Question Answering and Multi-Search Engines in Geo-Temporal Information Retrieval

Fernando S. Peregrino, David Tomás and Fernando Llopis

Department of Software and Computing Systems, University of Alicante
Carretera San Vicente del Raspeig s/n - 03690 Alicante (Spain)

Abstract. In this paper we present a complete system for the treatment of both geographical and temporal dimensions in text and its application to information retrieval. This system has been evaluated in both the *GeoTime* task of the *8th* and *9th NTCIR* workshop in the years 2010 and 2011 respectively, making it possible to compare the system to contemporary approaches to the topic. In order to participate in this task we have added the temporal dimension to our *GIR* system. The system proposed here has a modular architecture in order to add or modify features. In the development of this system, we have followed a *QA*-based approach as well as multi-search engines to improve the system performance.

Keywords: Geographical Information Retrieval, Geo-Tagging, Spacial Information, Temporal Information

1 Introduction

Information retrieval (*IR*) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)[**?**].

GIR is a specialization of *IR* with geographic metadata associated. *IR* systems usually see the documents as a collection or "bag of words". By contrast, *GIR* systems require semantic information, i.e. they need a place name or geographical feature associated with the document. Because of this, in *GIR* systems, it is common to separate the analysis and text indexing from the geographic indexing.

Temporal information is available in every document either explicitly, i.e., in the form of temporal expressions, or implicitly in the form of metadata. Recognizing such information and exploiting it for document retrieval and presentation purposes are important features that can significantly improve the functionality of search applications. Temporal Information Retrieval (*TIR*), analogously to *GIR*, is a specialization of Information Retrieval with temporal metadata associated.

The objective of this work is to adopt a first approach in the geo-temporal *IR* field, including the observation of how a basic *IR* system can be improved by embedding geo-temporal *IR* intelligence, and to identify what methods used in them have a better performance.

We have evaluated this approach according to the *GeoTime* task included in both the *NTCIR-8* and *NTCIR-9*[1] workshop. *GeoTime* for the *NTCIR Workshop* is an evaluation of Geographic and Temporal Information Retrieval "*NTCIR GeoTime*". The focus of this task is on searching with Geographic and Temporal constraints[**?**].

To that end, we have elaborated this paper to be structured as follow: In section 2, we provide a general description of the system, describing the topic storage architecture as well as the system operation. Subsequently, section 3 will outline the experiments and evaluations conducted. Finally, in section 4, we describe the conclusions and future work in this area.

2 System Description

For the creation of this *Geo-Temporal IR* system, we have chosen to implement it in a modular fashion with the intention of adding new components, testing and improving the existing ones.

Figure 1 shows the architecture of our system, its component modules and the data flow. This system works in three different phases: the first phase is represented by the solid lines which show the data flow that takes place in preprocessing time. On the other hand, broken lines represent the data flow which takes place in execution time. The second phase is represented by the thicker broken lines, those that process the topic, and the third phase is outlined by thinner broken lines, those which execute the query.

2.1 System Operation

As it was mentioned above, the system operation is divided into three phases: pre-processing and indexing the corpus, processing queries, and running queries.

Corpus Pre-process. Firstly, in this phase the lemmatized corpus is indexed in the search engine module. This module has two functionalities: to index the whole corpus, and to retrieve a set of relevant documents for a given query.

Initially, the search engine chosen for this system was *Lucene*[2]. We have included characteristics to this search engine, such as a stemming and stopword removal. The ranking function *Okapi BM25*[**?**] has been used to rank the results according to their relevance. Finally, it has been chosen to retrieve up to 1,000 relevant documents per query.

On the other hand, whilst the search engine is indexing, *Yahoo! Placemaker* obtains the geographic entities, and *FreeLing* gets the temporal expressions and the rest of named entities of the corpus. With all this information a new *XML* file is made for each corpus article. These *XML* documents will be useful to know the article relevance with respect to the query in the query runtime phase.

[1] `http://metadata.berkeley.edu/NTCIR-GeoTime/description.php`
[2] `http://lucene.apache.org/`

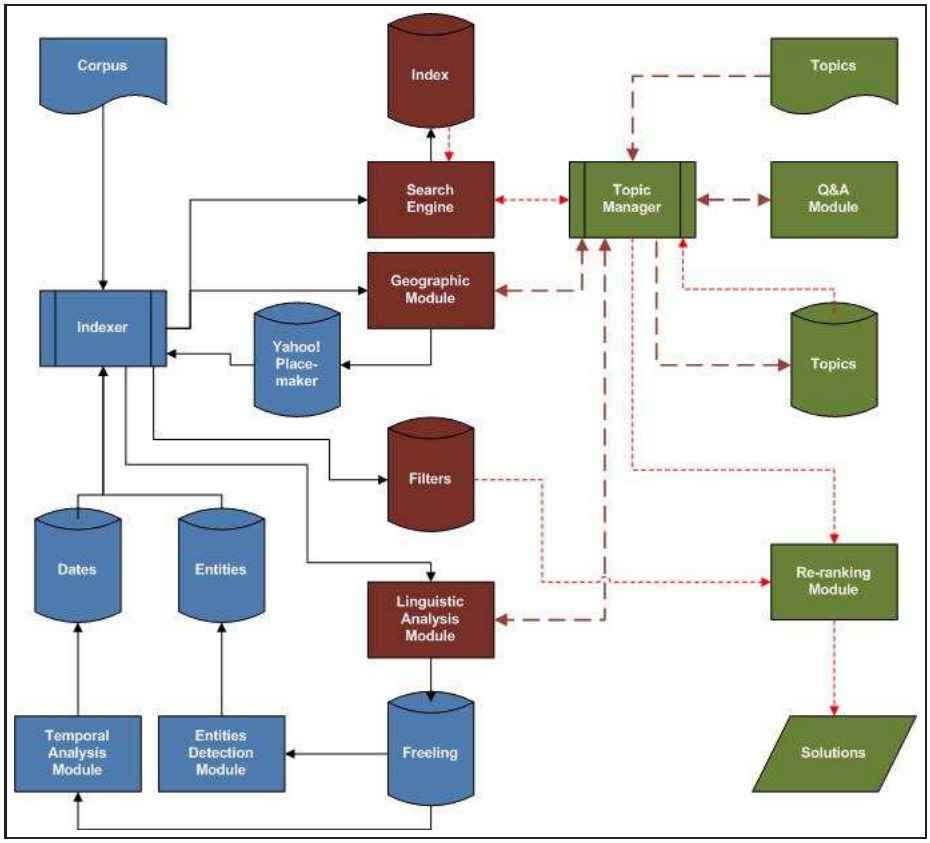

Fig. 1. Diagram of the workflow in the *GIR* system implemented.

Query Process. In this second phase, the topics are sent to the linguistic analysis module and to the *QA* module. Afterwards, our system sends every geographic reference obtained from the two previous modules to the geographic module in order to transform them to the *Yahoo!* unique identifier (*WOEID*). Finally, this data are stored, making a new *XML* file for each topic (this *XML* file is different to that one created for each corpus article). An example of this topic file can be seen in Figure 2, where the following sections can be observed:

- Search terms (<*search*>): all search term without stopwords.
- Lemmatized search terms (<*search_lemma*>): lemmatized search terms section.
- Filters (<*filters*>):
  - Descriptive part (<*description*>): dates, place names, and entities found in the descriptive part of the query.
  - Narrative part (<*narrative*>): Analogous to the previous one and, in addition, it has the geographical and temporal constraints.

```
<query id="GeoTime-0040">
  <search>Concorde crash</search>
  <search_lemma>concorde crash</search_lemma>
  <filters>
    <description>
      <entities>
        <item>concorde</item>
      </entities>
    </description>
    <narrative>
      <entities>
        <item>concorde</item>
      </entities>
      <commons>
        <item>crash</item>
        <item>airliner</item>
      </commons>
    </narrative>
    <yahoo>
      <dates>
        <item weight="1.0">[??:??/??/2000:??.??:??]</item>
        <item weight="0.8043478">[??:25/7/2000:??.??:??]</item>
        <item weight="0.6847826">[??:??/7/2000:??.??:??]</item>
        <item weight="0.1521739">[??:??/??/2003:??.??:??]</item>
        <item weight="0.07608695">[??:??/??/1976:??.??:??]</item>
        <item weight="0.07608695">[??:??/??/1969:??.??:??]</item>
        <item weight="0.06521739">[??:11/9/2001:??.??:??]</item>
        <item weight="0.06521739">[??:2/2/2010:??.??:??]</item>
        <item weight="0.04347826">[??:??/6/2000:??.??:??]</item>
        <item weight="0.04347826">[??:10/4/2003:??.??:??]</item>
      </dates>
      <dates_year>
        <item weight="1.0">[??:??/??/2000:??.??:??]</item>
        <item weight="0.098425195">[??:??/??/2003:??.??:??]</item>
        <item weight="0.08661418">[??:??/??/2010:??.??:??]</item>
        <item weight="0.05905512">[??:??/??/2001:??.??:??]</item>
        <item weight="0.05511811">[??:??/??/1969:??.??:??]</item>
        <item weight="0.03937008">[??:??/??/1976:??.??:??]</item>
        <item weight="0.023622047">[??:??/??/2008:??.??:??]</item>
        <item weight="0.01968504">[??:??/??/2011:??.??:??]</item>
        <item weight="0.007874016">[??:??/??/1985:??.??:??]</item>
        <item weight="0.007874016">[??:??/??/1979:??.??:??]</item>
      </dates_year>
      <dates_month>
        <item weight="1.0">[??:??/7/2000:??.??:??]</item>
        <item weight="0.06849315">[??:??/2/2010:??.??:??]</item>
        <item weight="0.05479452">[??:??/8/2000:??.??:??]</item>
        <item weight="0.047945205">[??:??/3/1969:??.??:??]</item>
        <item weight="0.047945205">[??:??/10/2003:??.??:??]</item>
        <item weight="0.047945205">[??:??/7/2001:??.??:??]</item>
        <item weight="0.04109589">[??:??/9/2001:??.??:??]</item>
        <item weight="0.034246575">[??:??/12/2010:??.??:??]</item>
        <item weight="0.02739726">[??:??/6/2011:??.??:??]</item>
        <item weight="0.02739726">[??:??/4/2003:??.??:??]</item>
      </dates_month>
      <locations>
        <item weight="1.0">615702</item>
        <item weight="0.49312714">23424819</item>
        <item weight="0.3676976"/>
        <item weight="0.1580756"/>
        <item weight="0.10584193">23424977</item>
        <item weight="0.0790378">44418</item>
        <item weight="0.06872852"/>
        <item weight="0.04467354">2384019</item>
        <item weight="0.030927835">24865675</item>
        <item weight="0.02749141">2459115</item>
      </locations>
    </yahoo>
  </filters>
</query>
```


Fig. 2. *XML* topic document sample.

  - ∗ Query expanding (<*commons*>). It has expanded entries of the most representative terms of the query to a possible future query expansion.
- *QA* (<*yahoo*>): It has the following extracted data from *Yahoo!*: dates, year and month dates, year dates, and toponyms. It has normalized the 10 more representative values for all four piece of date aforementioned. This data is obtained from the module of *Question Answering* which tries to obtain from the web geographic and temporal expressions that are relevant to the query. The process to get the expressions is:
  1. The query is sent to *Yahoo! Search BOSS*[3].

[3] *Yahoo! Search BOSS* (Build your Own Search Service) is *Yahoo!'s* open search and data services platform to build web-scale search products that utilize *Yahoo! Search* technology and data (`http://developer.yahoo.com/search/boss/`)

2. *Yahoo! Search BOSS* collects the first 1,000 snippets from the returned results.
3. All dates and places from these snippets are then extracted. In order to do this task, the open source language analysis tool *FreeLing*[4] is used.
4. The total number of occurrences is computed and normalized, obtaining the 10 most relevant for each of the following categories:
   (a) Completed or uncompleted dates (<dates>).
   (b) month and year (<dates_month>).
   (c) year (<dates_year>).
   (d) place names (<locations>). *FreeLing* assigns the same label to both a place name and other named entities, and in order to distinguish between them, we use a list of toponyms obtained from *GeoNames*[5]. Once we have separate the grain from the chaff, the locations are sent to *Yahoo! PlaceMaker* to get the *WOEID*.

Query Runtime. In this third and last phase, the system sends the query, which is the content of the tag <*search_lemma*> in the *XML* topic file (see Figure 2), to the search engine. The search engine returns 1,000 relevant ranked documents. The re-ranking module obtains the *XML* corpus files for each document returned by the search engine and this module re-ranks the documents matching the former rank from the search engine with the *XML* corpus, according to a weight function (the operation of this function is not going to be described here as this exceeds the scope of this paper).

3 Experimentation and Evaluation

In this section, on the one hand, we will describe both the metrics used to evaluate this system and the framework in which the evaluation was carried out. On the other hand, we will analyse the impact of the search engine and *QA* modules on the final results.

3.1 Metrics and Evaluation Framework

In this section, it how the system has been evaluated will be shown and the choice of an evaluation metric will be reasoned.

Evaluation Framework. Firstly, this system was assessed with the document collection of the task *GeoTime* included in the *NTCIR 2010*, which can be seen in [**?**]. The English collection used in this task consisted of 315,417 *New York*

[4] http://nlp.lsi.upc.edu/freeling/
[5] http://www.geonames.org/

Times stories for 2002-2005. Regarding topics, there were 25 which included geographical and temporal constraints, with both a descriptive and a narrative part (e.g. Descriptive part: "*When and where did a volcano erupt in Africa during 2002?*". Narrative part: "*The user would like to know the date in 2002 in which a volcano erupted in Africa. What was the name of the volcano and in which country is it located?*").

Secondly, the system was assessed in the following year task, i.e. the *NTCIR 2011 GeoTime*, which was similar to the previous one, with 25 new topics, and adding three more corpora for 1998-2001: *Mainichi Daily*, *Korea Times* and *Xinhua English*, for a total of 797,216 articles which cover the period 1998 to 2005.

Evaluation Metrics. To assess the result of this geo-temporal *IR* system we have chosen one of the metrics used in *NTCIR-GeoTime*, the *nDCG*[6] (*normalized Discounted Cumulative Gain*) [**?**]. We have chosen this metric because it is capable of doing gradual assessments, it means that not only can it tag a document as a relevant or irrelevant, but it gives a relevance degree. At *NTCIR-GeoTime* this metric was used with three different bases: 10, 100, and 1,000. We have chosen the base 1,000 for this metric (the same as the number of documents that we retrieve for a topic), which means that the function is not taking into account the position of a relevant document, but whether this document is retrieved (*Cumulative Gain*). This has been done because we are focusing on obtaining the biggest percentage of relevant documents rather that in getting an accurate ranking, such as will be shown in future work which will be carried out in the rest of the modules of the system.

3.2 Impact of the Components

In the next sections, the impact that the search engine and the *QA* modules have in the system will be shown and how they can obtain a considerable improvement.

Search Engine. As mentioned in the Section 2.1, this system highly depends on the search engine performance and, therefore, the first experiment carried out dealt with this module. The experiment took place in the *NTCIR 2011* framework, and it was observed that the coverage achieved by *Lucene* was just 55.7892%, so that lead to an experiment to test what would had happened if it had reached a wider coverage, the results of which are shown in Table 1. These results have been classified into three groups:

1. Topics which get a recall between 0% and 100%, all of them.
2. Topics which get a recall between 50% and 100%, 12 out of 25.

[6] *nDCG* measures the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated from the top of the result list to the bottom with the gain of each result discounted at lower ranks.

3. Topics which get a recall between 75% and 100%, 10 out of 25.

In each of these three groups, the percentage of document recall by each topic, and the *nDCG-1000* score achieved for the query can be observed. Finally, the average recall and score for the topics that fall into each of the three groups is obtained. The objective of this experiment was to see what would happen if there had been more recall by the search engine module, and the substantial improvement that could have been achieved can be appreciated in the last two rows of Table 1 (from a score of 0.3959 to 0.5607 or 0.6081, according to the minimum recall required).

Table 1. Recall and *nDCG-1000* scores achieved using only *Lucene* for each *NTCIR 2011* topic

	0% - 100%		50% - 100%		75% - 100%	
Topic	Recall	nDCG	Recall	nDCG	Recall	nDCG
GeoTime-0026	93.2945%	0.7730	93.2945%	0.7730	93.2945%	0.7730
GeoTime-0027	85.7143%	0.2576	85.7143%	0.2576	85.7143%	0.2576
GeoTime-0028	85.4839%	0.5846	85.4839%	0.5846	85.4839%	0.5846
GeoTime-0029	43.3566%	0.2806	-	-	-	-
GeoTime-0030	66.6667%	0.3467	66.6667%	0.3467	-	-
GeoTime-0031	36.6667%	0.2905	-	-	-	-
GeoTime-0032	35.0877%	0.3367	-	-	-	-
GeoTime-0033	74.4186%	0.5660	74.4186%	0.5660	-	-
GeoTime-0034	86.3636%	0.4655	86.3636%	0.4655	86.3636%	0.4655
GeoTime-0035	28.5714%	0.1031	-	-	-	-
GeoTime-0036	31.9149%	0.2849	-	-	-	-
GeoTime-0037	0.0000%	0.0000	-	-	-	-
GeoTime-0038	1.6908%	0.0317	-	-	-	-
GeoTime-0039	84.1202%	0.6174	84.1202%	0.6174	84.1202%	0.6174
GeoTime-0040	82.0755%	0.7887	82.0755%	0.7887	82.0755%	0.7887
GeoTime-0041	98.9362%	0.7117	98.9362%	0.7117	98.9362%	0.7117
GeoTime-0042	1.2739%	0.0145	-	-	-	-
GeoTime-0043	91.4894%	0.5294	91.4894%	0.5294	91.4894%	0.5294
GeoTime-0044	28.5714%	0.1920	-	-	-	-
GeoTime-0045	75.0000%	0.6110	75.0000%	0.6110	75.0000%	0.6110
GeoTime-0046	92.3077%	0.7454	92.3077%	0.7454	92.3077%	0.7454
GeoTime-0047	6.6667%	0.0174	-	-	-	-
GeoTime-0048	47.9167%	0.4963	-	-	-	-
GeoTime-0049	60.0000%	0.6509	60.0000%	0.6509	-	-
GeoTime-0050	57.1429%	0.2031	57.1429%	0.2031	-	-
Average Recall	**55.3770%**		**80.9295%**		**87.4785%**	
Average Score	**0.3959**		**0.5607**		**0.6081**	

Given that the coverage obtained by *Lucene* barely reached 50%, as can be seen in the penultimate row of the Table 1, and based on the work done

by [**?**], we decided to give the system an additional search engine, *Terrier*[7]. The *Bose-Einstein* (*Bo1*) query expansion model has been added to *Terrier*. In order to obtain a final normalized score for each document returned by both search engines, it was done as follow for each topic:

1. The maximum *Lucene* score value is obtained among all documents returned by it.
2. All documents scores returned by *Lucene* are divided between the value indicated in the previous step.
3. Similarly, the previous two steps are repeated for *Terrier*.
4. If there are documents returned either by *Lucene* and *Terrier*, both scores must be added.
5. Finally, the score of each document returned by the search engines mentioned above is divided by two, thereby obtaining a normalized value between 0 and 1.

Using both search engines, the recall improved from 55.377% to 87.0165% (Table 2). This recall increases the *nDCG-1000* score of the system from 0.3959 to 0.5921 by using only the *IR* module of the system.

Although Terrier alone achieved a recall comparable to the combination with Lucene, employing both search engines provided an improvement in 7 out of 25 topics. In the case of topic 44, this improvement was clearly significant (from 28.5714% to 46.9387%). Thus, this combination of search engines offers a more robust approach in order to retrieve the relevant documents that will be employed in the rest of the modules of the *GIR* system [**?**].

Question Answering Module. We performed a study on this module and noted that the *XML* documents created after the treatment of the topics (see Figure 2), in the part concerning to this module, which operation has been explained in Section 2.1 in the page 4, in the vast majority of cases, the temporal and/or the geographical part of the query were answered. For this reason it was decided to carry out an experiment where the 10 terms from the dates section (<*dates*>), complete or incomplete ones, and the 10 terms from the place names section (<*locations*>), all of them with their respective weights (*weight*), were added to the query which is run on the *Lucene* search engine. Later, the documents retrieve by *Lucene* would be joined to the *Terrier* ones, as explained in the Section 3.2 in the search engines experiment mentioned in page 7. As a result of this experiment the *nDCG-1000* score was increased from 0.5921 to 0.6206.

[7] `http://terrier.org/`

Table 2. Recall achieved using two search engines (*Lucene* and *Terrier*) for each *NTCIR 2011* topic

Topic	Lucene	Terrier	Lucene+Terrier
GeoTime-0026	93.2944%	98.5422%	98.8338%
GeoTime-0027	85.7142%	100%	100%
GeoTime-0028	85.4838%	99.1935%	99.1935%
GeoTime-0029	43.3566%	87.4125%	90.2097%
GeoTime-0030	66.6667%	85.7142%	85.7142%
GeoTime-0031	36.6667%	86.6667%	86.6667%
GeoTime-0032	35.087%	89.4736%	89.4736%
GeoTime-0033	74.4186%	100%	100%
GeoTime-0034	86.3636%	95.4545%	95.4545%
GeoTime-0035	28.5714%	76.1904%	76.1904%
GeoTime-0036	31.9148%	91.489%	91.489%
GeoTime-0037	0%	2.8571%	2.8571%
GeoTime-0038	1.6908%	68.5990%	68.8405%
GeoTime-0039	84.1201%	98.7124%	98.7124%
GeoTime-0040	82.075%	99.0566%	99.0566%
GeoTime-0041	98.9361%	100%	100%
GeoTime-0042	1.2738%	87.261%	87.8980%
GeoTime-0043	91.489%	100%	100%
GeoTime-0044	28.5714%	28.5714%	46.9387%
GeoTime-0045	75%	100%	100%
GeoTime-0046	92.3076%	96.1538%	98.7179%
GeoTime-0047	6.6667%	80%	80%
GeoTime-0048	47.9167%	77.0833%	79.1667%
GeoTime-0049	60%	100%	100%
GeoTime-0050	57.1428%	100%	100%
Average	**55.3770%**	**86.1557%**	**87.4330%**

4 Conclusions

In this first approach to geo-temporal *IR* systems, we have started from a *IR* system and we have added geographical intelligence. In addition, we have used a naive implementation to tackle the temporal dimension. In spite of this, we can draw the following conclusions.

In the future, the linguistic analysis module should be improved to have the ability to extract and/or filter better the information from the narrative part of the topics. Despite this, our system (*University of Alicante*) with only two search engines and *QA* techniques is able to obtain outstanding scores in the *NTCIR 2011 GeoTime* task, such as can be seen in [**?**] and in Figure 3[8].

As we have mentioned before, the *QA* module obtains a remarkable enrichment, therefore, we are exploring different *QA* techniques to use in the future.

[8] The scores from non completely automatic runs have been omitted

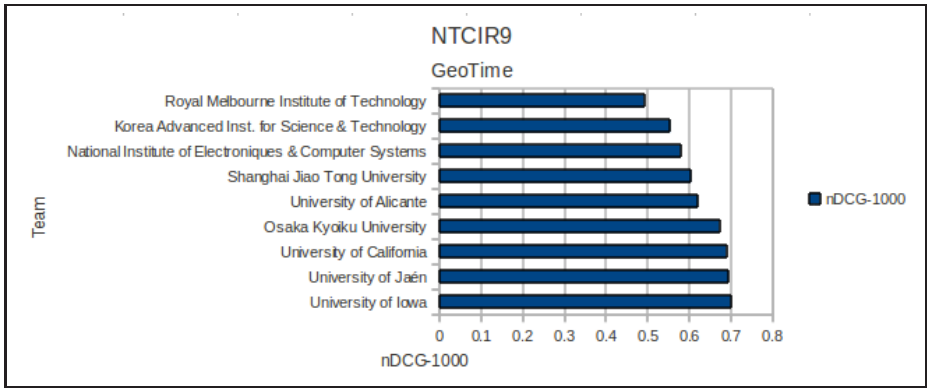

Fig. 3. Best *NTCIR 9* teams score.

In addition, given that good results were achieved by applying *QA* on *Lucene* query terms, as was seen in the Section 3.2 in the *QA* experiment in page 8, in a future experiment we will introduce the *QA* as *Terrier* query terms as well.

Focusing on the geographical module, currently we have two active fronts. On the one hand, we are exploiting more metadata from *Yahoo! Placemaker*, such as the general geographical scope of the document. On the other hand, we intend to fully develop the geographic module to be independent of applications which are subject to the restrictions of third parties.

Regarding Temporal Information Retrieval (*TIR*), a *TIR* system (*TIPSem*[9]) developed in our research group will be joined to this geo-temporal system in order to provide more temporal intelligence.

In future work, the usefulness of the rest of the components of the system such as the *entities detection module*, or the *Re-ranking module* will be analysed.

5 Acknowledgments

This research has been partially funded by the Spanish Government under project TEXTMESS 2.0 (TIN2009-13391-C04-01), and by the University of Alicante under project GRE10-33.

References

1. Christopher D Manning, Prabhakar Raghavan, and Hinrich Schtze. *An Introduction to Information Retrieval*. Number c. Cambridge University Press, 2009.
2. Fredric Gey, Ray R Larson, Jorge Machado, and Masaharu Yoshioka. *NTCIR9-GeoTime Overview - Evaluating Geographic and Temporal Search: Round 2*. 2011.
3. Stephen E Robertson, Steve Walker, and Micheline Hancock-Beaulieu. *Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Interactive*, pages 199–210. NIST, 1998.

[9] `http://gplsi.dlsi.ua.es/demos/TIMEE/`

4. Fredric Gey, Ray R Larson, Noriko Kando, Jorge Machado, and Tetsuya Sakai. *NTCIR-GeoTime Overview: Evaluating Geographic and Temporal Search*, pages 147–153. 2010.
5. Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20:422–446, October 2002.
6. J.M. Perea-Ortega. Recuperación de información geográfica basada en múltiples formulaciones y motores de búsqueda. *Procesamiento del Lenguaje Natural. N. 46 (2011). ISSN 1135-5948*, pages 131–132, 2010.
7. Fernando S. Peregrino, David Tomás, and Fernando Llopis. University of Alicante at NTCIR-9 GeoTime. *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, 2011.