

Temporal information needs in ResPubliQA: an attempt to improve accuracy. The UC3M participation at CLEF 2010

María Teresa Vicente-Díez, Julián Moreno Schneider, Paloma Martínez
Universidad Carlos III de Madrid

{tvicente, jmschnei, pmf}@inf.uc3m.es

Abstract. The UC3M team participates in 2010 in the second ResPubliQA evaluation campaign taking part in the monolingual Spanish task. On this occasion we have completely redesigned our Question Answering system, product of multiple efforts while being part of the MIRACLE team, by creating a whole new architecture. The aim was to gain in modularity, flexibility and evaluation capabilities that previous versions left pending. Despite its initial open-domain philosophy, the new system was tested by means of the JRC-Acquis and EUROPARL collections on the legal domain. We submitted two runs for the participation in the paragraph selection task. The main attempts in this campaign have focused on the study of the information needs concerning time. Starting from implementing a base system based on passage retrieval, we added temporal question analysis capabilities, temporal indexing to the collection, as well as some temporal filtering and reasoning features, getting a global accuracy of 0.51. In the second run we have implemented an answer analysis module based on n-gram analysis. The obtained results are slightly better, achieving a 0.52. We discuss the results found from each configuration when applied to the different temporal questions types.

Keywords: Question Answering, Temporal Indexing, Temporal Normalization, N-gram Similarity

1 Introduction

This paper describes the participation of the UC3M team in the second ResPubliQA evaluation exercise at CLEF 2010. Continuing with the aims of previous evaluations, new requirements were also included, such as the addition of new question types and a new document collection. Two main tasks were proposed, allowing to the participant systems to adopt a paragraph selection (PS) approach or an exact answer selection (AS) method. In our case, two runs for the Spanish monolingual subtask were submitted adopting the first strategy. The system follows a PS approach and is product of a complete redesign of our previous Question Answering (QA) system, adapting it to the new requirements.

Thus, this year the main challenge departed from a new system design in an attempt to explore modularity, simplicity, multilingualism and multi-collection support. The application domain (legal) has not changed this year but a new document collection has been added: the EUROPARL corpus.

In the past campaigns, our participation had outlined our growing interest on the management of temporal information and on their application to QA [1]. This year we have focused our work mainly on the study of the information needs concerning time. We will explore how temporal constraints can help to retrieve more accurate answers, filtering time-relevant information from other temporally out-of-scope information.

The question types that will be taken into account in this edition are Factoid, Definition, Reason-Purpose, Procedure, Opinion and Other. Based on our temporal-driven aims, our current system implements a question classifier attending to the temporal aspects uniquely of those questions concerning time.

Following the strategy of the previous campaign the system adopts a passage-oriented approach. Its responses are fixed again to the paragraph selection level although at the time of writing this article answer selection techniques are being developed. In this sense, the system follows an almost pure information retrieval scheme.

Indexing has being another point where we invested our effort. We developed specific indexes to satisfy temporal information needs, separating temporally restricted facts from the rest. This approach looks for gaining accuracy at the time of retrieving the candidate answer. Our experiments in this line have been successful and we have found configurations that performed substantially better than our baseline.

Finally, a global objective was to enlarge the capabilities of the QA system and advance towards an architecture that allows domain adaptation and multilingual processing.

The paper is structured as follows, Section 2 describes the system architecture with special attention paid to the novelties introduced this year, and Section 3 describes the submitted runs and analyzes the results. Finally, conclusions and future work are shown in Section 4.

2 System Description

The system architecture has been completely redefined and redesigned for this task. During the transition to a whole renewed platform we have reached a mixed structure between previous and future systems. It is still based on a pipeline which analyzes questions, retrieves documents and performs answer ranking based on n-gram similarity. The general architectural schema is shown in Figure 1.

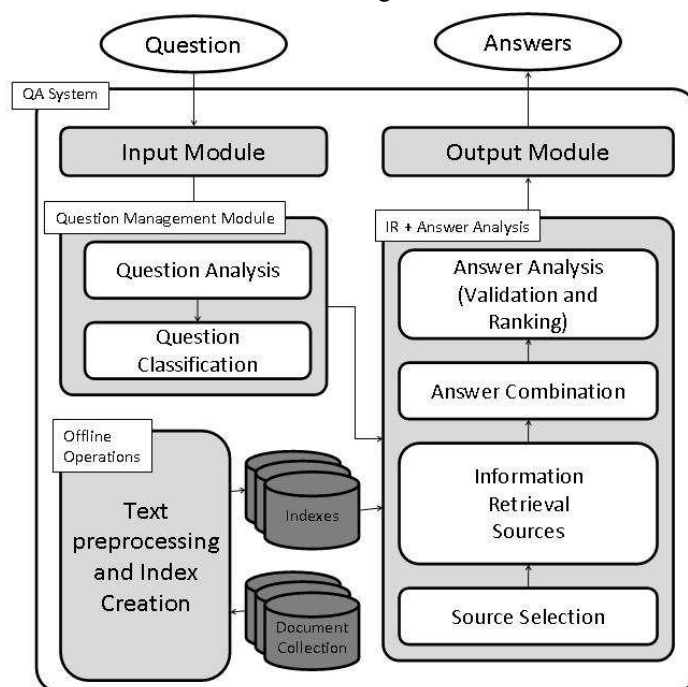


Figure 1. General System Architecture.

The main updates or additions performed in the system are outlined:

- Using a new architecture (still using some structural characteristics from previous version) developed to provide huge modularity and simplicity for implementation.
- Adding handling parsers for the new collection (EUROPARL).
- The evaluation procedure was modified to work with different measures like $c@1$, $top@n$ or the temporal questions classification percentage (see Section 2.6).
- New indexes implementing OKAPI BM-25 [2] have been created and tested.
- Some indexes containing only temporal constrained information were specifically created for responding to questions concerning time
- Implementation of an n-gram similarity ranking module.
- Temporal management was also added in two manners:
 - An automatic rule-based temporal expressions recognition and normalization (TERN) system [3] has been used as temporal analyzer for detection, extraction and resolution of expressions concerning time both in Question Analysis and Index Creation.
 - A temporal filtering module was created in order to discard the documents that do not fulfill the temporal constraints during the Answer Analysis step.
- Implementation of a new Temporal Question Classification module using the TERN analyzer.

2.1 Indexes

Indexes are crucial for QA since obtaining a good retrieval subsystem can considerably improve the final results of the system. Due to the addition of EUROPARL document collection, all IR indexes have been newly created using Lucene [4] as the IR engine. To accomplish the task of storing the relevant information as appropriately as needed, we have designed our indexes with the following characteristics:

- *Paragraph unit*: each document inside the collection has not been transformed into an index document. Instead of doing that, one index document has been created for each paragraph of the collections.

Besides of the paragraph text, the index document stores also other useful information about the document:

- The collection document name.
- The creation date of the documents in the collection.
- The name of the collection. This year the value can be: JRC-ACQUIS or EUROPARL.
- The language of the document (in this case always ES for Spanish).
- The text of the corresponding paragraph.
- The identification number of the paragraph inside the physical document.
- The initial and final dates for each temporal expression recognized by the TERN analyzer into the text of the paragraph (This will be explained deeply next).
- The analyzer used to preprocess the text before it is stored in the index has been a Snowball Analyzer whose main properties are:
 - Removes stopwords from the text.
 - Lowercases the text.
 - Stems the text.
 - Applies normalization (from StandardFilter¹) at each token.
- OKAPI BM-25 has been selected as the scoring strategy for the indexes. The algorithm implementation was developed and successfully tested in ResPubliQA 2009 by the UNED team [5].

Using the previous characteristics, we have developed two different indexes:

- **Baseline Index:** it is a simple paragraph index that stores all the fields explained previously except from 'InitialDate' and 'FinalDate' (for temporal expressions). It only indexes 'Text' for searching. As there is only one indexed field, the strategy used for searching is based on a simple query process that makes no difference between fields.
- **Temporal Index:** it consists of an index based on the previously described 'Baseline Index' where two more fields are added. Those fields are 'InitialDate' and 'FinalDate'. They correspond with the interval-based model applied by the TERN analyzer. For each paragraph all temporal expressions are recognized, normalized and resolved, if needed (relative expressions such as "tomorrow", incomplete dates such as "in January"), into a standardized interval format. As the result, a normalized date for the initial boundary of the interval is captured as the 'InitialDate' field. On the other hand, the final boundary of the normalized interval corresponds to the 'FinalDate' index field. In this case, besides of 'Text', 'InitialDate' and 'FinalDate' are also indexed fields for searching. Due to the increase in the number of indexed fields the query also changes. Two different new terms has been added to the query, one for searching in the 'InitialDate' field and another for searching in the 'FinalDate' field. This opens a range of reasoning possibilities during the retrieval almost only limited by implementation issues of the IR engine.

An important remark must be done at this point. While the baseline index incorporates a representation of all the collection documents, the temporal index only stores paragraphs containing some temporal restriction. Thus, it must be noted that it can not be used independently. When a search is launched, it is always used together with the baseline index in a retrieved documents fusion strategy. This strategy is based on a linear combination of the document scoring obtained from each index.

2.2 Question Classification

Question classification is an important part of QA because determining the type of question (or some specific characteristic which can be obtained during the classification process) can really help to increase the final performance.

As it was previously outlined, we have paid special attention to the development of a specific temporal question classifier, i.e. a classifier that determines if a question has some features that define it as a temporal type, as well as the kind of expected answer. In this sense, these features correspond with the two main temporal question categories according to the role of temporality in their resolution:

¹ http://lucene.apache.org/java/2_4_0/api/org/apache/lucene/analysis/standard/StandardFilter.html (12/07/2010)

- Temporally Restricted (TR) questions are those containing some time restriction: “¿Qué resolución fue adoptada por el Consejo el 10 de octubre de 1994?” (“What resolution was adopted by the Council on 10 October 1994?”)
- Questions with a Time-Expression Answer (TA) are those whose target is a temporal expression or a date: “¿Cuándo empieza la campaña anual de comercio de cereales?” (“When does the marketing year for cereals begin?”)

The temporal question classifier is based on a set of pre-defined rules. So that, if one or more temporal restrictions (temporal expressions) are found into the question, the classifier consider it as a TR question. For this category the focus of the query could be anything; the restriction will be helpful during the retrieval and the answer selection time for discarding chronologically non-relevant documents.

In the case of questions whose focus is a concrete moment, period or time unit, the classifier is able to detect some characteristic patterns that denote this category (TA). For instance, some predictable samples that fit these patterns are “When...?”, “At what hour...?”, or “At what time...?” questions. An example of this category would be in: “¿Cuándo fue presidente de Sudáfrica Nelson Mandela?” (“When was Nelson Mandela president of South Africa?”). In more, the classifier associates an expected granularity to the candidate documents accordingly to the precision of the question.

An example of rule for detecting time restrictions is shown in Table 1. The first row of the table represents the name of the rule. The second row specifies the normalization method that will be used once the expression is recognized. The third row specifies the type of the temporal expression and the annotation pattern. Finally, the fourth row shows the tokens that trigger the rule [3].

1. TEMPORAL_RULE(r1.3)
2. RULE=[[e/] [DIC(DIASEMANA)/] [dia/] DIC(DIA) de DIC(MES) DIC(PREP) METHOD(year)]
3. TEMPORAL_ANALYSIS_NORMALIZATION_TYPE=(abs_dia_mes_anio_3)
4. TEMPORAL_ANALYSIS_TYPE=(date:init:YYYY-MM-DD)

Table 1 Temporal restrictions detection pattern

Satisfactory classification results have been obtained following this naive approach, they will be presented in the corresponding section.

2.3 Temporal Management

It is a well-known aspect of all IR systems that they do not take advantage of all the semantic information that they manage. This is a pending issue that is being constantly faced up by many people and from many perspectives. One of the fields to be improved is the exploitation of temporal implicit information of the contents. In disciplines like QA such information can be very useful to improve the accuracy of the systems by discarding temporally non-relevant information, or by boosting candidates whose restrictions fit better with the temporal information needs, for instance.

In previous editions of CLEF we have already shown our interest on the management of temporal information and on their application to QA. This year we have stressed on this target, encouraged by those experiences. So in this campaign temporal management receives more emphasis and innovates in many senses.

First of all, the temporal information detection, resolution and annotation is based on a renewed TERN system developed for the TempEval track (inside the SemEval2010 Workshop). This decision was motivated by the need to improve the coverage of the original system [6] and test its performance against new datasets with a view to its integration in future domains of application. The EUROPARL corpus constituted a novelty to be taken into account. Main challenges were to move to a new temporal model where interval is considered as the basic time unit as well as the isolation of the internal representation of temporal information from the annotation schema.

Following the main aspects of the previous TERN system, the reference date used to resolve the relative expressions is taken from the creation date of each document of the collection. In the case of EUROPARL this date is specified inside the document, while in JRC-ACQUIS the creation year is taken from the document name but the creation month and day are supposed to be the 1st January.

Temporal management is applied at several points of the whole execution of the system. These different uses of the analyzer are described below:

- Question Analysis: the input question is analyzed and the normalized temporal expressions recognized are returned. From the point of view of classification, the existence of temporal expressions denotes that it is a TR question.
- Indexes Creation: temporal expressions in the paragraphs of the collection are recognized and stored (after resolution and normalization) in specific fields.
- Query Generation: through the formulation of a query to the IR engine for searching on the specific fields, looking for date matching ('InitialDate' and 'FinalDate'), a better document retrieval is performed.
- Answer Analysis: the retrieved documents (paragraphs) are analyzed and filtered accordingly to their suitability to the temporal restriction or scope detected in the question classification. If a document concerns to temporal interval that does not fit at all with the temporal restriction in the query it is strongly demoted. In the case of TA questions, documents whose temporal interval fits better with the granularity of the question are promoted.

2.4 Answer Combination

An answer combination strategy is needed since we use different information retrieval sources. As it was shown, our system uses two indexes to perform information retrieval. Each index returns its own list of candidates and it makes necessary to join these results.

Our proposal consists of a simple addition strategy, i.e. we make a linear combination of the score of the document from both indexes and then we reorder the resulting list. In (1) we present the equation used for the linear combination. As it can be observed, much more emphasis is given to the temporal index when a document satisfies not only the textual constraints of the query but also when it fits with its time constraints.

$$final_score = \alpha_1 \cdot baseline_score + \alpha_2 \cdot temporal_score \quad (1)$$

where $\alpha_1 = 1$ and $\alpha_2 = 4$

2.5 Ranking Based on N-gram Similarity

Ranking the retrieved documents of the IR engine is a really important part of a complete QA system. A proper ranking module can increase as much as decrease the final performance just due to a bad design criterion. In our case this module processes the resulting list of documents obtained from the IR module and ranks the document based on the n-gram similarity among the question and the specific document. The final scoring is calculated by means of a linear combination of three different and independent measures:

1. Lucene Scoring calculated by the Lucene index in the information retrieval module. (M1)
2. The N-Gram Scoring equation defined in [7]. (M2)
3. An own empirical equation (see (2)) that takes into account the number of n-grams in the question that are present in the document (but not caring about the frequency of appearance of these n-grams). (M3)

$$score = \beta \cdot \sum_{i=1}^K \left(\frac{i}{(K-i)} N_i \right)$$

where $\beta = \frac{1}{\sum_{j=1}^M (j)}$ is the normalization factor for the equation. (2)

In (2), M represents the number of grams in the question and $\delta_{K,i} = \frac{i}{(K-i)}$ is a weight factor where K is the number of grams in the question and N_i is the number of i-grams of the question that appear in the document. The linear combination used as the final score is presented in (3):

$$final_score = \alpha_1 \cdot M1 + \alpha_2 \cdot M2 + \alpha_3 \cdot M3 \quad (3)$$

Several configurations were tested during training, giving more and less weight to each scoring by varying the α_i coefficients. We obtained the best results taking the values:

$$\alpha_1 = 1; \alpha_2 = 1; \alpha_3 = 1;$$

However, we obtain identical results when applying other values to α_2 . In this manner, the linear combination is not influenced by the second factor (the N-Gram Scoring equation) and we can not conclude its contribution to the final results.

On the other hand, if more relevance is given to α_1 in comparison with α_3 , results are considerably worst.

This module provides the possibility of getting no answer as long as the resulting ranking score of every document is equals to zero. Then, the system is going to return a NOA answer.

2.6 Evaluation Module

The evaluation of the system is important for determining its performance. Our efforts while constructing the new architecture were focused on the ability of evaluating each part of the QA system. Evaluating separately question analysis (semantic analysis and classification), information retrieval or answer analysis, validation and ranking is the goal of our evaluation module.

In order to develop and test the system the gold standard from CLEF 2009 was used during training. The measure $c@1$ defined last year in QA@CLEF by the organization committee has been used as the main measure. Apart from that, we have implemented other proper measures:

- **Top@n** (4): determines the percentage of questions whose correct answer has been correctly selected (including paragraph number or not) among the first N candidate answers.

$$top@n = \frac{Correctly_Answered_in_n_first_candidates}{Total_Number_of_Questions} \cdot 100 \quad (4)$$

- **Temporal Question Classification Percentage** (TC). This measure estimates the performance of the question classifier module, determining two different values:
 - *True Positives* (5): percentage of questions tagged as temporally restricted in the gold standard that have been correctly classified as questions with a temporal answer (TA) or with a temporal restriction (TR).

$$Classifier_TruePositives_% = \frac{Correctly_Temporal_Classified_Questions}{Total_Number_Of_Temporal_Questions} \cdot 100 \quad (5)$$

- *False Positives* (6): percentage of questions not tagged as temporally restricted in the gold standard that have been wrongly classified as questions with a temporal answer (TA) or with a temporal restriction (TR).

$$Classifier_FalsePositives_% = \frac{Incorrectly_Temporal_Classified_Questions}{Total_Number_Of_Non-Temporal_Questions} \cdot 100 \quad (6)$$

- **Question Type Classification Measure** (QC) (7): determines the percentage of questions that have been correctly classified among the different types proposed for the ResPubliQA2010 track.

$$Question_Classification_% = \frac{Correctly_Classified_Questions}{Total_Number_Of_Questions} \cdot 100 \quad (7)$$

Periodically, the output and the XML logs of different executions were manually inspected to complete the evaluation process and to detect integration problems.

3 Experiments and Results

Several experiments were launched during the development of the new system to test the performance of the different configurations. Thanks to the ResPubliQA 2009 gold standard we could appreciate which adjustments provided the best results. Finally, the two configurations with the highest evaluation figures were chosen for the submission of the two runs (see Table 2 and Table 3). These are:

- **uc3m101PSeses**: the system is based on a pure IR scheme using paragraphs (passages) as documents. It uses the OKAPI BM-25 as IR index scorer. It includes temporal question analysis and classification. Moreover, it implements a combined search strategy between the baseline index and the temporal index. Finally, the system applies a temporal filter over the retrieved documents.
- **uc3m102PSeses**: the main configuration is similar to the first run, but in this case the system also implements the n-gram similarity over the answer analysis.

Some additional configurations without the use of the n-grams for answer selection and without using OKAPI BM-25 as index scorer were tested obtaining worse results. Therefore, we included these two modules in our start-up configuration.

Figure 2 and Figure 3 present the architectonical configuration of the system for the submitted runs.

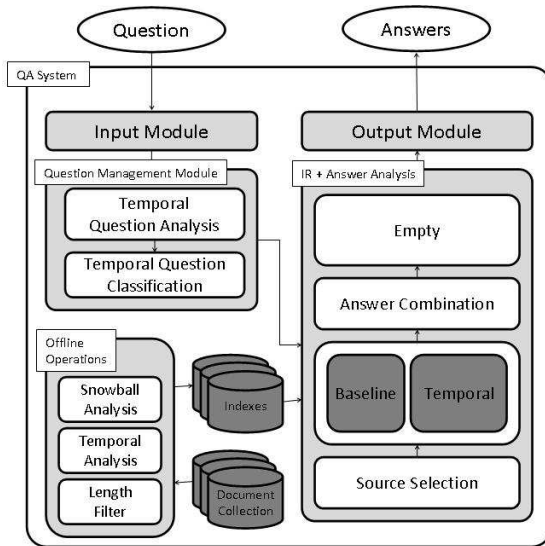


Figure 2. uc3m101PSeses run architecture

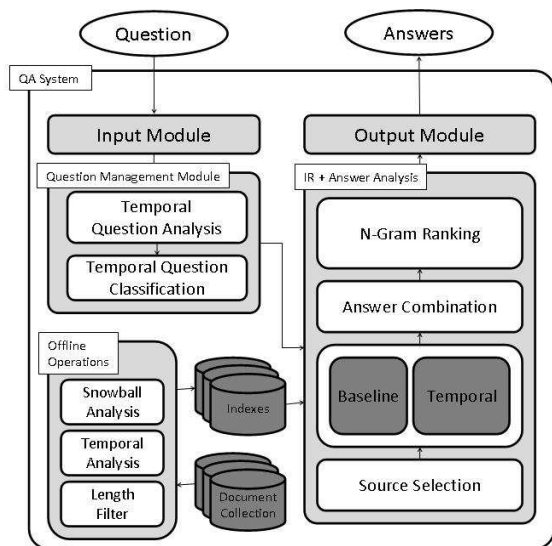


Figure 3. uc3m102PSeses run architecture

Firstly, we tested the performance of the question classifier against the gold standard question set, attaining a precision of 0.8941. The main cause of error was due to the lack of patterns to identify all types of temporal questions.

Accuracy	INFORMATION SOURCES	
	Baseline index	Baseline & Temporal Index
SYSTEM CONFIGURATION		
uc3m101PSeses config.	0.41	0.46
uc3m102PSeses config.	0.47	0.52

Table 2 Experimentation results over the complete gold standard

Accuracy	INFORMATION SOURCES	
	Baseline index	Baseline & Temporal Index
SYSTEM CONFIGURATION		
uc3m101PSeses config.	0.43	0.53
uc3m102PSeses config.	0.49	0.59

Table 3 Experimentation results over the temporal questions subset in the gold standard

The official results for the two selected runs are detailed in Table 4. Answer accuracy (Acc.) has been calculated as the ratio of questions correctly answered (Right) to the total number of questions. Only the first candidate answer is considered.

Name	Right	Wrong	Unansw. Right Candidate	Unansw. Wrong Candidate	Unansw. Empty Candidate	Acc.	Correctly discarded	c@1 measure
<i>uc3m101PSeses</i>	101	99	0	0	0	0.51	0	0.51
<i>uc3m102PSeses</i>	104	96	0	0	1	0.52	0	0.52

Table 4 Results for submitted runs

If we compare the results obtained both before and after the evaluation, we can conclude that the system has a very stable behavior. According to our expectations, the use of a combination strategy for merging the two information sources provides more accuracy, especially when managing questions concerning time.

We did not carry out an exhaustive analysis of performance over each question type of the total gold standard question set (120 questions of the corpus originally distributed in Spanish). In this way we mainly centered our study over the temporal questions subset. Due to the limitations of the number of temporal questions in this subset, we also translated the rest of temporal questions that were originally formulated in other languages. We achieved a total amount of 49 questions, what supposes a 40% of the initial corpus.

Another module that has contributed to improve the results has been the answer selection strategy based on n-gram similarity. The combination of the scoring measures and the tuning of their weights increased the official results by 0.01, but the increase is even greater when only dealing with temporal questions.

However, the analysis of the errors shows that further work is needed to be able to manage all the complexities of the domain. For example, a precise classification of questions is mandatory if a good answer selection is required. However, the process of finding the focus of the question which is crucial for question classification is specially error prone. Apart from temporal questions, other types would require further study of techniques that help to improve the classification of passages as bearing procedures, objectives, etc.

4 Conclusion and Future Work

The problem posed by the environment defined by the ResPublicQA task is still a challenge in many aspects: the addition of a new collection (i.e. EUROPARL), specific technical domain, multilingualism, etc.

Last year it was proved that passage retrieval works well for the JRC-Acquis collection. Moreover, the same retrieval scheme is also appropriate for the EUROPARL collection. As a direct consequence, the inclusion of the new document collection was an easy part of the task.

This year our work has focused on temporality issues. From the question analysis phase to the answer filtering step we have made big efforts on including a complete temporal perspective to the system taking advantage of the temporal properties of questions and documents. A manual evaluation of the results has proved that this domain (legal domain) is not good for testing temporality due to the large amount of dates present at each document and paragraph. It is really difficult to perform an accurate temporal filter to discriminate documents based on dates if those dates have a huge number of occurrences.

Concerning the evaluation framework, we have complemented the system with checkpoints that provide intermediate results. The system is able to evaluate partial results concerning the different parts of the whole system independently (question classifier, information retrieval module, etc...), allowing a progressive view of the QA process.

As future work, we are improving the answer filtering module by means of the integration of reasoning mechanisms based on the Allen intervals [8]. It will offer a way to discard those documents that, regarding to the query restriction, do not belong to the relevant time interval. Although we have implemented the temporal filtering module, due to time constraints we have not had enough time to test it completely.

Another line of research is the searching of an adequate certainty threshold for determining NOA answers at the stage of answer validation. This is an important part of QA systems that we have still not deeply investigated. Future work will focus on improving the N-Gram Scoring module by testing different threshold values to determine the NOA limit, and on creating different validation modules.

Finally, further work in the general architecture of the QA is expected to finalize the transition to the new architecture which simplifies the domain adaptation and multilingualism.

Acknowledgements

This work has been partially supported by the Research Network MA2VICMR (S2009/TIC-1542) and the project BRAVO (TIN2007-67407-C3-01).

References

- [1] Vicente-Díez, M.T., de Pablo-Sánchez, C., Martínez, P., Moreno, J. and Garrote, M. 2009. *Are Passages Enough? The MIRACLE Team Participation at QA@CLEF2009*. Multilingual Information Access Evaluation Vol. I Text Retrieval Experiments. 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfú, Greece, September 30 - October 2, 2009. LNCS. Peters, C.; Di Nunzio, G.M.; Kurimo, M.; Mandl, Th.; Mostefa, D.; Penas, A.; Roda, G. (Eds.) ISBN: 978-3-642-15753-0.
- [2] Pérez J. , Garrido G. , Rodrigo A., Araujo L., Peñas A. *Information Retrieval Baselines for the ResPubliQA task*. Working Notes for the CLEF 2009 Workshop, Corfu, Greece. ISBN:978-88-88506-84-5
- [3] Vicente-Díez, M.T., Moreno-Schneider, J., Martínez, P. *UC3M system: Determining the Extent, Type and Value of Time Expressions in TempEval-2*. In proceedings of the Semantic Evaluation – 2 (Semeval 2010), ACL Conference, Uppsala (Sweden), 2010.
- [4] *Apache Lucene project*. The Apache Software Foundation. <http://lucene.apache.org/>, visited 14/07/2010.
- [5] Pérez-Iglesias, J. Pérez-Agüera, J.R., Fresno, V. and Z. Feinstein, Y. *Integrating the Probabilistic Models BM25/BM25F into Lucene*. In CoRR, abs/0911.5046, 2009.
- [6] Vicente-Díez, M.T., de Pablo-Sánchez, C. y Martínez, P. *Evaluación de un Sistema de Reconocimiento y Normalización de Expresiones Temporales en Español*. Procesamiento del lenguaje natural. N. 39 (sept. 2007); pp. 113-120.
- [7] Montes-y-Gómez, M., Villaseñor-Pineda, L., Pérez-Coutiño, M., Gómez-Soriano, J. M. , Sanchos-Arnal, E. Rosso, P. *INAOE-UPV Joint Participation at CLEF 2005: Experiments in Monolingual Question Answering. Working Notes for the CLEF 2005 Workshop*. Vienna, Austria, September 2005.
- [8] Allen, J.F. 1983. *Maintaining knowledge about temporal intervals*. Communications of the ACM, 26 (11):832-843.