Assessing the Impact of Thesaurus-Based Expansion Techniques in QA-centric IR

Luís Sarmento[1], Jorge Teixeira[2] and Eugénio Oliveira[3]

Faculdade de Engenharia da Universidade do Porto
Laboratorio de Inteligência Artificial e Ciências de Computadores
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal
las@fe.up.pt[1] jft@fe.up.pt[2] eco@fe.up.pt[3]

Abstract. In this paper, we assess the impact of using thesaurus-based query expansion methods, at the Information Retrieval (IR) stage of a Question Answering (QA) system. We focus on expanding queries for questions regarding *actions* and *events*, where verbs have particularly important roles. Two different thesaurus are used: the OpenOffice thesaurus and an automatically generated verb thesaurus. The performance of thesaurus-based methods is compared against what is obtained by (i) executing *no expansion* and (ii) applying a simple query generalization method. Results show that using thesaurus-based approaches helps to improve retrieval *recall*, while keeping satisfactory *precision*. However, we confirm that positive impact for the final QA performance is mostly achieved due to increase in *recall*, which can also obtained by alternative and simpler methods. Nevertheless, thesaurus-based expansion helps controlling the number of text passages retrieved, thus selectively reducing the computational load in the answer extraction stage.

1 Introduction

One of the most obvious limitations of many automatic question answering (QA) systems is their relatively low recall: for a large proportion of questions many QA systems are unable to produce any answer at all. Some of the most frequent reasons have to do with insuccess at the Information Retrieval stage, i.e. with the inability to find text passages from which candidate answers can be found. Thus, there is much interest in solving recall problems at the IR stage of a QA system because they affect performance for *all types* of questions. However, QA-centric IR has a set of requirements that make it different from generic IR. First, in generic IR the retrieval unit is the *document*, while in QA-centric IR the unit of retrieval is usually a smaller text passage, such as a paragraph or a sentence. Second, in QA-centric IR, fine-tuned ranking is not as crucial and in general IR, because further filtering operations are performed down the QA pipeline. As mentioned in [1], in standard pipeline QA architectures improving *recall* in IR stage is often more important than improving *precision*: subsequent processing stages in the QA pipeline may filter out uninteresting text passages obtained, but they will never be able to extract the right answer candidates if the passage that contains the answer is not retrieved.

In this paper, we wish to extend previous work [2] on evaluating the impact of applying thesaurus-based query expansion techniques at the IR stage of a QA system, with the goal of improving the QA performance for factoid questions about *actions* or *events*, such as for example "Who killed J.F.K?" or "When did Brazil last won the World Cup?". These type of questions involve an explicit references to action through specific verbs (e.g. "to kill", "to win"), which have key roles in retrieving relevant text passages. One expects to increase the chances of finding correct answers if semantically equivalent verbs are used in the retrieval of text passages

2 Related Work

There have been several approaches to query expansion in QA-centric IR. One of the simplest is to apply a *stemming* procedure at indexing time that conflates morphological variations to the same index entry. At retrieval time, query terms are also stemmed and matched against the stems stored in the index. Another 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 [3], a component evaluation of the Esfinge QA system for Portuguese showed that *turning off* the stemming component *improved* slightly the results. Such slight improvement was observed for about half the types of factoid questions, except for date question ("When... ?") where performance dropped significantly when stemming was turned off. In [1] the authors conclude 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.

The work described in [4] 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. In [5], Wordnet is used to expand terms found in the question by all terms contained in their 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 also 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 [6]. 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. 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.

When resources like Wordnet of Cyc are not available, system may follow alternative approaches supported by statistical techniques. In [7] 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 question word can be expanded to a set of words that are expected to occur in the answer. In a second method a English-Chinese parallel corpus was used to learn English paraphrases. Query expansion was made by adding in the query the n-best paraphrases of the original terms.

3 Question-Answering Framework: RAPOSA

The Question Answering system that we will use to evaluate the impact of query expansion, RAPOSA, follows a classical pipeline architecture, composed of five main modules: the Question Parser, the Query Generator, the Passage Retriever, the Answer Extractor, and the Answer Selector. Since RAPOSA has been extensively described elsewhere ([8] and [9]), we will focus only on the Query Generator module. The *Query Generator* may generate queries according to two different strategies: using *pseudo-stemming* and using thesaurus-based expansion. Generation using *pseudo-stemming* involves a simple lexical process: for terms *not* identified as named-entities, the last 2-4 characters are stripped and substituted by wild-cards. Query generation using thesaurus-based expansion relies on a pre-existing verb thesaurus. For factoid questions that explicitly refer to *actions* or *events*, expansion is made by first taking the source verb and finding its lemma, v_s, and then using the verb thesaurus to find up to n verbs related to v_s: v_{r1}, v_{r2} ... v_{rn}. Finally pseudo-stemming is applied to terms in the query, including source verb v_s and related verbs v_{r1}, v_{r2} ... v_{rn}, in order to match most possible verb inflections.

4 Thesauri for Expansion

We have two thesaurus available for supporting verb expansion: the OpenOffice thesaurus for Portuguese and an automatically generated verb thesaurus. The OpenOffice thesaurus [1] contains 4002 synsets for adjectives, nouns and names. We took the verb synsets and indexed each verb in it to produce (verb → list of all equivalent verbs) mappings for all verbs. We obtained 2783 such mappings.

The automatically generated verb thesaurus was built following a simplified approach to that described in [10]. The basic principle is that "similar" word should have "similar" distributional properties under a given context. 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 to obtain a distributional description of verbs in Portuguese ([11]). N-gram information in this collection is not POS-tagged, but because verb forms in Portuguese are inflected, they can frequently be unambiguously distinguished using a dictionary. We used a dictionary to filter out ambiguous verb forms so that only the 3-grams (w_1, w_2, w_3) matching the following selection pattern were chosen: (w_1 = [unambiguous verb form] & w_2 = * & w_3 = *). Verb forms (at w_1) were lemmatized in order to obtain tuples of the form (verb lemma, w_2 w_3, frequency), and feature information from the various forms was merged. There are 173,607,555 distinct 3-grams available in n-gram database, and 14,238,180 (8.2%) matched the pattern, corresponding to 4,958 verbs. Verb v_i is described using a feature vector $[v_i]$ containing the pre-compiled information about co-occurring words. Vector features were weighted using by Mutual Information function, and vectors were then compared using the cosine-metric, to obtain the list of nearest neighbours. Verbs corresponding to the such nearest neighbours of $[v_i]$ are considered the "verb equivalents" of v_i. The current version of our automatically generated thesaurus can be queried and visualized via: `http://pattie.fe.up.pt/cgi-bin/word_map.pl`.

5 Experimental Setup

We took the CLEF 2007 and CLEF 2008 test sets, both having 200 question of several types (factoid, definition and enumeration), and we chose a subset of action/event-related factoid questions. To ensure that all test questions could potentially be answered we selected only those the we knew that our system could parse and extract candidate answers. We chose 27 question from each of the test sets whose expected answer type could be any of the following (see Table 1): date or time expression (DATE), an organization (ORG), a geo-political entity (GPE), a person (PER) or a quantity (QNT). For these types of questions, our

[1] Available from http://openthesaurus.caixamagica.pt/. Version used is dated from 2006-08-17

system relies on the *simplest* answer extraction strategy that we have available. When the question is parsed the expected type of answer is identified. Answer candidates are those entities extracted from the retrieved text passages whose type is compatible with the expected answer type. The final answer chosen is the *most frequent compatible candidate* found. The alternative method for answer extraction available in our QA system is based on context evaluation rules. It provides much higher precision but since we only have extraction rules for a small subset of questions, we cannot use it for a representative test.

We configured our system to answer test questions using 4 different options for query generation / expansion:

1. **Run** R_{ps}: in this run, queries are generated by pseudo-stemming. Up to a maximum of 150 text passages can be retrieved. This will be our baseline method.
2. **Run** R_{oo}: in this run, query expansion is made using the thesaurus expansion procedure described in Section 3, and the OpenOffice thesaurus. The verb is expanded up to a maximum of 14 related verbs options, and a maximum of 10 snippets are retrieved per verb. Thus, at most 150 text passages are retrieved.
3. **Run** R_{st}: this run is equivalent to R_{oo}, but the statistical thesaurus is used instead. Again, up to 14 expansion options are considered and no more that 150 text passages are retrieved.
4. **Run** R_0: in this run, we *remove* the verb from the query. Only the *argument* of the question (e.g. "J.F.K." in "Who killed J.F.K?") is used in the query. This method should provide maximum retrieval recall, although possible decreasing precision in retrieval. However, since further filtering will be done along the QA pipeline, this can be considered a realistic configuration for comparison purposes. Up to 150 text passages will be considered.

Answers were searched in the Wikipedia-derived collection provided by the CLEF organization. Answers were manually checked. We checked non-nil answers to see if they were *correct*, *incorrect* or *inexact* (i.e. only partially correct). *Unsupported answers* were considered *incorrect*. When the system was not able to produce any answer (i.e. produce the NIL answer) we checked whether the answer was present in the retrieved text passages but it was not extracted. In those cases, we can assume that the problem is related with the answer extraction.

test set	DATE	ORG	ORG/PER	PER	GPE	QNT	$\sum$
CLEF-2007	9	5	4	5	3	1	27
CLEF-2008	12	1	3	9	2	0	27

Table 1. The two sub-sets of action/event-related factoid questions used for testing.

6 Results and Analysis

Table 2 presents the results that runs R_{ps}, R_{oo}, R_{st} and R_0 obtained for the 54 questions in the test set. The first three columns report the results in case of non-nil answer (correct, incorrect and inexact). The forth column present the number of NIL answers, and explicitly shows number of cases where the answer *was present* in the text passages but the system was unable to extract it. The last column presents the number of cases where the found answer was inexact or was not extracted, emphasizing the cases where the retrieved text passages contained the correct answer but the extraction stage failed (partially or completely).

Run	Correct	Incorrect	Inex.	NIL (No Ext.)	$\sum$	Inex. + No Ext.
R_{ps}	4	10	1	39 (2)	54	3
R_{oo}	3	11	2	38 (2)	54	4
R_{st}	3	9	4	38 (4)	54	8
R_0	10	16	3	25 (5)	54	8

Table 2. Results obtained for the four query generation / expansion configurations

Run R_0 clearly outperforms all others both in the number of correct answers, and in the number of non-NIL answers. R_0 also produces more incorrect answers but relative increase in the number of correct answers is much higher. If we only consider correct answers none of the runs that use thesaurus expansion methods, R_{oo} and R_{st}, beats the baseline run, R_{ps}, that uses pseudo-stemming. The number incorrect and NIL answers also does not change significantly between runs R_{ps}, R_{oo} and R_{st}. The only significant difference is the aggregate number of *inexact answers* plus *not extracted* answers, where the figure for run R_{st} is higher that for runs R_{ps}, R_{oo}. This suggest that R_{st} was able to find the appropriate text passages in several cases but the extraction stage was unable to identify the correct answer.

Table 3 presents statistics about expansion and retrieval. The second column presents the average number of branches provided by the expansion mechanism. Obviously, both R_{ps} and R_0 only generate one query, so branching is 1. On the other hand, R_{st} generates the highest number of query branches (10.9). The third column presents the number of questions for which *no passages* were retrieved

Run	Avg. Branching	No Passages (in 54)	Avg. # Passages
R_{ps}	1	37	2.6
R_{oo}	3.1	35	3.4
R_{st}	10.9	30	7.7
R_0	1	19	24.6

Table 3. Retrieval Statistics

(out of 54 question). The last column indicates the average number of passages retrieved when at least one passage was retrieved. Again, R_0 allows retrieving more passages. R_{st} allows retrieving more passages than R_{oo} although the increase is proportionally lower than the corresponding increase in the branching factor. This suggests that some of the verbs provided by the statistic thesaurus might not be correct (or correlated with the argument of the question).

Generally, results confirm that retrieving and analyzing more passages does helps finding more corrects answers (higher recall). We verified that this seems to be specially the case when the number of passages referring to the argument of the question is *very low* (e.g. 1-3). In those situations, query expansion (by any method) helps finding the few decisive passages. R_0 clearly outperforms all others, but at the cost of processing many more text passages (even when limiting the maximum number of retrieved snippets to 150). Results of runs R_{ps}, R_{oo} and R_{st} do not vary significantly in terms of *correct* and *incorrect* answers. However, if we consider the aggregate number of inexact and not extracted answers (last column in Table 2) we see that R_{st} could potentially outperform both R_{ps} and R_{oo} if the extraction procedure was made more efficient. This can also be confirmed by the fact that R_{st} was able to retrieve passages for 5 questions more than R_{oo} and 7 more than R_{ps} (see Table 3). Unfortunately, the same can not be said for R_{oo} in relation to R_{ps}: the difference in performance is no significant. Apparently, thesaurus-based expansion is effective only when branching is relatively high, which is the case of R_{st}. Since thesaurus-based expansion uses closely related verbs, the number of passages retrieved and processed in subsequent stages of the QA pipeline does not grow in an uncontrolled fashion, as it does in run R_0 when arguments are very frequent entities.

7 Conclusion and Future Work

Results obtained with this set of questions and the Wikipedia collection, do not clearly demonstrate that executing thesaurus-based expansion in QA-centric IR is advantageous for the *overall* QA performance in comparison to not performing any query expansion at all. In fact, simpler "query expansion" methods may lead to better QA performance. Results suggest that thesaurus-based extraction improves recall at the IR-stage of the QA pipeline, while still keeping reasonable levels of precision. However, such improvement is only propagated to the *overall* QA performance if subsequent answer extraction procedures are also successful, which, unfortunately, is not always the case. The main advantage of using thesaurus-based expansion is that of selectively controlling the number of text passages retrieved, which helps reducing computational load further down the QA processing chain.

Future work will necessarily focus on problems related to the extraction of answer candidates. Additionally, we wish to improve our statistic thesaurus by using more linguistic information, namely POS tagging. This will allow us to find equivalence relations not only between simple verbs, but also between simple verbs and compound verbs, and paraphrase constructions. We wish to automat-

ically identify highly polysemous verbs so that we can later decide if candidates extracted from passages containing those verbs should be considered or not. From the point of view of query expansion itself we wish to experiment the impact of expanding the initial verb to larger sets, such as for example by also expanding each of the verbs obtained after expanding the initial verb, and using the much larger resulting set in the queries.

8 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. Bilotti, M.W., Katz, B., Lin, J.: 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)
2. Sarmento, L., Teixeira, J., Oliveira, E.: Experiments with query expansion in the raposa (fox) question answering system. In Borri, F., Nardi, A., Peters, C., eds.: Working Notes for the CLEF 2008 Workshop, Aarhus, Denmark (17-19 September 2008)
3. Costa, L., Sarmento, L.: 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)
4. Curtis, J., Matthews, G., Baxter, D.: 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)
5. Hovy, E., Gerber, L., Hermjakob, U., Junk, M., Lin, C.Y.: Question answering in webclopedia. In: Proceedings of the 9th Text REtrieval Conference, Gaithersburg, MD, USA (November 2000) 655–664
6. Negri, M.: Sense-based blind relevance feedback for question answering. In: SIGIR-2004 Workshop on Information Retrieval For Question Answering (IR4QA), Sheffield, UK (July 2004)
7. Riezler, S., Vasserman, A., Tsochantaridis, I., Mittal, V.O., Liu, Y.: 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)
8. Sarmento, L.: A first step to address biography generation as an iterative QA task. In Peters, C., Clough, P., Gey, F.C., Oard, D.W., Stempfhuber, M., Magnini, B., de Rijke, M., Gonzalo, J., eds.: 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)
9. Sarmento, L., Oliveira, E.: Making RAPOSA (FOX) smarter. In Nardi, A., Peters, C., eds.: Working Notes of the Cross-Language Evaluation Forum (CLEF) Workshop 2007, Budapest, Hungary (September 2007)
10. Lin, D.: Automatic Retrieval and Clustering of Similar Words. In: Proceedings of COLING-ACL 1998. Volume 2., Montreal (1998) 768–773

11. Sarmento, L.: BACO - A large database of text and co-occurrences. In Calzolari, N., Choukri, K., Gangemi, A., Maegaard, B., Mariani, J., Odjik, J., Tapias, D., eds.: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006), Genoa, Italy (22-28 May 2006) 1787–1790