Experiments with Query Expansion in the RAPOSA (FOX) Question Answering System

Luís Sarmento, Jorge Teixeira and Eugénio Oliveira
Faculdade de Engenharia da Universidade do Porto
las@fe.up.pt teixeira.jorge@fe.up.pt eco@fe.up.pt

Abstract

In this paper we present the results of applying a statistical query expansion method on the retrieval stage of a QA system for Portuguese (RAPOSA). Our approach involves expanding queries for event-related or action-related factoid questions using a verb thesaurus automatically generated using information extracted from large corpora. We show that our expansion approach improves QA recall when compared with applying expansion based on a simple form of stemming, while simultaneously requiring the analysis of only 30% as many text snippets. However, we were not able to outperform the recall obtained using an even simpler expansion method, which nevertheless achieves lower precision and requires analyzing many more text snippets. We conclude by observing that a more thorough analysis of the usefulness of our approach on QA performance requires improving other stages of the QA pipeline which currently impose significant limitations on the overall performance of the system.

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, Questions beyond factoids

1 Introduction and Motivation

One of the most obvious limitation in the performance of many automatic question answering (QA) systems is their relatively low recall: for a very large proportion of questions many QA systems are unable to produce any answer at all. There are many possible causes for this low recall. To name just a few of the most common:

1. inability to parse the question. The QA system does not have rules to decompose the input question and the QA process is thus immediately ended.

2. inability to find text passages where candidate answers can be found. After parsing the question and choosing the relevant terms / keywords, the system is not successful in its

information retrieval (IR) stage, and is unable to retrieve text passages for any subsequent information extraction procedure.

3. inability to extract answer candidates from retrieved text passages. Since no candidate is found no answer is produced.

Problem 1) can be more or less relevant depending on the specific application scenario. For example, when performing QA under restricted application scenarios, systems can usually be prepared to parse a relatively small number of types of questions that they are expected to be asked. In these cases, problems at this stage can usually be solved by adding more parsing rules. On the other hand, for unrestricted question answering scenarios, this problems becomes much more severe, since users can pose all sorts of question. Interactive question answering scenarios involve even more difficulties as they require the need to resolve possible co-references among question and answers. Additionally, robustness to typos and malformed questions is required in practice and may be quite difficult to achieve.

Solving problem 3) involves developing more sophisticated information extraction strategies, which sometimes require using additional knowledge resource such as lexical databases, ontologies, or pre-built factoid databases. Research in this field focus on applying techniques such as named-entity disambiguation, factoid extraction and semantic relation discovery improve question answering.

We will mainly focus on problem 2), the inability of a QA system to retrieve the appropriate text passages for extracting answer candidates. In the current paper, we will describe how we tried to increased the overall recall of our QA system RAPOSA (http://pattie.fe.up.pt/RAPOSA/) by applying query expansion techniques to improve recall in the IR stage. Following RAPOSA participation in 2006 [18] and 2007 [19] editions of QA-CLEF, we believed that tackling this problem should be the next step.

First of all, from a strategic point of view there is much interest in solving problems related to IR stage because they transversally affect overall performance for all types of questions. Second, the goal we should seek is very well defined. As mentioned in [2], in the standard pipeline QA architecture (used by RAPOSA), during the IR stage *recall* in retrieval is more important than *precision*: subsequent processing stages may filter out uninteresting text passages obtained, but they will unable to extract the right answer candidates if the passage that contains the answer is not retrieved. Therefore, the main goal should be to increase recall in the IR stage. Finally, the problem scope is well localized and constrained inside a single stage of the QA system. This allows easier testing because it involves changing only a very specific stage of the pipeline without the need to change any of the others (either before or after the IR stage).

2 Related Work

Efforts to increase the recall in traditional IR system have focused on trying to circumvent *morphological*, *lexical* and *semantic* differences between the query terms and the terms in the documents. One would expect that such general IR techniques would be suitable for improving QA performance. However, there are a few important difference between general IR and QA-centric IR. While in general IR the retrieval unit is the *document*, in QA-centric IR the unit of retrieval is usually much smaller, such as for example, a paragraph, a sentence or even a smaller text fragment. Also, in QA-centric IR very fine-tuned ranking is not as crucial and in general IR, because further filtering will be performed along the QA pipeline. These are two important differences which motivate additional and specific efforts in QA-centric IR.

From a purely morphological point of view, there have been two main approaches. The first is to apply a *stemming* procedure at indexing time that will conflate morphological variations to the same index entry. At retrieval time, query terms are also stemmed and matched against the stems stored in the index. The second alternative involves indexing document terms directly (no changes are made to terms), and performing *morphological expansion* of the query terms at retrieval time, so they can be matched to more (unstemmed) index entries.

The benefits for QA of using applying such morphological-based techniques are not clear. In [5], a component evaluation of the Esfinge QA system for portuguese showed that *turning off* the stemming component improved the results, although only slightly, when attempting to answer the CLEF 2005 question set. Such slight improvement was observed for about half the types of factoid questions. The only exception was the performance for date question ("When... ?") which dropped significantly when stemming was turned off. In [2] the authors studied the effect of stemming and morphological expansion on document retrieval for the purpose of answering factoid questions. They concluded that indexing stemmed word forms actually lead to a decrease document retrieval recall, when compared to baseline (no stemming nor expansion). On the other hand retrieval-time query expansion tends to increase document retrieval recall at the cost of bringing more irrelevant documents and placing relevant documents in lower ranks.

Another technique aiming at improving recall in QA systems involves *expanding terms* in query to lexically or semantically related ones. For example, each term in the query can be expanded to the set of all know synonyms, by terms that generalize or specialize the concept, or by other related terms. These type of semantic expansion techniques require specific language or knowledge resources such as lexical databases (e.g. Wordnet [7]) and ontologies (e.g.: Cyc [11]).

In [9], Wordnet is used to expand the query terms found in the question by all terms contained in its synsets. A Boolean search expression is made by combining all expanded terms in a logical OR. The authors observe that such a direct approach may bring problems when synonyms are highly polysemous words. For example "high" can be a possible synonym of "high school" but since it is much more frequent (and polysemous) it will make the original "high school" term relatively less significant in the search expression. To account for this problem, document ranking is made by pondering the original terms twice as much as the synonyms. However, problematic situations arise when the original word is itself polysemous, leading to totally inappropriate expansions.

An approach that tries to solve some of the problem generated by ambiguity is presented in in [13]. The proposed technique uses a combination of Blind Relevance Feedback (BRF) and Word-Sense Disambiguation (WSD) named Sense-based Blind Relevance Feedback (S-BRF). In a first step, sets of paragraphs are retrieved using several combinations of the original terms found in questions. In a second step, the retrieved paragraphs are subject linguistic analysis (POS-tagging, multi-word recognition, named-entity recognition) and to word-sense disambiguation over WordNet senses. For each of the original question terms, the *most frequent sense* found on the retrieved paragraphs is chosen. Query expansion is then made by expanding only the previously found sense, using WordNet hierarchy (synonyms, hypernyms, holonyms, etc.). S-BRF leads to an increase of 7% in the precision of retrieval of answer-bearing documents, in relation to results obtained using "standard" morphological query expansion.

The work described in [6] show an example of how Cyc can be used in query expansion in a QA system, the MySentient system. MySentient uses Cyc to expand terms to its synonyms (including acronym expansion), to its specializations or generalizations, to possible instances or classes (e.g. "MasterCard" is an *instance-of* "credit card"), and to concepts related by meronomy/holonomy (*is-part-of* or *is-composed-by*). The authors claim that such expansion procedures improve system performance, although no performance figures are given.

When resources like Wordnet of Cyc are not available, systems may use smaller hand-crafted ontologies ([16]), or follow alternative approaches supported by statistical techniques. In [15] two query expansion methods based on statistical machine translation models are proposed, although focusing on a different yet related problem: *answer retrieval*. In the first method, a "translation model" from question words to answers words was learned using a large corpus of question-answer pairs. Using such translation model, each word questions can be expanded to a set of words that are expected to occur in the answer. A second method a english-chinese parallel corpus was used to learn english paraphrases. Query expansion was then achieved by adding in the query the n-best paraphrases of the original terms. Authors report significant improvement over both methods over two alternative methods, [10] and [20].

We follow a different alternative to deal with the lack of standard Wordnet-like resources for doing query expansion in Portuguese. Our approach consists in automatically building thesaurus using statistical processing of a large corpus. Such thesaurus should contain information about

synonyms or strongly related words, which can then be used to expand queries. Several different approaches have been tried for automatic building thesaurus from corpora. These usually involve either finding distributional similarities between words and clustering (e.g. [12] [3]), or mining text with patterns (manually defined or automatically learned) that allow identifying specific semantic relations (e.g. [14] [8]). Since the goal using the thesaurus is query expansion within a QA-pipeline, we believe that the problem of thesaurus generation can in fact be relaxed: having very precise semantic (namely synonymy) might not be an absolute requirement. We have thus used a rather simple method, to be described in Section 4, for building such "relaxed thesaurus".

3 RAPOSA

The architecture of RAPOSA has been described in [19]. Briefly, RAPOSA is a pipeline QA systems composed by six main modules:

1. **Question Parser**: identifies the type of question, the expected semantic type(s) of the answer, its arguments, possible restrictions and other relevant keywords. Morphological analysis is made using JSpell ([1]).

2. **Query Generator**: the Query Generator selects which term from the question must necessarily occur in target text snippets and which terms are optional.

3. **Snippet Searcher**: takes the queries and searches several available text bases to retrieve text snippets where candidate answer may eventually be found. The retrieval unit is a *text snippet*, which, depending on the text base queried, can be a sentence or a paragraphs (but not usually a complete document).

4. **Answer Extractor**: tries to identify candidate answers in text snippets using two possible strategies. The first one is based on a set of *context evaluation rules* that search for given answer patterns. The second is called *simple type checking strategy* and extracts the most frequently found candidates whose type is compatible with the expected semantic type of the answer.

5. **Answer Fusion**: the role of the Answer Fusion module is to cluster lexically different but possible semantically equivalent (or overlapping) answers in to a single "answer group". At this moment, this module is not yet developed and it simply outputs previously chosen candidates.

6. **Answer Selector**: selects one of the candidate answers produced by the Answer Fusion module and choses the supporting text / answer justification among previously extracted text snippets.

The focus of our work this year has been the Query Generator. Up to CLEF 2007 this module had a very simple role. Using the information given by the Question Parser, the Query Generator selected terms that are required to be found in text snippets where answer might eventually be found, and which terms are optional. For example, all named-entities found in questions are *required* to occur in the text snippets.

A very rudimentary query expansion technique was also applied in this module. For terms that were not identified as named-entities, suffixes were stripped to obtain a simple form query expansion by *pseudo-stemming*. Suffixes were considered to be the last 2-4 characters of terms with more the 5 characters long. These were substituted by wild-cards in order to generalize the query and obtain more text snippets to be further analyzed down the pipeline.

Obviously, being a purely lexical transformation process, this method has several limitations and problems. In fact, *pseudo-stemming* should have all the possible problems of standard stemming, worsened by the fact the our current retrieval system is not using any form of relevance ranking because it is a simple text database system. If too many snippets are returned, which

can happen when pseudo-stemming generates a very frequent stem, only the first text snippets are kept and used for answer extraction. The problem is that keeping only the first N_{max} snippets increases the chances of not forwarding any relevant information to the Answer Extraction module. We believe this to be the one of the major sources of nil (and also incorrect) answers in RAPOSA. In the next section we will explain how we attempted to solve this problem.

4 Expansion using an Automatically Generated Thesaurus

In this work, we focused on expanding queries for answering factoid questions related to actions or events, such as for example "Who killed J.F.K?" or "When did Brazil last won the World Cup?". In this type of questions, in which an action or event that is central to the answer is directly or indirectly mentioned, verbs have the key role in retrieving the relevant text snippets for finding answer candidates. Therefore, expanding the verb to semantically equivalent verbs, and ideally also to verbal and nominal paraphrases, should help increasing retrieval recall without adding too many irrelevant or noisy text snippets. For instance, for the question "Who killed J.F.K?", answers could be found in texts containing both "J.F.K." and forms of the verb "to kill" but also in texts where semantically equivalent or related verbs occur, such as "to murder", "to assassinate", "to shoot", etc. Such expansion requires a large coverage verb thesaurus.

4.1 Building a Verb Thesaurus

For building a verb thesaurus for Portuguese we followed a simplified approach of that described in [12]. The basic principle is that "similar" word should have "similar" distributional properties under a given context. Such context can be defined, for example, by the set of grammatical relations that such word establishes with other words, or even simpler, by the set of words with which the word co-occurs within a predefined lexical window (e.g. two words to each side).

For the case of verbs in Portuguese, one can intuitively see that much of the information capable of describing the semantic properties of a verb can be found in the two following words. Within this context we can observe many of the more relevant verb-object relations as well as the most typical adverbial constructions. We used n-gram information compiled from a large web-corpus of about 1000 million words [17] to obtain a distributional description of verbs in portuguese. N-gram information is not POS-tagged but we used the following regular expression pattern over the 4-gram list:

```
token1 = "para" & token2 ends_with(ar|er|ir|or) & token3 = * & token4 = *
```

to constrain 4-grams so that token at position 2 almost surely referes to a verb in the infinitive form (equivalent in english "[to] [verb in infinitive] [*] [*]"). Thus, for the list of mined verbs we obtained the distributional profile of the two following words which can be represented by tuples of the form (verb, *token3*, *token4*, frequency). We had 293,130,369 distinct 4-grams available from which 435,702 (approx. 0.15%) matched the previous pattern. These 4-grams correspond to 6862 distinct words at position *token2*, corresponding mostly to verbs, as expected.

Using such information we can now describe each verb v_i using a feature vector $[v_i]$ containing information about the previously found co-occurring words. $[v_i]$ belongs to a space with 174,764 dimensions, each dimension being defined by a distinct co-occurring bigram (found at positions *token3 token4*). Features were weighted using Mutual Information [4]. This weighting function allows to consider global information about the features in each vector, demoting the importance of features that occur in many vectors, and that should be considered less relevant. Therefore, Mutual Information helps to reduce the influence of noisy features.

The next step for building a thesaurus is to find the top-k closest vectors for each vector (i.e. for each verb). Vectors were compared using the cosine metric. Although we are dealing with a moderate size the vector set (less than 7000 vectors), performing an "all-against-all" comparison between the vectors can still take too long. Alternatively, we built a feature index and computed the contribution of each feature to the similarity of each pair of vectors that shares such feature.

All features that occurred in more than 5% of the items were considered noisy and were left out. These very frequent features represent a significant part of the computational effort because they are common to many vectors. However, they tend to have a very low weight when pondered by Mutual Information, so in practice they should only contribute marginally to the values of similarity. Globally, this feature filtering procedure leads to substantial computational savings. and allowed us to quickly compute a very close approximation of the results that would be obtained if an "all-against-all" comparison was followed. The current version of our automatically generated thesaurus can be queried and visualized via: `http://pattie.fe.up.pt/cgi-bin/tep/word_map.pl`.

4.2 Query Expansion Procedure

We only apply query expansion to factoid questions that explicitly refer to an *action* or to an *event*. For these question the verb has the central role in finding the answer, so providing alternatives for it should lead to improvement in retrieval recall. For example, we try to apply query expansion to questions such as "Em que ano *houve* um terramoto no Irão", "Quando *começou* o Neolítico?" "Quantas vezes *ganhou* Portugal a Taça Davis?", but not "Qual a capital de estado de Nova York?" (examples taken from QA-CLEF 2008 question set).

After the question being processed by the Question Parsed module, the type of the question and all its main components have been identified. Then for factoid questions referring to an action or event, the following procedure is executed to expand the verb:

1. take the verb and find its radical;
2. using the statistical thesaurus, find $n_{top} = 5$ related verbs;
3. apply pseudo-lematization to source and related verbs by substituting last character by a wild card (to match most possible verb inflections);

For example, for the question "Quantas vezes *ganhou* Portugal a Taça Davis", the previous procedure would be instantiated to:

1. "ganhou": "ganhar";
2. "ganhar": "poupar", "vencer", "conquistar", "perder", "ter" ("angariar", "dar", "disputar")
3. 'ganh*", "poup*", "venc*", "conquist*", "perd*", "ter"

All the options that result from expansion are then combined in a boolean OR to make the actual query. One can see from the results of Step 2 that there are some possibly problematic situation in the set of expanded verbs but two of the 5 expansion options - "vencer" and "conquistar" - seem to be clearly correct. Problematic situations that can occur include those related to the ambiguity of the source verb ("poupar") or the difficulty in identifying opposite senses / antonyms ("perder").

For the sake of comparison, if we performed expansion using the OpenOffice thesaurus for Portuguese - a manually created thesaurus - the verb "ganhar" would have only one expansion, "lucrar", which almost surely would not provide any benefit to retrieval, because it refers to a different sense of the original verb. This example clearly shows the type of problems that may arise from using manually created dictionaries for query expansion, and illustrates how statistical thesaurus, despite obvious errors, may actually lead to higher recall and a much less biased semantics.

5 Results at QA@CLEF 2008

In order to test the impact of our statistical query expansion approach in the performance of RAPOSA, we submitted two runs for evaluation at the 2008 CLEF QA track. RAPOSA was configured to execute one of two types of query expansion for factoid action / event factoid questions only (as explained before). The two distinct runs this year are:

- Run R^0: in this run, query expansion is made through pseudo-stemming, i.e. by substituting the last characters of the verb by a wild-card. This was the method used in QA@CLEF 2007 version of RAPOSA, and in the current evaluation should be considered the *baseline* run. Up to a maximum of 150 snippets could be retrieved and analyzed.
- Run R^+: in this run, query expansion is made using the procedure described in Section 4.3: the verb is expanded to 5 statistically related verb options using the thesaurus and then pseudo-lematization is applied to each of the resulting options. A maximum of 25 snippets could be retrieved per expanded query, leading to a maximum of 150 snippets (for the original verb + up to 5 expanded verb options).

The global results obtained in CLEF 2008 for these runs were:

Run	Right	Wrong	ineXact	Unsup.	Accuracy
R^0	25	169	4	2	12.5 %
R^+	29	165	4	2	14.5 %

Table 1: Global results obtained on each run, R^0 and R^+, for the 2008 test set

When looking at results from Table 1 it is important to take into account that RAPOSA is not trying to answer all questions in the test set. RAPOSA is not considering list questions (10 in 2008 test set) nor questions that require anaphoric resolution. For each "cluster" of anaphorically related questions, RAPOSA only tries to answer the first question in the cluster, because is the only one that is not dependent on anaphoric resolution. Basically, this means that RAPOSA did not even try to answer 51 dependent questions from the 2008 test set. In the following sections we will focus our analysis taking into account these factors. Still, we will make a brief comparison between the performance of RAPOSA in 2007 and 2008.

5.1 2008 vs. 2007

The results in 2008 were somehow disappointing because RAPOSA was able to correctly answer only 25 questions in run R^0, whereas in 2007 RAPOSA answered 38 questions, with the exact same working configuration. When comparing the 2007 and 2008 test sets we noticed that there is not significant difference between the number of factoid and the number of definition questions in both test sets. Also, the number of anaphorically related questions is approximately the same: 50 in 2007 vs. 51 in 2008.

A closer look at the 2007 and 2008 results shows that RAPOSA performance improved slightly this year for factoid questions. However performance for definitions questions dropped abruptly: from 16 correct answers in 2007 to just 4 in 2008. This suggests that this year's test set has harder definition questions than in 2007. The most important difference seems to be that the pattern of definition questions changed significantly in 2008. While most definition question in 2007 where person related definitions (e.g.: "Quem é George Vassiliou?" / "Who is George Vassiliou?"), the 2008 definition questions addressed canonical definitions (e.g. "O que é uma cítara?" / "What is a zither?"). In our opinion this is a more realistic scenario for QA evaluation, and is very appropriate for the Wikipedia collection. However, our work this year did not focus on these issues and, thus, RAPOSA was not prepared for answering canonical definitions. That will be subject of future work.

5.2 Evaluating Query Expansion

Since our query expansion method is to be applied only to a specific type of factoid questions, for the purposes of evaluating query expansion it only makes sense to observe results over that smaller subset of factoid question in test set. Nevertheless, Table 2 shows the performance of run R^0 and R^+ over all 162 factoid questions contained in the 2008 test set. From all 162 factoid

Run	Right	Wrong	ineXact	Unsup.	Accuracy(all)
R^0	21	138	1	2	12.96 %
R^+	25	134	1	2	15.43 %

Table 2: Results obtained for factoid questions, R^0 and R^+, for the 2008 test set

questions RAPOSA was only able to parse 90 questions, i.e. only 56%. This relatively low parsing performance is due to the lack of a more complete base of parsing rules and to some unexpected encoding problems. For the correctly parsed questions, our query expansion method could be applied in 41 questions. Table 3 contains some statistics regarding performance on those 41 questions, differentiating results of using each of the two individual text collections - the XML dump of portuguese Wikipedia [1] and CHAVE news collection - for extracting answers candidates. Columns indicate then number of questions for which no text snippet was found, *"0 snip."*, the total number of snippets found for all questions, *"# snip."*, the average number of snippets anaylized for each question (i.e. when at least one snippet was found), *"avg. snip."*, the number of nil answers, *"nil answers"*, and the number of correct non-nil answers.

Collection	mode	0 snip.	# snip.	avg. snip.	nil ans.	correct non-nil ans.
Wikipedia	no expansion	28	296	22.77	31	1
Wikipedia	expansion	21	154	7.7	24	5
CHAVE	no expansion	29	303	25.25	32	1
CHAVE	expansion	24	173	10.18	27	1

Table 3: Statistics for the 41 factoid questions when verb is removed from the query

From Table 3 one can make the following observations:

- query expansion allowed RAPOSA to find up to 4 additional answers, despite the fact that much less snippets were retrieved (and analyzed) for both collections;
- query expansion reduced significantly the number of questions for which no single text snippet was found;
- query expansion reduced significantly the number of nil answers, although this only allowed to improve the number of correct non-nil answers when the Wikipedia collection was used for extracting results;
- query expansion seems to have more success when used in the Wikipedia collection: all relevant parameters improved relatively more for Wikipedia than for CHAVE collection, when query expansion was used, but we are not sure if additional answers could in fact be found in CHAVE.

For better understanding why query expansion helped we looked in more detail to 4 questions that were correctly answered when using query expansion and Wikipedia. Table 4 shows the questions, the set of expanded verbs and the number of snippets retrieved. Verbs that helped retrieving snippets from which the correct answer was extracted are written in bold.

For question 0015 and 0091, query expansion allowed retrieving one relevant snippet for each of the indicated expanded verbs. In the case of question 0015, the connection between the expanded verb and the original verb is very strong (they are *quasi-synonym* in this context) and the positive effects of query expansion on the result are easy to understand. For question 0091 the verb at stake, *ficar*, is highly polysemous so the expansion provided by the automatically generated

[1] Available from the University of Amsterdam: `http://ilps.science.uva.nl/WikiXML`

#	Question	Verb Expansion	# snip.
0007	Em que ano **houve** um terramoto no Irão?	*ter*, garantir, obter, permitir, estabelecer	1
0015	Quem **escreveu** Fernão Capelo Gaivota?	ler, criar, ver, **publicar**, seleccionar	1
0063	Quem **criou** Descobridores de Catan?	construir, desenvolver, obter, produzir, *ter*	4
0091	Em que ilha **fica** Sapporo?	**estar**, viver, andar, **trabalhar**, entrar	2

Table 4: The 4 additional questions that were correctly answered in R^+.

thesaurus looks less accurate, yet still quite reasonable. Nevertheless, it was enough for retrieving two snippets where the correct answer could be found.

For questions 0007 and 0063, the snippets that provided to the right answers were apparently retrieved when using the verb "ter". However, because our current retrieval system actually ignores any search term with less that 4 character, expansion with "ter" is virtually equivalent to *excluding* the verb from the query. This lead to the generation of less restrictive queries containing only the arguments of the question ("terramoto and Irão" and "Descobridores de Catan"), allowing to retrieve more snippets, in which the correct answers ended up being found.

Thus, we decided to experiment what happens if we totally remove the verb from the query, for each of the 41 question at test. Results are presented in Table 5. For the Wikipedia collection the correct answers found are exactly the same as the ones found with query expansion. For CHAVE collection we actually found one additional correct answer. In both cases, when comparing with results obtained with query expansions, there are also much less nil answers but RAPOSA ended up analyzing about 3.5 times more snippets.

Collection	mode	0 snip.	# snip.	avg. snip.	nil ans.	correct non-nil ans.
Wikipedia	verb removal	15	689	26.5	19	5
CHAVE	verb removal	16	901	36.0	21	2

Table 5: Statistics for the 41 factoid questions in which our query expansion method could be applied, when applying *verb removal* instead of query expansion

5.3 Discussion

Results confirm that retrieving and analyzing more snippets does helps RAPOSA finding more corrects answers (higher recall). This seems to be more the case when the number of existing snippets available for extracting answers to a given question is *very low*. In those situations, if the answer is in fact included in one of the few retrieved snippets, RAPOSA seems to be able to find the it, specially because there are also less chances of choosing a wrong answer among the few possible candidates found. Questions 0007, 0015, 0063 and 0091 illustrate such type of situations.

The big question is: is our query expansion approach useful? We first need to focus on the different reasons why query expansion helped. There were two cases where query expansion was clearly successful: question 0015 and 0091. For questions 0009 and 0063 our expansion method indirectly caused the removal of the verb from the query (due to the minimum 4 character threshold on our retrieval index) and this lead to retrieving more snippets and the correct answer. We shall consider these two later cases as resulting from luck, so we will not consider them as successful examples of our expansion method. We believe, however, that a different indexing mechanism (combined with a better strategy for building the statistical thesaurus) could eventually help to

solve this problem and, thus increase the number of cases where our expansion approach can be considered valid.

Thus, if we consider only the cases where our expansion method clearly worked as intended, we conclude that query expansion is marginally beneficial in comparison with performing no expansion. When comparing with simple verb removal, query expansion does not help to achieve as many correct answers, but it also does not require analyzing so many text snippets for extracting candidates answers. However, because RAPOSA is producing many nil and incorrect answers it becomes very hard to access if expansion is really beneficial: do the much fewer snippets retrieved really contain the correct answers, even when RAPOSA is not able to find them (i.e. the problems occur later in the pipeline)? What would be the effect of query expansion if RAPOSA was able to correctly parse more questions? And, if another thesaurus was built, from a larger corpus or using linguistically informed methods, would that improve the results significantly?

At this point we can only conclude that there are too many important and basic limitations in RAPOSA at the level of question parsing, candidate extraction and candidate selection, to allow a thorough evaluation of the query expansion method we propose. Nevertheless, we believe that our expansion method can help improve RAPOSA performance when more of these problems are solved.

6 Conclusion and Future Work

In this paper we presented a query expansion system intended to improve recall in answering a event-related or action-related factoid questions. Expansion is achieved by using a verb thesaurus, automatically generated from corpora. We have showed that our expansion method provides a marginal increase in the recall of our question answering system when compared with not using any form of expansion. Query expansion seems to help RAPOSA in finding additional correct answers, while reducing the number of text snippets retrieved and analyzed. However, because of many other limitations in RAPOSA, it is not yet clear how much contribution is actually provided by our method, when compared to the extremely simple strategy of increasing recall by removing the verb from the query.

Future work will necessarily focus on improving specific stages of RAPOSA, namely implementing better Question Parsing and Snippets Searcher modules. We are currently implementing from scratch a generic wide-spectrum semantic analyzer system for portuguese, which will replace our current named-entity recognition system. The new analyzer will help to improve recall of the question parsing module so that we can increase the number of questions RAPOSA tries to answer. Also, a better analysis will help to achieve more efficient text preprocessing and indexing. We are also experimenting using native XML databases for storing pre-processed source collections, and for retrieving text snippets based on new semantic annotations. After these improvements in RAPOSA we will repeat these experiments with query expansion, possibly with thesaurus generated using information gathered by our new semantic parser.

7 Acknowledgments

This work was partially supported by grant SFRH/BD/ 23590/2005 from Fundação para a Ciência e Tecnologia (Portugal), co-financed by POSI.

References

[1] J.J. Almeida and Ulisses Pinto. Jspell – um módulo para análise léxica genérica de linguagem natural. In *Actas do X Encontro da Associação Portuguesa de Linguística*, pages 1–15, Évora 1994, 1995.

[2] Matthew W. Bilotti, Boris Katz, and Jimmy Lin. What works better for question answering: Stemming or morphological query expansion? In *Proceedings of the Information Retrieval for Question Answering (IR4QA) Workshop. SIGIR 2004*, Sheffield, England, July 2004.

[3] Timothy Chklovski and Patrick Pantel. VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. In *In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, Barcelona, Spain, 2004.

[4] Kenneth Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22?29, 1990.

[5] Luís Costa and Luís Sarmento. Component evaluation in a question answering system. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*, Genoa, Italy, May 2006.

[6] Jon Curtis, Gavin Matthews, and David Baxter. On the effective use of cyc in a question answering system. In *IJCAI Workshop on Knowledge and Reasoning for Answering Questions (KRAQ'05)*, Edinburgh, Scotland, 2005.

[7] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database.* The MIT Press, Cambridge, MA, 1998.

[8] Marti A. Hearst. Automatic Acquisition of Hyponyms from Large Text Corpora. In *COLING*, pages 539–545, 1992.

[9] Edward Hovy, Laurie Gerber, Ulf Hermjakob, Michael Junk, and Chin-Yew Lin. Question answering in webclopedia. In *Proceedings of the 9th Text REtrieval Conference*, pages 655–664, Gaithersburg, MD, USA, November 2000.

[10] Valentin Jijkoun and Maarten de Rijke. Retrieving answers from frequently asked questions pages on the web. In *CIKM '05: Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pages 76–83, New York, NY, USA, 2005. ACM Press.

[11] Douglas B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.

[12] Dekang Lin. Automatic Retrieval and Clustering of Similar Words. In *Proceedings of COLING-ACL 1998*, volume 2, pages 768–773, Montreal, 1998.

[13] Matteo Negri. Sense-based blind relevance feedback for question answering. in , sheffield, uk, july. In *SIGIR-2004 Workshop on Information Retrieval For Question Answering (IR4QA)*, Sheffield, UK, July 2004.

[14] Patrick Pantel and Deepak Ravichandran. Automatically Labeling Semantic Classes. In *HLT-NAACL*, pages 321–328, 2004.

[15] Stefan Riezler, Alexander Vasserman, Ioannis Tsochantaridis, Vibhu O. Mittal, and Yi Liu. Statistical machine translation for query expansion in answer retrieval. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, June 23-30 2007.

[16] José Saias and Paulo Quaresma. The Senso question answering approach to portuguese qa@clef-2007. In *Proceedings of Cross Language Evaluation Forum (CLEF 2007)*, Budapest, Hungary, September 2007.

[17] Luís Sarmento. BACO - A large database of text and co-occurrences. In Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odjik, and Daniel Tapias, editors, *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*, pages 1787–1790, Genoa, Italy, 22-28 May 2006.

[18] Luís Sarmento. A first step to address biography generation as an iterative QA task. In Carol Peters, Paul Clough, Fredric C. Gey, Douglas W. Oard, Maximilian Stempfhuber, Bernardo Magnini, Maarten de Rijke, and Julio Gonzalo, editors, *7th Workshop of the Cross-Language Evaluation Forum, CLEF 2006. Alicante, Spain, September 2006. Revised Selected papers*, Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, 2007.

[19] Luís Sarmento and Eugénio Oliveira. Making raposa (fox) smarter. In Alessandro Nardi and Carol Peters, editors, *Working Notes of the Cross-Language Evaluation Forum (CLEF) Workshop 2007*, Budapest, Hungary, September 2007.

[20] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceeding of SIGIR'96*, Zurich, Switzerland, 1996.