INAOE's Participation at QA@CLEF 2007

Alberto Téllez, Antonio Juárez, Gustavo Hernández, Claudia Denicia, Esaú Villatoro, Manuel Montes, Luis Villaseñor

Laboratorio de Tecnologías del Lenguaje
Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Mexico.
{albertotellezv, antjug, ghernandez, cdenicia, villatoroe, mmontesg, villasen}@inaoep.mx

Abstract

This paper describes the system developed by the Language Technologies Lab of INAOE for the Spanish Question Answering task at CLEF 2007. The presented system is centered in a full data-driven architecture that uses information retrieval and machine learning techniques to identify the most probable answers to definition and factoid questions respectively. The major quality of our system is that it mainly relies on the use of lexical information and avoids applying any complex language processing resource such as POS taggers, named entity classifiers, parsers or ontologies. Experimental results indicate that our approach is very effective for answering definition questions from Wikipedia. In contrast, they also reveal that it is very difficult to respond factual questions from this resource solely based on the use of lexical overlaps and redundancy.

Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [Database Managment]: Languages—Query Languages

General Terms

Measurement, Performance, Experimentation

Keywords

Question Answering for Spanish, Lexical Information, Information Retrieval, Machine Learning.

1 Introduction

Question Answering (QA) has become a promising research field whose aim is to provide more natural access to information than traditional document retrieval techniques. In essence, a QA system is a kind of search engine that allows users to pose questions using natural language instead of an artificial query language, and that returns exact answers to the questions instead of a list of entire documents.

Current developments in QA tend to use a variety of linguistic resources to help in understanding the questions and the documents. The most common linguistic resources include: part-of-speech taggers, parsers, named entity extractors, dictionaries, and WordNet [2, 3, 4, 6]. Despite of the promising results of these approaches, they have two main inconveniences. On the one hand, the construction of such linguistic resources is a very complex task, and on the other hand, their performance rates are usually not optimal.

In contrast to these recent developments that point to knowledge rich methods (that are intrinsically language and domain dependent), in this paper we present a straightforward QA approach that avoids using any kind of linguistic resource, and therefore, that can be –in theory– applied to answer questions in several languages. This approach is mainly supported on two simple ideas. First, questions and answers are commonly expressed using the same set of words, and second, different kind of questions requires different kind of methods for adequate answer extraction.

In particular, the developed QA system is based on a full data-driven approach that exclusively uses *lexical information* in order to determine relevant passages as well as candidate answers. Furthermore, this system is divided in two basic components; one of them focuses on definition questions and applies traditional *information retrieval* techniques, whereas the other one centers on factoid questions and uses a supervised *machine learning* strategy. This system continues our last year work [5]; however it incorporates some new elements. For instance, it takes advantage of the structure of the document collection (Wikipedia in this case) to easily locate definition phrases, and it also applies a novel technique for query expansion based on association rule mining [1] in order to enhance the recovery of relevant passages.

The following sections give some details on the proposed system. Sections 2 and 3 describe the subsystems for answering definition and factoid questions respectively. Then, section 4 describes the system's adaptations required to deal with group of related questions. Later on, section 5 presents our evaluation results. Finally, section 6 discusses some general conclusions about our participation at QA@CLEF 2007.

2 Answering Definition Questions

Our method for answering definition questions uses *Wikipedia* as target document collection. It takes advantage of two known facts: (1) Wikipedia organizes information by topics, that is, each document concerns one single subject and, (2) the first paragraph of each document tend to contain a short description of the topic at hand. This way, it simply retrieves the document(s) describing the target term of the question and then returns some part of its initial paragraph as answer.

Figure 1 shows the general process for answering definition questions. It consists of three main modules: target term extraction, document retrieval and answer extraction. The following sections briefly describe these modules.

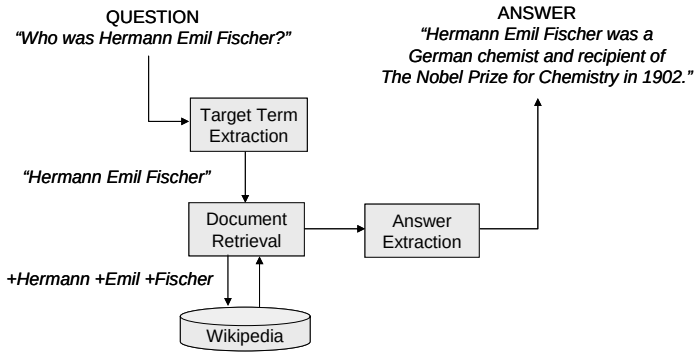

Figure 1. Process for answer definition questions

2.1 Finding Relevant Documents

In order to search in Wikipedia for the most relevant document to the given question, it is necessary to firstly recognize the target term. For this purpose our method uses a set of manually constructed regular expressions such as: *"What|Which|Who|How"+"any form of verb to be"+<TARGET>+"*?", *"What is a <TARGET> used for?"*, *"What is the purpose of <TARGET>?"*, *"What does <TARGET> do?"*, etc.

Then, the extracted target term is compared against all document names and the document having the greatest similarity is recovered and delivered to the answer extraction module. It is important to mention that, in order to favor the retrieval recall, we decided using the document names instead of the document titles since they also indicate their subject but normally they are more general (i.e., titles tend to be a subset of document names).

In particular, our system uses the *Lucene*[1] information retrieval system for both indexing and searching.

2.2 Extracting the Target Definition

As we previously mentioned, most Wikipedia's documents tend to contain a brief description of its topic in the first paragraph. Based on this fact, our method for answer extraction is defined as follows:

1. Consider the first sentence of the retrieved document as the target definition (the answer).
2. Eliminate all text between parenthesis (the goal is to eliminate comments and less important information).
3. If the constructed answer is shorter than a given specified threshold[2], then aggregate as many sentences of the first paragraph as necessary to obtain an answer of the desire size.

For instance, the answer for the question "*Who was Hermann Emil Fischer?*" (refer to Figure 1) was extracted from the first paragraph of the document "Hermann_Emil_Fischer": "*Hermann Emil Fischer* (October 9, 1852 – July 15, 1919) *was a German chemist and recipient of the Nobel Prize for Chemistry in 1902*. Emil Fischer was

1 http://lucene.apache.org/

2 For the experiments reported in section 4 we defined this threshold equal to 70 characters. This number was estimated after a manual analysis of several Wikipedia's documents.

born in Euskirchen, near Cologne, the son of a businessman. After graduating he wished to study natural sciences, but his father compelled him to work in the family business until determining that his son was unsuitable".

3 Answering Factoid Questions

Figure 2 shows the general process for answering factoid questions. This process considers three main modules: *passage retrieval*, where the passages with the major probability to contain the answer are recovered from the document collections; *question classification*, where the type of expected answer is determined; and *answer extraction*, where candidate answers are selected using a machine-learning approach, and the final answer recommendation of the system is produced. The following sections describe each of these modules.

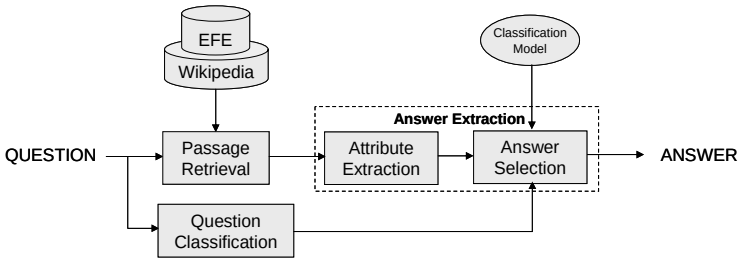

Figure 2. Process for answering factoid questions

3.1 Passage Retrieval

This module aims, as we previously mentioned, to recover a set of relevant passages from all target document collections, in this particular case, the EFE news collection and Wikipedia. It is primary based on a traditional vector-space-model retrieval system, but also incorporates a novel query expansion approach. Figure 3 shows the general scheme of this module. It considers four main processes: association rule mining, query generation, passage retrieval, and passage integration.

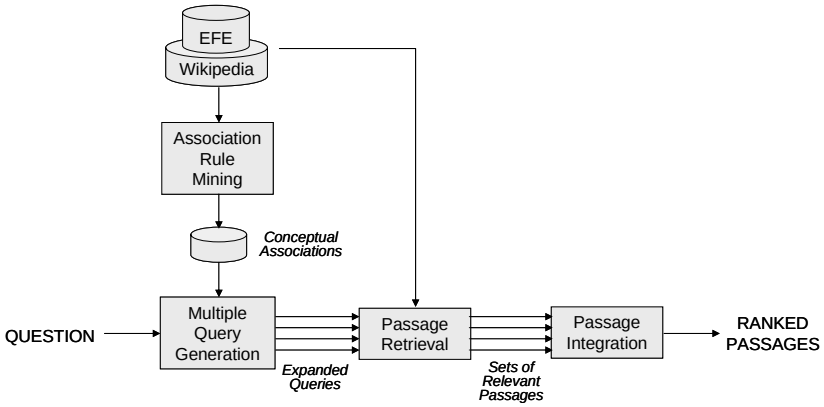

Figure 3. General Process for Passage Retrieval

Association rule mining. This process is done offline. Its purpose is to obtain all pairs of highly related concepts (i.e., named entities) from a given document collection. It considers that a concept A is related or associated to some other concept B (i.e., A → B), if B occurs in σ% of the documents that contains A.

In order to discover all association rules satisfying a specified σ-threshold this process applies the well-known Apriori algorithm [1]. Using this algorithm it was possible to discover association rules such as "Churchill → Second World War" and "Ernesto Zedillo → Mexico".

Query generation. This process uses the discovered association rules to automatically expand the input question. Basically, it constructs four different queries from the original question. The first query is the set of key-

words (for instance, the set of named entities) from the original question, whereas the other three queries expand the first one by including some associated concept[3].

For instance, given a question such as "*Who was the president of Mexico during the Second World War?*", this process generates the following four queries: (1) "Mexico Second World War", (2) "Mexico Second World War 1945", (3) "Mexico Second World War United States", and (4) "Mexico Second World War Pearl Harbor".

Passage retrieval.[4] The purpose of this process is to recover the greatest number of relevant passages from the target document collections (EFE and Wikipedia). In order to do that it retrieves passages using all generated queries.

Passage integration. This process combines the retrieved passages into one single set. Its objective is to sort all passages in accordance with a homogeneous weighting scheme. The new weight of passages is calculated as follows:

$$w_p = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{|G_i|}\sum_{y\in G_i} C(p,y)\right)$$

Where w_p is the new weight of passage p, n indicates the number of words of the reference question, G_i is the set of all n-grams of size i from the question, and $C(p,y)$ is equal to 1 if the question n-gram y occurs in the passage p, otherwise it is equal to 0. This new weighting scheme favors those passages sharing the greatest number of n-grams with the question.

3.2 Question Classification

This module is responsible to define the semantic class of the answer of the given question. The idea is to know in advance the type of the expected answer in order to reduce the searching space to only those information fragments related to this specific semantic class.

Our prototype implements this module following a direct approach based on *regular expressions*. It only considers three general semantic classes for the type of expected answer: date, quantity and name (i.e., a proper noun).

3.3 Answer Extraction

Answer extraction aims to establish the best answer for a given question. It is based on a *supervised machine learning* approach. It consists of two main modules, one for attribute extraction and other one for answer selection. These modules were taken from our last year prototype [5].

Attribute extraction. First, the set of recovered passages are processed. The purpose is to identify all text fragments related to the semantic class of the expected answer. This process is done using a set of regular expression that allows identifying proper names, dates and quantities. Each identified text fragment is considered a "candidate answer".

In a second step, the lexical context of each candidate answer is analyzed with the aim of constructing its formal representation. In particular, each candidate answer is represented by a set of 17 attributes, clustered in the following groups:

1. Attributes that describe the complexity of the question. For instance, the length of the question (number of non-stopwords).
2. Attributes that measure the similarity between the context of the candidate answer and the given question. Basically, these attributes considers the number of common words, word lemmas and named entities (proper names) between the context of the candidate answer and the question. They also take into consideration the density of the question words in the answer context.
3. Attributes that indicate the relevance of the candidate answer in reference to the set of recovered passages. For instance, the relative position of passage that contains the candidate answer as well as the redundancy of the answer in the whole set of passages.

Answer Selection. This module selects from the set of candidate answers the one with the maximum probability of being the correct answer. This selection is done by a machine learning method, in particular, by a Naïve Bayes classifier.

---

[3] These concepts must be associated with all keywords of the given question.

[4] This process was carried out by the *Lucene* information retrieval system.

It is important to mention that the classification model (actually, we have three classifiers, one for each kind of answer) was constructed using as training set the questions and documents from previous CLEFs.

4 Answering Lists of Questions

This year's evaluation includes a new challenge: groups of related questions, where the first one indicates the focus of the group and the rest of them are somehow dependent from it. For instance, the pair of questions "*When was Amintore Fanfani born?*", and "*where was he born?*".

Our approach for answering this kind of questions is quite simple. It basically considers the enrichment of dependent questions by adding some keywords as well as the answer from the first (head) question.

The process for answering list of related questions is as follows:

1. Handle head questions as usual (refer to sections 2 and 3).
2. Extract the set of keywords (in our case the set of named entities) from the head question. This process is done using a set of regular expressions.
3. Add to all dependent questions the set of keywords and the extracted answer from the head question.
4. Handle enriched dependent questions as usual (refer to sections 2 and 3).

For instance, after this process the example question "*where was he born?*" was transformed to the enriched question "*where he was born? + Amintore Fanfani + 6 February 1908*".

5 Evaluation Results

This section presents the experimental results about our participation at the monolingual Spanish QA track at CLEF 2007. This evaluation exercise considers two basic types of questions, definition and factoid. However, as we mentioned in section 4, this year there were also included some groups of related questions.

From the given set of 200 test question, our QA system treated 34 as definition questions and 166 as factoid. Table 1 details our general accuracy results.

Table 1. System's general evaluation

	Right	Wrong	Inexact	Unsupported	Accuracy
Definition	30	-	4	-	88.23%
Factoid	39	118	3	6	23.49%
TOTAL	69	118	7	6	34.50%

|

It is very interesting to notice that our method for answering definition questions is very precise. It could answer almost 90% of the questions; moreover, it never replies wrong or unsupported answers. This result evidenced that Wikipedia has some inherent structure, and that our method could effectively take advantage of it.

On the other hand, Table 1 also shows that our method for answering factoid questions was not completely adequate (it only could answer 23% of this kind of questions). Taking into consideration that this method obtained 40% of accuracy on last year exercise [5], we presume that this poor performance was caused by the inclusion of Wikipedia. Two characteristics of Wikipedia damage our system's behavior. First, it is much less redundant that general news collections; and second, its style and structure makes lexical contexts of candidate answers less significant that those extracted from other free-text collections.

Table 2. Evaluation details about answering groups of related questions

	Right	Wrong	Inexact	Unsupported	Accuracy	NIL	
						Right	Wrong
Head questions	64	95	6	5	37.65%	3	35
Dependent questions	5	23	1	1	16.67%	0	5

Finally, Table 2 shows some results about the treatment of groups of related questions. It is clear that the proposed approach (refer to section 4) was not useful for dealing with dependent questions. The reason of this poor performance is that only 37% of head questions were correctly answered, and therefore, in the majority of the cases dependent questions were enriched with erroneous information.

6 Conclusions

This paper presented a QA system that allows answering factoid and definition questions. This system is based on a lexical data-driven approach. Its main idea is that the questions and their answers are commonly expressed using almost the same set of words, and therefore, it simply uses lexical information to identify the relevant passages as well as the candidate answers.

The proposed method for answering definition questions is quite simple; nevertheless it allowed achieving very high precision rates. We consider that its success is mainly attributable to its capability to take advantage of the style and structure of Wikipedia (the used target document collection). On the contrary, our method for answering factoid question was not equally successful. Paradoxically, the style and structure of Wikipedia caused detriment in most of its internal processes, since they are mainly based on lexical overlap and redundancy.

With respect to the treatment of groups of related questions our conclusion is that the achieved poor performance (16% in dependent questions) was consequence of a cascade error, in view of the fact that only 37% of head questions were correctly answered, and therefore, most dependent questions were expanded using incorrect information.

Acknowledgements. This work was done under partial support of CONACYT (Project Grant 43990). We also like to thanks to the CLEF organizing committee as well as to the EFE agency for the resources provided.

References

1. Agrawal R., and Srikant R. *Fast Algorithms for Mining Association Rules*. Proceedings of the 20th. VLDB Conference. Santiago de Chile, Chile, 1994.
2. De-Pablo-Sánchez C., González-Ledesma A., Martinez-Fernández J.L., Guirao J.M., Martinez P. and Moreno A., *MIRACLE's 2005 Approach to Cross-Lingual Question Answering*, In Working notes for the Cross Language Evaluation Forum Workshop (CLEF 2005), Vienna, Austria, September 2005.
3. Ferrés D. Kanaan S., González E., Ageno Al, Rodríguez H. and Turmo J., *The TALP-QA System for Spanish at CLEF-2005*, In Working notes for the Cross Language Evaluation Forum Workshop (CLEF 2005), Vienna, Austria, September 2005.
4. Gómez-Soriano J.M., Bisbal-Asensi E., Buscaldi D., Rosso P. and Sanchos-Arnal E., *Monolingual and Cross-language QA using a QA-oriented Passage Retrieval System*, In Working notes for the Cross Language Evaluation Forum Workshop (CLEF 2005), Vienna, Austria, September 2005.
5. Juárez-Gonzalez A., Téllez-Valero A., Denicia-Carral C., Montes-y-Gómez M., Villaseñor-Pineda L. *INAOE at CLEF 2006: Experiments in Spanish Question Answering*. Working Notes of CLEF-2006 Workshop. Alicante, Spain, September 2006.
6. Roger S., Ferrández S., Ferrández A., Peral J., Llopis F., Aguilar A. and Tomás D., *AliQAn, Spanish QA System at CLEF-2005*, In Working notes for the Cross Language Evaluation Forum Workshop (CLEF 2005), Vienna, Austria, September 2005.