporal expressions, they can be expressed in a wide range of ways, they occur infrequently, and they cannot readily be interpreted to calendar dates.

8.1 Summary

We developed an approach for automatically extracting these named temporal expressions from Wikipedia, and we developed a named temporal expression corpus using linked data. This then helped train classifiers for automatically recognising (and thus discovering) named temporal expressions, with reasonable success (64.7% F1 measure). We also extracted interpretation rules for these expressions, allowing them to be converted to calendar dates, and used these to extend an existing state-of-the-art system. This augmented system had improved performance on existing temporally-annotated corpora.

8.2 Resources

The mined expressions and the annotated sentences extracted from Gigaword are made available via an author's website.³ Further, the TIMEN rules for normalising named timexes are also released, to be included in TIMEN.

8.3 Future Work

Building basic approaches to timex normalisation is no longer an interesting or useful task. Multiple actively-maintained, state-of-the-art tools address this problem, achieving good performance. However, as with many natural language processing problems, diminishing returns are being seen in the field. Therefore, next efforts must address the temporal expressions that we cannot yet already detect and interpret.

It is of interest to consider the automatic extraction of named timex resolution rules, perhaps using the most important timexes (Strötgen et al., 2012) from articles describing the corresponding occasion. It is also relevant to merge our named timex corpus with existing timex corpora

³See <http://derczynski.com/sheffield/>

(e.g. Derczynski et al. (2012)), after annotating the conventional timexes in our named timex training data. Such a corpus could be extended by extracting sentences that cite the Wikipedia or DBpedia entries corresponding to named timexes. Evaluation against such a resource is less likely to over-report the variety of expressions recognised by timex annotation systems, and can provide a solid base for future wide-coverage approaches to temporal expression recognition.

Decomposing the complex temporal annotation task so that it can be reliably crowdsourced would enable the construction of more resources. Using human computation like this is also likely to be useful in named timex sense disambiguation and interpretation, making it a promising source of more and better data.

Acknowledgments

This work was partially supported by funding from CHIST-ERA grant No. EP/K017896/1, uComp (www.ucomp.eu). The authors thank Barry Norton of Ontotext for helpful discussions on resolving events on the semantic web. The first two authors would also like to thank Aarhus University for their support.

References

- G. Angeli, C. D. Manning, and D. Jurafsky. 2012. Parsing time: Learning to interpret time expressions. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 446–455. Association for Computational Linguistics.
- C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. 2009. DBpedia-A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165.
- T. Brants and A. Franz. 2006. The Google Web 1T 5-gram corpus version 1.1. *LDC2006T13*.
- R. Campos, G. Dias, A. M. Jorge, and C. Nunes. 2012. Enriching temporal query understanding through date identification: how to tag implicit temporal queries? In *Proceedings of the 2nd Temporal Web Analytics Workshop*, pages 41–48. ACM.
- H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. 2002. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.
- H. Cunningham, V. Tablan, A. Roberts, and K. Bontcheva. 2013. Getting More Out of Biomedical Documents with GATE’s Full Lifecycle Open Source Text Analytics. *PLoS computational biology*, 9(2):e1002854.
- L. Derczynski, H. Llorens, and E. Saquete. 2012. Massively Increasing TIMEX3 Resources: A Transduction Approach. In *Proceedings of the international Conference on Language Resources and Evaluation*, pages 3754–3761.
- L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson. 2005. TIDES 2005 standard for the annotation of temporal expressions.
- J. R. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.
- D. Graff, J. Kong, K. Chen, and K. Maeda. 2003. English Gigaword. *Linguistic Data Consortium, Philadelphia*.
- B. Han, D. Gates, and L. Levin. 2006. From language to time: A temporal expression anchorer. In *Temporal Representation and Reasoning, 2006. TIME 2006. Thirteenth International Symposium on*, pages 196–203. IEEE.
- R. Jones and F. Diaz. 2007. Temporal profiles of queries. *ACM Transactions on Information Systems (TOIS)*, 25(3):14.
- T. Kiss and J. Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525.
- H. Llorens, L. Derczynski, R. Gaizauskas, and E. Saquete. 2012a. TIMEN: An Open Temporal Expression Normalisation Resource. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*.
- H. Llorens, E. Saquete, and B. Navarro-Colorado. 2012b. Applying semantic knowledge to the automatic processing of temporal expressions and events in natural language. *Information Processing & Management*.
- D. Metzler, R. Jones, F. Peng, and R. Zhang. 2009. Improving search relevance for implicitly temporal queries. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 700–701. ACM.
- M. A. Montemurro. 2001. Beyond the Zipf-Mandelbrot law in quantitative linguistics. *Physica A: Statistical Mechanics and its Applications*, 300(3):567–578.

- S. Nunes, C. Ribeiro, and G. David. 2008. Use of temporal expressions in web search. In *Advances in Information Retrieval*, pages 580–584. Springer.
- G. Pass, A. Chowdhury, and C. Torgeson. 2006. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems*.
- J. Pustejovsky, J. Castano, R. Ingria, R. Sauri, R. Gaizauskas, A. Setzer, G. Katz, and D. Radev. 2003. TimeML: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.
- H. Reichenbach. 1947. The tenses of verbs. In *Elements of Symbolic Logic*. Dover Publications.
- M. Sabou, K. Bontcheva, and A. Scharl. 2012. Crowdsourcing research opportunities: lessons from natural language processing. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, page 17. ACM.
- R. Shaw, R. Troncy, and L. Hardman. 2009. LODÉ: Linking Open Descriptions of Events. In *Proceedings of the Asian Semantic Web Conference*, volume LNCS 5926, pages 153–167. Springer.
- M. Steedman. 2011. Romantics and revolutionaries. *Linguistic Issues in Language Technology*, 6(11).
- J. Strötgen and M. Gertz. 2010. HeidelTime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324. Association for Computational Linguistics.
- J. Strötgen, O. Alonso, and M. Gertz. 2012. Identification of top relevant temporal expressions in documents. In *Proceedings of the 2nd Temporal Web Analytics Workshop*, pages 33–40. ACM.
- B. Urgun, C. E. Dyreson, R. T. Snodgrass, J. K. Miller, N. Kline, M. D. Soo, and C. S. Jensen. 2007. Integrating multiple calendars using τ ZAMAN. *Software: Practice and Experience*, 37(3):267–308.
- N. UzZaman, H. Llorens, L. Derczynski, M. Verhagen, J. F. Allen, and J. 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*.
- M. Verhagen, R. Saurí, T. Caselli, and J. Pustejovsky. 2010. SemEval-2010 task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62. Association for Computational Linguistics.