A Few Examples Go A Long Way

Constructing Query Models from Elaborate Query Formulations

Krisztian Balog
kbalog@science.uva.nl

Wouter Weerkamp
weerkamp@science.uva.nl

Maarten de Rijke
mdr@science.uva.nl

ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam

ABSTRACT

We address a specific enterprise document search scenario, where the information need is expressed in an elaborate manner. In our scenario, information needs are expressed using a short query (of a few keywords) together with examples of key reference pages. Given this setup, we investigate how the examples can be utilized to improve the end-to-end performance on the document retrieval task. Our approach is based on a language modeling framework, where the query model is modified to resemble the example pages. We compare several methods for sampling expansion terms from the example pages to support query-dependent and query-independent query expansion; the latter is motivated by the wish to increase "aspect recall," and attempts to uncover aspects of the information need not captured by the query.

For evaluation purposes we use the CSIRO data set created for the TREC 2007 Enterprise track. The best performance is achieved by query models based on query-independent sampling of expansion terms from the example documents.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.4 [**Information Systems Applications**]: H.4.2 Types of Systems; H.4.m Miscellaneous

General Terms

Algorithms, Measurement, Performance, Experimentation

Keywords

Enterprise search, query modeling, query expansion, language models

1. INTRODUCTION

Query modeling has been a topic of active research for many years. One popular way of enriching the user's (usually sparse) query, and thus obtaining a more detailed specification of the underlying information need is through query expansion, by selecting terms from documents that are known, believed or assumed to be relevant. In the absence of explicit user feedback, the canonical approach is to treat the top-ranked documents retrieved in response to a query as if they had been marked relevant by the user.

Our work takes place in an enterprise setting, where users are more willing than, say, average web search engine users, to express their information need in a more elaborate form than by means of a few key words. In our scenario users have to create overview pages of the information available within the enterprise on a given topic, and the search engine should help them identify *key references*, pages on the intranet of the enterprise that should be linked to by a good overview page. The additional information that our users provide consists of a small number of example key references. We refer to those documents as *sample documents*.

An important research goal in this paper is to devise a way of using these rich specifications of the user's information need, consisting of a query and sample documents, in a theoretically transparent manner. We address this goal while working within the generative language modeling (LM) approach to retrieval. Here, one usually assumes that the relevance of a document is correlated with the likelihood of the query [9, 16, 17], builds a language model from each document, and ranks documents based on the probability of the document model generating the query. Feedback documents are assumed to be relevant, which often entails that the generation probabilities are (re-)estimated (using the feedback documents). The implicit nature of relevance within the LM framework has attracted some criticism; see, e.g., [23]. This criticism has been addressed in various proposals, including ones that consider not only document models, but also a language model based on the request, i.e., a query model [14], relevance models [15], and parsimonious language models [10].

In this paper we use sample documents to explicitly model relevance and an important goal for us is to develop methods for accurately estimating sampling probabilities. We assume that the query, sample documents and relevant documents are all coming from an unknown relevance model R. Lavrenko and Croft [15] used two methods to build a relevance model θ_R, where $P(t|\theta_R)$ is the relative frequency with which we expect to see term t during repeated independent random sampling of words from all of the relevant documents (see Section 6.1 below). Both approaches assume conditional dependence between the query and the terms t selected for expansion. While this dependence assumption may be appropriate in some cases (especially if the query is the only expression of the information need that we have), we want to be able to lift it. The reason for this is as follows. "Aspect recall" is an important cause of failure of current IR systems [5], one that tends to be exacerbated by today's query expansion approaches: key aspects of the user's information need may be completely missing from the pool of top-ranked documents, as this pool is usually query-biased

and (to keep precision reasonable) often small, and, hence, tends to only reflect aspects covered by the original query itself [12]. In a scenario such as ours, where a user provides a query plus sample documents, we expect the sample documents to provide important aspects not covered by the query. Hence, we want to avoid biasing the expansion term selection toward the query and thereby possibly loosing important aspects.

Our main contribution is a theoretically justified model for estimating a relevance model when training material (in the form of sample documents) is available, a model that is fully general in that we can sample expansion terms either independent of, or dependent on, the query. Our model has two main components, one for estimating (expansion) term importance, and one for estimating the importance of the documents from which expansion terms are selected—we consider various instantiations of these components, including ones where document importance estimations are done in a query independent manner, based on sample documents.

We use data provided by the TREC 2007 Enterprise track to evaluate our models. We compare them against standard blind relevance feedback approaches (where expansion terms are selected from a query-biased set of documents) and against relevance models based on the sample documents. Assuming independence between query and sampling terms leads to expanded queries that outperform a high performing baseline with no query expansion, as well systems that perform standard query expansion. Unbiased query expansion improves "aspect recall" by bringing in more "rare" relevant documents, that are not identified by the standard (query-biased) expansion methods that we consider.

The remainder of the paper is organized as follows. In Section 2 we discuss related work. In Section 3 we detail our retrieval approach and describe our take on query modeling. In Section 4 we describe our experimental setup and in Section 5 we establish a baseline, using the sample documents to maximize query log likelihood. Then, in Section 6 we detail several query models, which we evaluate in Section 7. We follow with an analysis in Section 8 and a conclusion in Section 9.

2. RELATED WORK

Query modeling, i.e., transformations of simple keyword queries into more detailed representations of the user's information need (e.g., by assigning (different) weights to terms, expanding the query, or using phrases), is often used to bridge the vocabulary gap between query and document collection. Many expansion techniques have been proposed, and they mostly fall into two categories, i.e., global and local. The idea of *global* analysis is to expand the query using global collection statistics based, for instance, on a co-occurrence analysis of the entire collection. Thesaurus- and dictionary-based expansion as, e.g., in [18], also provide examples of the global approach.

We focus on *local* approaches to query expansion, that use the top retrieved documents as examples from which to select terms to improve the retrieval performance [19]. In the setting of language modeling approaches to query expansion, the local analysis idea has been instantiated by estimating query language models [13, 24] or relevance models [15] from a set of feedback documents. Yan and Hauptmann [25] explore query expansion in the setting of multimedia retrieval. Our work goes beyond this work by dropping the assumption that query and expansion terms are dependent.

"Aspect recall" has been identified in [5, 8]. Kurland et al. [12] provide an iterative "pseudo-query" generation technique to uncover aspects of a query, using cluster-based language models.

At the TREC 2007 Enterprise track, several teams experimented with the use of sample documents for the document search task,

using a language modeling setting [4, 11, 21] or using ideas reminiscent of resource selection [6], or using the document structure in various ways [1, 7]. Some groups experiment with the use of sample documents, but the difference between the best performance with sample documents and the best performance without sample documents was modest [2].

3. RETRIEVAL MODEL

In this section we derive our ranking mechanism. We bring query-likelihood LM approaches and relevance models to a common ground, and show that both lead to the same scoring function, although the theoretical motivation behind them is different.

3.1 Query Likelihood

In case of the query likelihood (also referred as standard LM) approach, documents are ranked according to the likelihood of them being relevant given the query $P(D|Q)$. Instead of calculating this probability directly, we apply Bayes' rule and rewrite it to

$$P(D|Q) = \frac{P(Q|D) \cdot P(D)}{P(Q)}. \tag{1}$$

The probability of the query $P(Q)$ can be ignored for the purpose of ranking documents, which leaves us with

$$P(D|Q) \propto P(D) \cdot P(Q|D). \tag{2}$$

Assuming that query terms are independent from each other, we estimate $P(Q|D)$ by taking the product across terms in the query. Substituting this into Eq. 2 we obtain

$$P(D|Q) \propto P(D) \cdot \prod_{t \in Q} P(t|D)^{n(t,Q)}. \tag{3}$$

Here, $n(t, Q)$ is the number of times term t is present in the query Q. This is the multinomial view of the document model, i.e., the query Q is treated as a sequence of independent terms [9, 16, 22].

To prevent numerical underflows, we perform this computation in the log domain (thus compute the log-likelihood of the document being relevant to the query) and rewrite our equation as

$$\log P(D|Q) \propto \log P(D) + \sum_{t \in Q} n(t, Q) \cdot \log P(t|D). \tag{4}$$

Next, we generalize $n(t, Q)$ so that it can take not only integer but real values. This will allow more flexible weighting of query terms. We replace $n(t, Q)$ with $P(t|\theta_Q)$, which can be interpreted as the weight of term t in query Q. We will refer to θ_Q as *query model*. We generalize $P(t|D)$ to a *document model*, $P(t|\theta_D)$, and arrive at our final formula for ranking documents:

$$\log P(D|Q) \propto \log P(D) + \sum_{t \in Q} P(t|\theta_Q) \cdot \log P(t|\theta_D) \tag{5}$$

Two important components remain to be defined, the query model and the document model. Before doing so, we point out a relation between our ranking formula in Eq. 5 and relevance models.

For relevance language modeling one assumes that for every information need there exists an underlying relevance model R, and the query and documents are random samples from R, see Figure 1. We view documents and queries as samples from R, however, the two sampling processes do not have to be the same (i.e., $P(t|R)$ does not need to be the same as $P(t|Q)$ or $P(t|D)$, where D is a relevant document). The query model θ_Q is to be viewed as an approximation of R. Estimating $P(t|\theta_Q)$ in a typical retrieval setting is problematic because we have no training data. (Below, however, we will use the sample documents for this purpose, see Section 6.2.) Documents and queries are represented by a multinomial probability distribution over the vocabulary of terms. Documents are ranked based on their similarity to the query model.
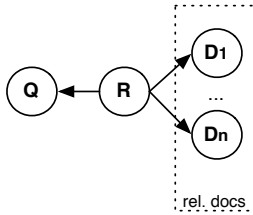
Figure 1: The query and relevant documents are random samples from an underlying relevance model R.

The Kullback-Leibler divergence between the query and document models can then be used to provide a ranking of documents:

$$D(\theta_Q || \theta_D) = -\sum_t P(t|\theta_Q) \cdot \log P(t|\theta_D) + cons(Q). \quad (6)$$

The document-independent constant $cons(Q)$ (the entropy of the query model) can be dropped, because it does not affect the ranking of documents; see [14, 26]. If we assume a uniform prior in Eq. 5, maximizing the query log-likelihood in Eq. 5 provides the same document ranking as minimizing the KL-divergence (Eq. 6).

3.2 Document Modeling

The move from $P(t|D)$ to the document model $P(t|\theta_D)$ in Eq. 4 and 5 is motivated by sparseness issues. To be able to rank documents using Eq. 4, we need to estimate $P(t|D)$, the probability that t would be observed during repeated random sampling from the document model. The maximum likelihood (ML) estimate of a term provides the simplest method for inferring an empirical document model: $P(t|D) = n(t, D)/\sum_{t'} n(t', D)$. If one or more query terms do not appear in the document, it will be assigned a zero probability (Eq. 3).

Nonetheless, creating a document model θ_D can resolve the zero probability problem, by smoothing the ML estimate such that for every term t, $P(t|\theta_D) > 0$. The document model is built up from a linear combination of the empirical estimate, $P(t|D)$, and the maximum likelihood estimate of the term, given the collection model $P(t|C)$, using the coefficient λ to control the influence of each:

$$P(t|\theta_D) = (1-\lambda) \cdot P(t|D) + \lambda \cdot P(t|C). \quad (7)$$

We discuss the problem of estimating the smoothing parameter λ— and exploit sample documents for this purpose—in Section 5.

3.3 Query Modeling

As to the query model, we consider several flavors. Our *baseline query model* consists of terms from the topic title only, and assigns the probability mass uniformly across these terms:

$$P(t|\theta_Q) = P(t|Q) = \frac{n(t, Q)}{\sum_{t'} n(t', Q)} \quad (8)$$

As before, $n(t, Q)$ is the frequency of term t in Q.

The baseline query model has two potential issues. Not all query terms are equally important, hence, we may want to reweigh some of the original query terms. Also, $P(t|Q)$ is extremely sparse, and, hence, we may want to add new terms (so that $P(t|\theta_Q)$ amounts to query expansion), and for this purpose we will again use the sample documents; see Sections 6.2 and 6.3.

Much of the paper is devoted to investigating ways of constructing the query model θ_Q that approximates the true relevance model R accurately. In [15] two methods are presented that estimate relevance models by constructing topic models from the topic title only without training data; in this paper, we examine theoretically justi-

fied ways of estimating the relevance model when training data (in the form of sample documents) is available.

4. EXPERIMENTAL SETUP

We addressed the following research questions: Can sample documents be used to estimate the amount of smoothing applied? How does using sample documents compare to blind relevance feedback? Expansion terms in the case of standard blind relevance feedback are dependent on the original query. How does lifting this assumption affect retrieval performance? To address our research questions we ran experiments using the CSIRO Enterprise Research Collection (CERC), a crawl of *.csiro.au (public) web sites conducted in March 2007. The crawl has 370,715 documents, with a total size 4.2 gigabytes [3].

In the 2007 edition of the TREC Enterprise track, CERC was used as the document collection [2]. CSIRO's science communicators played an important role in topic creation. They, the envisaged end-users of systems taking part in the TREC Enterprise track, read and create outward-facing web pages of CSIRO to enhance the organization's public image and promote its expertise. A total of 50 topics were created by the science communicators; systems had to return "key references" for these topics, i.e., pages that should be linked to by a good overview page.

Assessment was done by the TREC 2007 Enterprise track participants. Judgments were made on a three-point scale: 2: highly likely to be a "key reference;" 1: a candidate key page, or otherwise informative to help build an overview page, but not highly likely; 0: not a "key reference," because, e.g., not relevant, off-topic, not an important page on the topic, on-topic but out-of-date, not the right kind of navigation point, or too informal or too narrow an audience. All non-judged documents are considered as irrelevant. For our experiments we used the official qrels released after TREC 2007, consisting of 50 topics, but with the sample documents removed from the runs and from the set of relevant documents.

We scored our retrieval output both using the possibly relevant and the highly relevant levels, using mean average precision and mean reciprocal rank.

5. ESTABLISHING A BASELINE

5.1 Parameter Estimation

In order to establish a baseline, we need a reasonable estimate of λ (in Eq 7) for those documents that are likely to be relevant to a given query, since they are the ones we are interested in.

When training data (in the form of topics and corresponding relevance judgments) is available, we can estimate λ empirically on a set of training topics, and then apply this value on the test set. This way the same amount of smoothing is applied for all queries. When such a set of training topics is not available, one approach is to use automatic relevance feedback [20]. We perform an initial guess for λ (e.g., $\lambda = 0.5$) and assume that the top M retrieved documents are relevant to the query. These M top-scoring documents then become the set we use to estimate λ.

Given our setting (with sample documents available), we will view these as documents relevant to a given query, and use them for learning settings for the λ parameter. However, instead of estimating a uniform λ, we estimate a query-dependent λ_Q. Below, we present two unsupervised methods that can accurately estimate this value, and deliver the same performance as the empirical estimate.[1]

---

[1]Recall that sample documents (which we use for estimation) are removed from the runs and from the relevance judgments; in particular, they are not part of the test data that we use in the estimation

Method	(possibly) relevant						(highly) relevant					
	MAP	bpref	P@5	P@10	P@20	MRR	MAP	bpref	P@5	P@10	P@20	MRR
EMP_BEST ($\lambda = 0.6$)	.3599	.3856	.6320	.6080	.5660	.7200	.3150	.3964	.4920	.4380	.3800	.6361
Using example documents												
MAX_AP	.3517	.3812	.6040	.5840	.5470	.7017	.3092	.3901	.4600	.4120	.3660	.6131
MAX_QLL	**.3576**	**.3853**	**.6120**	**.6000**	**.5610**	**.7134**	**.3143**	**.4013**	**.4880**	**.4360**	**.3770**	**.6326**

Table 1: Comparison of the two parameter (λ) estimation methods (MAX_AP, MAX_QLL) and the empirical estimate (EMP_BEST). All results are evaluated against the (TREC) relevance judgments. The row in boldface row will serve as a baseline.

Maximizing Average Precision.

In our first technique for estimating λ_Q (called MAX_AP) we view the sample documents as if they were the only relevant documents given the query. The process for each query Q is as follows: 1. For each $\lambda_Q \in (0, 1)$ (with steps δ); 2. Run retrieval using the parameter λ_Q; 3. Calculate the average precision (AP) of the sample documents; 4. Select λ_Q that maximizes AP. Formally:

$$\lambda_Q = \arg \max_\lambda AP(\lambda, Q, S). \qquad (9)$$

Maximizing Query Log Likelihood.

Our second technique sets for estimating λ_Q (called MAX_QLL) sets λ_Q to the value that maximizes the log-likelihood of the query Q, given a set of sample documents S:

$$\lambda_Q = \arg \max_\lambda \sum_{D \in S} \sum_{t \in Q} \log((1 - \lambda) \cdot P(t|D) + \lambda \cdot P(t|C)) \quad (10)$$

5.2 Evaluation

In order to evaluate the two approximation methods presented above, we first perform an empirical exploration of a query-independent smoothing parameter λ. That is, we iterate over possible λ values in steps of $\delta = 0.01$ and calculate the mean average precision (MAP) on the entire set of topics:

$$\lambda = \arg \max_\lambda \frac{\sum_Q AP(\lambda, Q)}{|Q|} \qquad (11)$$

We refer to this value as the best empirical estimate (EMP_BEST). Figure 2 displays the results, using both possibly and highly relevant assessments. There is a broad range of settings where performance levels close to the maximum are achieved; the maximum AP scores are reached around $\lambda = 0.6$, with a substantial drop in performance for $\lambda \geq 0.8$.

Next, we use $\lambda = 0.6$ and compare our approximation methods against this baseline; see Table 1. Our estimation methods for λ_Q are effective in estimating λ. MAX_QLL performed slightly better than MAX_AP, but the differences are not significant.[2]

5.3 Wrap-up

We have fixed our baseline retrieval approach. We set the smoothing parameter using an estimation method that exploits sample documents (MAP_QLL). Although this method uses only a handful of sample documents per query (3.6 on average), the performance is as good as that of the empirical best; moreover, it can be computed more efficiently. For the remainder of the paper, this (with smoothing determined using MAP_QLL) serves as our baseline.

6. REPRESENTING THE QUERY

We now consider different ways of representing the query. For comparison purposes, we first consider standard blind relevance

---

[2]We use the two-tailed paired T-test for significance testing. We consider differences with $p < 0.01$ significant.
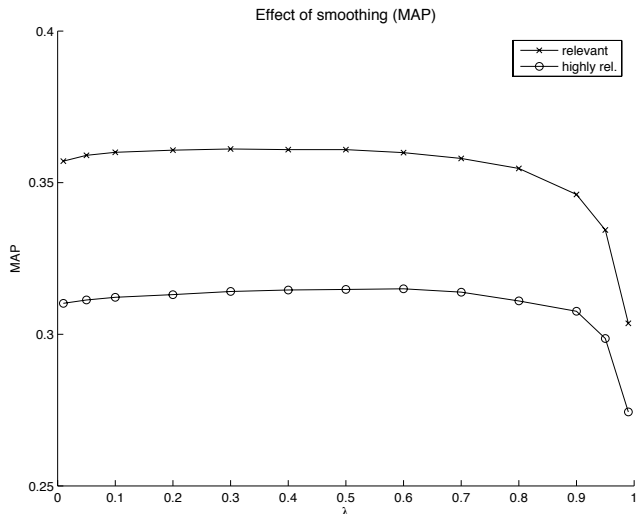

Figure 2: Effect of smoothing; MAP plotted against the weight (λ) of the collection model; results on two relevance levels.

feedback using relevance models as defined in [15]. Next, we use the same methods but instead of selecting expansion terms from the top ranked documents in an initial retrieval run, we select them from the sample documents. These expansion methods both assume that expansion terms are dependent on the query; after that, we provide a model according to which we can sample terms from the sample documents both independent of and dependent on the original query. The output of these methods is an *expanded query model* $\hat{Q}$.

Next, we combine the selected query terms with the terms from the original query; this is also done in the original query expansion papers (see, e.g., [19]) and in query modeling methods based on language models (see, e.g., [12]) and prevents the topic to shift (too far) away from the original user information need. We use Eq. 12 to mix the original query with the expanded query.

$$P(t|\theta_Q) = (1 - \mu) \cdot P(t|\hat{Q}) + \mu \cdot P(t|Q), \qquad (12)$$

where $P(t|Q)$ and $P(t|\hat{Q})$ are the probability of term t given the original query Q (see Eq. 8) and the expanded query $\hat{Q}$, respectively. The expanded query models $\hat{Q}$ are evaluated in Section 7.1, and their combinations with the original query (by performing an empirical exploration of μ), are presented in Section 7.2.

6.1 Feedback Using Relevance Models

One way of expanding the original query is by using blind relevance feedback: assume the top M documents to be relevant given a query. From these documents we sample terms that are used to form the expanded query model $\hat{Q}$. Lavrenko and Croft [15] suggest a reasonable way of obtaining $\hat{Q}$, by assuming that $P(t|\hat{Q})$ can be approximated by the probability of term t given the (original)

query Q. We can then estimate $P(t|\hat{Q})$ using the joint probability of observing t together with the query terms $q_1, \ldots, q_k \in Q$, and dividing by the joint probability of the query terms:

$$P(t|\hat{Q}) \approx \frac{P(t, q_1, \ldots, q_k)}{P(q_1, \ldots, q_k)} \qquad (13)$$

$$= \frac{P(t, q_1, \ldots, q_k)}{\sum_{t'} P(t', q_1, \ldots, q_k)}, \qquad (14)$$

In order to estimate the joint probability $P(t, q_1, \ldots, q_k)$, Lavrenko and Croft [15] propose two methods; they differ in the independence assumptions that are being made:

RM1 It is assumed that t and q_i are sampled independently and identically to each other; therefore, their joint probability can be expressed as the product of the marginals:

$$P(t, q_1 \ldots q_k) = \sum_{D \in M} P(D) \cdot P(t|D) \cdot \prod_{i=1}^{k} P(q_i|D), \quad (15)$$

where M is the set of feedback documents.

RM2 The second method tackles a different sampling strategy, and we assume that query words $q_1, \ldots, q_k$ are independent of each other, but we keep their dependence on t:

$$P(t, q_1 \ldots q_k) = P(t) \cdot \prod_{i=1}^{k} \sum_{D \in M} P(D|t) \cdot P(q_i|D). \quad (16)$$

That is, the value $P(t)$ is fixed according to some prior, then the following process is performed k times: a document $D \in M$ is selected with probability $P(D|t)$, then the query word q_i is sampled from D with probability $P(q_i|D)$.

RM1 can be viewed as sampling of all query terms conditioned on t: a strong mutual independence assumption, compared to the pairwise independence assumptions made by RM2. Empirical evaluations reported in [15] found that RM2 is more robust, and performs slightly better that RM1. Our experiments below confirm this.

6.2 Relevance Models from Sample Documents

Next, we follow the approach of the previous section and apply relevance models to the sample documents. Instead of performing an initial retrieval run to obtain a set of feedback documents, we use the sample documents and observe the co-occurrence of term t with query terms $q_1, \ldots, q_k$ in the sample documents. I.e., we set $M = S$. For RM1, we also need to make an extra assumption, viz. that all sample documents are equally important: $P(D) = 1/|S|$.

6.3 A Query Model from Sample Documents

Now we introduce a new model based on sampling from documents that are assumed to be relevant. Unlike with the methods considered above, the sampling can be done both independent of, and dependent on, the original query. Our approach to constructing the expanded query $\hat{Q}$ is the following. First, we estimate a "sampling distribution" $P(t|S)$ using sample documents $D \in S$. Next, the top K terms with highest probability $P(t|S)$ are taken and used to formulate the expanded query $\hat{Q}$:

$$P(t|\hat{Q}) = \sum_{t \in K} \frac{P(t|S)}{\sum_{t'} P(t'|S)}. \qquad (17)$$

Calculating the sampling distribution $P(t|S)$ can be viewed as the following generative process: 1. Let the set of sample documents S be given; 2. Select a document D from this set S with probability $P(D|S)$; and 3. From this document, generate the term t with probability $P(t|D)$. By summing over all sample documents, we obtain $P(t|S)$. Formally, this can be expressed as

$$P(t|S) = \sum_{D \in S} P(t|D) \cdot P(D|S) \qquad (18)$$

For estimating the term importance, $P(t|D)$, we consider three natural options:

- Maximum likelihood estimate of a term (EX-QM-ML)

$$P(t|D) = P_{ML}(t|D) = \frac{n(t, D)}{\sum_{t'} n(t', D)} \qquad (19)$$

- Smoothed estimate of a term (EX-QM-SM)

$$P(t|D) = P(t|\theta_D) = (1 - \lambda) \cdot P_{ML}(t|D) + \lambda \cdot P_{ML}(t|C) \qquad (20)$$

- Use the ranking function proposed by Ponte and Croft [17] for unsupervised query expansion (EX-QM-EXP)

$$s(t) = \log \frac{P_{ML}(t|D)}{P_{ML}(t|C)} \qquad (21)$$

and set $P(t|D) = s(t)/\sum_{t'} s(t')$.

The probability $P(D|S)$ expresses the importance of sample document D given the samples S. I.e., this is a weight that determines how much a term $t \in D$ will contribute to the sampling distribution $P(t|S)$. We consider three options for estimating $P(D|S)$:

- Uniform: $P(D|S) = 1/|S|$, all sample documents are assumed to be equally important. We assume conditional independence between the original query terms $q \in Q$ and the "expanded term" t. This can safely be done, since the original query terms are preserved in $P(t|\theta_Q)$ because of the smoothing (see Eq. 12).

- Query-biased: $P(D|S) \propto P(D|Q)$. A document's importance is approximated by its relevance to the original query.

- Inverse query-biased: $P(D|S) \propto 1 - P(D|Q)$. We reward documents that bring in aspects different from the query.

7. EXPERIMENTAL EVALUATION

7.1 Expanded Query Models

We start by evaluating the relevance models using blind feedback (Section 6.1). We explore the number of feedback documents that need to be taken into account (note that the number of terms extracted is $K = 10$). In Figure 3 (Left) the performance of query expansion using BFB-RM1 and BFB-RM2 on different numbers of feedback documents ($|M|$) is shown. A smaller number of feedback documents gives better performance on MAP for both models; best performance is achieved with only 5 feedback documents.

Next, we construct relevance models on the sample documents using relevance models (Section 6.2). EX-RM2 fails on two topics (1 and 11), while topic 45 does not have any sample documents. The influence of the number of selected terms K on retrieval performance for EX-RM1 and EX-RM2 is displayed in Figure 3 (Center). The best performance is achieved when selecting 15 terms for EX-RM1 and 25 for EX-RM2.

Finally, we explore the number of selected terms K for our query models generated from sample documents (Section 6.3). Results are displayed in Figure 3 (Right).

Table 2 records our baseline performance (which is similar to the median achieved at TREC 2007) and summarizes the results for
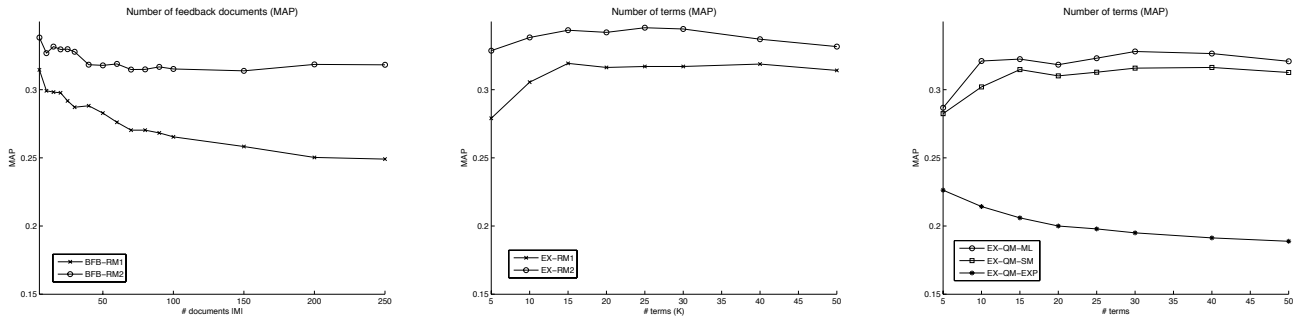
Figure 3: (Left) BFB-RM, MAP against the number of feedback documents used for query models construction. (Center) and (Right) MAP against the number of terms selected for query models construction; (Center): EX-RM, (Right) EX-QM.

model	K	(possibly) relevant		(highly) relevant	
		MAP	MRR	MAP	MRR
baseline		.3576	.7134	.3143	.6326
BFB-RM1	10	.3145	.6326	.2679	.5335
BFB-RM2	10	.3382	.6683	.2845	.5609
EX-RM1	15	.3193	.8794	.2813	.7695
EX-RM2	25	.3454	.8596	.3111	.8169
EX-QM-ML	30	.3280	.8508	.2789	.7093
EX-QM-SM	40	.3163	.8050	.2822	.7133
EX-QM-EXP	5	.2263	.6131	.2062	.5854

Table 2: Performance of the expanded query model $\hat{Q}$.

the expanded query model $\hat{Q}$, together with the number K of feedback terms used. The query models based on query-dependent sampling of expansion terms (BFB and EX) perform closer to the baseline than those based on query-independent sampling (in terms of MAP). EX-QM-ML and EX-QM-SM are able to add more terms without hurting performance than EX-RM1 and EX-RM2, thereby allowing more aspects to be retrieved.

7.2 Combination with the Original Query

Next, we combine the expanded query $\hat{Q}$ and the original query Q, where the parameter μ controls the weight of the original query (see Eq. 12). We perform a sweep on μ to determine the optimal mixture weight of the original query. The results are in Figure 4.

model	μ	(possibly) relevant		(highly) relevant	
		MAP	MRR	MAP	MRR
baseline		.3576	.7134	.3143	.6326
BFB-RM1	0.6	.3677	.6703	.3171	.5772
BFB-RM2	0.6	.3797	.6905	.3296	.6033
EX-RM1	0.4	.4264*	.8808*	.3758*	.8259*
EX-RM2	0.4	.4273*	.9029*	.3833*	.8473*
EX-QM-ML	0.5	.4449*	.8533*	.3951*	.7911*
EX-QM-SM	0.5	.4406*	.8771*	.3955*	.8035*
EX-QM-EXP	0.7	.4016*	.8148	.3520	.7603*

Table 3: Performance of the baseline run, relevance models on blind feedback documents and sample documents, and query models on sample documents using optimal K and λ settings for each model. Results marked with * are significantly different from the baseline.

The best results together with the optimal μ values are listed in Table 3. Here we see two of the query models based on query-independent sampling outperforming all other query models (in

terms of (possibly) relevant MAP), although the differences between the best relevance model (EX-RM2) and our best query model (EX-QM-ML) are not significant.

7.3 The Importance of a Sample Document

Finally, we evaluate the three options we considered for estimating the importance of a sample document ($P(D|S)$); see Section 6.3. Table 4 lists the results. Non-uniform document importance settings tend to hurt MAP performance, for two of the three flavors of term importance estimations (ML, SM); the query-biased setting has an early precision enhancing effect, boosting MRR scores for all term importance estimations methods.[3]

$P(D|S)$	(possibly) relevant		(highly) relevant	
	MAP	MRR	MAP	MRR
EX-QM-ML				
Uniform	.4449	.8533	.3951	.7911
$P(D|Q)$	.4294	.8810	.3871	.8399
$1-P(D|Q)$	.4184	.8268	.3681	.7376
EX-QM-SM				
Uniform	.4406	.8771	.3955	.8035
$P(D|Q)$	.4189	.8950	.3831	.8533
$1-P(D|Q)$	.4264	.8248	.3755	.7375
EX-QM-EXP				
Uniform	.4016	.8148	.3520	.7603
$P(D|Q)$	.4026	.8383	.3544	.7803
$1-P(D|Q)$	.3988	.7928	.3503	.7411

Table 4: Importance of a sample document.

8. ANALYSIS/DISCUSSION

8.1 Topic-level comparison

So far, we have looked at results at an aggregate level. Next, we continue the comparison by looking at the topic-level performance. Figure 5 presents the difference in average precision of the best performing query generation methods (BFB-RM2, EX-RM2, and EX-QM-ML) against the baseline. Most topics gain from the query models, although there are always some topics that are hurt. Clearly, EX-RM2 and EX-QM-ML have bigger gains than BFB-RM2; possibly relevant and highly relevant assessments yield similar patterns.

Next, we zoom in on two example topics, where these methods display interesting behavior. The first example concerns the topic

---

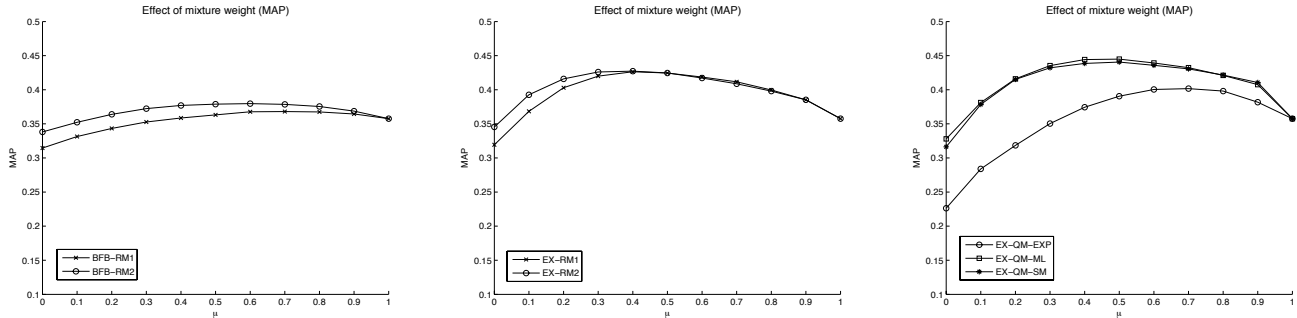[3]Only the difference between the $P(D|Q)$ and $1-P(D|Q)$ versions of EX-QM-SM is significant.

Figure 4: MAP is plotted against the weight (μ) of the original query. (Left): BFB-RM. (Center): EX-RM. (Right): EX-QM.
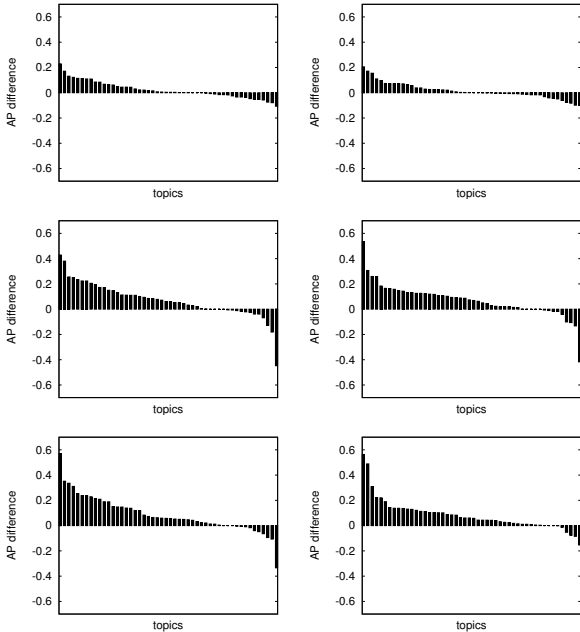

Figure 5: AP differences between baseline and (Top): BFB-RM2, (Middle): EX-RM2, (Bottom): EX-QM-ML, on (Left): possibly, and (Right): highly relevant.

machine vision; Table 5 reports the MAP scores, and Table 6 displays the top 10 terms for the query models constructed for the topic *machine vision*, with EX-QM-ML and EX-RM2 performing much better than BFB-RM2. EX-QM-ML is mostly on target (with a shift to surveillance and security), while the other two models display a shift to a far broader topical area.

relevance	BFB-RM2	EX-RM2	EX-QM-ML
possibly	.0722	.1283	.2848
highly	.0696	.1552	.3062

Table 5: Performance on topic # 32.

The next example, *termites*, shows a different behavior, with BFB-RM2 beating EX-QM-ML, which in turn beats EX-RM2. Table 7 reports the MAP scores, and Table 8 displays the top 10 terms for query models constructed for this topic. We see topic drift for EX-RM2 and EX-QM-ML, but many on target terms for BFB-RM2.

8.2 Sampling Conditioned on the Query

Interestingly, when we compare two document importance estimation methods (query-biased and inverse query-biased) and two

$P(t|\theta_Q)$	t	$P(t|\theta_Q)$	t	$P(t|\theta_Q)$	t
0.4123	vision	0.2707	vision	0.2796	vision
0.3935	machine	0.2641	machine	0.2762	machine
0.0336	csiro	0.0735	csiro	0.0513	csiro
0.0303	image	0.0267	projects	0.0248	image
0.0302	toolbox	0.0256	high	0.0224	vehicles
0.0227	robot	0.0245	research	0.0220	safe
0.0221	information	0.0239	systems	0.0214	cam
0.0204	control	0.0223	development	0.0178	traffic
0.0202	visual	0.0204	computing	0.0176	technology
0.0147	object	0.0191	performance	0.0173	camera

Table 6: Query models for topic # 32 "machine vision". (Left) BFB-RM2; (Center) EX-RM2; (Right) EX-QM-ML.

relevance	BFB-RM2	EX-RM2	EX-QM-ML
possibly	0.7971	0.1205	0.2342
highly	0.8107	0.1886	0.4520

Table 7: Performance on topic # 36.

term selection methods (EX-QM-SM and EX-QM-ML), we see a mostly balanced picture; see Figure 6. For some topics the query-biased document importance works best (promoting aspects covered by the query), while for others inverse query-biased works best (promoting aspects not covered by the query that comes with the topic). On average, though, the query-independent sampling delivers the best performance; see Table 4.
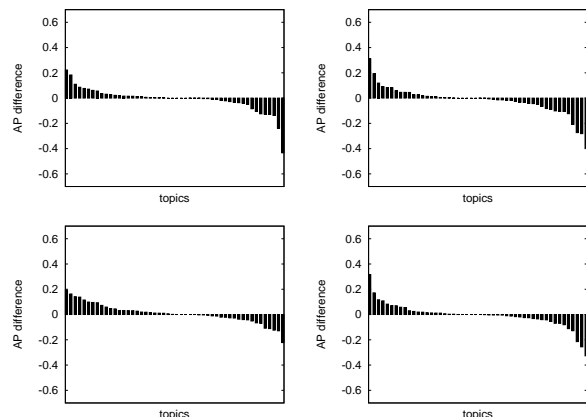

Figure 6: AP differences between query-biased ("baseline") and inverse query-biased document sampling methods. (Top): EX-QM-ML, (Bottom): EX-QM-SM, on (Left): possibly, and (Right): highly relevant.

Let us return to the issue of aspect recall. We have seen that using query models leads to better ranking of documents. Looking

$P(t\|\theta_Q)$	t	$P(t\|\theta_Q)$	t	$P(t\|\theta_Q)$	t
0.7405	termites	0.4729	termites	0.5653	termites
0.0401	csiro	0.0452	site	0.0299	site
0.0388	wood	0.0443	information	0.0292	information
0.0316	food	0.0412	legal	0.0281	legal
0.0314	termite	0.0410	notice	0.0281	notice
0.0258	vibrations	0.0404	disclaimer	0.0271	disclaimer
0.0242	blocks	0.0402	privacy	0.0271	privacy
0.0231	species	0.0381	web	0.0252	drywood
0.0228	australian	0.0378	subject	0.0243	statement
0.0217	made	0.0378	drywood	0.0173	subject

Table 8: Query models for topic # 36 "termites". (Left) BFB-RM2; (Center) EX-RM2; (Right) EX-QM-ML.

at the individual documents returned by each model, we find that using blind relevance feedback, recall either decreases (BFB-RM1; over all queries, BFB-RM1 retrieves 2,564 highly relevant documents vs. 2,763 for the baseline; see Table 9) or only marginally increases (BFB-RM2; 2,816 vs. 2,763). On the other hand, expanding the query based on the example documents can help to capture on average 10% more relevant documents than the baseline, on both relevance levels; see Table 9. Importantly, there is a number of doc-

relevance	baseline	BFB-		EX-		EX-QM-		
		RM1	RM2	RM1	RM2	ML	SM	EXP
possibly	5,445	5,238	5,582	5,951	5,882	6,052	5,953	5,671
highly	2,763	2,564	2,816	2,954	2,929	3,047	3,019	2,823

Table 9: Number of relevant documents retrieved.

uments that are found only when sampling is done independent of the query (EX-QM-*). Consider topic #32 (*machine vision*) again. First, the number of relevant documents found for this topic are the following: baseline: 53, BFB-RM2: 54, EX-RM2: 54, and EX-QM-ML: 62. The additional documents are identified through the new terms introduced by our query models, as is clearly illustrated in Table 6: the terms *cam* and *camera* are captured only by EX-QM-ML. In sum, then, our sampling method from sample documents does indeed pick up different aspects of the topic, and as such, helps improve "aspect recall."

9. CONCLUSIONS

We introduced a method for sampling query expansion terms in a query-independent way, based on sample documents that reflect aspects of the user's information need that are not captured by the query. We described various versions of our expansion term selection method, based on different term selection and document importance weighting methods, and compared them against more traditional query expansion methods that select expansion terms in a query-biased manner.

Evaluating our methods on the TREC 2007 Enterprise track test set, we found that our expansion method outperforms a high performing baseline as well as standard language modeling based query expansion methods. Our analysis revealed that our query-independent expansion method does help to address the "aspect recall" problem, and helped to identify relevant documents that are not identified by the other query models that we considered.

As to future work, we see a number of other ways of exploiting sample documents provided for a topic. One possibility is to look at other features of these example documents, including layout, link structure, document structure, etc. and favor documents in the ranking that share the same characteristics. Another possibility is to combine terms extracted from blind feedback documents, together with terms from sample documents.

11. REFERENCES

[1] P. Bailey, D. Agrawal, and A. Kumar. TREC 2007 Enterprise Track at CSIRO. In *TREC 2007 Working Notes*, 2007.

[2] P. Bailey, N. Craswell, A. P. De Vries, and I. Soboroff. Overview of the TREC 2007 Enterprise Track. In *TREC 2007 Working Notes*, 2007.

[3] P. Bailey, N. Craswell, N. Soboroff, and A. de Vries. The CSIRO enterprise search test collection. *ACM SIGIR Forum*, 41, 2007.

[4] K. Balog, K. Hofmann, W. Weerkamp, and M. de Rijke. The University of Amsterdam at the TREC 2007 Enterprise Track. In *TREC 2007 Working Notes*, 2007.

[5] C. Buckley. Why current IR engines fail. In *SIGIR '04*, pages 584–585, 2004.

[6] Y. Fu, Y. Xue, T. Zhu, Y. Liu, M. Zhang, and S. Ma. THUIR at TREC 2007: Enterprise Track. In *TREC 2007 Working Notes*, 2007.

[7] D. Hannah, C. Macdonald, J. Peng, B. He, and I. Ounis. University of Glasgow at TREC 2007: Experiments in Blog and Enterprise Tracks with Terrier. In *TREC 2007 Working Notes*, 2007.

[8] D. Harman and C. Buckley. The NRRC reliable information access (RIA) workshop. In *SIGIR '04*, pages 528–529, 2004.

[9] D. Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, University of Twente, 2001.

[10] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *SIGIR '04*, pages 178–185, 2004.

[11] H. Joshi, S. D. Sudarsan, S. Duttachowdhury, C. Zhang, and S. Ramasway. UALR at TREC-ENT 2007. In *TREC 2007 Working Notes*, 2007.

[12] O. Kurland, L. Lee, and C. Domshlak. Better than the real thing? In *SIGIR '05*, pages 19–26, 2005.

[13] J. Lafferty and C. Zhai. Probabilistic relevance models based on document and query generation. In *Language Modeling for Information Retrieval*. Springer, 2003.

[14] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *SIGIR '01*, pages 111–119, 2001.

[15] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01*, pages 120–127, 2001.

[16] D. Miller, T. Leek, and R. Schwartz. A hidden Markov model information retrieval system. In *SIGIR '99*, pages 214–221, 1999.

[17] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98*, pages 275–281, 1998.

[18] Y. Qiu and H.-P. Frei. Concept based query expansion. In *SIGIR '93*, pages 160–169, 1993.

[19] J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice Hall, 1971.

[20] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.

[21] H. Shen, G. Chen, H. Chen, Y. Liu, and X. Cheng. Research on Enterprise Track of TREC 2007. In *TREC 2007 Working Notes*, 2007.

[22] F. Song and W. B. Croft. A general language model for information retrieval. In *CIKM '99*, pages 316–321, 1999.

[23] K. Sparck Jones, S. E. Robertson, D. Hiemstra, and H. Zaragoza. Language modelling and relevance. In W. B. Croft and J. Lafferty, editors, *Language Modeling for Information Retrieval*, pages 57–71. 2003.

[24] T. Tao and C. Zhai. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR '06*, pages 162–169, 2006.

[25] R. Yan and A. Hauptmann. Query expansion using probabilistic local feedback with application to multimedia retrieval. In *CIKM '07*, pages 361–370, 2007.

[26] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01*, pages 403–410. ACM, 2001.