# Query Expansion Using Term Distribution and Term Association

Dipasree Pal, Mandar Mitra Indian Statistical Institute, Kolkata

Kalyankumar Datta Jadavpur University, Kolkata

#### Abstract

Good term selection is an important issue for an automatic query expansion (AQE) technique. AQE techniques that select expansion terms from the target corpus usually do so in one of two ways. Distribution based term selection compares the distribution of a term in the (pseudo) relevant documents with that in the whole corpus / random distribution. Two well-known distribution-based methods are based on Kullback-Leibler Divergence (KLD) [8] and Bose-Einstein statistics (Bo1) [1]. Association based term selection, on the other hand, uses information about how a candidate term co-occurs with the original query terms. Local Context Analysis (LCA) [31] and Relevance-based Language Model (RM3) [15] are examples of association-based methods. Our goal in this study is to investigate how these two classes of methods may be combined to improve retrieval effectiveness. We propose the following combination-based approach. Candidate expansion terms are first obtained using a distribution based method. This set is then refined based on the strength of the association of terms with the original query terms. We test our methods on 11 TREC collections. The proposed combinations generally yield better results than each individual method, as well as other state-of-the-art AQE approaches. En route to our primary goal, we also propose some modifications to LCA and Bo1 which lead to improved performance.

# 1 Introduction

Consider a user's query Q and a relevant document D from a document collection. Q and D may use different vocabulary to refer to the same concept. Information Retrieval (IR) systems that rely solely on keyword-matching may not detect a match between Q and D, and may therefore not retrieve D in response to Q. This is the well-known vocabulary mismatch problem in IR.

A good retrieval system must solve this problem by bridging the vocabulary gap that exists between useful documents and the user's query. Query Expansion (QE) is an important technique that attempts to increase the likelihood

of a match between the query and relevant documents by adding related terms (called *expansion terms*) to a user's query.

A wide variety of methods for Automatic Query Expansion (AQE) have been proposed over the last 15–20 years. These methods find related terms from different sources such as the target corpus, linguistic resources like Wordnet [14], thesauri [22], ontologies [5], the World Wide Web, Wikipedia [18] and query logs [12]. A recent survey of such techniques can be found in [9]. Of all these techniques, methods that use the target corpus as a source of expansion terms are among the most widely used because they are simple and require no additional resources.

Target-corpus-based AQE techniques can be broadly classified into two groups: distribution based and association based. Distribution based methods select terms by comparing the distribution of a term in the (pseudo) relevant documents with its distribution in the whole corpus. Broadly, such methods select terms that are more likely to occur in the (pseudo) relevant documents than in a document chosen randomly from the entire corpus. On the other hand, association based methods select expansion terms on the basis of their association (or co-occurrence) with all query terms. A term that tends to co-occur with all / many of the query terms is regarded as a good expansion term.

While a number of distribution / association-based QE techniques have been shown to be effective on average (i.e. when their overall performance across a large set of queries is measured), the impact of different QE techniques on individual queries can vary greatly.

	Baseline	Assoc. based	Distr. based
MAP	0.218	0.250 (+14.8)	0.257 (+ 18.0%)
Better on		81 queries	91 queries

Table 1: Potential improvement obtainable in principle by judiciously choosing QE techniques

Table 1 shows the Mean Average Precision (MAP) scores for three retrieval methods on TREC queries 301–450 (for more details, please see Section 4): a baseline strategy that uses original, unexpanded queries, and representative distribution-based [8] and association-based [31] QE methods. The QE methods are superior to the baseline on average, but they result in decreased performance for a number of queries. Further, while the overall performance figures for these two QE methods are comparable, each of these methods outperforms the other on about half the queries used in this experiment.

As these two methods work in different ways, our hypothesis is that if we combine these two methods by considering both distribution information and association information, we should be able to improve overall performance. In this study, therefore, we investigate the possibility of improving retrieval effectiveness by combining association- and distribution-based QE approaches. We first select two well-known, representative method from each category, viz. LCA [31], RM3 [15] (association-based) and KLD [8], Bo1 [1] (distribution-based). Next, we introduce some simple modifications in the basic formulae of some of these

methods in order to improve their performance. We verify that these modifications indeed result in better retrieval effectiveness. Finally, the two approaches are combined as follows: we select a relatively large number of candidate expansion terms using the distribution based method. Some of these are filtered out using information from the association based method. The refined set is finally used for query expansion.

We test our combined method on eleven TREC collections. Our proposed method yields significant improvements on all collections over a baseline that uses the original, unexpanded queries. More importantly, the combined methods yield improvements over the individual AQE methods for most of the collections.

In summary, this study makes the following contributions.

- It proposes refinements for some well-known QE methods.
- It demonstrates that a combination of distribution based and association based methods outperforms the individual methods as well as state-of-the-art QE methods, such as the approaches proposed in [15, 1].

In the next section, we discuss the relationship between this study and related work. Section 3 briefly reviews the existing AQE methods that are used in this study, our modifications of these methods, as well as the proposed method for combining AQE techniques. Section 4 describes the experimental setting that we used. Results comparing the proposed methods with existing ones are presented in Section 5. Finally, Section 6 summarizes some related issues that need to be studied in future work.

# 2 Related work

Early work on automatic query expansion dates back to the 1960s. Rocchio's relevance feedback method [29] is still used in its original and modified forms for AQE. The availability of the TREC collections, and the widespread success of AQE on these collections stimulated further research in this area. Carpineto and Romano [9] provide a recent and comprehensive survey of AQE techniques. We focus here on some important AQE techniques that are either distribution-or association-based.

Association-based QE techniques. Early work on association-based AQE includes "concept-based" QE [26] and phrasefinder [11]. Both methods make use of term co-occurrence information extracted from a corpus. Local context analysis (LCA) [31, 30] is another well-known method that also selects expansion terms based on whether they have a high degree of co-occurrence with all query terms. However, in LCA, co-occurrence information is obtained from a set of top-ranked documents retrieved in response to the original query, rather than the whole target corpus. Relevance-based language models [17] constitute another, more recent, co-occurrence based approach. This method is based on the Language Modeling framework. The query and relevant documents are all assumed to be generated from an underlying relevance model. This model

is estimated based on (only) the pseudo relevant documents for a particular query. This approach was subsequently refined by AbdulJaleel et al. [15]. The refinement, called RM3, incorporates the original query when estimating the relevance model. According to a comparative study by Lv et al. [19, 23], RM3 is the most effective and robust among a number of state-of-the-art AQE methods. RM3 is frequently used as a baseline against which several recent QE methods have been compared [23, 20, 3, 6, 16].

Distribution-based QE techniques. As early as 1978, Doszkocs [13] proposed the interactive use of an associative dictionary that was constructed based on a comparative analysis of term distributions. Also well known is Robertson's analysis of term selection for query expansion [27]. More recently, Carpineto et al. [8] proposed an effective QE method based on information theoretic principles. This method uses the Kullback-Leibler divergence (KLD) between the probability distributions of terms in the relevant (or pseudo-relevant) documents and in the complete corpus.

Amati [1] proposes a new distribution based method which uses Bose-Einstein statistics. This method also calculates the divergence between the distribution of terms in the pseudo relevant document set and a random distribution.

Efforts have also been made to combine AQE methods in various ways to improve retrieval effectiveness. Carpineto et al. [10] combined the scoring functions of a number of methods, all of them distribution-based, to obtain improvements. In contrast, we combine a distribution-based method with an association-based method (based on our belief that these two classes of methods offer different advantages). Also, rather than combining scores, we use one method to refine the set of terms selected by the other. Our approach is somewhat similar in spirit to a method proposed by Cao et al. [7], in which terms selected using standard pseudo relevant feedback (PRF) are refined using a classifier that is trained to differentiate between useful and harmful candidate expansion terms. Our work is most strongly related to that of Pérez-Agüera and Araujo [24], who also combine co-occurrence-based and distribution-based methods. The combination is relatively straightforward: one method is used for term selection and the other for weighting. Word co-occurrence is measured using the Tanimoto coefficient. Distributional differences are measured based on KLD or Bose-Einstein statistics. The methods are tested on a relatively small Spanish dataset. We use the well-known LCA and RM3 method (instead of Tanimoto coefficient) to quantify term association. Also, instead of simply using one method for term selection and the other for weighting, we combine both methods for selection. Finally, we test our method on a number of large TREC datasets.

 $<sup>^{1}</sup>$  Of course, this can also, strictly speaking, be regarded as a combination where one component is very highly weighted.

# 3 Methods

#### 3.1 Basic Methods

We first review KLD, Bo1, LCA and RM3, the existing methods that form the base of our approach.

#### 3.1.1 Distribution based method I: KLD

The approach proposed by Carpineto et al. [8] is one of the two distribution based term ranking methods used in this study. In this method, all terms in the pseudo relevant set are treated as candidate expansion terms. Let R and C represent the (pseudo) relevant documents (PRD) and the whole corpus respectively. We use  $p_r$  and  $p_c$  to denote the unigram probability distribution of terms in R and C respectively;  $p_r$  and  $p_c$  are calculated as shown in Equations 1 and 2 (tf(t, d)) represents the term frequency of term t in document d).

$$p_r(t) = \frac{\sum_{d \in R} tf(t, d)}{\sum_{d \in R} \sum_{t' \in d} tf(t', d)}$$
(1)

$$p_c(t) = \frac{\sum_{d \in C} tf(t, d)}{\sum_{d \in C} \sum_{t' \in d} tf(t', d)}$$
(2)

The contribution of a term to the divergence between  $p_r$  and  $p_c$  is given by Equation 3. Terms for which this contribution is the largest are selected as expansion terms.

$$S(t) = p_r(t) * \log \frac{p_r(t)}{p_c(t)}$$
(3)

S(t) is also used as the term weight of a candidate expansion term t.

In our experiments with KLD (and other methods), we use Equations 4 to 6 to merge the original query terms with the candidate expansion terms to formulate the final expanded query. The weights of original query terms are normalized using the maximum original query term weight (Eqn. 4); weights of expansion terms are similarly normalized (Eqn. 5). These weights are simply added together to obtain the final weight of a term t in the expanded query (Eqn. 6).

$$score_{orig}(t) = \frac{1 + \log(tf(t,Q))}{1 + \max_{t' \in Q} \log(tf(t',Q))}$$
(4)

$$score_{exp}(t) = \frac{S(t)}{\max_{t' \in d \in PRD} S(t')}$$
 (5)

$$score(t) = score_{orig}(t) + score_{exp}(t)$$
 (6)

#### 3.1.2 Distribution based method II: Bo1

The second, more recent, distribution based term ranking model we considered is Bo1, which is the most effective variant of the Divergence From Randomness (DFR) term weighting model [25, 21]. In this model based on Bose-Einstein statistics, the informativeness of a term t is measured by the divergence between its distribution in the top ranked documents and a random distribution. Specifically, the score of a candidate expansion term t is given by

$$S(t) = \left(\sum_{d \in PRD} t f(t, d)\right) * \log_2\left(\frac{1 + f_{avg}(t, C)}{f_{avg}(t, C)}\right) + \log_2\left(1 + f_{avg}(t, C)\right)$$
(7)

where

$$f_{avg}(t,C) = \sum_{d \in C} tf(t,d)/N$$
(8)

denotes the average term frequency of t in the collection (N is the number of documents in the collection). As in Section 3.1.1, we use Equations 4–6 to merge the original query with the expansion terms and formulate the new expanded query.

#### 3.1.3 Modified Bo1

Taking the Bo1 formula as a starting point, we modify it as follows to obtain a more effective scoring function for an expansion term t. First, an occurrence of t in a top-ranked document is considered more important than an occurrence in a lower ranked document. Thus, instead of using tf(t,d) directly, we scale the term frequencies by the normalized similarity score of the corresponding document. We then incorporate inverse collection frequency information as shown in Equation 10.

While the tf factor in Equation (10) is indicative of the distribution of t in the top ranked set, the ictf factor reflects the distribution of the term in the collection.

$$ictf(t) = \log_{10}\left(\frac{1}{p_c(t)}\right)$$
 (9)

$$S(t) = \sum_{d \in PRD} \left( tf(c, d) * \frac{\operatorname{Sim}(d, Q)}{\max_{d' \in PRD} \operatorname{Sim}(d', Q)} \right) * \frac{ictf(t)}{1 + ictf(t)}$$
(10)

Finally, Equations 4 to 6 are used to merge the original query with the expansion terms.

#### 3.1.4 Association based method I: LCA

LCA [31] is one of the most well-known association based term selection methods. This method also considers all terms from the top ranked set as candidate expansion terms. Equations 11 through 14 show how the co-occurrence is calculated for a candidate term t and a query Q consisting of terms  $q_1, \ldots, q_k$  ( $N_t$ 

has its obvious meaning, PRD denotes the set of pseudo-relevant documents, n = |PRD|,  $\delta$  is set to 0.1 as suggested in [31]).

$$idf_t = min(\log_{10}(N/N_t)/5.0, 1.0)$$
 (11)

$$co(t, q_i) = \sum_{d \in PRD} tf(t, d) * tf(q_i, d)$$
(12)

$$codegree(t, q_i) = \frac{\log_{10}(co(t, q_i) + 1) * idf_t}{\log_{10}(n)}$$

$$(13)$$

$$S(t) = \sum_{i=1}^{k} i df_{q_i} * \log_{10}(\delta + codegree(t, q_i))$$
(14)

The T terms with the highest S(t) scores are selected as expansion terms. Finally, the j-th "best" term is weighted according to Equation 15.

$$score_{exp}(t) = 1.0 - \frac{0.9 * j}{T} \tag{15}$$

We did not use noun-phrases or passage level retrieval, since the authors show that these refinements do not have much impact. Our experiments confirm that our implementation yields very similar results for the collections and settings mentioned in [31].

#### 3.1.5 Modified LCA

Our implementation of the above formulae did not yield the expected improvements. A failure analysis suggests that Equation 12 might be the culprit. For example, consider the TREC4 query: "How has affirmative action affected the construction industry?". Two terms papuc (Pennsylvania Public Utility Commission) and limerick are very highly ranked among candidate terms by Equation 14, even though these are not useful expansion terms. This is because in one top document, the word 'papuc' occurs 21 times and a query word ('construction') occurs 35 times. The multiplication of raw term frequencies in Equation 12 results in a very high weight for the term 'papuc'. A similar problem occurs in case of 'limerick', which occurs 17 times in one document.

Our hypothesis is that the number of co-occurrences of a term pair can only be as large as the minimum term frequency of the two terms under consideration. We also hypothesize that co-occurrences in a document are more important if the document is "close" (or similar) to the query. Finally, we use the idf factor<sup>2</sup> for a candidate expansion term when calculating its co-occurrence (Equation 17); it is no longer used when calculating co-degree (Equation 18). Equations 16–19 define our modified approach for calculating the association between a candidate term t and the query Q.

$$idf_t = \log_{10} \frac{N - N_t + 0.5}{N_t + 0.5} \tag{16}$$

 $<sup>^2</sup>$ Note that we use Robertson's idf formula [28] (Equation 16) instead of Equation 11.

$$co(t, q_i) = \sum_{d \in PRD} \left( \min(tf(t, d), tf(q_i, d)) * \max(idf_{t \vee q_i}, 0) * \frac{\operatorname{Sim}(d, Q)}{\max_{d' \in PRD} \operatorname{Sim}(d', Q)} \right)$$

$$(17)$$

where  $idf_{t\vee q_i}$  denotes the idf of term t or  $q_i$ , based on whose term frequency is minimum in document d.

$$codegree(t, q_i) = \frac{\log_{10}(co(t, q_i) + 1)}{\log_{10}(n)}$$
(18)

$$S(t) = \sum_{i=1}^{k} i df_{q_i} * \log_{10}(\delta + codegree(t, q_i))$$
(19)

As before, the T terms with the highest association scores (S(t)) are selected as expansion terms. The final term weights in the expanded query are determined using Equations 4 to 6.

#### 3.1.6 Association based method II: RM3

Relevance-based language models [17, 15] constitute a more recent association-based approach. In this approach, the association S(t) between a word t and a query  $Q = q_1, \ldots, q_k$  can be measured by  $P(t, q_1, \ldots, q_k)$ , the joint probability of observing the word together with the query words, when these words are all sampled from an (unknown) relevance model. This relevance model consists of a finite universe  $\mathcal{M}$  of unigram distributions each of which corresponds to a (pseudo) relevant document. Under the assumption that  $t, q_1, \ldots, q_k$  are independently and identically sampled from  $M \in \mathcal{M}$ ,

$$S(t) = P(t, q_1, ..., q_k)$$

$$= \sum_{M \in \mathcal{M}} P(M)P(t|M) \prod_{i=1}^{k} P(q_i|M)$$

$$= \frac{1}{\#PRD} \sum_{d \in PRD} \left( \frac{tf(t, d)}{|d|} \times \prod_{i=1}^{k} \frac{tf(q_i, d) + \mu P(q_i|C)}{|d| + \mu} \right), \quad (20)$$

where  $\mu = 2500$  is a smoothing parameter, and  $P(q_i|C) = p_c(q_i)$ . Equations 21 to 23 show, as before, how the expanded query terms are added to the original query. This implementation duplicates the LEMUR RM3 method. However, we used the i.i.d. sampling approach instead of the conditional sampling method recommended in [17], since this gave us better results.

$$score_{exp}(t) = \frac{S(t)}{\sum_{d \in PRD} \sum_{t' \in d} S(t')}$$
 (21)

$$score_{orig}(t) = \frac{tf(t,Q)}{|Q|}$$
 (22)

$$score(t) = \alpha * score_{exp}(t) + (1 - \alpha) * score_{orig}(t), \text{ where } 0 \le \alpha \le 1$$
 (23)

# 3.2 Combining association based method with distribution based method

Section 3.1 reviews two different types of query expansion methods. In this section, we describe a hybrid approach that combines the above methods to improve retrieval effectiveness.

We conducted some preliminary experiments to explore various ways to combine individual methods. Our first attempt involved simply adding up the normalized weights of the expansion terms as computed by the individual methods. This particular method did not perform better than the individual methods. Next, we tried to apply the methods sequentially: the original query is expanded using one of the methods, and the expanded query is then used as the initial query for the other method and expanded further. This approach also results in a performance drop. The final approach that we tried also applies the methods sequentially, but in a different way. One of the methods is used first to create a large expanded query. This query is then refined (instead of being expanded further) using the other method. This method turns out to work well, and yields significant improvements over the individual methods.

We can see from Table 5 that the distribution-based methods generally perform better than the association-based methods on most of the test collections used in our experiments. We therefore choose a distribution-based method KLD (Equation 3) or Bo1 (Equation 10) — to first select (and weight) a relatively large number of candidate terms that occur preferentially in a few topranked documents, where the proportion of relevant documents is expected to be high. This set is then refined using co-occurrence information: terms that do not co-occur significantly with original query terms are discarded. Conversely, candidate expansion terms that are relatively poorly ranked by the distributionbased method have a chance to be included in the final query if they adequately co-occur with the original query terms. More precisely, the candidate terms are re-ranked using an association-based method — our modified version of LCA (Equation 19) or RM3 (Equation 20) — that looks at a larger number of topranked documents. The top T terms from this re-ranked list are chosen as the final expansion terms. However, we retain the weights of these terms as determined by the distribution based method. As before, the final term weights in the expanded query are determined using Equations 4 to 6.

# 4 Experimental Setup

Table 2 lists the details of the test collections used in our experiments. As reallife queries are very short, we used only the title field of all these queries, except for the TREC4 queries, which contain only the description field. Many of the

Table 2: Test collections

Query Id.	# of Queries	Documents
• 0	11	
TREC123	150	TREC disks 1, 2
51 - 200		
TREC4	49	TREC disks 2, 3
202-250		
TREC5	50	TREC disks 2, 4
251-300		
TREC678	150	TREC disks 4, 5 - CR
301 - 450		
ROBnew	100	TREC disks 4, 5 - CR
601 - 700		
TREC910	100	WT10G
451-550		

queries thus contain only one term, and most of the remainder are no longer than three words; only the TREC4 queries are longer.

We used the TERRIER<sup>3</sup> retrieval system for our experiments. At the time of indexing, stopwords are removed and Porter's stemmer is used as preprocessing. All documents and queries are indexed using single terms, no phrases are used. The IFB2 variant of the Divergence From Randomness model [2] — a relatively recent model that performs well across test collections — is used for term-weighting in all our experiments as it performs better compared to the other variants available within TERRIER. Parameters are set to the default values used in TERRIER.

Results are evaluated using standard evaluation metrics (Mean Average Precision (MAP), precision at top 10 ranks (P@10), and overall recall (number of relevant documents retrieved)). Additionally, for each expansion method, we report the percentage of queries for which the method resulted in an improvement in MAP of more than 5% over the baseline (no feedback).

# 5 Experimental Results

We now present experimental results for the QE methods described in Section 3. The first set of results presented in Section 5.1 pertain to our implementation of well-known QE methods, as well as the proposed refinements to these methods. Section 5.2 corresponds to the combination-based method described in Section 3.2.

**Notation.** We use the following labels to denote various techniques in tables / figures. In the following tables, results that are statistically significantly better (as determined by a two-tailed paired t-test with a confidence level of 95%) than the baseline (no feedback), KLD, Bo1new, LCAnew and RM3 are marked with the superscripts B, k, b, l and r respectively.

<sup>&</sup>lt;sup>3</sup>http://terrier.org/

Name	Description	Details in
KLD	Our implementation of the KLD method	Section 3.1.1
Bo1	Our implementation of the Bo1 method	Section 3.1.2
Bo1new	Modified Bo1 method	Section 3.1.3
LCA	Our implementation of the LCA method	Section 3.1.4
LCAnew	Modified LCA method	Section 3.1.5
RM3	Our implementation of the RM3 method	Section 3.1.6
KLDLCA	Combination of KLD and LCAnew	)
KLDRM3	Combination of KLD and RM3	(0.4:
Bo1LCA	Combination of Bo1new and LCAnew	Section 3.2
Bo1RM3	Combination of Bo1new and RM3	J

Table 3: Labels for various QE methods

## 5.1 Experiment 1: modified methods

Baselines. For comparison, we use the following baselines.

- 1. No feedback. The original, unexpanded queries are used for retrieval using the baseline method described in Section 4.
- 2. Bo1. For this method, Amati [1] suggested adding T=10 expansion terms from the top D=3 documents. We use T=40 and D=10 instead, since we wanted a larger number of candidate terms, particularly for the combination-based method. Our experiments confirm that we get comparable results with these parameter settings.
- 3. LCA. To determine the parameters for LCA, we used the TREC678 collection as a "tuning" dataset, as TREC678 is comparatively recent, and contains a large set of queries. We varied the number of top-ranked documents (D) from 10 to 50 in steps of 10, while the number of expansion terms (T) was varied from 5 to 50 in steps of 5. Xu and Croft [31] recommended using D=70 and T=70. In our setup, however, a setting of D=10 documents and T=40 expansion terms works well. Figure 1 shows that these settings work well in terms of MAP. We use these values on all collections used in our experiments. A similar exercise suggests that the same settings can be used for LCAnew as well.

LCAnew. Our first goal is to verify that the proposed modifications to the LCA formula actually yield improvements in retrieval performance. Table 4 shows that, with 'title only' queries, our implementation of the original LCA formula results in a drop in MAP for almost all collections. Only for the TREC4 collection (in which queries consist of a description only), a marginal improvement is observed, suggesting that the original method works better for longer queries. Indeed, the experiments by Xu and Croft all used relatively long queries, e.g., the full TREC3 queries (including title, description and narrative fields), the TREC4 queries, and the description field of TREC5 queries.

Compared to the original formula, the modified formula results in significant improvements in MAP across all data sets. On the ROBnew collection in

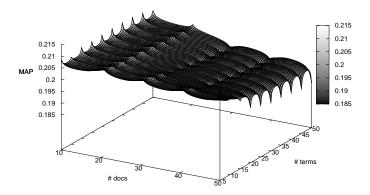


Figure 1: Performance (MAP) of LCA on TREC678 for different parameter settings.

particular, it performs very well, outperforming the original method by nearly 24%. For the TREC4 corpus, an improvement of about 11.41% (over LCA) is observed. The modifications thus seem to be effective for both short as well as relatively longer queries. The LCAnew method is also better in terms of P@10, number of relevant documents retrieved, and robustness.

Bo1new. Table 4 shows that Bo1new gives better results than Bo1 across all test collections. For the TREC123, ROBnew, and TREC910 collections, these improvements are significant. The modified method also yields better P@10, and appears to be more robust across all datasets. With regard to the number of relevant documents retrieved, Bo1new is better on most collections. Overall, Bo1new appears to be a superior alternative to Bo1 in all respects.

Thus, based on Table 4, we conclude that LCAnew and Bo1new are more effective, and can be used in place of LCA and Bo1.

## 5.2 Experiment 2: combination methods

As explained in Section 3.2, in the combination-based approach, we first select a large set of candidate terms (T=100) from D=10 documents using a distribution-based QE method. The association of these candidate terms with the query terms is computed using the top D'=50 documents<sup>4</sup>, and the best T'=40 terms (as determined by an association-based method) are included in the final query. We report results for a total of  $2\times 2=4$  combinations: KLDLCA (LCAnew with KLD), KLDRM3 (RM3 with KLD), Bo1LCA (LCAnew with Bo1new), and Bo1RM3 (RM3 with Bo1new).

 $<sup>^4</sup>$ Measuring association scores over the top 30–50 documents works about equally well.

Dataset	Measure	Baseline	LCA	LCAnew	Bo1	Bo1new
TREC123	MAP	0.218	0.213	$0.254^{B*}$	0.272	$0.277^{B*}$
			(-2.4)	(16.4)	(24.4)	(26.6)
	P@10	0.481	0.472	0.520	0.531	0.545
			(-1.8)	(8.2)	(10.4)	(13.4)
	#rel_ret	16536	15714	17475	18227	18242
			(-5.0)	(5.7)	(10.2)	(10.3)
	> baseline on	0	35	54	58	62
TREC4	MAP	0.217	0.219	$0.244^{B}$	0.256	$0.259^{B}$
			(1.1)	(12.6)	(17.8)	(19.5)
	P@10	0.461	0.400	0.496	0.441	0.467
			(-13.3)	(7.5)	(-4.4)	(1.3)
	#rel_ret	3482	3507	3691	3854	3768
			(0.7)	(6.0)	(10.7)	(8.2)
	> baseline on	0	38	57	55	57
TREC5	MAP	0.157	0.130	0.152*	0.166	0.168
			(-17.6)	(-3.1)	(5.4)	(7.0)
	P@10	0.286	0.210	0.238	0.248	0.270
			(-26.6)	(-16.8)	(-13.3)	(-5.6)
	#rel_ret	1936	1894	2053	2194	2183
	, ,,		(-2.2)	(6.0)	(13.3)	(12.8)
	> baseline on	0	20	38	42	44
TREC678	MAP	0.218	0.209	$0.250^{B*}$	0.255	$0.257^{B}$
	2010		(-4.2)	(14.8)	(16.8)	(17.7)
	P@10	0.431	0.379	0.420	0.427	0.436
		=00=	(-12.2)	(-2.6)	(-0.9)	(1.1)
	#rel_ret	7287	7367	8152	8529	8463
	. 1 11		(1.1)	(11.9)	(17.0)	(16.1)
	> baseline on	0	36	52	53	60
ROBnew	MAP	0.278	0.264	$0.327^{B*}$	0.307	$0.331^{B*}$
	D@10	0.401	(-5.0)	(17.6)	(10.3)	(19.0)
	P@10	0.421	0.385	0.452	0.394	0.433
	// 1 /	9007	(-8.6)	(7.2)	(-6.5)	(2.9)
	#rel_ret	2887	2864	3009	3178	3202
	S. 1 15	0	(-0.8) 36	(4.2) 53	(10.1)	(10.9)
TREC910	> baseline on MAP	0.195		0.175	48 0.189	56 0.202*
1 KEC910	MAP	0.195	0.155 (-20.6)			
	P@10	0.307	0.231	(-10.6) 0.291	(-3.3) 0.284	(3.5) $0.304$
	1 @10	0.507	(-24.9)	(-5.3)	(-7.6)	(-1.0)
	#rel_ret	3770	3440	3646	3974	3948
	#101=100	3110	(-8.8)	(-3.3)	(5.4)	(4.7)
	> baseline on	0	27	33	41	45

Table 4: Improvements on different datasets obtained by modifying LCA / Bo1. The "> baseline on" line shows the %-age of queries on which each method beats the baseline by > 5%. A \* denotes an improvement (over original formula) that is statistically significant.

Baselines. We compare the combination-based methods with the following baselines.

- 1. No feedback. Same as in Section 5.1
- 2. KLD. We find that a setting of D = 10 top-ranked documents and T = 40 expansion terms works well for KLD across collections. This is in agreement with the observations of Carpineto et al.[8].

Note that the results presented here correspond to our implementation of KLD within TERRIER. While our implementation provides better results than TERRIER's native implementation of KLD, we were not able to exactly replicate the results reported in [8]. This is likely due to differences between the retrieval functions, indexing or query processing. For example, using full queries (title, desc and narr) on the TREC8 collection, and BM25 as the base term-weighting formula, we get MAP scores of 0.2992 for KLD (compared to a baseline of 0.2625). When using the IFB2 model, however, the baseline is higher (MAP = 0.2753), but KLD appears less effective (MAP = 0.2850).

- 3. Bo1new, LCAnew. As discussed in Section 5.1, for these methods also, we use D=10 documents, and T=40 terms.
- 4. RM3. We use D=50 documents (as suggested in [17]) and T=50 terms. We set the Dirichlet smoothing parameter  $(\mu)$  to 2500 and the interpolation parameter to 0.5, based on the default settings for these parameters in Lemur <sup>5</sup>. As before, we used the TREC678 collection to verify that these parameter values work well for us. In fact, for a number of datasets, our results for RM3 are superior to those reported in other recent papers ([4], for example).

Table 5 shows that the proposed combined approaches are statistically significantly better than the no-feedback method across all test collections except for TREC5 and TREC910. More importantly, the combined methods consistently work better than the individual QE methods involved in the combination, as well as most of the other standard QE methods. These differences are, by and large, statistically significant, with only a few exceptions. Overall, while RM3 seems to be the best in terms of P@10 in most of the cases, the combination based methods are generally the best on all other measures. We now briefly discuss each combination in turn.

KLDLCA. KLDLCA is better than KLD or LCAnew alone on all measures, and across all datasets. For 5 out of the 6 collections, the combination yields significant improvements in MAP over KLD or LCAnew or both. It is interesting to note that for the sixth collection (TREC5), LCAnew results in a drop in performance compared to the no-expansion baseline. However, the combinations KLDLCA and Bo1LCA perform better than the baseline as well as KLD.

<sup>&</sup>lt;sup>5</sup>http://www.lemurproject.org/

Dataset	Measure	Baseline	KLD	Bo1new	LCAnew	RM3	KLDLCA	KLDRM3	Bo1LCA	Bo1RM3
TREC123	MAP	0.218	0.274	0.277	0.254	0.249	$0.280^{Blr}$	$0.277^{Blr}$	$0.285^{Bklr}$	$0.284^{Blr}$
			(25.4)	(26.6)	(16.4)	(14.1)	(28.0)	(26.8)	(30.6)	(29.8)
	P@10	0.481	0.537	0.545	0.520	0.511	0.537	0.541	0.553	0.540
			(11.8)	(13.4)	(8.2)	(6.2)	(11.8)	(12.5)	(15.0)	(12.3)
	#rel_ret	16536	18299	18242	17475	17702	18585	18438	Ì870Í	18639
			(10.7)	(10.3)	(5.7)	(7.1)	(12.4)	(11.5)	(13.1)	(12.7)
	> baseline on	0	62	62	54	64	67	65	67	68
TREC4	MAP	0.217	0.261	0.259	0.244	0.252	$0.279^{Bkblr}$	$0.265^{B}$	$0.273^{Bl}$	$0.265^{B}$
			(20.2)	(19.5)	(12.6)	(15.9)	(28.7)	(22.3)	(25.6)	(21.9)
	P@10	0.461	0.455	0.467	0.496	0.516	0.498	0.480	0.498	0.502
			(-1.3)	(1.3)	(7.5)	(11.9)	(8.0)	(4.0)	(8.0)	(8.8)
	#rel_ret	3482	3815	3768	3691	3689	3882	3781	3846	3775
			(9.6)	(8.2)	(6.0)	(5.9)	(11.5)	(8.6)	(10.5)	(8.4)
	> baseline on	0	57	57	57	75	55	59	57	61
TREC5	MAP	0.157	0.168	0.168	0.152	0.170	0.171	$0.172^{k}$	$0.174^{l}$	0.173
			(6.9)	(7.0)	(-3.1)	(8.2)	(9.0)	(9.2)	(10.4)	(9.9)
	P@10	0.286	0.268	0.270	0.238	0.336	0.274	0.280	0.290	0.304
			(-6.3)	(-5.6)	(-16.8)	(17.5)	(-4.2)	(-2.1)	(1.4)	(6.3)
	#rel_ret	1936	2184	2183	2053	2077	2218	2166	2226	2184
			(12.8)	(12.8)	(6.0)	(7.3)	(14.6)	(11.9)	(15.0)	(12.8)
	> baseline on	0	42	44	38	50	52	50	48	48
TREC678	MAP	0.218	0.257	0.257	0.250	0.230	$0.266^{Bkblr}$	$0.260^{Br}$	$0.265^{Bblr}$	$0.259^{Br}$
			(18.0)	(17.7)	(14.8)	(5.6)	(22.0)	(19.2)	(21.6)	(18.7)
	P@10	0.431	0.438	0.436	0.420	0.435	0.441	0.431	0.435	0.428
			(1.6)	(1.1)	(-2.6)	(0.8)	(2.2)	(0.0)	(0.8)	(-0.8)
	#rel_ret	7287	8556	8463	8152	7617	8567	8552	8570	8449
		_	(17.4)	(16.1)	(11.9)	(4.5)	(17.6)	(17.4)	(17.6)	(15.9)
	> baseline on	0	52	60	52	45	57	57	61	58
ROBnew	MAP	0.278	0.312	0.331	0.327	0.305	$0.326^{Bkr}$	$0.322^{Bk}$	$0.341^{Bkbr}$	$0.341^{Bkbr}$
			(12.2)	(19.0)	(17.6)	(9.8)	(17.2)	(15.9)	(22.5)	(22.6)
	P@10	0.421	0.405	0.433	0.452	0.442	0.438	0.424	0.455	0.455
			(-3.8)	(2.9)	(7.2)	(5.0)	(4.1)	(0.7)	(7.9)	(7.9)
	#rel_ret	2887	3172	3202	3009	3002	3173	3160	3214	3218
	s 1 12		(9.9)	(10.9)	(4.2)	(4.0)	(9.9)	(9.5)	(11.3)	(11.5)
TED FI COA O	> baseline on	0	52	56	53	56	55	57	62	63
TREC910	MAP	0.195	0.193	0.202	0.175	0.211	$0.204^{kl}$	$0.210^{kl}$	$0.207^{kl}$	$0.213^{kl}$
	D@10	0.007	(-1.1)	(3.5)	(-10.6)	(8.0)	(4.7)	(7.4)	(6.0)	(9.1)
	P@10	0.307	0.293	0.304	0.291	0.329	0.313	0.309	0.320	0.313
	// 1 /	9770	(-4.6)	(-1.0)	(-5.3)	(7.0)	(2.0)	(0.7)	(4.3)	(2.0)
	#rel_ret	3770	3987	3948	3646	3889	4021	3992	4016	4018
	> basalina		(5.8)	(4.7) 45	(-3.3)	(3.2)	(6.7)	(5.9)	(6.5)	(6.6) 48
	> baseline on	0	44	45	33	53	51	50	53	48

Table 5: Improvements on different datasets obtained by combining association based and distribution based QE methods. (The "> baseline on" line shows the %-age of queries on which each method beats the baseline by > 5%.)

In general, the combination also seems to be safer, in the sense that combination-based expansion usually hurts fewer queries than expansion using either KLD or LCAnew. On a related note, a query wise analysis of the TREC678 dataset shows that out of the 150 queries in this collection, there are 59 queries on which KLD outperforms KLDLCA (with an average improvement in MAP of 0.0148), but KLDLCA does better than KLD on 85 queries, and improves MAP by 0.0255 on average. Similarly, LCAnew performs better than KLDLCA on 68 queries (average improvement in MAP = 0.0360), whereas KLDLCA wins on 81 queries (average improvement in MAP = 0.0594).

It is particularly encouraging that KLDLCA is also generally better than the two other state-of-the-art QE methods, RM3 and Bo1new, on all measures and across all datasets. The only exceptions are: RM3 yields better P@10 on TREC4,TREC5, ROBnew and TREC910 and superior MAP for TREC910, while Bo1 outperforms KLDLCA on P@10 for TREC123, on MAP for ROBnew, and on the number of relevant documents retrieved for ROBnew.

KLDRM3. KLDRM3 also yields better MAP than either KLD or RM3 on all collections (but neither difference is statistically significant for TREC4). It is also better than the other individual QE methods (LCAnew and Bo1new) on all corpora except ROBnew, where Bo1new outperform KLDRM3. This method is among the safest: only Bo1new yields improvements on marginally more queries for the TREC678 collection; on all other datasets, expansion by KLDRM3 improves performance on more queries than any other method.

Bo1LCA, Bo1RM3. Both methods yield improvements (often significant) in MAP compared to all individual QE methods. Indeed, with a few exceptions, Bo1LCA is better than all individual QE methods for all the datasets and on all the measures.

#### 5.3 Discussion

The results in the preceding section confirm our hypothesis that, on average, distribution and association based methods work well together. For queries such as 321 (Women in Parliaments), the combination works as expected. Both LCAnew (AP = 0.2531) and KLD (AP = 0.2611) select and assign relatively high weights to specific names such as mashokw, jankowska, starkova, fedulova. When LCAnew is used to filter terms based on association information obtained from 50 documents, these terms are eliminated, and retrieval effectiveness goes up (AP = 0.3629).

More interesting are queries where the combination fails. Query 350 (Health and  $Computer\ Terminals$ ) is one such example for which LCAnew (AP = 0.5911) and KLD (AP = 0.4512) both do reasonably well, but AP drops to 0.4007 for KLDLCA. For this particular query, filtering candidates terms using association information results in the elimination of a number of good expansion terms.

Unfortunately, no general pattern seems to be discernible for such queries where a combination is inferior to either or both of its ingredients.

# 6 Conclusion

In this study, our objective was to combine distribution based and association based query expansion methods. Using a number of standard test collections, we have shown that distribution based QE can be improved by using an association based method to refine term selection. The proposed combination gives better results than each individual method, as well as other state-of-the-art approaches.

En route to this goal, we also proposed some modifications to a few well-known QE methods which lead to improved performance. This may be regarded as an additional contribution of this paper.

In future work, we intend to do a more comprehensive study by investigating other combinations of QE methods.

# References

- [1] G. Amati. Probability Models for Information Retrieval Based on Divergence from Randomness. University of Glasgow, 2003.
- [2] Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst., 20:357–389, October 2002.
- [3] Michael Bendersky, Donald Metzler, and W. Bruce Croft. Parameterized concept weighting in verbose queries. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 605–614, New York, NY, USA, 2011. ACM.
- [4] Michael Bendersky, Donald Metzler, and W. Bruce Croft. Parameterized concept weighting in verbose queries. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 605–614, New York, NY, USA, 2011. ACM.
- [5] J. Bhogal, A. Macfarlane, and P. Smith. A review of ontology based query expansion. *Inf. Process. Manage.*, 43(4):866–886, July 2007.
- [6] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '08, pages 243–250, New York, NY, USA, 2008. ACM.
- [7] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '08, pages 243–250, New York, NY, USA, 2008. ACM.

- [8] Claudio Carpineto, Renato de Mori, Giovanni Romano, and Brigitte Bigi. An information-theoretic approach to automatic query expansion. *ACM Trans. Inf. Syst.*, 19(1):1–27, 2001.
- [9] Claudio Carpineto and Giovanni Romano. A survey of automatic query expansion in information retrieval. ACM Comput. Surv., 44(1):1:1-1:50, January 2012.
- [10] Claudio Carpineto, Giovanni Romano, and Vittorio Giannini. Improving retrieval feedback with multiple term-ranking function combination. *ACM Trans. Inf. Syst.*, 20(3):259–290, 2002.
- [11] Bruce Croft and Jing Yufeng. An association thesaurus for information retrieval. In *RIAO*, pages 146–161, 1994.
- [12] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. Probabilistic query expansion using query logs. In *Proceedings of the 11th international* conference on World Wide Web, WWW '02, pages 325–332, New York, NY, USA, 2002. ACM.
- [13] Tamas E. Doszkocs. An associative interactive dictionary for online bibliographic searching. In *Jerusalem Conference on Information Technology*, pages 489–492, 1978.
- [14] Hui Fang. A re-examination of query expansion using lexical resources. In *In Proceedings of ACL-08: HLT*, pages 139–147, 2008.
- [15] Nasreen Abdul Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. Umass at trec 2004: Novelty and hard. In TREC, 2004.
- [16] Eyal Krikon, Oren Kurland, and Michael Bendersky. Utilizing inter-passage and inter-document similarities for reranking search results. ACM Trans. Inf. Syst., 29(1):3:1–3:28, December 2010.
- [17] Victor Lavrenko and W. Bruce Croft. Relevance-based language models. In *SIGIR*, pages 120–127, 2001.
- [18] Yinghao Li, Wing Pong Robert Luk, Kei Shiu Edward Ho, and Fu Lai Korris Chung. Improving weak ad-hoc queries using wikipedia asexternal corpus. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 797–798, New York, NY, USA, 2007. ACM.
- [19] Yuanhua Lv and ChengXiang Zhai. A comparative study of methods for estimating query language models with pseudo feedback. In *Proceedings* of the 18th ACM conference on Information and knowledge management, CIKM '09, pages 1895–1898, New York, NY, USA, 2009. ACM.

- [20] Yuanhua Lv and ChengXiang Zhai. Positional relevance model for pseudorelevance feedback. In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, SIGIR '10, pages 579–586, New York, NY, USA, 2010. ACM.
- [21] Craig Macdonald, Ben He, Vassilis Plachouras, and Iadh Ounis. University of glasgow at trec 2005: Experiments in terabyte and enterprise tracks with terrier. In Ellen M. Voorhees and Lori P. Buckland, editors, *TREC*, volume Special Publication 500-266. National Institute of Standards and Technology (NIST), 2005.
- [22] Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. Combining multiple evidence from different types of thesaurus for query expansion, 1999.
- [23] Jun Miao, Jimmy Xiangji Huang, and Zheng Ye. Proximity-based rocchio's model for pseudo relevance. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 535–544, New York, NY, USA, 2012. ACM.
- [24] José R. Pérez-Agüera and Lourdes Araujo. Comparing and combining methods for automatic query expansion. *CoRR*, abs/0804.2057, 2008.
- [25] Vassilis Plachouras, Ben He, and Iadh Ounis. University of glasgow at trec 2004: Experiments in web, robust, and terabyte tracks with terrier. In Ellen M. Voorhees and Lori P. Buckland, editors, *TREC*, volume Special Publication 500-261. National Institute of Standards and Technology (NIST), 2004.
- [26] Yonggang Qiu and Hans-Peter Frei. Concept based query expansion. In Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '93, pages 160– 169, New York, NY, USA, 1993. ACM.
- [27] S. E. Robertson. On term selection for query expansion. J. Doc., 46(4):359–364, January 1991.
- [28] Stephen Robertson. Understanding inverse document frequency: On theoretical arguments for idf. *Journal of Documentation*, 60:2004, 2004.
- [29] Gerard Salton, editor. The SMART Retrieval System Experiments in Automatic Document Processing. Prentice Hall, Englewood, Cliffs, New Jersey, 1971.
- [30] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *SIGIR*, pages 4–11, 1996.
- [31] Jinxi Xu and W. Bruce Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18(1):79–112, 2000.