

Finding Salient Dates for Building Thematic Timelines

Rémy Kessler
LIMSI-CNRS
Orsay, France
kessler@limsi.fr

Xavier Tannier
Univ. Paris-Sud,
LIMSI-CNRS
Orsay, France
xtannier@limsi.fr

Caroline Hagège
Xerox Research Center Europe
Meylan, France
hagege@xrce.xerox.com

Véronique Moriceau
Univ. Paris-Sud, LIMSI-CNRS
Orsay, France
moriceau@limsi.fr

André Bittar
Xerox Research Center Europe
Meylan, France
bittar@xrce.xerox.com

Abstract

We present an approach for detecting salient (important) dates in texts in order to automatically build event timelines from a search query (e.g. the name of an event or person, etc.). This work was carried out on a corpus of newswire texts in English provided by the Agence France Presse (AFP). In order to extract salient dates that warrant inclusion in an event timeline, we first recognize and normalize temporal expressions in texts and then use a machine-learning approach to extract salient dates that relate to a particular topic. We focused only on extracting the dates and not the events to which they are related.

1 Introduction

Our aim here was to build thematic timelines for a general domain topic defined by a user query. This task, which involves the extraction of important events, is related to the tasks of *Retrospective Event Detection* (Yang et al., 1998), or *New Event Detection*, as defined for example in Topic Detection and Tracking (TDT) campaigns (Allan, 2002).

The majority of systems designed to tackle this task make use of textual information in a bag-of-words manner. They use little temporal information, generally only using document metadata, such as the document creation time (DCT). The few systems that do make use of temporal information (such as the now discontinued Google timeline), only extract absolute, full dates (that feature a day, month and year). In our corpus, described in Section 3.1, we found that only 7% of extracted temporal expressions are absolute dates.

We distinguish our work from that of previous researchers in that we have focused primarily on extracted temporal information as opposed to other textual content. We show that using linguistic temporal processing helps extract important events in texts. Our system extracts a maximum of temporal information and uses only this information to detect salient dates for the construction of event timelines. Other types of content are used for initial thematic document retrieval. Output is a list of dates, ranked from most important to least important with respect to the given topic. Each date is presented with a set of relevant sentences.

We can see this work as a new, easily evaluable task of “date extraction”, which is an important component of timeline summarization.

In what follows, we first review some of the related work in Section 2. Section 3 presents the resources used and gives an overview of the system. The system used for temporal analysis is described in Section 4, and the strategy used for indexing and finding salient dates, as well as the results obtained, are given in Section 5¹.

2 Related Work

The ISO-TimeML language (Pustejovsky et al., 2010) is a specification language for manual annotation of temporal information in texts, but, to the best of our knowledge, it has not yet actually been used in information retrieval systems. Neverthe-

¹This work has been partially funded by French National Research Agency (ANR) under project Chronolines (ANR-10-CORD-010). We would like to thank the French News Agency (AFP) for providing us with the corpus.

less, (Alonso et al., 2007; Alonso, 2008; Kanhabua, 2009) and (Mestl et al., 2009), among others, have highlighted that the analysis of temporal information is often an essential component in text understanding and is useful in a wide range of information retrieval applications. (Harabagiu and Bejan, 2005; Saquete et al., 2009) highlight the importance of processing temporal expressions in Question Answering systems. For example, in the TREC-10 QA evaluation campaign, more than 10% of questions required an element of temporal processing in order to be correctly processed (Li et al., 2005a). In multi-document summarization, temporal processing enables a system to detect redundant excerpts from various texts on the same topic and to present results in a relevant chronological order (Barzilay and Elhadad, 2002). Temporal processing is also useful for aiding medical decision-making. (Kim and Choi, 2011) present work on the extraction of temporal information in clinical narrative texts. Similarly, (Jung et al., 2011) present an end-to-end system that processes clinical records, detects events and constructs timelines of patients' medical histories.

The various editions of the TDT task have given rise to the development of different systems that detect novelty in news streams (Allan, 2002; Kumaran and Allen, 2004; Fung et al., 2005). Most of these systems are based on statistical bag-of-words models that use similarity measures to determine proximity between documents (Li et al., 2005b; Brants et al., 2003). (Smith, 2002) used spatio-temporal information from texts to detect events from a digital library. His method used place/time collocations and ranked events according to statistical measures.

Some efforts have been made for automatically building textual and graphical timelines. For example, (Allan et al., 2001) present a system that uses measures of pertinence and novelty to construct timelines that consist of one sentence per date. (Chieu and Lee, 2004) propose a similar system that extracts events relevant to a query from a collection of documents. Important events are those reported in a large number of news articles and each event is constructed according to one single query and represented by a set of sentences. (Swan and Allen, 2000) present an approach to generating graphical timelines that involves extracting clusters of noun phrases and named entities. More recently, (Yan et

al., 2011b; Yan et al., 2011a) used a summarization-based approach to automatically generate timelines, taking into account the evolutionary characteristics of news.

3 Resources and System Overview

3.1 AFP Corpus

For this work, we used a corpus of newswire texts provided by the AFP French news agency. The English AFP corpus is composed of 1.3 million texts that span the 2004-2011 period (511 documents/day in average and 426 millions words). Each document is an XML file containing a title, a date of creation (DCT), set of keywords, and textual content split into paragraphs.

3.2 AFP Chronologies

AFP "chronologies" (textual event timelines) are a specific type of articles written by AFP journalists in order to contextualize current events. These chronologies may concern any topic discussed in the media, and consist in a list of dates (typically between 10 and 20) associated with a text describing the related event(s). Figure 1 shows an example of such a chronology. Further examples are given in Figure 2. We selected 91 chronologies satisfying the following constraints:

- All dates in the chronologies are between 2004 and 2011 to be sure that the related events are described in the corpus. For example, "*Chronology of climax to Vietnam War*" was excluded because its corresponding dates do not appear in the content of the articles.
- All dates in the chronology are anterior to the chronology's creation date. For example, the chronology "*Space in 2005: A calendar*", published in January 2005 and listing scheduled events, was not selected (because almost no rocket launches finally happened on the expected day).
- The temporal granularity of the chronology is the day. For example, "*A timeline of how the London transport attacks unfolded*", relating the events hour by hour, is not in our focus.

```

<NewsML Version="1.2">
  <NewsItem xml:lang="en">
    <HeadLine>Key dates in Thailand's political crisis</HeadLine>
    <DateId>20100513T100519Z</DateId>
    <NameLabel>Thailand-politics</NameLabel>
    <DataContent>
      <p>The following is a timeline of events since the protests began, soon after Thailand's Supreme Court confiscated 1.4 billion dollars of Thaksin's wealth for abuse of power.</p>
      <p>March 14: Tens of thousands of Red Shirts demonstrate in the capital calling for Abhisit's government to step down, [...]</p>
      <p>March 28: The government and the Reds enter into talks but hit a stalemate after two days [...]</p>
      <p>April 3: Tens of thousands of protesters move from Bangkok's historic district into the city's commercial heart [...]</p>
      <p>April 7: Abhisit declares state of emergency in capital after Red Shirts storm parliament.</p>
      <p>April 8: Authorities announce arrest warrants for protest leaders.</p>
      ...
    </DataContent>
  </NewsItem>
</NewsML>

```

Figure 1: Example of an AFP manual chronology.

For learning and evaluation purposes, all chronologies were converted to a single XML format. Each document was manually associated with a user search query made up of the keywords required to retrieve the chronology.

3.3 System Overview

Figure 3 shows the general architecture of the system. First, pre-processing of the AFP corpus tags and normalizes temporal expressions in each of the articles (step ① in the Figure). Next, the corpus is indexed by the Lucene search engine² (step ②).

Given a query, a number of documents are retrieved by Lucene (③). These documents can be filtered (④), and dates are extracted from the remaining documents. These dates are then ranked in order to show the most important ones to the user (⑤), to-

²<http://lucene.apache.org>

- Chronology of 18 months of trouble in Ivory Coast
- Chechen rebels' history of hostage-takings
- Iraqi political wrangling since March 7 election
- Athletics: Timeline of men's 800m world record
- Major accidents in Chinese mines
- Space in 2005: A calendar
- Developments in Iranian nuclear standoff
- Chronology of climax to Vietnam War
- Timeline of ex-IMF chief's sex attack case
- A timeline of how the London transport attacks unfolded

Figure 2: Examples of AFP chronologies.

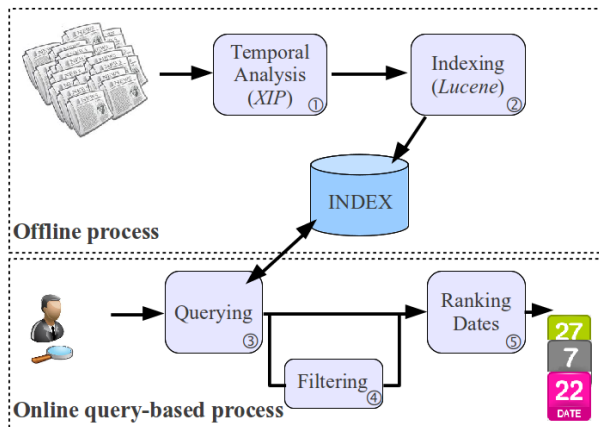


Figure 3: System overview.

gether with the sentences that contain them.

4 Temporal and Linguistic Processing

In this section, we describe the linguistic and temporal information extracted during the pre-processing phase and how the extraction is carried out. We rely on the powerful linguistic analyzer XIP (Aït-Mokhtar et al., 2002), that we adapted for our purposes.

4.1 XIP

The linguistic analyzer we use performs a deep syntactic analysis of running text. It takes as input XML files and analyzes the textual content enclosed in the various XML tags in different ways that are specified in an XML guide (a file providing instructions to the parser, see (Roux, 2004) for details). XIP performs complete linguistic processing ranging from tokenization to deep grammatical dependency analysis. It also performs named entity recog-

nition (NER) of the most usual named entity categories and recognizes temporal expressions. Linguistic units manipulated by the parser are either terminal categories or chunks. Each of these units is associated with an attribute-value matrix that contains the unit's relevant morphological, syntactic and semantic information. Linguistic constituents are linked by oriented and labelled n-ary relations denoting syntactic or semantic properties of the input text. A Java API is provided with the parser so that all linguistic structures and relations can be easily manipulated by Java code.

In the following subsections, we give details of the linguistic information that is used for the detection of salient dates.

4.2 Named Entity Recognition

Named Entity (NE) Recognition is one of the outputs provided by XIP. NEs are represented as unary relations in the parser output. We used the existing NE recognition module of the English grammar which tags the following NE types: **location names**, **person names** and **organization names**. Ambiguous NE types (ambiguity between type **location** or **organization** for country names for instance) are also considered.

4.3 Temporal Analysis

A previous module for temporal analysis was developed and integrated into the English grammar (Hagège and Tannier, 2008), and evaluated during TempEval campaign (Verhagen et al., 2007). This module was adapted for tagging salient dates. Our goal with temporal analysis is to be able to tag and normalize³ a selected subset of temporal expressions (TEs) which we consider to be relevant for our task. This subset of expressions is described in the following sections.

4.3.1 Absolute Dates

Absolute dates are dates that can be normalized without external or contextual knowledge. This is the case, for instance, of “*On January 5th 2003*”. In these expressions, all information needed for normalization is contained in the linguistic expression.

³We call *normalization* the operation of turning a temporal expression into a formatted, fully specified representation. This includes finding the absolute value of relative dates.

However, absolute dates are relatively infrequent in our corpus (7%), so in order to broaden the coverage for the detection of salient dates, we decided to consider relative dates, which are far more frequent.

4.3.2 DCT-relative Dates

DCT-relative temporal expressions are those which are relative to the creation date of the document. This class represents 40% of dates extracted from the AFP corpus. Unlike the absolute dates, the linguistic expression does not provide all the information needed for normalization. External information is required, in particular, the date which corresponds to the moment of utterance. In news articles, this is the DCT. Two sub-classes of relative TEs can be distinguished. The first sub-class only requires knowledge of the DCT value to perform the normalization. This is the case of expressions like *next Friday*, which correspond to the calendar date of the first Friday following the DCT. The second sub-class requires further contextual knowledge for normalization. For example, *on Friday* will correspond either to *last Friday* or to *next Friday* depending on the context where this expression appears (e.g. *He is expected to come on Friday* corresponds to *next Friday* while *He arrived on Friday* corresponds to *last Friday*). In such cases, the tense of the verb that governs the TE is essential for normalization. This information is provided by the linguistic analysis carried out by XIP.

4.3.3 Underspecified Dates

Considering the kind of corpus we deal with (news), we decided to consider TEs whose granularity is at least equal to a day. As a result, TEs were normalized to a numerical YYYYMMDD format (where YYYY corresponds to the year, MM to the month and DD to the day). In case of TEs with a granularity superior to the day or month, DD and MM fields remain unspecified accordingly. However, these underspecified dates are not used in our experiments.

4.4 Modality and Reported Speech

An important issue that can affect the calculation of salient dates is the modality associated with timestamped events in text. For instance, the status of a salient date candidate in a sentence like “*The meet-*

ing takes place on Friday” has to be distinguished from the one in “The meeting should take place on Friday” or “The meeting will take place on Friday, Mr. Hong said”. The time-stamped event *meeting takes place* is factual in the first example and can be taken as granted. In the second and third examples, however, the event does not necessarily occur. This is expressed by the modality introduced by the modal auxiliary *should* (second example), or by the use of the future tense or reported speech (third example). We annotate TEs with information regarding the factuality of the event they modify. More specifically, we consider the following features:

Events that are mentioned in the future: If a time-stamped event is in the future tense, we add a specific attribute MODALITY with value FUTURE to the corresponding TE annotation.

Events used with a modal verb: If a time-stamped event is introduced by a modal verb such as *should* or *would*, then attribute MODALITY to the corresponding TE annotation has the value MODAL.

Reported speech verbs: Reported speech verbs (or verbs of speaking) introduce indirect or reported speech. We dealt with time-stamped events governed by a reported speech verb, or otherwise appearing in reported speech. Once again, XIP’s linguistic analysis provided the necessary information, including the marking of reported speech verbs and clause segmentation of complex sentences. If a relevant TE modifies a reported speech verb, the annotation of this TE contains a specific attribute, DECLARATION=“YES”. If the relevant TE modifies a verb that appears in a clause introduced by a reported speech verb then the annotation contains the attribute REPORTED=“YES”.

Note that the different annotations can be combined (e.g. modality and reported speech can occur for a same time-stamped event). For example, the TE *Friday* in “The meeting should take place on Friday, Mr. Hong said” is annotated with both modality and reported speech attributes.

4.5 Corpus-dependent Special Cases

While we developed the linguistic and temporal annotators, we took into account some specificities of our corpus. We decided that the TEs *today* and

```
<DCT value="20050105"/>
<EC TYPE="TIMEX" value="unknown">The year
2004</EC> was the deadliest <EC TYPE="TIMEX"
value="unknown">in a decade</EC> for journalists
around the world, mainly because of the number of reporters
killed in <EC TYPE="LOCORG">Iraq</EC>, the
media rights group <EN TYPE="ORG">Reporters
Sans Frontieres</EN> (Reporters Without Bor-
ders) said <EC TYPE="DATE" SUBTYPE="REL"
REF="ST" DECLARATION="YES" value
="20050105">Wednesday</EC>.
```

Figure 4: Example of XIP output for a sample article.

now were not relevant for the detection of salient dates. In the AFP news corpus, these expressions are mostly generic expressions synonymous with *nowadays* and do not really time-stamp an event with respect to the DCT. Another specificity of the corpus is the fact that if the DCT corresponds to a Monday, and if an event in a past tense is described with the associated TE *on Monday* or *Monday*, it means that this event occurs on the DCT day itself, and not on the Monday before. We adapted the TE normalizer to these special cases.

4.6 Implementation and Example

As said previously, a NER module is integrated into the XIP parser, which we used “as is”. The TE tagger and normalizer was adapted from (Hagège and Tannier, 2008). We used the Java API provided with the parser to perform the annotation and normalization of TEs. The output for the linguistic and temporal annotation consists in XML files where only selected information is kept (structural information distinguishing headlines from news content, DCT), and enriched with the linguistic annotations described before (NEs and TEs with relevant attributes corresponding to the normalization and typing). Information concerning modality, future tense and reported speech, appears as attributes on the TE tag. Figure 4 shows an example of an analyzed excerpt of a news article.

In this news excerpt, only one TE (*Wednesday*) is normalized as both *The year 2004* and *in a decade* are not considered to be relevant. The first one being a generic TE and the second one being of granularity superior to a year. The annotation of the relevant TE has the attribute indicating that it time-stamps an event realized by a reported speech verb. The nor-

malized value of the TE corresponds to the 5th of January 2005, which is a Wednesday. NEs are also annotated.

In the entire AFP corpus, 11.5 millions temporal expressions were detected, among which 845,000 absolute dates (7%) and 4.6 millions normalized relative dates (40%). Although we have not yet evaluated our tagging of relative dates, the system on which our current date normalization is based achieved good results in the TempEval (Verhagen et al., 2007) campaign.

5 Experiments and Results

In Section 5.1, we propose two baseline approaches in order to give a good idea of the difficulty of the task (Section 5.4 also discusses this point). In Section 5.2, we present our experiments using simple filtering and statistics on dates calculated by Lucene. Finally, Section 5.3 gives details of our experiments with a learning approach. In our experiments, we used three different values to rank dates:

- $occ(d)$ is the number of textual units (documents or sentences) containing the date d .
- Lucene provides ranked documents together with their relevance score. $luc(d)$ is the sum of Lucene scores for textual units containing the date d .
- An adaptation of classical $tf.idf$ for dates:

$$tf.idf(d) = f(d) \cdot \log \frac{N}{df(d)}$$

where $f(d)$ is the number of occurrences of date d in the sentence (generally, $f(d) = 1$), N is the number of indexed sentences and $df(d)$ is the number of sentences containing date d .

In all experiments (including baselines), timelines have been built by considering only dates between the first and the last dates of the corresponding manual chronology. Processing runs were evaluated on manually-written chronologies (see Section 3.2) according to Mean Average Precision (MAP), which is a widely accepted metric for ranked lists. MAP gives a higher weight to higher ranked elements than lower ranked elements. Significance of evaluation results are indicated by the p -value results of the Student’s t-test ($t(90) = 1.9867$).

Baselines “only DCTs”			
Model	BL_{DCT}^{occ}	BL_{DCT}^{luc}	$BL_{DCT}^{tf.idf}$
MAP Score	0.5036	0.5521	0.5523
Baselines “only absolute dates”			
Model	BL_{abs}^{occ}	BL_{abs}^{luc}	$BL_{abs}^{tf.idf}$
MAP Score	0.2627	0.2782	0.2778
Baselines “absolute dates or alternatively DCTs”			
Model	BL_{mix}^{occ}	BL_{mix}^{luc}	$BL_{mix}^{tf.idf}$
MAP Score	0.4005	0.4110	0.4135

Table 1: MAP results for baseline runs.

5.1 Baseline Runs

BL_{DCT} . Indexing and search were done at **document level** (*i.e.* each AFP article, with its title and keywords, is a document). Given a query, the top 10,000 documents were retrieved. In these runs, **only the DCT** for each document was considered. Dates were ranked by one of the three values described above (occ , luc or $tf.idf$) leading to runs BL_{DCT}^{occ} , BL_{DCT}^{luc} and $BL_{DCT}^{tf.idf}$.

BL_{abs} . Indexing and search were done at **sentence level** (document title and keywords are added to sentence text). Given a query, the top 10,000 sentences were retrieved. **Only absolute dates** in these sentences were considered. We thus obtained runs BL_{abs}^{occ} , BL_{abs}^{luc} and $BL_{abs}^{tf.idf}$.

Note that in this baseline, as well as in all the subsequent runs, the information unit was the sentence because a date was associated to a small part of the text. The rest of the document generally contained text that was not related to the specific date.

BL_{mix} . Same as BL_{abs} , except that sentences containing no absolute dates were considered and associated to the DCT.

Table 1 shows results for these baseline runs. Using only DCTs with Lucene scores or $tf.idf(d)$ already yielded interesting results, with MAP around 0.55.

5.2 Salient Date Extraction with XIP Results and Simple Filtering

In these experiments, we considered a Lucene index to be built as follows: each document was taken to

Model	MAP Score	Model	MAP Score
<i>Salient date runs with all dates</i>			
SD^{luc}	0.6962	$SD^{tf.idf}$	0.6982
<i>Salient dates runs with filtering</i>			
SD_R^{luc}	0.6975	$SD_R^{tf.idf}$	0.6996
SD_F^{luc}	0.6967	$SD_F^{tf.idf}$	0.6993 **
SD_M^{luc}	0.6978	$SD_M^{tf.idf}$	0.7005 *
SD_D^{luc}	0.7066 **	$SD_D^{tf.idf}$	0.7091 **
SD_{FMD}^{luc}	0.7086 **	$SD_{FMD}^{tf.idf}$	0.7112 **
SD_{RFMD}^{luc}	0.7127 **	$SD_{RFMD}^{tf.idf}$	0.7146 **

Table 2: MAP results for salient date extraction with XIP and simple filtering. The significance of the improvement due to filtering wrt no filtering is indicated by the Student t-test (*: $p < 0.05$ (significant); **: $p < 0.01$ (highly significant)). The improvement due to using $tf.idf(d)$ as opposed to $occ(d)$ is also highly significant.

be a **sentence** containing a **normalized date**. This sentence was indexed with the title and keywords of the AFP article containing it. Given a query, the top 10,000 documents were retrieved. Combinations between the following filtering operations were possible, by removing all dates associated with a reported speech verb (R), a modal verb (M) and/or a future verb (F). All these filtering operations were intended to remove references to events that were not certain, thereby minimizing noise in results.

These processing runs are named SD runs, with indices representing the filtering operations. For example, a run obtained by filtering modal and future verbs is called $SD_{M,F}$. In all combinations, dates were ranked by the sum of Lucene scores for these sentences (luc) or by $tf.idf^4$.

Table 2 presents the results for this series of experiments. MAP values are much higher than for baselines. Using $tf.idf(d)$ is only very slightly better than luc . Filtering operations bring significant improvement but the benefits of these different techniques have to be further investigated.

5.3 Machine-Learning Runs

We used our set of manually-written chronologies as a training corpus to perform machine learning experiments. We used IcsiBoost⁵, an implementa-

⁴We do not present runs where dates are ranked by the number of times they appear in retrieved sentences (occ), as we did for baselines, since results are systematically lower.

⁵<http://code.google.com/p/icsiboost/>

tion of adaptative boosting (AdaBoost (Freund and Schapire, 1997)).

In our approach, we consider two classes: *salient dates* are dates that have an entry in the manual chronologies, while *non-salient dates* are all other dates. This choice does, however, represent an important bias. The choices of journalists are indeed very subjective, and chronologies must not exceed a certain length, which means that relevant dates can be thrown away. These issues will be discussed in Section 5.4.

The classifier instances were not all sentences retrieved by the search engine. Using all sentences would not yield a useful feature set. We rather aggregated all sentences corresponding to the same date before learning the classifier. Therefore, each instance corresponded to a single date, and features were figures concerning the set of sentences containing this date.

Features used in this series of runs are as follows:

1. Features representing the fact that the more a date is mentioned, the more important it is likely to be: 1) Sum of the Lucene scores for all sentences containing the date 2) Number of sentences containing the date 3) Ratio between the total weights of the date and weights of all returned dates 4) Ratio between the frequency of the date and frequency of all returned dates;
2. Features representing the fact that an important event is still written about, a long time after it occurs: 1) Distance between the date and the most recent mention of this date 2) Distance between the date and the DCT;
3. Other features: 1) Lucene's best ranking of the date 2) Number of times where the date is absolute in the text 3) Number of times where the date is relative (but normalized) in the text 4) Total number of keywords of the query in the title, sentence and named entities of retrieved documents 5) Number of times where the date modifies a reported speech verb or is extracted from reported speech.

We did not aim to classify dates, but rather to rank them. Instead, we used the predicted probability $P(d)$ returned by the classifier, and mixed it with the Lucene score of sentences, or with date $tf.idf$:

Model	MAP Score
<i>Machine-Learning Runs</i>	
ML_{base}^{luc}	0.7033
ML^{luc}	0.7905 **
$ML^{tf.idf}$	0.7918 **

Table 3: MAP results for salient date extraction with machine-learning. ML_{base}^{luc} used Lucene scores and only the first set of features described above. ML^{luc} and $ML^{tf.idf}$ used the three sets of features. They are both highly significant under the *t-test* ($p \approx 6.10^{-4}$) wrt respectively SD^{luc} and $SD^{tf.idf}$.

$$score(d) = P(d) \times val(d)$$

where $val(d)$ is either $luc(d)$ or $tf.idf(d)$.

Because the task is very subjective and (above all) because of the low quantity of learning data, we preferred not to opt for a “learning to rank” approach.

We evaluated this approach with a classic 4-fold cross-validation. Our 91 chronologies were randomly divided into 4 sub-samples, each of them being used once as test data. The final scores, presented in Table 3, are the average of these 4 processes. As shown in this table, the learning approach improves MAP results by about 0.05 point.

5.4 Discussion and Final Experiment

Chronologies hand-written by journalists are a very useful resources for evaluation of our system, as they are completely dissociated from our research and are an exact representation of the output we aim to obtain. However, assembling such a chronology is a very subjective task, and no clear method for evaluation agreement between two journalists seems immediately apparent. Only experts can build such chronologies, and calculating this agreement would require at least two experts from each domain, which are hard to come by. One may then consider our system as a useful tool for building a chronology more objectively.

To illustrate this point, we chose four specific topics⁶ and showed one of our runs on each topic to an AFP expert for these subjects. We asked him to assess the first 30 dates of these runs.

⁶Namely, “Arab revolt timeline for Morocco”, “Kyrgyzstan unrest timeline”, “Lebanon’s new government: a timeline”, “Libya timeline”.

Topic	AP_C	AP_E
<i>Morocco</i>	0.5847	0.5718
<i>Kyrgyzstan</i>	0.6125	0.9989
<i>Libya</i>	0.7856	1
<i>Lebanon</i>	0.4673	0.7652

Table 4: Average precision results for manual evaluation on 4 topics, against the original chronologies (AP_C), and the expert assessment (AP_E).

Table 4 presents results for this evaluation, comparing average precision values obtained 1) against the original, manual chronologies (AP_C), and 2) against the expert assessment (AP_E). These values show that, for 3 runs out of 4, many dates returned by the system are considered as valid by the expert, even if not presented in the original chronology.

Even if this experiment is not strong enough to lead to a formal conclusion (*post-hoc* evaluation with only 4 topics and a single assessor), this tends to show that our system produces usable outputs and that our system can be of help to journalists by providing them with chronologies that are as useful and objective as possible.

6 Conclusion and Future Work

This article presents a task of “date extraction” and shows the importance of taking temporal information into consideration and how with relatively simple temporal processing, we were able to indirectly point to important events using the temporal information associated with these events. Of course, as our final goal consists in the detection of important events, we need to take into account the textual content. In future work, we envisage providing, together with the detection of salient dates, a semantic analysis that will help determine the importance of events. Another interesting direction in which we soon aim to work is to consider all textual excerpts that are associated with salient dates, and use clustering techniques to determine if textual excerpts correspond to the same event or not. Finally, as our news corpus is available both for English and French (comparable corpus, not necessarily translations), we aim to investigate cross-lingual extraction of salient dates and salient events.

References

- Salah Aït-Mokhtar, Jean-Pierre Chanod, and Claude Roux. 2002. Robustness beyond Shallowness: Incremental Deep Parsing. *Natural Language Engineering*, 8:121–144.
- James Allan, Rahul Gupta, and Vikas Khandelwal. 2001. Temporal summaries of new topics. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 10–18.
- James Allan, editor. 2002. *Topic Detection and Tracking*. Springer.
- Omar Alonso, Ricardo Baeza-Yates, and Michael Gertz. 2007. Exploratory Search Using Timelines. In *SIGCHI 2007 Workshop on Exploratory Search and HCI Workshop*.
- Omar Rogelio Alonso. 2008. *Temporal information retrieval*. Ph.D. thesis, University of California at Davis, Davis, CA, USA. Adviser-Gertz, Michael.
- Regina Barzilay and Noemie Elhadad. 2002. Inferring Strategies for Sentence Ordering in Multidocument News Summarization. *Journal of Artificial Intelligence Research*, 17:35–55.
- Thorsten Brants, Francine Chen, and Ayman Farahat. 2003. A system for new event detection. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '03, pages 330–337, New York, NY, USA. ACM.
- Hai Leong Chieu and Yoong Keok Lee. 2004. Query based event extraction along a timeline. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 425–432.
- Yoav Freund and Robert E. Schapire. 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139.
- Gabriel Pui Cheong Fung, Jeffrey Xu Yu, Philip S. Yu, and Hongjun Lu. 2005. Parameter free bursty events detection in text streams. In *VLDB '05: Proceedings of the 31st international conference on Very large data bases*, pages 181–192.
- Caroline Hagège and Xavier Tannier. 2008. XTM: A Robust Temporal Text Processor. In *Computational Linguistics and Intelligent Text Processing, proceedings of 9th International Conference CICLing 2008*, pages 231–240, Haifa, Israel, February. Springer Berlin / Heidelberg.
- Sanda Harabagiu and Cosmin Adrian Bejan. 2005. Question Answering Based on Temporal Inference. In *Proceedings of the Workshop on Inference for Textual Question Answering*, Pittsburg, Pennsylvania, USA, July.
- Hyuckchul Jung, James Allen, Nate Blaylock, Will de Beaumont, Lucian Galescu, and Mary Swift. 2011. Building timelines from narrative clinical records: initial results based-on deep natural language understanding. In *Proceedings of BioNLP 2011 Workshop*, BioNLP '11, pages 146–154, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Nattiya Kanhabua. 2009. Exploiting temporal information in retrieval of archived documents. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2009, Boston, MA, USA, July 19–23, 2009, page 848.
- Youngho Kim and Jinwook Choi. 2011. Recognizing temporal information in korean clinical narratives through text normalization. *Health Inform Res*, 17(3):150–5.
- Giridhar Kumaran and James Allen. 2004. Text classification and named entities for new event detection. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 297–304. ACM.
- Wei Li, Wenjie Li, Qin Lu, and Kam-Fai Wong. 2005a. A Preliminary Work on Classifying Time Granularities of Temporal Questions. In *Proceedings of Second international joint conference in NLP (IJCNLP 2005)*, Jeju Island, Korea, oct.
- Zhiwei Li, Bin Wang, Mingjing Li, and Wei-Ying Ma. 2005b. A Probabilistic Model for Restrospective News Event Detection. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Salvador, Brazil. ACM Press, New York City, NY, USA.
- Thomas Mestl, Olga Cerrato, Jon Ølnes, Per Myrseth, and Inger-Mette Gustavsen. 2009. Time Challenges - Challenging Times for Future Information Search. *D-Lib Magazine*, 15(5/6).
- James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. Iso-timeml: An international standard for semantic annotation. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).
- Claude Roux. 2004. Annoter les documents XML avec un outil d'analyse syntaxique. In *11ème Confrence annuelle de Traitement Automatique des Langues Naturelles*, Fès, Morocco, April. ATALA.

- Estela Saquete, Jose L. Vicedo, Patricio Martínez-Barco, Rafael Muñoz, and Hector Llorens. 2009. Enhancing QA Systems with Complex Temporal Question Processing Capabilities. *Journal of Artificial Intelligence Research*, 35:775–811.
- David A. Smith. 2002. Detecting events with date and place information in unstructured text. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 191–196, New York, NY, USA. ACM.
- Russell Swan and James Allen. 2000. Automatic generation of overview timelines. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '00*, pages 49–56, New York, NY, USA. ACM.
- Marc Verhagen, Robert Gaizauskas, Franck Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. SemEval-2007 - 15: TempEval Temporal Relation Identification. In *Proceedings of SemEval workshop at ACL 2007*, Prague, Czech Republic, June. Association for Computational Linguistics, Morristown, NJ, USA.
- Rui Yan, Liang Kong, Congrui Huang, Xiaojun Wan, Xiaoming Li, and Yan Zhang. 2011a. Timeline generation through evolutionary trans-temporal summarization. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, Edinburgh, UK*, pages 433–443.
- Rui Yan, Xiaojun Wan, Jahna Otterbacher, Liang Kong, Xiaoming Li, and Yan Zhang. 2011b. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29, 2011*, pages 745–754.
- Y. Yang, T. Pierce, and J. G. Carbonell. 1998. A study on retrospective and on-line event detection. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, August. ACM Press, New York City, NY, USA.