QA on the Web: A Preliminary Study for Spanish Language

Alejandro Del-Castillo-Escobedo
INAOE, Mexico
delca@inaoep.mx

Manuel Montes-y-Gómez
INAOE, Mexico
DSIC, UPV, Spain
mmontes@dsic.upv.es

Luis Villaseñor-Pineda
INAOE, Mexico
villasen@inaoep.mx

Abstract

Finding accurate information on the web has become a challenge due to the increment in the number of documents available on line. Current search engines retrieve relevant documents to general –often short– user queries, but fail extracting answers to simple factual questions in natural language. This paper presents the basis of a statistical question answering system capable to find answers to factual questions in Spanish language from the web. This approach is supported on data redundancy rather than on sophisticated linguistic analyses of either questions and candidate answers. Preliminary results show that it is feasible to find concise and accurate answers from the web to factual questions made in Spanish language. The study also concludes that the available Spanish documents in the web are redundant enough in order to apply statistical methods like those described in this document in order to provide better mechanisms for information access.

1. Introduction

Nowadays there is a large amount of digital documents accessible from the web. These documents may satisfy almost every information need. However, without the appropriate mechanisms that help users to find the required information when they need it, all these documents are practically useless. In order to solve this dilemma several information access approaches have emerged. Two popular examples are: information retrieval (IR) and question answering (QA).

Information retrieval [2] addresses the problems associated with the retrieval of documents from a collection in response to a user query. The goal of an IR system is to search a text collection and return as result a subset of documents ordered by decreasing likelihood of being relevant to the given query. The most popular IR systems are the search engines for the web. For instance, Google, Altavista and Yahoo.

The current IR systems allow finding relevant documents for a given user need, but are incapable to return a concise answer for a specific information request [8]. The alternative to IR systems for resolving specific questions are the question answering (QA) systems. These systems are capable to answer questions formulated by the users in natural language. For instance, given the question like "Where is the Amparo Museum located?", a QA system responds "Puebla" instaed of returning a list of related documents to the Amparo Museum.

Recent developments in QA are mainly focused on answering factual questions (those having a simple named entity as the answer), and are mainly suited to English as the target language.

This paper presents the basis of a statistical QA system capable to find answers to factual questions in Spanish language from the web. This system is supported on the idea that the questions and their answers are commonly expressed using (almost) the same set of words. Therefore the answers may be extrated using simple lexical pattern matching methods, rather than sophisticated linguistic analyses of either questions and documents.

The rest of the paper is organized as follows. Section 2 briefly presents the current approaches on QA. Section 3 shows the architecture of our QA system and describes the methods for question reformulation and answer extraction. Section 4 presents some experimental results, and finally, section 5 exposes our conclusions and future work.

2. Related work

The first QA systems used information retrieval techniques to retrieve the most relevant text passages based on the keywords of questions and documents [1,6,7]. These kind of systems performed relatively well when retrieving 250-byte passages, but less well when they attempted to locate the concrete answers (restricted to 50-byte long).

Current aproaches use a variety of linguistic resources to help in understanding the questions and the matching

sections in the documents. The most common linguistic resources include: part-of-speech tagging, parsing, named entity extraction, semantic relations, dictionaries, and WordNet [5,9,10,14,16,17]. Despide of the promising results of these approaches, they have two main inconvenients: (i) the construction of such linguistic resources is a very complex task; and (ii) these resources are highly binding to a specific lenguage.

In recent years, the combination of the web growth and the explosive demand for better information access has motivated the interest in QA systems for the web [12,15]. Current approaches of QA on the web use a varity of linguistic resources for processing the questions and the web documents. However, the size of the web complicates their usage. As a result, new probabilistic methods based on the web redundancy have emerged [3,4,11,13].

This paper presents a statistical QA system capable to find answers to factual questions in Spanish language from the web. This system is primarily based on [3]. Its main idea is that the questions and their answers are commonly expressed using the same words, and that the probability of finding a simple (lexical) matching between them increases with the redundancy of the target collection. Therefore, given a question, our system generates several query reformulations manipulating the order of the words from the question. Then it sends each reformulation to a search engine, and collects the returned snnippets (document summaries). Finally, it extracts the most frequent n-grams (sequences of words) from the snnippets. Each n-gram is defined as a possible answer to the given question.

The present work extents that of Brill [3] in the sense that it studies the application of this approach for questions and documents in Spanish language. The main difference is on the query reformulation method. While Brill uses a lexicon to determine the part-of-speech of the question words as well as its morphological variants, we construct the quey reformulations just manipulating the word order without using any previous knowledge about the words. In addition, in order to compensate the lack of linguistic information, we propose three new statistical methods to extract the possible answers (i.e. to rank the frequent n-grams) from the snnippets.

3. System architecture

The figure 1 shows the general architecture of the proposed system. It consists of three major modules:

- Query reformulation
- Snnippets recollection
- Answer extraction

These modules are described in the following subsections.

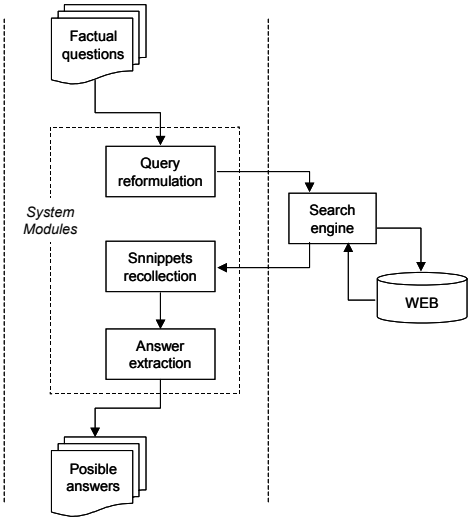

Figure 1. System arquitecture

3.1 Query reformulation

Given a question, this module generates a set of query reformulations. These reformulations are expressions that were probably used to write down the expected answer.

We performed several experiments in order to determine the most general and useful reformulations. The following paragraphs present those with the better results. All the cases are illustrated for the question: "¿Quién obtuvo el premio Nóbel de la paz en 1992?" (*Who received the Nobel Peace Prize in 1992?*).

In the algorithms described below, we represent a question Q as a set of words, i.e., $Q = \{w_0, w_1, \ldots, w_{n-1}\}$. Here w_0 corresponds to the wh-word, and n indicates the number of words of the question. On the other hand, we represent a query reformulation R as a symbol string. This string consists of words, spaces, and quotation marks, and it satisfies the format of a conventional search engine query. For instance the reformulation $R = w_i\ w_j$ corresponds to the query w_i AND w_j, while the reformulation R = "$w_i\ w_j$" to the query "$w_i\ w_j$".

First reformulation: "bag of words"

This reformulation is the set of non stop-words[1] of the question. It is built as follows:

1. For each $w_i \in Q \mid i \geq 1$
2. If w_i is not a stop word
3. $R \leftarrow w_i$
4. Save R

[1] The list of stop-words contains all the prepositions, conjuntions and articles.

The reformulation generated for the example query is:
obtuvo premio Nóbel paz 1992
(*received Nobel Peace Prize 1992*)

Second reformulation: "verb movement"

One of our first observations after examine a list of factual questions was that the verb is frequently used right after the wh-word. We also know that in order to transform an interrogative sentence into a declarative one is necessary to eliminate the verb, or to move it to the final position of the sentence. The resulting sentence is expected to be more abundant in the web that the original one.

In order to take adventage of this phenomenon, but without using any kind of linguistic resource, we propose to build a set of query reformulations eliminating, or moving to the end of the sentence, the first and second words from the question. The reason to include the second word was to consider the cases when an auxiliar verb exists.

The procedure to build these reformulations is as follows:

1. Set w_0 = ""
2. R = "$w_1\ w_2 \ldots\ w_{n-1}$"
3. Save R
4. For i = 1 to 2
5. R = "$w_{i+1}\ w_{i+2} \ldots\ w_{n-1}$"
6. Save R
7. R = "$w_{i+1}\ w_{i+2} \ldots\ w_{n-1}\ w_{i-1}\ w_i$"
8. Save R

Two examples of these kind of reformulations are:
"el premio Nóbel de la paz en 1992 obtuvo"
("*the Nobel Peace Prize in 1992 received*")
"premio Nóbel de la paz en 1992"
("*Nobel Peace Prize in 1992*")

Thrid reformulation: "components"

In this case the question is divided in components. A component is an expression delimited by a preposition. Therefore, a question Q with m prepositions is represented by a set of components C = $\{c_1, c_2, \ldots, c_{m+1}\}$. Each component c_i is a substring (subset of words) of the original query. New reformulations are defined combining these components as follows:

1. Determine the set of components C from Q
2. R = "c_1" "c_2" … "c_{m+1}"
3. Save R
4. For each permutation C′ of C
5. R = "$c'_1\ c'_2 \ldots\ c'_{m+1}$"
6. Save R

Some examples of this kind of query reformulations are:
"obtuvo el premio Nóbel" "de la paz" "en 1992"
("*received the Nobel Prize*" "*of Peace*" "*in 1992*")
"de la paz obtuvo el premio Nóbel en 1992"
("*of Peace received the Nobel Prize in 1992*")
"en 1992 obtuvo el premio Nóbel de la paz"
("*in 1992 received the Nobel Peace Prize*")

Fourth reformulation: "components without the first word"

In order to construct this set of reformulations we eliminate the main verb of the question (commonly expressed by the word w_1), and then we apply the method of reformulations by components. Some examples of these reformulations are:
"en 1992 el premio Nóbel de la paz"
("in 1992 *the Nobel Peace Prize*")
"el premio Nóbel" "de la paz" "en 1992"
("*the Nobel Prize*" "*of Peace*" "*in 1992*")

Fifth reformulation: "components without the first and second words"

In this case we suppose the presence of an auxiliar verb. Thus it is necessary to delete the words w_1 and w_2 from the question, and then to apply the method of reformulations by components. Two examples of this kind of reformulations are:
"premio Nóbel de la paz en 1992"
("*Nobel Peace Prize in 1992*")
"en 1992 premio Nóbel de la paz"
("*in 1992 Nobel Peace Prize*")

It is evident from the previous examples that some reformulations have not sense. For instance, "en 1992 de la paz obtuvo el premio Nóbel" ("in 1992 of Peace received the Nobel Prize"). However, the probability of finding snnippets from them is very low. On the contrary, most reformulations are syntactically correct, and they produce several snnippets containing the desiered answer.

It is also important to notice that some reformulations allow collecting snnippets using a synonim of the main verb of the question. For instance, the query reformulation "premio Nóbel de la paz en 1992" ("*Nobel Peace Prize in 1992*") lets to extract an snnippet with the phrase: "ganó el premio Nóbel de la paz en 1992" ("won the Nobel Peace Prize in 1992").

3.2 Snnippets recollection

Once the set of reformulations has been generated and sent to a search engine (currently we are using Google), this module collects the returned snnippets. Here is an example of an snnippet collected from the reformulation "el premio Nobel de la Paz en 1992" ("*the Nobel Peace Prize in 1992*"):

Edicion Especial Aniversario - 30 Años
… Ganador del premio Nobel de la Paz (1993). 7.- *Rigoberta Menchu* (1959) Lider indigena guatemalteca, recibio el premio Nobel de la Paz en 1992 …

3.3 Answer extraction

This module extracts all the n-grams (from unigrams to pentagrams) from the collected snnippets. Each n-gram is defined as a possible answer to the given question. Then, using some statistical criteria, it ranks the n-grams by decreasing likelihood of being the correct answer. The top five n-grams are selected as possible answers.

Following we describe three different methods for the n-gram extraction and ranking.

Method of relative frequency:

1. Extract the twenty most frequent unigrams.
2. Compute the relative frequency of each unigram.
   If G_1 is the set of frequent unigrams, and f_w indicates the frequency of occurence of the unigram $w \in G_1$, then the relative frequency of w is computed as follows:
   $$P_w = \frac{f_w}{\sum_{i \in G_1} f_i}$$
3. Determine all the n-grams, from bigrams to pentagrams, built from the set frequent unigrams.
4. Rank the n-grams based on their relative frequency.
   The relative frequency of a n-gram $g = (w_1 \ldots w_n)$ is computed as follows:
   $$P_g = \frac{1}{n} \sum_{i=1}^{n} p_{w_i}$$
   Here P_{w_i} indicates the relative frequency of the unigram $w_i \in g$.
5. Select the top five n-gramas as possible answers.

Applying this method we obtained these answers:

Menchu	0.05541
Rigoberta Menchu	0.05074
Rigoberta	0.04607
Rigoberta Menchu recibio	0.04005
guatemalteca Rigoberta Menchu	0.03860

Method of regular expressions:

1. Extract the twenty most frequent unigrams that satisfy a given typografic criteria (i.e., words starting with an uppercase letter, numbers and names of months).
2. Determine all the n-grams, from bigrams to pentagrams, built from the set frequent unigrams.
3. Rank the n-grams, in decreasing order, based on the number of words.
4. Select the top five n-grmas as possible answers.

For the example question we obtained the following n-grams:

Rigoberta Menchu Tum
Rigoberta Menchu Recibio
Rigoberta Menchu
Menchu Tum
Menchu Recibio

Method of regular expressions plus a compensated frequency:

1. Extract the twenty most frequent unigrams that satisfy a given typografic criteria (i.e., words starting with an uppercase letter, numbers and names of months).
2. Determine all the n-grams, from bigrams to pentagrams, built from the set frequent unigrams.
3. Rank the n-grams based on their compensated relative frequency.
   The compensated relative frequency of a n-gram $g(n) = (w_1 \ldots w_n)$ is computed as follows[2]:
   $$P_{g(n)} = \sum_{i=1}^{n} \sum_{j=1}^{n-i} \frac{f_{j(i)}}{\sum_{\forall x \in G_i} f_{x(i)}}$$
   where G_i is the set of n-grams of size i, $|G_i|$ indicates the cardinality of this set, j(i) is an n-gram j of size i contained in g(n), and $f_{j(i)}$ is the frequency of occurrence of this n-gram.
4. Select the top five n-grmas as possible answers.

Applying this method we obtained the following answers:

Rigoberta Menchu	0.07418
Rigoberta Menchu Tum	0.05753
Menchu	0.05541
Rigoberta Menchu Recibio	0.05143
Rigoberta	0.04607

It is important to notice that the method of relative frequency favors the short n-grams, while the method of regular expressions the large ones. The last method, the method of regular expressions plus a compensated frequency, combines the adventajes of both previous methods. However it applies a compensation factor in order to avoid favoring the short answers. This method produced the best results in our experiments.

4 Experiments

For experimental evaluations we used a set of 40 factual questions in Spanish language. These questions are of four types: who-questions, when-questions, where-questions and what-questions. Their answers were manually determined. Some examples of these questions are:

¿Quién es el Gobernador del Banco de México?
(Who is the governor of the Bank of México?)

[2] We introduce the notation g(n) for the sake of simplicty. In this case g(n) indicates a n-gram g of size n.

¿Cuándo fue lanzado el Apolo 11?
(When was the Apolo 11 launched?)
¿Dónde está la Laguna del Carpintero?
(Where is located the Carpintero lake?)
¿Cuál es el símbolo químico del Oro?
(What is the chemical symbol of gold?)

All runs were completely automatic. We used the system described at section 3. For each question we generated the five different kinds of query reformulations, and for each reformulation we collected, when posible, 50 snnippets. Finally we reported the top five better answers for each question.

The following tables show our results. Each table focuses on a different answer extration method, and compares the performance of all types of query reformulation approaches.

We reported the Mean Reciprocal Rank (MRR) of the first correct answer as well as the proportion of the questions correctly answered (i.e., the precision).

The MRR is computed as follows:

$$MRR = \frac{1}{n}\sum_{i=1}^{n} r_i$$

where n is the total number of test questions and r_i is the reciprocal of the rank (position in the answer list) of the first correct answer. For instance, if the correct answer is on the second position then $r_i = 0.5$, and if it is on the third then $r_i = 0.33$. In the case that the correct answer does not occur in the list of the top five n-grams, then $r_i = 0$.

Table 4. Results using the method of relative frequency

	QUESTION REFORMULATION				
QUESTION	Bag of words	Components	Components without the first word	Components without the first and second words	Verb movement
Who-question	60%	**70%**	50%	50%	**70%**
When-question	**70%**	10%	30 %	40%	50%
Where-question	**50%**	20%	40%	30%	**50%**
What-question	20%	0%	**40%**	20%	20%
Precision	**50%**	25%	40%	35%	48%
MRR	**0.3379**	0.1571	0.2271	0.1821	0.2613

Table 5. Results using the method of regular expressions

	QUESTION REFORMULATION				
QUESTION	Bag of words	Components	Components without the first word	Components without the first and second words	Verb movement
Who-question	80%	**90%**	80%	**90%**	80%
When-question	80%	10%	40 %	50%	**90%**
Where-question	**90%**	20%	10%	60%	**90%**
What-question	60%	30%	70%	**90%**	60%
Precision	78%	38%	63%	73%	**80%**
MRR	**0.5329**	0.2833	0.3817	0.4479	0.5196

Table 6. Results using the method of regular expressions plus a compensated frequency

QUESTION	QUESTION REFORMULATION				
	Bag of words	Components	Components without the first word	Components without the first and second words	Verb movement
Who-question	90%	80%	100%	**100%**	**100%**
When-question	**70%**	10%	40 %	50%	50%
Where-question	**100%**	30%	70%	60%	**100%**
What-question	80%	40%	80%	**90%**	70%
Precision	**85%**	40%	73%	75%	80%
MRR	0.6821	0.4	0.6404	0.6542	**0.7175**

From the results we conclude that:

- There is not a query reformulation method that produces the best results for all the kinds of questions. For instance, at table 4, the method of components obtained the best results for the who-questions, but the worst for the rest of the cases.
- The method of regular expressions plus a compensated frequency produced the best results, with a MRR as high as 0.717. However this method was not the best option for all the cases. For instance, it just answered 70% of the when-questions, while the method based on regular expressions responds the 80%.

5 Conclusions

This paper presents the basis of a statistical QA system capable to find answers to factual questions in Spanish language from the web. The main idea of the system is that the questions and their answers are commonly expressed using the same words, and that the probability of finding a simple (lexical) matching between them increases with the redundancy of the target collection.

The experiments have shown the huge potential of this apporach, which allow to find the answers without using any kind of linguistic resource.

Our experments also indicate that there is not a unique combination of query reformulation and answer extraction methods that can resolve all the kinds of questions. Our idea is to apply the best combination in accordance to the type of question.

As future work we plan to study the bahavior of the proposed system over a bigger question data set, such as those used in the CLEF competition. We are also interested in determining the correlation between the number of the extracted snnippets and the precision results.

6 Acknowledgements

We would like to thank CONACyT for partially supporting these work under grants 43990 and U39957-Y, and to the Secretaría de Estado de Educación y Universidades de España.

7 References

[1] J. Allan, M. Connel, W. Croft, F. Feng, D. Fisher and X. Li. "INQUERY and TREC-9", TREC-10, 2000.

[2] R. Baeza, B. Ribeiro (1999). Modern information retrieval. ACM Press, New York, Addison-Wesley, 1999.

[3] E. Brill, J. Lin, M. Banko, S. Dumais and A. Ng. "Data-intensive question answering". In TREC 2001, 2001.

[4] S. Buchholz. "Using grammatical relations, answer frequencies and the World Wide Web for TREC question answering". In TREC 2001, 2001.

[5] J. Chen, A. Diekema, M. Taffet, N. McCracken, N. Ozgencil, O. Yilmazel and E. Liddyl. "Question answering:CNLP at the TREC-10 question answering track". In TREC 2001, 2001.

[6] G. Cormack, A. Clarke, C. Palmer and D. Kisman. "Fast Automatic Pasaje Ranking (MultiText Experiments for TREC-8)". In TREC-8, 1999.

[7] M. Fuller, M. Kaszkiel, S. Kimberly, J. Sobel, R. Wilson and M. Wu. "The RMIT/CSIRO Ad Hoc, Q&A, Web, Interactive, and Speech Experiments at TREC-8". In TREC-8, 1999.

[8] L. Hirshman and R. Gaizauskas. "Natural Language Question Answering: The View from Here". Natural Language Engineering, Vol. 7, 2001.

[9] E. Hovy, L. Gerber, U. Hermajakob, M. Junk and C. Lin. "Question answering in Webclopedia". In TREC-9, 2000.

[10] E. Hovy, U. Hermajakob and C. Lin. "The use of external knowledge in factoid QA". In TREC 2001, 2001.

[11] C. Kwok, O. Etzioni and D. Weld. “Scaling question answering to the Web”. In the proceedings of the WWW Conference, 2001.
[12] J. Lin. “The Web as a resource for question answering: perspectives and challenges”. In Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002), 2002.
[13] G. Neumann and F. Xu. “Mining Answers in German Web Pages”. In Proceedings of The International Conference on Web Intelligence (WI 2003), Halifax, Canada, October 2003.
[14] J. Prager, E. Brown, A. Coden and D. Radev. “Question Answering by Predictive Annotation”. In Proceedings of SIGIR’2000, 2000.
[15] D. Roussinov, J.A. Robles-Flores. “Web Question Answering: Technology and Business Applications”. Proceedings of the Tenth Americas Conference on Information Systems, New York, New York, August 2004.
[16] M. Vargas-Vera and E. Motta. AQUA- Ontology-Based Question Answering System. In MICAI-2004, LNCS 2972, Springer, 2004.
[17] J. Vicedo. “Los Sistemas de Búsquedas de Respuesta desde una Perspectiva Actual”. Revista Iberoamericana de Inteligencia Artificial, 2004.