Tracking Out-of-date Newspaper Articles

Frederik Cailliau[1,2], Aude Giraudel[3] and Béatrice Arnulphy[4]

[1] Sinequa Labs, 12 Rue d'Athènes, F-75009 Paris
[2] LIPN-Univ. de Paris-Nord, 99 av. Jean-Baptiste Clément, F-93430 Villetaneuse
[3] Syllabs, 15 rue Jean-Baptiste Berlier, F-75013 Paris
[4] LIMSI-CNRS, BP 133, F-91403 Orsay*
cailliau@sinequa.com giraudel@syllabs.com beatrice.arnulphy@limsi.fr

Abstract. Local newspapers rely on their local correspondents to bring you the freshest news of your home town. Some of the articles written by these correspondents are not published immediately, but are put aside to be placed in a later edition. One of the challenges the editor in chief is confronted with, is to publish only up-to-date information about a local event. In this paper we present a system that tracks out-of-date newspaper articles to prevent their publishing. It firstly dates the event the article is talking about. The date detection grammar is written in a single but complex finite state automaton based on linguistic pattern matching. Secondly, it computes an absolute date for each relative date. A freshness score can then be deduced from the time difference between the extracted and the publication date. The system has been tested on several French local newspaper corpora. Baseline for the temporal extraction is our standard date extraction that was developed for general purposes.

Keywords: extraction of temporal information, named entities, events, information retrieval, newspaper articles

1 Introduction

Local daily newspapers rely on their network of correspondents to cover local events. Once revised by a journalist, these texts are able to be published as newspaper articles. Everyday the editor in chief validates or assembles the pages that will be published in the next edition. One of the challenges he is confronted with, is to validate only those articles covering hot news or coming events and to replace out-of-date articles by more recent ones. Unfortunately there is no easy or quick way to perform this task. To judge whether the article talks about a recent or a coming event, the editor has to read most of the article, since the day of writing is not a reliable indicator, when given. Even if most of the articles may be short, this activity is too time-expensive to be executed in the short delay before going to press. The editor's

* Some of the work in this article was done in collaboration with Béatrice Arnulphy during her internship at Sinequa in the summer of 2006 as a graduate student of the French university *Université de Provence*.

© A. Gelbukh (Ed.)
Advances in Computational Linguistics.
Research in Computing Science 41, 2009, pp. 277-288
Received 09/11/08
Accepted 15/12/08
Final version 05/02/09

efficiency relies on his or hers capability of rapidly dating the event an article is talking about.

Sinequa† is a French software editor of search technology, and several of its press-related clients expressed the need to facilitate this process. Therefore, we have enriched its search technology with a tool that performs automatic date and period detection and calculates the difference with the publication date as to give a *freshness* score to each article. That score can then be indexed as meta-data.

The paper is organized as follows. After a section on related work, we present a typology of temporal markers that appear in newspaper articles and their corresponding patterns that can be used to date the article's contents. The technology used to tackle the date extraction problem is detailed in section 4. We then dedicate a section on the conversion of the relative dates and the calculation of the difference with the publication date. The final section presents an evaluation of our tool on several local newspaper corpora.

2 Related Work and Working Hypothesis

Well performing Named Entities Recognition (NER) systems are around for quite some time. Back in 1997, the best system at the English NER task in the MUC-7 evaluation showed recognition results close to those of the human annotators: the best system performed a 93.39 score where the "worst" human annotator did 96.95 [1]. NER remains a domain of high interest in the community, as shows e.g. the existence of the second HAREM contest for Portuguese [2] and the 2007 and 2008 ACE (Automatic Content Extraction) evaluations organized by the NIST [3]. Part of the sustaining interest is due to the desire of building robust NER systems for more than one language. The systems that participated at ACE 2007 EDR (Entity Detection and Recognition) task could run their system on evaluation corpora of broadcast news, newswires and weblogs in English, Chinese and Arabic. Three other corpora were available for English: broadcast conversations, telephone and Usenet. The results show that an overall progress is still to be made.

Learning-based probabilistic systems are expected to perform badly when confronted to an unknown text type. But rule-based systems, as report [4], show the same behavior when confronted to other text types: a recognition rate of 90% on written documents of newspaper genre drops to 50% when executed on more informal text. The authors' conclusion is that if we want to get near to human-level scores, rule-based systems have to take into account specific characteristics of the corpus. Our system being rule-based, its results on texts of the same genre are quite interesting in this perspective. [5] shows that open system architecture is a way to go to tackle the problem of the different text types.

ACE 2007 presented a task on the detection of temporal expressions. The standard used for the annotation is TIMEX2 [6]. Currently, our time extractions comply with an internal standard in XML, but could be converted into the TIMEX2 format. Aside some exceptions, they cover a subset of the TIMEX2 expressions. We did e.g. not

† See http://www.sinequa.com/ for more information.

include any proper nouns (like *New Year's Eve*, *All Saint's Day*) although we probably should have, nor hour expressions (*8:00*) as lexical triggers. Some adjectives (*biannual*, *daytime*) and adverbs (*currently*, *monthly*) of the standard are clearly not taken into account, while others are (*next*), but only in co-occurrence with other triggers.

Temporal processing of news has been the subject of [7] who detects temporal expressions in print and broadcast news for event ordering. They use a basic set of handcrafted rules refined by machine-learning results. [8] extracts temporal information of any document in order to obtain the temporal coverage of its topics, called event-time.

Finally, TimeML [9] provides an annotation scheme for identified events in a text document to be oriented on a timeline. It is the recognized standard for inclusion of all temporal references in order to build a general model of text semantics. Parent et al. (2008) have reported work done on French, but this kind of complete temporal analysis is too rich for our purpose. It is certainly very important for language modeling and has a direct application in Question Answering [10].

Our approach is engineering-driven, rule-based and corpus-oriented. The framework is clearly determined by the application and the extraction rules are motivated by the corpus study. These restrictions led us to the following working hypothesis.

1. A newspaper article is self-sufficient: it usually talks about one or more events that will be dated within the same article.
2. An article can contain more than one date. The one closest to publication date is probably closely related to the main event and therefore the only one of interest for the calculation of the article's freshness score.
3. An article mainly focuses on one event. All dates within one article are somehow related to this event. If this is not the case, the article is out of our scope.
4. Simple regular expression date detection is of no use as we need to situate all dates, eg. *Thursday*, in respect to the writer's temporal perspective: for the freshness score, we need to know whether it happened last Thursday or will happen next Thursday. A complete temporal analysis would in the mean time be too ambitious for our purpose.

3 Typology of the Extracted Temporal References

Of all temporal references discourse is made of, we will only detect those that give us a precise time indication about the article's event. We will call them *direct* in opposition to *indirect* temporal references. Indirect references give a circumstantial description and their conversion into an absolute temporal expression would ask for a deep syntactic and semantic understanding combined with some ontology-based calculation: e.g. *during the last presidential elections*. As indicated before, this is far too ambitious for the aimed application. Instead, we extract all direct references and use verbal tense as a temporal marker to situate the reference future or past to the writer's perspective.

The extracted temporal references are illustrated in Table 1.

Table 1. Typology of extracted temporal references

Type	Subtype	Example	Verb tense	Translation
Explicit relative references	TODAY	aujourd'hui; ce matin	-	today, this morning
	YESTERDAY	hier	-	Yesterday
	DAY_BEFORE_YESTERDAY	avant-hier	-	the day before yesterday
	TOMORROW	demain	-	Tomorrow
	DAY_AFTER_TOMORROW	après-demain	-	the day after tomorrow
Isolated days	DAY	dimanche	-	Sunday
	DAY_BEFORE	dimanche dernier	-	last Sunday
		dimanche	past	Sunday
	DAY_AFTER	dimanche prochain	-	next Sunday
		dimanche	present, future	Sunday
Weekends	WEEKEND_BEFORE	le week-end dernier	-	last weekend
		ce week-end	past	this weekend
	WEEKEND_AFTER	le week-end prochain	-	next weekend
		ce week-end	present, future	this weekend
Explicit date references	COMPLETE_DATE	21 juin 2007	-	21 June 2007
	DATE_BEFORE	lundi 3 mars; jusqu'au 12 janvier; depuis le 26 juin	past	Monday 3 March; until the 1st of April; since the 26th of June
	DATE_AFTER	lundi 3 mars; jusqu'au 12 janvier; depuis le 26 juin	present, future	Monday 3 March; until the 1st of April; since the 26th of June
General references	GEN_DATE	lundi 3 mars; 2004	-	Monday 3 March; 2004

The alphanumeric temporal references in table 1 are recognized by the patterns described in table 2. Some special restrictions concern the extraction of the months: they are solely extracted when preceded by the French preposition *en* (in), itself not preceded by a past participle.

Table 2. Alphanumeric temporal reference patterns

Day of the week	Day of the month	Month	Year	Example
x	x	x	x	lundi 3 mars 1981
	x	x	x	3 mars 1981
x	x	x		lundi 3 mars
	x	x		3 mars
		x	x	mars 1981
x				lundi
			x	1981

The following two temporal reference patterns are deliberately not extracted:

1. day of the week + day of the month (with no indication of the month) : e.g. *mardi 3* (= on Tuesday, the 3rd);

2. *le* + day of the week, which is used in French for events recurring every week: e.g. *le lundi* (= every Monday);

Special restrictions were introduced on the recognition of *aujourd'hui* (= today) and *hier* (= yesterday) because they can sometimes occur in idiomatic expressions like in *d'hier et d'aujourd'hui*, which can signify "of yesterday and today", as well as "of all times". Contextual recognition of these two references requires them to co-occur with a small and incomplete list of event verbs (eg. *débuter* = to start; *organiser* = to organize) and expressions (eg. *avoir lieu = to take place*).

Durations are only partially taken into account: only the last reference is extracted. Eg. *du vendredi 6 au 8 juin* (from the 6th to the 8th Juin) + past tense : *8 juin* is extracted as DATE_BEFORE ; *du 6 au 8 juin* + present/future tense : *8 juin* is extracted as DATE_AFTER.

Sequences of month days are treated in the same way as the durations, eg. *le 6, 7 et 8 février* (the 6th, 7th and 8th February): *8 février* is either extracted as preceding or following the publication date.

Depending on the verb tense, sequences of separate weekdays are not treated in the same way. If the tense is past, the last one is extracted, if it is present or future, the first one is. This way, only the weekday closest to the publication day is extracted. E.g. *La conférence aura lieu lundi, mardi, mercredi …* (The conference takes place on Monday, Thursday, Wednesday): *lundi* is extracted.

4 Extraction Technology

We use the standalone version of the information extraction suite of Sinequa's search technology. It provides a rule- and lexicon-based morpho-syntactic analysis as well as disambiguation based on a general language model. Entity extraction is performed on the tagger's output with an in-house proprietary transducer technology. Patterns can be defined using word forms, lemmas and morpho-syntactic categories. These can be negated and combined. Word forms can be expressed by regular expressions and categories are dictionary-based (e.g. *adj* for the grammatical category adjective, *s* for singular) or computed (e.g. *has_vowel* for words with at least one vowel). For performance reasons, verb tense is provided by morpho-syntactic analysis rather than syntactic parsing.

Subtypes can be defined for the different paths. They are heavily exploited to provide the relative to absolute date conversion module with the necessary information. They also indicate whether the extracted date is situated before or after the writing date. Fig. 1 presents a screenshot of the transducer containing all of the extraction patterns. It contains 388 states and 668 arcs. The initial state is at the center and eleven final states are dispersed on the border of the image. The number of match-paths totals the impressive number of 6174.

The transducer used for the baseline extraction in the evaluation counts 67 nodes and 138 arcs. It represents 736 match-paths.

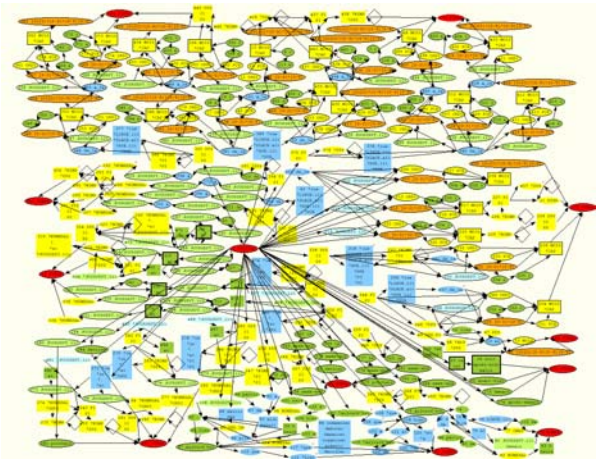

Fig. 1. Transducer for temporal reference extraction

5 Converting Extracted Dates into Absolute Temporal Expressions

The various temporal references extracted by the automaton can be absolute or relative. The global objective of the work is to give an absolute date to each newspaper article. Absolute dates are dates independent from any relative marker and can be listed into a calendar. Typical pattern for absolute dates is the following sequence: day of the month + month + year. All references should thus be transformed to absolute dates. The techniques used to compute absolute dates differ between types of relative references. All are presented in the next sections.

5.1 Explicit Relative Dates

Some relative dates are extracted with sufficient precision so as to easily deduce an absolute date. Those references belong to the extracted subtypes TODAY, YESTERDAY, DAY_BEFORE_YESTERDAY, TOMORROW and DAY_AFTER_TOMORROW. The absolute date is just the result of the addition or subtraction of 0 to 2 days to the publication date.

5.2 Isolated Days

The case of isolated days like *le mercredi dernier* (= last Wednesday) or *il arrivera lundi* (= he will arrive on Monday) that respectively refer to the subtypes DAY_BEFORE and DAY_AFTER implies a computation of the number of days between extracted reference and publication day.

The articles' metadata provide the publication date as an absolute date. It is formatted as following: YYYY-MM-DD, where Y=year, M=month and D=day of the month. To be able to compare the extracted day of the week with the given day of the month, we must transform the least into a day of the week. The technique used to achieve this transformation is the use of the famous Zeller formula. This formula computes the day of the week corresponding to a given date:

$$\text{Day} = (j+[2.6m-0.2] + a + [a/4] + [c/4] - 2c - (1+b)[m/11] \pmod 7 \quad (\mathbf{1})$$

- *j* is the day of the month
- *m* is the month number (march being the first month)
- c is the hundreds of the year
- *a* is the year in the century
- *b=1* for bissextile years

The return value is a number from 0 to 6, 0 corresponding to Sunday.

The extracted day is then mapped to the same numbering system and the absolute day is given after calculation of the distance between the two days. The computation is different whether the extracted day is situated before or after the publication day.

$$\text{DAY\_AFTER} : d = (\text{extracted\_day} - \text{publication\_day}) \pmod 7 \quad (\mathbf{2})$$

$$\text{DAY\_BEFORE} : d = -((\text{publication\_day} - \text{extracted\_day}) \pmod 7)$$

5.3 Weekends

Dates referring to weekends (subtypes WEEKEND_BEFORE and WEEKEND_AFTER) are problematic as they correspond to 2 days. Given that we want to give a unique absolute date to each extracted reference, we have to decide between the 2 weekend days. Our decision was to take the day nearest to the publication day in order to get the minimum distance between the two dates. As a consequence, the extracted references "last weekend" and "next weekend" will respectively refer to "last Sunday" and "next Saturday". The exact distance to the publication date is computed using again the Zeller formula on publication date and a simple difference between the two normalized dates.

5.4 Explicit and Absolute Dates

Some temporal references are already given in the article in their absolute date form: (day of the week +) day + month + year, e.g. "January the 5th 2008". These do not require any specific transformation.

Some others are formed without an explicit mention of the year (subtypes DATE_BEFORE and DATE_AFTER). We made the hypothesis that the absence of the year implies a time distance of less than a year to the publication date. As a consequence, the absolute date is computed by considering the difference between days and month for both dates.

6 Giving an Absolute Date to Newspaper Articles

Given that we have computed an absolute date for each extracted temporal reference, the objective is now to give a unique date to the corresponding article. The case where only one date has been extracted is trivial as the extracted date is assigned to the article. The problem is harder when for an article comes several or no dates.

6.1 Choosing between Multiple Dates

When several dates have been extracted from the article, we decided to assign the date nearest to the publication date. We assume here that previous dates are used to put the context and later ones to give a deadline. The nearest date is usually the reference that triggers off the recounted event.

Another problem persists: two different dates can be extracted and have the same time distance to the publication date (a previous one and a later one). To follow the hypothesis previously put forward, we assume that the previous date refers to the context of the event while the later one is likely to represent the event to come. We thus chose to assign to the article the later one.

6.2 Absence of Extracted Dates

The absence of extracted temporal references is a definitive obstacle to article dating. This does not necessarily mean that the article is not dated. Some or more cases could have not been predicted in our study. We only can say that we did not found any explicit reference in order to date the article with precision.

The exact dating is thus excluded but there are some key elements that could provide some information about the article. Verb tenses are precious temporal references. Present and future tenses refer to an up-to-date event. The past tense is trickier. If the first sentence uses the present perfect, the following not present perfect verb will determine if the article should be located in the past or in the future.

There is no perfect solution in the absence of exact dating to decide whether the article is up-to-date or not.

7 Evaluation on Four French Local Newspapers

The evaluation of our approach is done on a large corpus of four French local newspapers. The objective of the evaluation is two-fold: First, we intend to give a quantitative evaluation about the performances of our extraction technology. The highlight is put on the impact on article dating results with a special focus on error analysis. We then propose a task-oriented comparison to our standard date extraction technology.

7.1 Corpus Presentation

The test corpus is made of 4 subcorpora totalizing 71942 articles. Each subcorpus corresponds to a local newspaper from a different French region. The articles are in the XML format. All articles are well structured and the textual content of the news is well delimited.

Table 3. Corpus of 4 French local newspapers

	Number of documents	Total size (in Mo)
Newspaper A	6202	14.2
Newspaper B	12809	21.5
Newspaper C	19601	38.8
Newspaper D	33330	83
Total	**71942**	**157.5**

Only the smallest of the subcorpora, newspaper A, served as development corpus for the establishment of the extraction patterns.

7.2 Extraction Results

Table 4 lists the distribution of subtype extractions in the different newspapers. In spite of some local disparities in the distribution of subtype extractions, our results show certain regularity.

The subtype GEN_DATE is the most represented. As this subtype corresponds to a date not precise enough to give an absolute date to the extracted reference, this means that, on average, one third of extracted references do not allow to date the article.

The other extracted references correspond to precise dates and are shared out evenly between dates referring to the past and dates referring to the future, with a short lead for previous dates.

Table 4. Distribution of subtypes extractions in the corpus (in %)

	News A	News B	News C	News D
GEN_DATE	30.7	31.1	40.8	43.6
COMPLETE_DATE	1.9	2.4	2.6	3.5
TODAY	1.8	2.2	5.3	3.7
YESTERDAY	10.4	4.7	9.9	13.3
DAY_BEFORE_YESTERDAY	0.1	0.1	0.1	0.1
TOMORROW	4.7	1.4	5.8	3
DAY_AFTER_TOMORROW	0.1	0.1	0.1	0.1
DAY	0.4	1.5	1.2	0.5
DAY_BEFORE	25.2	21.8	8.4	10
DAY_AFTER	12.8	8.8	5.8	5.4
WEEKEND_BEFORE	0.9	1.3	0.5	0.6
WEEKEND_AFTER	0.5	0.8	0.6	0.5
DATE_BEFORE	5	10.9	8.2	7.4
DATE_AFTER	5.7	13	11.1	8.2

7.3 Dating of the Articles

If we exclude newspaper A that was used to develop the transducer, we observe that 30% to 40% of the articles are undated (see Table 5). These results are not surprising as approximately one third of the extracted dates do not refer to absolute dates.

Nevertheless, results show that undated articles can be divided into two categories: articles without extracted dates and articles with imprecise extracted dates. The case of the absence of dates has previously been developed (see 4.2), so we will say no more about it.

Table 5. Distribution of undated articles

	Articles with no dates	Articles with no precise dates	Total of undated articles
News A	2.9 %	2.4 %	5.3 %
News B	23.4 %	7.6 %	31 %
News C	22.1 %	20.7 %	42.8 %
News D	18.4 %	18.3 %	36.7 %

Imprecise dates. When we took a closer look into the results, we noticed that there are roughly two major patterns where the system was unable to compute an absolute date. The first corresponds to a sequence of four figures and is meant to match any year. The second matches months introduced by “in”. These references are not intended to give an absolute date but are key elements to temporally situate the date in the present or in the past

7.4 Analysis of Extraction Errors

Some extraction patterns have a tendency to be error-prone. The first unreliable pattern tries to match all occurrences of years i.e. every sequence of four figures (e.g. “1324”). The problem detected for this pattern is that we have not set any time window and almost all sequences of 4 figures are extracted. Tests on the corpus show that on average 50% of these matches do not refer to a plausible year (before year 1950 and after year 2015).

The use of verbal tenses is another source of errors that is much more difficult to comprehend. The extraction technique does not use syntactic analysis. As a consequence, the verb chosen to temporally locate the event is not always the right one. The extraction of subtypes DATE_BEFORE and DATE_AFTER may thus be erroneous. Analysis on the corpus reveals that about 15% of those extractions give the wrong position to the publication date.

Even if those problems have to be corrected, we have noticed no consequence on article dating. In fact, extractions of years do not refer to absolute dates and are thus not used for article dating. The errors on verbal tenses may be more harmful but our distance calculation makes up for it and always gives the exact difference between extracted date and publication date when complete dates are extracted, which is the case here.

7.5 Comparison with our Standard Date Extraction

A comparison of our approach to date articles and the use of standard date extraction has been achieved to highlight the differences between the two techniques and explain the necessity to develop our article dating methodology.

The number of extractions resulting from both approaches is shown in Table 6. The total amount of extractions is equivalent but the nature of the extracted references is significantly different as shown by the number of common extractions.

Table 6. Comparison of extractions between standard date and article dating extractions

	Number of extractions with standard date automaton	Number of extractions with article dating automaton	Number of common extractions
News A	28811	25448	16825
News B	44847	27378	18653
News C	58043	49210	26785
News D	108994	107015	59050

Extraction differences. The differences between the two extraction techniques are of two natures. On the one hand, some subtypes are not recognized by the "standard" approach, e.g. DAY and WEEKEND_BEFORE. The non extracted subtypes generally correspond to complex structures. On the other hand, the "standard dates" not extracted by the article dating transducer are mainly references that cannot be converted into absolute dates and have therefore been deliberately excluded from recognition.

Dating usability. The major difference between the two approaches lies in the capacity to locate the extracted references. The article dating approach can not only extract temporal references but also situate them in respect to the publication date. This information is not provided by the standard date extraction technique which makes it unusable for our purpose.

8 Conclusion and Perspectives

In this paper, we have studied the impact of temporal reference extraction on an article dating task. The results show that, when present, any extracted reference can assign an absolute date to the article corresponding to the moment when the related event takes place. In spite of the good precision of the technique, the amount of undated articles is still a problem. An interesting lead, not available in our corpora, would have been to select articles only written by local correspondents to get better results.

For genericity concerns, one would expect the use of handcrafted rules to give somehow more corpus independent results than with a learning-based method. Our results show that even handcrafted rules are corpus-dependant, since the difference between the test corpus and the other corpora was too significant to be a coincidence.

It would be most interesting to compare the results of both methods on several corpora.

To enhance the precision of our approach, the use of syntactic analysis instead of the morpho-syntactic one has to be tested. It would be of great interested to select the exact dates referring to the related event. Finally, not all references should be considered at the same level. An in-depth study of the type of speech containing a temporal reference is needed to get a better analysis of what reference to extract.

References

1. Proceedings of the Message Understanding Conference 7, MUC-7, http://www-nlpir.nist.gov/related_projects/muc/proceedings/ne_english_score_report.html
2. Mota, C., Santos, D.: Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: Actas do Encontro do Segundo HAREM (Aveiro, 7 de Setembro de 2008). Linguateca (2008).
3. ACE, http://www.nist.gov/speech/tests/ace/
4. Poibeau, T., Kosseim, L.: Proper Name Extraction from Non-Journalistic Texts. In W. Daelemans, K. Sima'an, J. Veenstra and J. Zavrel (eds.) Computational Linguistics in the Netherlands. Selected Papers from the Eleventh CLIN Meeting, p. 144-157, Amsterdam/New York (2001)
5. Maynard, D., Tablan, V., Ursu, C., Cunningham, H., Wilks, Y.: Named entity recognition from diverse text types. In Recent Advances in Natural Language Processing 2001 Conference. Tzigov Chark, Bulgaria (2001)
6. Ferro, L., Gerber, L., Mani, I., Sundheim, B. and Wilson G. *TIDES 2005 Standard for the Annotation of Temporal Expressions*. April 2005, Updated September 2005 (2005)
7. Mani, I., Wilson, G.: Robust temporal processing of news. In Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, pp. 69-76. Morristown, NJ (2000)
8. Llidó, D., Llavori, R. B., and Cabo, M. J.: Extracting Temporal References to Assign Document Event-Time Periods. In *Proceedings of the 12th international Conference on Database and Expert Systems Applications* (September 03 - 05, 2001). H. C. Mayr, J. Lazanský, G. Quirchmayr, and P. Vogel, Eds. Lecture Notes In Computer Science, vol. 2113, pp. 62-71. Springer-Verlag, London (2001)
9. Pustejovsky J., Ingria R., Sauri R., Castano J., Littman J., Gaizauskas R., Setzer A., Katz G. & Mani I: The specification language TimeML. In I. Mani, J. Pustejovsky & R. Gaizauskas, Eds., The Language of Time: A Reader. Oxford University Press (2005).
10. Pustejovsky, J., Knippen, R., Littman, J. Saurí, R. Temporal and Event Information in Natural Language Text. Computers and the Humanities, Volume 39, Numbers 2-3, May 2005, pp. 123-164(42). Springer (2007)