Scoring Missing Terms in Information Retrieval Tasks

Egidio Terra
School of Computer Science
University of Waterloo
Waterloo, Canada
elterra@uwaterloo.ca

Charles L.A. Clarke
School of Computer Science
University of Waterloo
Waterloo, Canada
claclark@plg2.uwaterloo.ca

ABSTRACT

An usual approach to address mismatching vocabulary problem is to augment the original query using dictionaries and other lexical resources and/or by looking at pseudo-relevant documents. Either way, terms are added to form a new query that will be used to score all documents in a subsequent retrieval pass, and as consequence the original query's focus may drift because of the newly added terms. We propose a new method to address the mismatching vocabulary problem, expanding original query terms only when necessary and complementing the user query for missing terms while scoring documents. It allows related semantic aspects to be included in a conservative and selective way, thus reducing the possibility of query drift. Our results using replacements for the *missing query terms* in modified document and passages retrieval methods show significant improvement over the original ones.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Performance, experimentation

Keywords

Automatic Query Expansion, Document Retrieval, Passage Retrieval

1. INTRODUCTION

A user query for a retrieval system expresses both the user's information need and the knowledge he/she has about the query topic. All these surrounding factors in an Information Retrieval setting makes it hard to capture the marginal aspects of a query. In particular, a word used in a query can have different meanings or have other words that may replace it in documents. This causes problems such as query drift and mismatching vocabulary that deteriorate the accuracy of the retrieval process. One way to address the mismatching problem is through automatic query expansion (AQE), where new terms are added to create a new expanded query to be submitted to the retrieval engine [2, 3, 12, 17, 24, 27]. On the other hand, AQE increases the chances for query drift [8, 15].

An alternative to handle the problem of mismatching vocabularies is through the use of translation language models and the methods used in Cross-Language Information Retrieval (CLIR). The lack of query terms in the documents is addressed by using one or more words in the document as a translation for query terms [1, 9, 11, 19]. In a sense, the translation of document words into query terms is not the same thing as expanding the query with extra terms. Translation focuses on replacing the query term while the AQE focuses is on complementing the query with some other aspects.

We take a different approach to address the mismatching vocabulary problem. Unlike AQE and translation models, instead of augmenting the query to score documents, we use the original query and replace missing terms only when necessary. The idea is to use the original query terms to score documents as long as possible. This can be viewed as a kind of translation, however we do not try to translate query terms that are present in the document and, depending on how we choose the replacement terms, we can also capture relationship types other than translation.

Since the vocabulary changes from one document to another, it is likely that our approach will score documents using different queries from the original but forming a new query with as minimal change as possible from the original user query. In the same situation, traditional AQE will use one query for all documents, regardless of the mismatching vocabulary problem. In order to prevent the original query terms from being outweighed by replacement terms, we adjust the weights of replacement terms based on their relatedness to the missing query term. While our approach is a form of query expansion, it does not exclude the possibility that a traditional AQE could be performed later in the retrieval process.

We apply the new method to passage retrieval and document retrieval, as described in section 3. The method to find replacements for the missing query terms is described in section 4. Our empirical results and discussions are presented in section 5.

CIKM'04, November 8–13, 2004, Washington, DC, USA.

2. RELATED WORK

Information Retrieval models, with the exception of classic boolean model, allow documents to be scored when not all the query terms are present. In general, the score of the document is given by weights assigned to query terms present in it. In the vector space model [18] a missing term will have zero value in the document vector, thus contributing no weight towards the document score. In the *tf.idf* probabilistic models [12], a missing term will not count either since its term frequency is zero. In these models a rather common approach to handle mismatched vocabulary is to use pseudo-relevance feedback [2, 3, 12, 17, 24].

In language models, instead of using maximum likelihood estimators, the term frequencies are smoothed in order to assign some probability mass for terms missing in documents [16]. Pseudo-relevance feedback is also used in language modeling [13, 27], normally by expanding the query term set to form a query language model.

One particular language model, the statistical translation model for IR, is related to the work presented in this paper [1]. It is inspired by statistical translation models for natural language and relies on the idea of parallel corpora, where it assumes there exists some alignment between texts. When adapted to IR, the translation is made from a document to a query and the retrieval process comprises word translation from document into query terms by means of translation probabilities. The relevance of the document is assumed to be monotonically increasing to the likelihood of generating (translating) the query from the document. The translation probabilities enable the use of all query terms for every document, even when they are not present in the document. Berger *et al.* propose two translation models for Information Retrieval [1], and both models (1 and $1'$) compute the weight for every query term as the sum of the product of the translations of every document word into the query term and document frequency. The use of all words as a possible translation of a query term is a way to capture all possible alignments between the document and the query. This also has the effect of relating all terms in the query and document, even when they are not related. It is also interesting to note that the queries are expanded to form a query model before the actual "translation" (i.e. scoring) occur, which can also lead to query drift.

In CLIR, the query is specified in one language and documents in another. As a consequence, the query terms will not be present in the documents. To address the cross language aspect, a common approach is to translate the query into document language [19]. Darwish and Oard use the idea of replacement of query terms by document words at query-time in CLIR and in the retrieval of scanned OCR documents [9]. In their CLIR application a handful of translation resources, such as dictionaries and parallel corpora was used. A parallel corpus was used in their OCR application, having on one side the corrected digital version of the document and on the other the version resulting from OCR (containing errors), and these translation resources were then used in a document retrieval task.

In monolingual information retrieval the idea of translation is not quite natural. It is arguable that one synonym may translate to its counterparts; however, that is not same as AQE. Rather, in AQE, the expansion terms tend to complement the original query terms by including not only synonyms but also other types of relationships, such as morphological variants of the term, and also other semantic relations (e.g. hyponyms, hypernyms and many others). Furthermore, the translation models rely either on the availability of alignments, such as in CLIR, or on brute force alignments, such as the statistical translation model for IR. On the other hand, AQE methods add terms in heuristic ways that may cause query drift, particularly if the documents used for expansion are not relevant.

In another related work, Xu and Croft [25] use a corpus based approach to filter errors from stemmers, providing a query expansion less prone to query drift. The classes of related words resulting from this filtered stemmed were then used in traditional automatic query expansion.

3. MODIFIED RETRIEVAL METHODS

Two probabilistic models, one passage retrieval method and one document retrieval method, are modified in order to accommodate non-zero scoring of missing terms. For these methods, it is desirable to make as few changes as possible in order to prevent query drift.

3.1 Passage Retrieval

We use the passage retrieval component of MultiText. It has been successfully applied to question answering [5, 4, 14] and pseudo-relevance feedback [26]. From a query $Q = \{t_1, t_2, .., t_k\}$ let $T \subseteq Q$. Given an extent of text comprising all words in the interval (u, v). The extent length is $l = v - u + 1$ and the probability of $P(t, l)$ that the extent contains one or more occurrences of t is

$$\begin{aligned} P(t,l) &= 1 - (1 - p_t)^l \\ &= 1 - (1 - lp_t + O(p_t^2)) \\ &\approx lp_t. \end{aligned}$$

The probability that an extent (u, v) contains all the terms from T is then

$$\begin{aligned} P(T,l) &= \textstyle\prod_{t_i in T} P(t,l) \\ &= \textstyle\prod_{t \in T} lp_t \\ &= l^{|T|} \textstyle\prod_{t \in T} p_t. \end{aligned}$$

The estimation of p_t is given by the Maximum Likelihood Estimator (MLE) for t in the collection

$$p_t = f_t/N$$

where f_t is the collection frequency of t and N is the collection size in words. The score for an extent of length l containing the terms in T is the self-information of $P(T, l)$

$$\sum_{t_i \in T} \log(N/f_t) - |T| \log(l) \qquad (1)$$

The score is higher for short passages containing all terms in T and there is a trade-off on the number of terms and size of the passage.

For the original passage retrieval method presented by [5], an efficient algorithm to retrieve all passages comprising 1 to $|Q|$ query terms is presented by Clarke *et al.* [6]. The running time to extract all extents containg the terms in T is $O(|Q|\mathcal{J}_l log(N))$ where $|Q|$ is the total number of query terms, $\mathcal{J}_l$ is the number of extents containing $|T|$ query

Word	Frequencies
New	104,483,262
York	12,205,261
population	4,854,401
demographic	428,641

Table 1: Corpus Individual Frequencies

terms and N is the corpus size in words. The algorithm is based on the positions of query terms, checking for close occurrence of other query terms and skipping repetitions of the same term. This algorithm benefits from the sorted poisition entries in the inverted list used to index the underlying collection and quickly locate terms.

To accommodate scoring of missing terms, the modified version only considers the whole query Q since every extent has a representative for missing query terms. We assume $p_{t,t} = p_t$ if the term t is present in the extent. If the term t is not in the document a replacement term r will be used. The weight of the replacement is the conditional probability $p_{t|r}$, which is calculated by estimating the maximum likelihood for p_r from the corpus and estimating the joint probability by

$$p_{t,r} = f_{r,t}/N'$$

where $f_{r,t}$ is the joint frequency and N' is the total number of pairs considered for the joint frequency in the corpus.

We take a winner-takes-it-all approach and choose the best r in the extent,

$$\arg\max_{r \in (u,v)} \; p_{t|r}$$

Finally, the modified version of equation 1 using replacements is given by

$$\sum_{t_i \in Q} \log(N/f_{ti}) \cdot p_{ti|r} - |Q| \log(l) \tag{2}$$

We should note that since every extent has a representative for a query term, we can make arbitrary decisions on the extent size. This creates a trade-off between extent size and replacement quality. On the other hand, the fact that any extent can have a representative does not allow us to use the efficient algorithm existing for the original method. Instead of selecting the extent in sub-linear time complexity (log of the corpus size) as the original method, our approximation extracts the passages in linear time.

3.2 Document Retrieval

For document retrieval, we use Okapi BM25 formula [12], a *tf.idf* model that uses a bag-of-words approach. The weights of query terms are calculated from the collection, and relevancy is used if available. The document score is the sum of weights of query term in that document. Specifically, given an query $Q = \{t_1, t_2, .., t_k\}$, a document d is assigned the score

$$\sum_{t_i \in Q'} w^{(1)} \frac{(k_1+1)df_{ti}}{K+df_{ti}} \frac{(k_3+1)q_{ti}}{k_3+q_{ti}} + k_2 \cdot |Q| \cdot \frac{avdl-dl}{avdl+dl}$$

Pair	1–3	4–40	41–∞
New York	11,784,589	3,365,934	8,215,334
population & demographic	10,507	89,772	485,491

Table 2: Corpus Frequencies of Pairs at specific distance intervals

where

$$\begin{array}{rcl}
w^{(1)} & = & \log \dfrac{(r_{ti}+0.5)/(R-r_{ti}+0.5)}{(d_{ti}-r_{ti}+0.5)/(D-d_{ti}-R+r_{ti}+0.5)} \\
Q' & = & \text{subset of unique terms in } Q \\
D & = & \text{number of documents in the collection} \\
d_{ti} & = & \text{\# documents containing the term } t_i \\
q_{ti} & = & \text{frequency of } t_i \text{ in the query } Q \\
df_{ti} & = & \text{frequency of } t_i \text{ in the document } d \\
dl & = & \text{document length in words} \\
avdl & = & \text{average document length in the collection} \\
R & = & \text{\# relevant documents for the query} \\
r_{ti} & = & \text{\# relevant documents containing } t_i \\
K & = & k_1((1-b)+b \cdot dl/avgdl) \\
k_1, b, k_2, k_3 & = & \text{query nature and database parameters}
\end{array}$$

In cases where relevance information is not available, the values of R and r_{ti} are set to zero. Usual values for query nature and database parameters are $k_1 = 1.2$, $b = 0.75$, $k_2 = 0$, and $k_3 = \infty$, as result the main *tf.idf* components are kept in the short version of the formula:

$$\sum_{t_i \in Q'} \log \frac{D}{d_{ti}} \cdot q_{ti} \cdot \frac{(k_1+1)df_{ti}}{K+df_{ti}} \tag{3}$$

To allow missing query terms to be scored we modified the short formula (equation 3) by adding the relatedness factor for term r as a replacement for term t_i in similar fashion to passage retrieval. We calculate the conditional $p_{ti|r}$ by the maximum likelihood of p_r and the joint probability :

$$p_{ti,r} = f_{r,ti}/N'$$

where $f_{r,ti}$ is the joint frequency and N' is the total number of pairs considered for the joint frequency in the corpus.

As in the modified passage retrieval method, we use the best replacement, thus the term r is

$$\arg\max_{r \in d_m} \; p_{ti|r}$$

where d_m is a document that does not contain t_i.

Our modified version of BM25 uses a modified *idf*,

$$\sum_{t_i \in Q'} \log(p_{ti|r} \cdot \frac{D}{d_{ti}}) \cdot q_r \cdot \frac{(k_1+1)df_r}{K+df_r} \tag{4}$$

Equation 4 is similar to the modified *tf.idf* presented by Darwish and Oard [9] and used in CLIR and OCR retrieval:

$$tf_i = \sum_{k \in R(t_i)} tf_k \cdot w_k$$

and

$$idf_i = 1/\sum_{k \in R(t_i)} d_k \cdot w_k$$

where tf_i and tf_k are frequencies of terms i and k in the document being scored, d_k is the number of documents containing k, w_k is the replacement weight of the term k and $R(t_i)$ is the set of replacements for t_i. While BM25 is a *tf.idf* formula, it has a more sophisticated handling of document size and term frequencies. Apart from that, there some other major differences between our modified BM25 and Darwish and Oard's formula. First, they recommend using the replacement weight twice, once in the *tf* component and another in the *idf*. The way the replacements are computed also differs from our method, which is explained in section 4. A last major difference is the fact that the original terms are not present in the scored documents both in CLIR and OCR, thus they do not need handle the case when the query term is present.

4. FINDING TERM REPLACEMENTS

To prevent the query drift, it is desirable to have a replacement term that represents the original term's abstract concept when used in the context specified by the user query. The actual type of semantic relationship is not easily predicted, can be just a synonym or a hypernym, or any other. We use a simple statistical approach as laid out by Firth [10], which considers the co-occurrence of related words to be a result of their *mutually expectancy* (i.e. high frequency co-occurrence is indicative of some sort of lexical-semantic relationship). Empirical experiments confirm this as a language phenomena, such as the one in which synonyms are guessed by their co-occurrence [21, 22, 23].

We use pointwise mutual information (PMI) as the similarity measure to score relatedness between pairs of word.

$$PMI(w_1, w_2) = log \frac{P_{w1,w2}}{P_{w1} P_{w2}} \tag{5}$$

The reason for choosing PMI is two fold. First, it was demonstrated to be effective for language phenomena [22]. Second, it has a relationship with *idf*. This relationship comes from the assumption that $P_{w,w} = P_w$, thus

$$\begin{aligned} PMI(w,w) &= log \ \tfrac{P_{w,w}}{P_w \cdot P_w} \\ &= log \tfrac{P_w}{P_w \cdot P_w} \\ &= -log \ P_w \\ &= idf_w \end{aligned} \tag{6}$$

TREC10 test set		
Method	% Coverage	% Correct
Original	96.1	87.97
Replacement	96.1	89.21

TRECs 9-12 test set		
Method	% Coverage	% Correct
Original (1+ missing)	89.9	85.88
Replacement (1+ missing)	94.5	87.88
Original (all)	94.5	89.59
Replacement (all)	95.3	89.50

Table 3: Replacement Method results

Method	Coverage
IBM	92.87%
SiteQ	92.59%
ISI	91.44%
Alicante	90.97%
MultiText	89.81%

Table 4: Top five passage retrieval in Tellex *et al.*

In the case of the pair of words w_1 and w_2, the maximum value for the pointwise mutual information is bounded by $PMI(w_1, w_2) \leq idf_{w1}$ and $PMI(w_1, w_2) \leq idf_{w2}$. This can be easily verified since the PMI formula has maximum value when the joint probability is equal to the smallest marginal (if marginals are different). Therefore, we can use *idf* to normalize the PMI for a given word we want to replace

$$CondPMI(w_1, w_2) \ = \ \frac{log \ (P_{w1,w2})/(P_{w1} \cdot P_{w2})}{log \ (1)/(P_{w1})} \tag{7}$$

which is monotonic to

$$\frac{(P_{w1,w2})/(P_{w1} \cdot P_{w2})}{1/P_{w1}} \ = \ P_{w1|w2}$$

Thus, if we fix one word, in this case the missing query term, we can rank the affinity of remaining words of the vocabulary. Since the goal is to find a replacement for one query term at each time, the denominator of the equation 7 is fixed for every replacement. We should note that there is a problem with the normalization in the conditional PMI. The problem occurs when PMI is negative, in which case we just set it to zero. Setting the negative value to zero could be avoided if we offset both *idf* and PMI by the minimal PMI value. We ignore negative PMI and set its value to zero, thus we use a self-regulated cut-off for the minimal value for a conditional PMI. We assume that any word in the document with a negative PMI with respect to the missing query term is not a good candidate for replacement.

For estimation of $P_{w1,w2}$ use the maximum likelihood :

$$P_{w1,w2} = f_{w1,w2}/N' \tag{8}$$

where the joint-frequency $f(w_1, w_2)$ is the number of the co-occurrences of w_1 and w_2 at distances ranging from four to 40 words apart. The lower cut-off prevents phrasal relationships (e.g. if the term "New" is a query term but "York" is not, then the latter is probably not a good replacement for the first). As most of the co-occurrences of "New" and "York" happen at distance one, then this cut-off will avoid this bias for pairs in the same phrase. The frequencies values for "New" and "York" and for the pair "demographic" and "population" are shown in the tables 1 and 2 over a terabyte corpus. The pairs counting in table 2 do not include nesting, thus "new New York" will count only once towards the joint frequency. Terra and Clarke [22] showed that windows of 32 words are a good setting for an upper bound on the distance. Our upper cut-off was arbitrarily set close to it (40). The value of N' is the size of the window times the corpus size $(36 \cdot N)$.

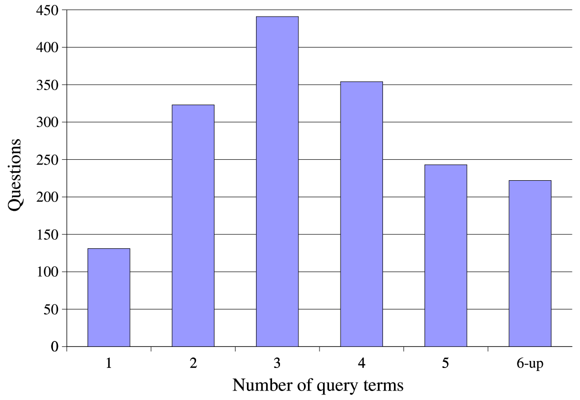

Figure 1: QA query terms histogram

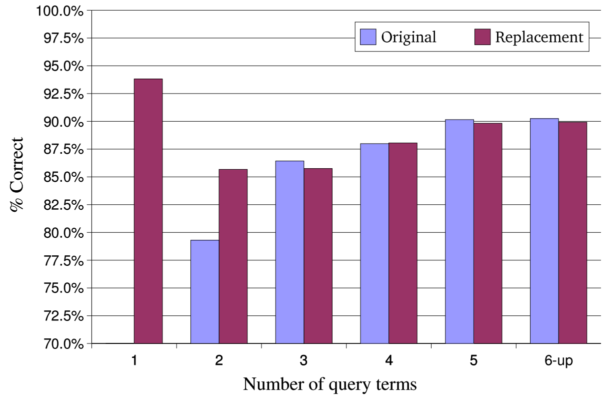

Figure 2: % correct passages by # original query terms

5. EMPIRICAL EVALUATION

For the passage and document retrieval experiments, all the replacements were calculated using the statistics of a terabyte corpus of Web data crawled from the general web in 2001 [4, 22]. The crawl was conducted using a breadth-first search from a initial seed set of URLs representing the home page of 2392 universities and other educational organizations. No duplicate pages were included and the crawler also did not allow a large number of pages from the same site to be downloaded simultaneously. Pages with duplicate content were eliminated. Overall, the collection contains 53 billion words and 77 million documents.

5.1 Passage Retrieval

We assess the performance of the modified passage retrieval method using TRECs 9 to 12 QA tests sets. TREC 9 contains some question variants, with some rewording of questions. Those are questions are left out because most of important query terms are the same. The remaining 1,732 questions with known answers in the TREC official collections were used. As we are particularly interested in the evaluation of the passage retrieval method, we only extract passages from documents in TREC collections (TIPSTER/TREC disks 1–5 and AQUAINT) that contain answer to questions. A similar approach was used in Tellex *et al.* [20]. For these 1,732 questions, the total number of relevant documents is 10,561.

The queries used in passage retrieval methods were generated from questions by simple stopwords exclusion. The query size distribution is given by figure 1. Since most of the queries are short, a missing query term can harm the effectiveness of the passage retrieval. We perform automatic judgments in this evaluation, using the regular expression patterns available from the NIST website[1]. We consider a passage correct if it matches the pattern for the question.

For each pair ¡query number,relevant document¿ we find the best passages using the original and the modified methods. For the modified one, we scan the whole document to find the best scoring passage among all possible candidates using equation 2. For every candidate passage we want a representative for each query term to be present. The number of candidate passages is $O(|D_i|^2)$ for each document, but since the goal of passage retrieval is to find a fragment of text smaller than the whole document, we limit our reported passages to 170 words for comparison purposes. Tellex *et al.* [20] used snippets of 1000 bytes in a similar passage retrieval evaluation (170 words $\sim$ 1000 bytes using our tokenizer). Every 170 words passage has a smaller fragment we call a "hotspot", that contains all the query term representatives and we seek representatives in hotspots of 20 words using a sliding window. Limiting the size of the hotspot is necessary to prevent representatives from being located too far apart, preventing less stronger representatives to be used even if they are close to other query terms. This makes the number of passages $O(|D_i|)$ but we may discard some passages with a better score. The best hotspot in the document is later extended to 170 words. The choice of hotspot size is a trade-off between execution time and effectiveness.

The baseline is the original passage retrieval method using the scoring function of equation 1. To evaluate the difference between the two methods, we first compute the effectiveness measures when at least one of the query terms is missing in the passages retrieved using the original method. Since we retrieve exactly one passage from each document, we can compare the passages from the two methods side by side. Figure 2 plots the % of correct passages by the number of original query terms. It shows only passages where at least one original terms is missing. The y-axis is the percent of *correct* passages, i.e. containing the answer for the question. For instance, for the more than 300 questions that have query size of 2, the original method retrieves 79% of passages correctly (in this case the passages contains exactly one query term). The modified method replaces the missing term with another in the document and improves the percent of correct passages to around 86%.

The improvements are higher for short queries, comprised of one or two query terms. For queries of size one, a missing term means no information is available to select a passage in the original method; in this case our new method of replacement can only improve the results. The replacements also help for queries of size two. When more query terms are available, replacements do not help or harm. Using Wilcoxon signed rank test, the difference in the percentage of correct passages is significant at 99% confidence level.

We also calculated the *coverage*, the percentage of the 1,714 questions where at least one retrieved passage contains the answer [7]. As many Question Answering systems

[1] trec.nist.gov

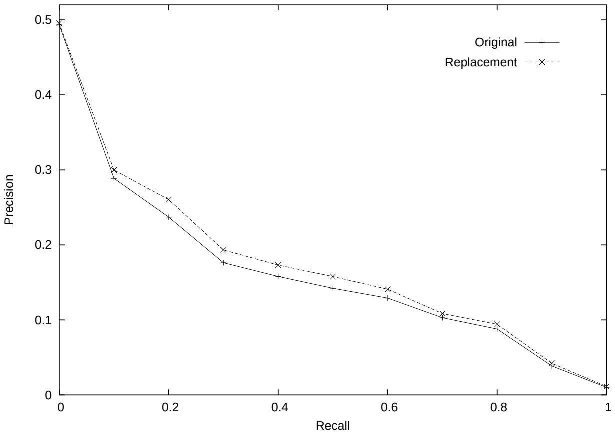

Figure 3: Interpolated Precision-Recall for Topics 51-100 on SJMN

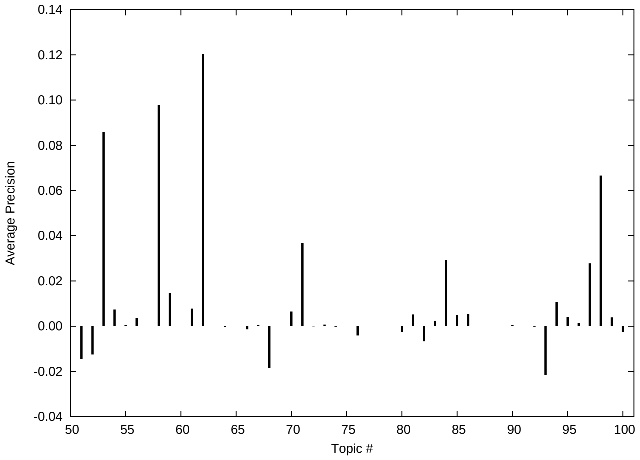

Figure 4: Difference in Average Precision

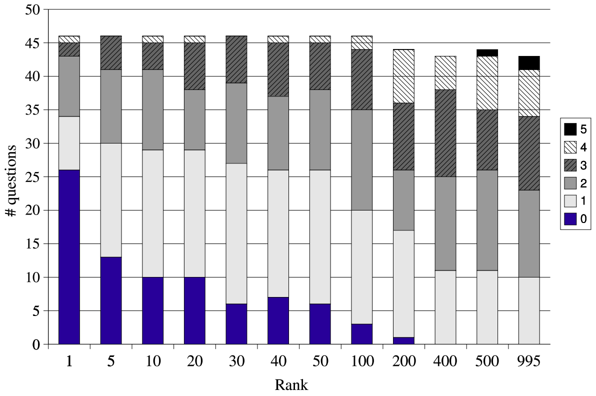

Figure 5: Rank by # missing terms - original

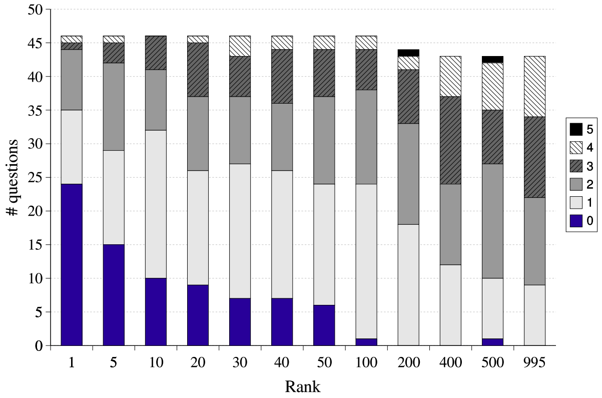

Figure 6: Rank by # missing terms - replacement

use the output of the passage retrieval to feed an answer extraction component, it is important to have at least one passage containing the answer so that the answer extraction module can have chance to find it.

The new method provides a better coverage than the original baseline method. Table 3 shows the coverage performance when at least one query term is missing. There is a substantial improvement in the coverage when using the replacement method.

We further compare the results of our new method with the evaluation presented by Tellex *et al.* [20], where different passage retrieval methods were evaluated using the TREC10 QA test set. Tellex *et al.* report effectiveness by means of Mean Reciprocal Rank (MRR) and the percent of incorrect *questions* (instead of passages). The MRR is calculated by averaging the inverse rank of the first correct answer to each question. It is not clear that MRR is appropriate for evaluating the passage retrieval component of a QA system. It is an intuitive measure if considered in terms of the end-user. However, in typical QA systems, the passages are going to be further processed by an answer extraction component, thus their retrieval rank may not be as important as it would be for the end-user. For this reason, we do not report MRR. The latter measure, percent of incorrect questions, is the complement of *coverage* (i.e. 1-*coverage*), thus the results are directly comparable. The reported coverage by Tellex *et al.* [20] is reproduced in Table 4. The coverage is higher in our experiments and the differences can be explained by two factors: Tellex *et al.* use *idf* in equation 1, which is not appropriate since in its derivation the collection frequency is used (rather than document frequency); the statistics used in both original and modified passage retrieval, and reported in Table 3, are drawn from the terabyte corpus and not from TREC collections.

5.2 Document Retrieval

For document retrieval, our evaluation was performed on the ad hoc queries corresponding to TREC topics 51–100. The target corpus was the San Jose Mercury News subcollection of TIPSTER/TREC disk 3, containing 90,257 documents. The queries were extracted from the title field, stopwords removed — Stemming was not used.

As the *tf.idf* models score only documents containing at least one of the query terms, the number of documents that can be scored is normally smaller than the size of all documents in the collection. For the case where query terms can be replaced, this limitation is obviously not present. However, it is less likely that all query terms need to be replaced, but that depends on agreement of the vocabularies used in the query and the collection. All the available relevance

Table 5: Replacements in topic 51

Query Term	Replacement	COND-PMI
Airbus	aeroflot	0.2060
Airbus	mcdonnell	0.1680
Airbus	aerospace	0.1234
subsidies	subsidized	0.1808
subsidies	revamping	0.1135
subsidies	taxpayers	0.0444

Table 6: Replacements in topic 62

Query Term	Replacement	COND-PMI
coups	coup	0.1598
coups	dessalines	0.1721
coups	honasan	0.2376
etat	choonhavan	0.0889
etat	gqozo	0.1251
etat	aristide	0.0273

Table 7: Replacements in topic 71

Query Term	Replacement	COND-PMI
incursions	gissin	0.2554
incursions	infiltrations	0.2042
incursions	incursion	0.1735
incursions	militants	0.1122
incursions	militia	0.0662
border	Mexicans	0.0032

Table 8: Replacements in topic 53

Query Term	Replacement	COND-PMI
buyouts	buyout	0.3224
buyouts	divestitures	0.1862
buyouts	mergers	0.1756
leveraged	buyout	0.1667
leveraged	takeovers	0.1587
leveraged	mergers	0.1050

Table 9: Replacements in topic 68

Query Term	Replacement	COND-PMI
hazards	carcinogenicity	0.5195
hazards	hazardous	0.0913
diameter	vesicle	0.4657
diameter	pipe	0.0218
fine	allo	0.4109
fibers	asbestosis	0.1773

Table 10: Replacements in topic 94

Query Term	Replacement	COND-PMI
aided	conspired	0.0867
aided	autocad	0.1150
aided	drafting	0.1294
crime	hacking	0.0443
crime	crimes	0.1256
crime	burglaries	0.1530

judgments were performed only on documents that contain original or expanded query terms, and due to these reasons, our evaluation use the documents that contain at least one query term. As a result, four topics (57, 75, 77 and 78) were discarded from our evaluation since they always have the original term present and exactly one word in title field, thus our method will score documents the same way the original method does. Two other topics — 65 and 88 — were not considered since they do not have any document judged relevant in the SJMN sub-collection. The remaining 44 topics were used in our evaluation.

For each document, every original query term is weighted as in the normal BM25 formula. If the query term is not present, all the words in the document are considered for replacement and the corresponding weight is calculated by using equation 7. The best replacement is selected for each missing query term and final document score is given by equation 4.

Some examples of replacements are shown in tables 5 to 10. Some replacements are just morphological variants of the original term but some semantic relationships are present too. In topic 94, no relevant document had the query term *computer* replaced. The representative terms for query term *aided* were not as good as the ones used for the original term *crime*. Replacement in topic 62 tend to focus on the people involved in Coups d'etat, and as in topic 94 one term is always present in the relevant judgments - *Military*. This shows that some query terms are really important in the query and documents not containing them are less likely to be relevant.

The precision-recall curves of the original and the modified formula with replacements are depicted in figure 3. There is a consistent improvement over the original BM25 and the difference in the mean average precision between the original and the modified methods is statistically significant at 99% level using Wilcoxon signed rank test. The analysis of the average precision in the individual topics, depicted in figure 4, shown that in many topics the precision improved substantially. In fact, 28 out of the 44 topics improved on average 0.0206, four stayed the same and in 12 topics where the precision drop the reduction was on average 0.0058.

It is interesting to note that the recall in the replacement method improved as well, from 1227 to 1315 relevant document retrieved, which corresponds to retrieving 8.70% of the remaining relevant documents not retrieved in the original Okapi BM25 (at 1000 documents). A run with all terms stemmed also improved mean average precision but maintained the recall at exactly the same level of the original method.

We performed a failure analysis on the four topics responsible for the big drops in average precision: 51, 52, 68 and 93. In two of them, topics 52 - South African Sanctions; and 93 - What Backing Does the National Rifle Association Have?, the replacement of components of a phrase were responsible for the decline in performance. This problem can be addressed by using the noun phrases from queries. In topic 51 - Airbus Subsidies, the replacements for the proper name Airbus harmed the average precision. In topic 68 Health Hazards from Fine-Diameter Fibers, the replacements for Fine-Diameter were not helpful, whereas Fibers and Hazards had good replacements in asbestosis and carcinogenicity.

An alternative way to see the differences between the original and the method with replacements is to look at their rankings. Figure 5 plots different rank positions in the original Okapi BM25 method, and Figure 6 shows the same cut points in the new method with replacements. The cumulative bars indicate how many missing query terms the documents ranked at that position have. For example, in the original method, 26 topics had documents with no missing query terms at rank 1 (figure 5). In the new method, this number is reduced to 24. The new method of replacement shuffles the ranking since now every document has its own query term representative, and there is a slight tendency of documents not containing all of the query terms to move up in the rank. This effect is not stronger because we consider original query terms more important. Nevertheless, we can see that the number of queries ranking the documents with no missing query term at position one is reduced between the two methods. We also see a document with all query terms being ranked at position 500 in the new method where in the original okapi the same does not occur.

6. CONCLUSIONS

We presented a new method to score missing terms in information retrieval tasks. We modify two original scoring functions, one for passage retrieval and the other for document retrieval, to include new terms in a conservative way while scoring documents. We change the user query only when necessary, and do not include terms unless they are absent in the document (or passages). This strategy guarantees every query term with a representative in each document. Our empirical results show that the methods work well in practice, improving both recall and precision in document retrieval. It also increases the accuracy of passage retrieval evaluated in terms of question answering.

Acknowledgments

We thank Peter Turney and the National Research Council for allowing the use of their computer resources to generate our replacements. This work was partly financed by PUC/RS-Brazil and Ministry of Education of Brazil through CAPES agency.

7. REFERENCES

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229, 1999.

[2] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC 3. In *In proceedings of Third Text REtrieval Conference*, Gaithersburg, MD, 1994.

[3] D. Carmel, E. Farchi, Y. Petruschka, and A. Soffer. Automatic query refinement using lexical affinities with maximal information gain. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 283–290, Tampere, Finland, 2002.

[4] C. Clarke, G. Cormack, M. Laszlo, T. Lynam, and E. Terra. The impact of corpus size on question answering performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 369–370, Tampere, Finland, 2002.

[5] C. L. A. Clarke, G. V. Cormack, and T. R. Lynam. Exploiting redundancy in question answering. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 358–365. ACM Press, 2001.

[6] C. L. A. Clarke, G. V. Cormack, T. R. Lynam, and E. Terra. *Advances in Open Domain Question Answering*, chapter Question answering by passage selection. Kluwer Academic Publishers. To appear, 2004.

[7] K. Collins-Thompson, E. Terra, J. Callan, and C. L. A. Clarke. The effect of document retrieval quality on factoid question answering. In *ACM SIGIR Conference on Research and development in Information Retrieval*, 2004.

[8] C. J. Crouch, D. B. Crouch, Q. Chen, and S. J. Holtz. Improving the retrieval effectiveness of very short queries. *Information Processing and Management*, 38(1):1–36, 2002.

[9] K. Darwish and D. W. Oard. Probabilistic structured query methods. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 338–344, 2003.

[10] J. Firth. *Studies In Linguistic Analisys*, chapter A Synopsis of Linguistic Theory, 1930-1955, pages 1–32. Basil Blackwell, Oxford, 3rd edition, 1957.

[11] J. Gao, M. Zhou, J.-Y. Nie, H. He, and W. Chen. Resolving query translation ambiguity using a decaying co-occurrence model and syntactic dependence relations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 183–190, 2002.

[12] K. S. Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: development and comparative experiments - part 1 and 2. *Information Processing and Management*, 36(6):779–808; 809–840, 2000.

[13] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, 2001.

[14] J. Lin, A. Fernandes, B. Katz, G. Marton, and S. Tellex. Extracting answers from the web using data annotation and knowledge mining techniques. In *The Eleventh Text REtrieval Conference (TREC 2002)*, Gaithersburg, MD, 2002.

[15] M. Mitra, A. Singhal, and C. Buckley. Improving automatic query expansion. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214. ACM Press, 1998.

[16] J. M. Ponte and W. B. Croft. A language modeling

approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, 1998.

[17] J. Rocchio. *The SMART Retrieval System: Experiments in Automatic Document Processing*, chapter Relevance feedback in information retrieval, pages 313–323. Prentice-Hall Inc., 1971.

[18] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[19] P. Schäuble and P. Sheridan. Cross-language information retrieval (clir) track overview. In *The Sixth Text REtrieval Conference (TREC 6)*, Gaithersburg, MD, 1997.

[20] S. Tellex, B. Katz, J. Lin, A. Fernandes, and G. Marton. Quantitative evaluation of passage retrieval algorithms for question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 41–47, 2003.

[21] E. Terra and C. L. Clarke. Fast computation of lexical affinity models. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, 2004.

[22] E. Terra and C. L. A. Clarke. Frequency estimates for statistical word similarity measures. In *Proceedings of Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting*, pages 244–251, Edmonton, Alberta, 2003.

[23] P. D. Turney. Mining the Web for synonyms: PMI–IR versus LSA on TOEFL. In *Proceedings of European Conference on Machine Learning-2001*, pages 491–502, 2001.

[24] J. Xu and B. Croft. Improving the effectiveness of information retrieval with Local Context Analysis. *ACM Transactions on Information Systems*, 18(1):79–112, 2000.

[25] J. Xu and W. B. Croft. Corpus-based stemming using cooccurrence of word variants. *ACM Trans. Inf. Syst.*, 16(1):61–81, 1998.

[26] D. L. Yeung, C. L. A. Clarke, G. V. Cormack, T. R. Lynam, , and E. Terra. Task-specific query expansion (multitext experiments for trec 2003). In *2002 Text REtrieval Conference*, Gaithersburg, MD, 2003.

[27] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 403–410, 2001.