Building an XML framework for Question Answering

David Tomás José L. Vicedo Maximiliano Saiz Rubén Izquierdo
Department of Software and Computing Systems
University of Alicante, Spain
{dtomas,vicedo,max,ruben}@dlsi.ua.es

Abstract

This paper describes the novelties introduced in the Question Answering system developed in the Natural Language Processing and Information Systems Group at the University of Alicante for QA@CLEF 2005 campaign with respect to our previous participations. Thinking of future developments, this year we have designed a modular framework based on XML that will easily let us integrate, combine and test system components based on different approaches. In this context, several modifications have been done. Mainly, a new machine learning based question classification module has been added and tested, the document retrieval engine has been changed and several new heuristics for answer extraction have been applied. We took part in the monolingual Spanish task.

Categories and Subject Descriptors

H.3.2 [**Information Storage and Retrieval**]: Information Storage—*File organization*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Natural Language*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language parsing and understanding*

General Terms

Experimentation, Performance, Design

Keywords

Question answering, Question classification, Multilingual, XML

1 Introduction

Most of the Question Answering systems are based on pipeline architecture comprising three main stages: question analysis, document retrieval and answer extraction. These three tasks can be isolated in different modules, so that the development of each one could be set apart and afterward integrated as a whole system. In order to achieve this goal, we have developed an XML framework that facilitates the communication between the different components of the system, so that we can easily substitute and test new modules into the general framework for further development.

Besides that, the system has suffered several modifications with respect to previous competitions [1] [2] in the different stages of the question answering process. First, for question analysis we have added a machine learning based question classification system [3] that can be trained with corpora in different languages for multilingual purpose. Secondly, for document retrieval we have

moved to Xapian[1], an open source probabilistic information retrieval library, highly adaptable and also flexible enough to cope with documents in different languages. Finally, several new heuristics have been added in the answer extraction process in order to improve answer candidate weighting and narrow the set of possible answers.

This paper is organized as follows: in section 2 we describe the system architecture detailing the novelties included this year; section 3 outlines the XML framework that allows the communication between modules; section 4 presents and analyses the results obtained at QA@CLEF 2005 Spanish monolingual task, paying special attention at the question classification module performance; finally, in section 5 we discuss the main challenges for future work.

2 System Description

Our approach has evolved from the system developed in our research group [2], where new components and old ones have been fully integrated in a brand new XML framework designed for combining QA processes in a multilingual environment.

The system follows the classical three-stages pipeline architecture mentioned above. In the question analysis stage, queries proposed to the system are analyzed and useful information for next modules is extracted, like keywords or question type. Document retrieval module takes the keywords extracted from the query in the previous stage and returns a set of relevant documents related to these terms. The documents obtained are the input for the final step, the answer extraction stage, which processes these texts in order to locate and present the final answer. Figure 1 shows the system architecture. Next paragraphs describe each module in detail.

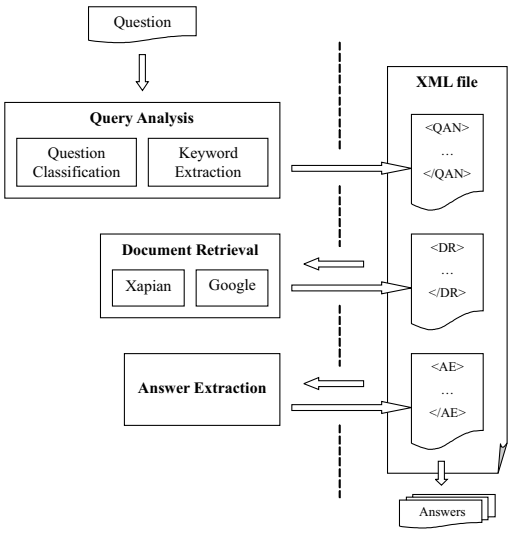

Figure 1: System architecture

2.1 Question Analysis

This stage carries out two processes: question classification and keyword selection. The first one detects the sort of information claimed by the query, mapping the question into a previously defined taxonomy. Otherwise, keyword selection chooses meaningful terms from the query that helps to locate the documents that are likely to contain the answer.

This year we have replaced the former question classification module, based on hand made lexical patterns, with a new one based on machine learning [3]. After defining the possible an-

[1] http://www.xapian.org

swer types (NUMBER, DATE, LOCATION, PERSON, ORGANIZATION, DEFINITION and OTHER), we trained the system with an annotated corpus made up of questions from Question Answering Track in TREC[2] 1999 to 2003 and CLEF 2003 to 2004, to sum up 2793 training questions in Spanish. Thus there is no need to manually tune the module since all the knowledge necessary to classify the questions is automatically acquired. To apply the system to different languages we only have to change the training corpus. In section 4 we present detailed performance results of this particular module.

On the other hand, the keyword extraction module remains the same for this year competition [2], using hand made lexical patterns in order to obtain useful information (keywords and definition terms). Freeling Spanish Lemmatizer [4] is applied to the terms selected and lemmas are stored for further use.

2.2 Document Retrieval

To accomplish this task we use two different search engines: Xapian and Google[3]. Xapian performs document retrieval over the entire EFE Spanish document collection. The lemmas of the keywords detected in the question analysis stage are used to retrieve the 50 topmost relevant documents from the EFE collection.

In parallel, the same keyword list (not lemmatized this time) is sent to Google search engine through its Web API[4], selecting the 50 top ranked short summaries returned. We store this information for later use as a statistical indicator of answer correctness.

As a novelty, we introduced last year the use of English search to improve the retrieval task. The original Spanish question is translated into English via SysTran[5] online translation service. Keywords are extracted and sent to Google, selecting again the 50 top ranked short English summaries returned that are later used to weight possible answers in the extraction module. This special search is only performed if the question is mapped to type NUMBER, DATE, PERSON or ORGANIZATION, the classes that are likely to have a language independent answer: numbers, dates, people and company names tend to keep unchanged through languages.

2.3 Answer Extraction

In this stage a single answer is selected from the list of relevant documents retrieved from the EFE Spanish corpus. At this point we have the following information: keywords and definition terms from the query, the set of relevant Spanish documents retrieved from the EFE corpus, the set of relevant summaries in Spanish retrieved from Google and, depending on the question class, a set of summaries in English also retrieved from the Web. The system uses all this information to select the correct answer. The set of possible answers is formed extracting all the n-grams (unigrams, bigrams and trigrams in our experiments) from the relevant documents in the EFE collection.

Although the general process is similar to the one we used in previous competitions and already explained in detail in [1], new information has been added to improve the filtering and the final answer selection step.

The following filters are applied to the set of possible answers so that we can narrow down the solution space:

- Part of speech and query class: depending on the query class, we reject the answers that do not contain a proper part of speech. For instance, if the query class is PERSON we expect the answer to be a proper noun, not an adjective or a verb.
- Query terms: we reject all the possible answers that contain any keyword or that only contain definition terms.

[2] Text REtrieval Conference, http://trec.nist.gov
[3] http://www.google.com
[4] http://www.google.com/api
[5] http://www.systransoft.com

- Stopwords: all the answers that start or end with a stopword are rejected.

Once the filtering process is done, remaining candidate answers are scored taking into account the following information:

- The sentence where the answer appears is scored depending on the number of keywords and definition terms that co-occur. We refer to this weighting value in the formula below as w_s.
- The frequency (w_f) of the answer in the documents and summaries obtained in the retrieval stage. Here we have three different values: the frequency in the documents retrieved by Xapian, the frequency in the Spanish web summaries and, depending on the query type, the frequency in the English web summaries. The value w_f is computed like this:

$$w_f = \frac{\log(\frac{docFrec}{docMax}) + log(\frac{gooFrec}{gooMax}) + log(\frac{gooEnFrec}{gooEnMax})}{n} \quad (1)$$

  where numerators inside logarithm functions represent the word frequency in the Spanish corpus, in the Spanish web summaries and in the English web summaries respectively. Denominators represent the frequency of the most frequent word in the corresponding collections, being used to normalize the final values. The global frequency w_f is normalized through n, that represents the number of IR processes taking place: 2 or 3, depending on whether we use Google English summaries or not.
- The distance (w_d) or number of words between the possible answer and the keywords and definition terms co-occurring in the same sentence.
- The size (number of words) of the answer (w_n). For instance, DEFINITION questions are more likely to have long answers, while NUMBER answers tend to be shorter.

All the weights obtained are normalized in order to obtain a final value between 0 and 1, determining the confidence of the system in the answer selected. The final answer score (as) is computed as follows:

$$as = 0.2 \cdot w_s + 0.3 \cdot w_f + 0.2 \cdot w_d + 0.3 \cdot w_n \quad (2)$$

3 The XML Framework

Once detailed the different stages of the Question Answering system, we are describing the XML framework where all the process takes place. The eXtensible Markup Language (XML) is a general-purpose markup language that has become a standard de facto in inter-system communication, being widely used to facilitate data sharing between applications. We have used it to exchange information between the different modules in our system, building a framework were individual components can be easily interchanged. Thus, new modules can be developed separately and later used in place of old ones in the framework for testing purpose. In order to change a module, we only have to make sure that it fits de XML specification for that process.

We have associated an XML tagset for each stage of the process. Every module adds the XML fragment generated to a common file where the following modules can extract the information required to perform. So, what we finally get is a sort of log file that stores the complete question answering process in XML format. This file can be used to save time testing individual modules, as we have the information needed already stored in the file. For instance, if we just want to test the answer extraction module, we wouldn't need to execute the previous processes as the information might be already stored in the file because of a previous run.

Although our run was limited to Spanish monolingual task, the framework is prepared to store information in different languages together for multilingual purpose.

In the question analysis stage, we must store information about query terms and question type detected. For instance, the question "¿En qué provincia está Atapuerca?" ("Which province is Atapuerca in?") generates the following XML fragment:

```
<QAN>
  <CLASS type="LOCATION"/>
  <TERMS lang="en">
    <TERM type="DT" wf="Which" lemma="which" pos="WDT"/>
    <TERM type="DT" wf="province" lemma="province" pos="NN"/>
    <TERM type="SW" wf="is" lemma="be" pos="VBZ"/>
    <TERM type="KW" wf="Atapuerca" lemma="Atapuerca" pos="NP"/>
    <TERM type="SW" wf="in" lemma="in" pos="IN"/>
  </TERMS>
  <TERMS lang="sp">
    <TERM type="SW" wf="En" lemma="en" pos="SPS00"/>
    <TERM type="DT" wf="qué" lemma="qué" pos="DT0CN0"/>
    <TERM type="DT" wf="provincia" lemma="provincia" pos="NCFS000"/>
    <TERM type="SW" wf="está" lemma="estar" pos="VMIP3S0"/>
    <TERM type="KW" wf="Atapuerca" lemma="Atapuerca" pos="NP00000"/>
  </TERMS>
</QAN>
```

QAN tag indicates the kind of process taking place (question analysis). CLASS tag indicates the query type detected by the question classification module. TERMS stores the information on question terms, telling us if they are keywords (KW), stopwords (SW) or definition terms (DT). It also stores the word itself, the lemma and the part of speech detected by Freeling. The attribute lang indicates the language of the query, so that we can store multilingual information for every question.

This snippet is included in the common XML file and passed to the document retrieval module, which reads the information required in that process. As a result we obtain the relevant documents retrieved by the search engine (in this this case Xapian and Google), which are stored as an XML fragment in the common file. In the example above:

```
<DR>
  <ENGINE type="Xapian" lang="sp">
    <DOC name="EFE19950904-02045">
      <PARAGRAPH>J.M. Bermúdez, uno de los investigadores [...]</PARAGRAPH>
      <PARAGRAPH>El integrante del equipo de Atapuerca [...]</PARAGRAPH>
      ...
    </DOC>
    ...
  </ENGINE>
  <ENGINE type="Google" lang="sp">
    <DOC name="1">
      <PARAGRAPH>En la provincia de Burgos Atapuerca [...]</PARAGRAPH>
    </DOC>
    ...
  </ENGINE>
  <ENGINE type="Google" lang="en">
    <DOC name="1">
      <PARAGRAPH>The Sierra de Atapuerca, east of Burgos [...]</PARAGRAPH>
    </DOC>
    ...
  </ENGINE>
</DR>
```

Table 1: Detailed results for Spanish monolingual run

Accuracy (%)			
Factoid	**Definition**	**Temporally restricted**	**Overall**
28.81	46.00	25.00	32.50

Table 2: Detailed accuracy on factoid questions and on the whole set

Question type	Number of questions	Number correct	Accuracy (%)
Factoid	150	115	76.67
Factoid + Definition	200	165	82.50

DR tag stands for the name of the process (document retrieval). Each ENGINE tag indicates the name of the search engine and the language of the documents returned. Inside this tag the different documents are separated in DOC tags, and these are subdivided in PARAGRAPH tags, that evidently represent paragraphs.

Finally, the answer extraction module gets information from both QAN and DR fragments to select the candidate answers. Therefore another XML snippet is generated and added to the common file. In the example above:

```
<AE>
  <ANSWER lang="sp" doc="EFE19950620-13256" cert="0.895063">Burgos</ANSWER>
  <ANSWER lang="sp" doc="EFE19950907-04306" cert="0.432019">León</ANSWER>
  <ANSWER lang="sp" doc="EFE19950907-04306" cert="0.311245">Granada</ANSWER>
  ...
</AE>
```

AE tag represents the kind of process (answer extraction). Each ANSWER tag stores a possible answer, indicating its language, the document where it was found and the confidence of the system.

Another benefit of this XML framework is that additional tags could be added on demand if extra information storing is required for new modules, having not to change the old modules performance as the original structure remains the same.

4 Results

This year we submitted one run for the Spanish monolingual task. Table 1 shows the overall and detailed results obtained for each CLEF question type.

These results are very similar to the ones obtained in last year competition [2]. The main goal this year was the design of the XML framework for future developments and the inclusion of a new question classification module based on machine learning. In this sense results are encouraging as there seems to be no lost of performance due to the new module, having the additional benefit of being easily adaptable to new languages for multilingual purpose.

Table 3 shows the detailed performance of the question classification process, but not the impact in the whole Question Answering system. DEFINITION questions are not included in this comparison as this type of query was identified by CLEF organization before question answering process took place. Thus we present the results for the classification of the 150 factoid questions in the set. Table 2 reveals that almost 77% of the factoid questions were correctly classified (up to 82.5% if we also consider DEFINITION questions), quite promising for a system trained on surface text features aiming to reach language independence.

Table 3: Detailed precision and recall on factoid question classification

Class	**Precision**	**Recall**
PERSON	1	0.793
NUMBER	0.913	0.875
OTHER	0.769	0.905
LOCATION	0.897	0.813
DATE	0.739	0.85
ORGANIZATION	0.947	0.75

5 Future Work

In this paper we have described the novelties introduced in our Question Answering system for QA@CLEF 2005 competition. Mainly, a new XML framework has been introduced laying the foundations for future developments. In this framework we can easily introduce new modules and substitute old ones for testing purpose. This year we have introduced a new question classification module that can be trained with different languages, proving to be as competitive as other state-of-the-art systems. We also introduced a new IR engine that can be easily adapted to new languages.

Therefore, the main goal is to continue the gradual development and integration of new multilingual modules in order to have a system that can deal with many different languages at the same time. To sum up, this can be considered the first step of a full multilingual framework for QA.

6 Acknowledgements

This work has been developed in the framework of the project CICYT R2D2 (TIC2003-07158-C04) and it has been partially funded by the Spanish Government through the grant BES-2004-3935.

References

[1] Vicedo, J.L., Izquierdo, R., Llopis, F. and Muñoz, R.: Question answering in Spanish. In: CLEF, editor, Proceedings CLEF-2003 Lecture Notes in Computer Science, Tronheim, Norwey, August 2003.

[2] Vicedo, J.L., Saiz, M. and Izquierdo, R.: Does English help Question Answering in Spanish? In: CLEF, editor, Proceedings CLEF-2004 Lecture Notes in Computer Science, Bath, UK, September 2004.

[3] Tomás, D., Bisbal, E., Vicedo, J.L., Moreno, L. and Suárez, A.: Una aproximación multilingüe a la clasificación de preguntas basada en aprendizaje automático. XXI Conference of the Spanish Society for Natural Language Processing, 2005 (in print).

[4] Carreras, X., Chao, I., Padró, L. and Padró, M.: Freeling: An Open-Source Suite of Language Analyzers. In: Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04). Lisbon, Portugal. 2004.