Exploiting External Collections for Query Expansion

WOUTER WEERKAMP, University of Amsterdam
KRISZTIAN BALOG, NTNU Trondheim
MAARTEN DE RIJKE, University of Amsterdam

A persisting challenge in the field of information retrieval is the vocabulary mismatch between a user's information need and the relevant documents. One way of addressing this issue is to apply query modeling: to add terms to the original query and reweigh the terms. In social media, where documents usually contain creative and noisy language (e.g., spelling and grammatical errors), query modeling proves difficult. To address this, attempts to use external sources for query modeling have been made and seem to be successful. In this article we propose a general generative query expansion model that uses external document collections for term generation: the External Expansion Model (EEM). The main rationale behind our model is our hypothesis that each query requires its own mixture of external collections for expansion and that an expansion model should account for this. For some queries we expect, for example, a news collection to be most beneficial, while for other queries we could benefit more by selecting terms from a general encyclopedia. EEM allows for query-dependent weighing of the external collections.

We put our model to the test on the task of blog post retrieval and we use four external collections in our experiments: (i) a news collection, (ii) a Web collection, (iii) Wikipedia, and (iv) a blog post collection. Experiments show that EEM outperforms query expansion on the individual collections, as well as the Mixture of Relevance Models that was previously proposed by Diaz and Metzler [2006]. Extensive analysis of the results shows that our naive approach to estimating query-dependent collection importance works reasonably well and that, when we use "oracle" settings, we see the full potential of our model. We also find that the query-dependent collection importance has more impact on retrieval performance than the independent collection importance (i.e., a collection prior).

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

General Terms: Experimentation, Performance, Theory

Additional Key Words and Phrases: Query modeling, external expansion, blog post retrieval

This research was supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430 (GALATEAS), the 7th Framework Program of the European Commission, grant agreement no. 258191 (PROMISE), the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments under project nr STE-09-12, The Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.814, 623.061.815, 640.004.802, 380-70-011, 727.011.005, 612.001.116, the Center for Creation, Content and Technology (CCCT), the Hyperlocal Service Platform project funded by the Service Innovation & ICT program, the WAHSP and BILAND projects funded by the CLARIN-nl program, the Dutch national program COMMIT, and by the ESF Research Network Program ELIAS.
Authors' addresses: W. Weerkamp (corresponding author), ISLA, University of Amsterdam; email: w.weerkamp@uva.nl; K. Balog, Department of Computer and Information Science, NTNU, Trondheim; M. de Rijke, ISLA, University of Amsterdam.

DOI 10.1145/2382616.2382621 http://doi.acm.org/10.1145/2382616.2382621

ACM Reference Format:
Weerkamp, W., Balog, K., and de Rijke, M. 2012. Exploiting external collections for query expansion. ACM Trans. Web 6, 4, Article 18 (November 2012), 29 pages.
DOI = 10.1145/2382616.2382621 http://doi.acm.org/10.1145/2382616.2382621

1. INTRODUCTION

Searching for information has become one of the main online activities. In 2012, Pew-Internet[1] reported that 91% of online adults use search engines to find information on the Web and 54% do so once a day or more, indicating the importance of search engines in our daily live activities. The field of information retrieval, as search is more formally known, focuses on developing (models for) systems that inform users on the existence (or nonexistence) and whereabouts of documents relating to the user's request [Lancaster 1968]. One of the grand challenges in information retrieval is to bridge the vocabulary gap between a user and her information need on the one hand and the relevant documents on the other [Baeza-Yates and Ribeiro-Neto 2011]. To clarify this point, consider the two information needs and the request, or query, to which they are translated in Table I.

We find that, besides the vocabulary gap, by simplifying the information need to a short keyword query, much information about which documents are to be considered relevant is lost. In case of the first information need, relevant documents could focus on topics that were addressed in the speech (e.g., economics, homeland security) or could mainly be about the person addressing the nation (e.g., speaking style, clothing). The keyword query, however, fails to address these specific topic aspects. Something similar happens for the second information need. Here, relevant documents should be about Shimano products, but these are very diverse, ranging from fishing to cycling equipment, each having a very different vocabulary.

In information retrieval we often apply *query expansion* as a technique to bridge the vocabulary gap between the query and relevant documents. Query expansion is the modification of the original query by adding and reweighing terms. In case of the first example from Table I, we could add terms like "bush," "president," or "terrorism" to the query, while assigning the highest weight to the original query. For the second example, we could add terms like "products," "fishing," and "cycling."

In general, query expansion helps more queries than it hurts [Balog et al. 2008; Manning et al. 2008], leading to better overall results. Several attempts have been made to decide on a per-query basis whether or not to use query expansion [Cronen-Townsend et al. 2004; He and Ounis 2007], thereby reducing the number of queries that are negatively affected by query expansion. One common issue with query expansion is topic drift, the introduction of new query terms that lead the expanded query away from the original information need. In the case of our *state of the union* example, we could expand the query with "film," "capra," and "thorndyke," causing the query to drift away from the 2006 State of the Union by President Bush towards the 1948 film from Frank Capra about Kay Thorndyke.

1.1. Information Retrieval in Social Media

In this article we focus on a particular type of content, namely user-generated content. With the rise in popularity of social media, like blogs, microblogs, and forums, the

[1]http://pewinternet.org/Reports/2012/Search-Engine-Use-2012

Table I. Two Examples of Information Need and Query

Information need	Query
Find documents on President Bush's 2006 State of the Union address.	state of the union
Provide documents on equipment using the brand name Shimano.	shimano

amount of information stored in these platforms' (user-generated) content has grown rapidly. Information retrieval in social media has become an important research area [Weerkamp 2011], which was boosted by the introduction of the Blog track at the Text REtrieval Conference (TREC) in 2006 [Ounis et al. 2007]. One of the tasks in this track was blog post retrieval, that is, finding relevant blog posts for a given topic (query). It is this task that we focus on in the remainder of this article.

In the setting of blogs or other types of social media, bridging the vocabulary gap between information need and relevant documents becomes even more challenging than usual. This has two main causes: (i) the spelling errors, unusual, creative, or unfocused language usage resulting from the lack of top-down writing rules and editors in the content creation process, and (ii) the (often) limited length of documents generated by users. Query expansion should therefore be beneficial in the setting of social media.

When working with user-generated content, expanding a query with terms taken from the very corpus in which one is searching (in our case, a collection of blog posts) tends to be less effective [Arguello et al. 2008; Jijkoun et al. 2010]; besides topic drift being an obvious problem, the text quality and creative language cause expansion terms to be less informative than necessary for successful query expansion. To counter both these issues and to be able to arrive at a richer representation of the user's information need, various authors have proposed to expand the query against an external corpus, that is, a corpus different from the target (user-generated) corpus from which documents need to be retrieved.

Our aim in this work is to define and evaluate a generative model for expanding queries using external collections. We propose a retrieval framework in which dependencies between queries, documents, and expansion collections are explicitly modeled. One of the reasons behind proposing our framework is that the "ideal" external collection to extract new query terms from is dependent on the query. Mishne and de Rijke [2006] examined queries submitted to a blog search engine and found many to be either news-related *context* queries (that aim to track mentions of a named entity) or *concept* queries (that seek posts about a general topic). For context queries such as *cheney hunting* (TREC topic 867) a news collection is likely to offer various (relevant) aspects of the topic, whereas for a concept query such as *jihad* (TREC topic 878) a knowledge source (e.g., Wikipedia) seems an appropriate source of terms that capture aspects of the topic.

We seek to answer the following question in this article: Can we define a generative model for query expansion using external collections? In answering this question, we also seek to answer the following questions.

(1) How does our model relate to the Mixture of Relevance Models originally proposed by Diaz and Metzler [2006]?
(2) Can we effectively apply external expansion in the retrieval of (user-generated) blog posts?
(3) Does conditioning the external collection on the query help improve retrieval performance?
(4) Which of the external collections is most beneficial for query expansion in blog post retrieval?

(5) Does our model show similar behavior across topics or do we observe strong per-topic differences?

In Section 2 we review previous research in the area of query expansion and the use of external collections. The most important sections are Sections 3 and 4, in which we introduce our retrieval framework and query modeling approach. Section 5 details how various components of the framework are estimated and in Section 6 we discuss the experimental setup used to test our framework. We give the results of our framework in Section 7 and analyze the results in detail in Section 8. Finally, we draw conclusions in Section 9.

2. RELATED WORK

To bridge the vocabulary gap between the query and the document collection we often use query modeling. Query modeling consists of transformations of simple keyword queries into more detailed representations of the user's information need, for example, by assigning (different) weights to terms, expanding the query with terms related to the query, or using phrases. Many query expansion techniques have been proposed and they mostly fall into two categories, that is, global analysis and local analysis. 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 (e.g., Qiu and Frei [1993]) also provide examples of the global approach.

Our focus is on *local* approaches to query expansion, that use the top retrieved documents as examples from which to select terms to improve retrieval performance [Rocchio 1971]. In the setting of language modeling approaches to query expansion, the local analysis idea has been instantiated by estimating additional query language models [Lafferty and Zhai 2003; Tao and Zhai 2006] or relevance models [Lavrenko and Croft 2001] from a set of feedback documents. Yan and Hauptmann [2007] explore query expansion in a multimedia setting. Meij et al. [2009] introduce a model that does not depend solely on each feedback document individually nor on the set of feedback documents as a whole, but combines the two approaches. Balog et al. [2008] compare methods for sampling expansion terms 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. Kurland et al. [2005] also try to uncover multiple aspects of a query and to that end they provide an iterative "pseudo-query" generation technique, using cluster-based language models.

2.1. External Query Expansion

The use of external collections for query expansion has a long history; see, for example, Kwok et al. [2001] and Sakai [2002]. Diaz and Metzler [2006] were the first to give a systematic account of query expansion using an external corpus in a language modeling setting, with the goal of improving the estimation of relevance models. As will become clear in Section 4, Diaz and Metzler [2006]'s approach is an instantiation of our general model for external expansion.

Typical query expansion techniques, such as pseudo-relevance feedback, using a blog or blog post corpus do not provide significant performance improvements and often dramatically hurt performance. For this reason, query expansion using external corpora has been a popular technique at the TREC Blog track [Ounis et al. 2007]. For blog post retrieval, several TREC participants have experimented with expansion against external corpora, usually a news corpus, Wikipedia, the Web, or a mixture of these [Ernsting et al. 2008; Java et al. 2007; Zhang and Yu 2007]. For the blog finding

task introduced in 2007, TREC participants again used expansion against an external corpus, usually Wikipedia [Balog et al. 2009; Elsas et al. 2008a; Ernsting et al. 2008; Fautsch and Savoy 2009]. The motivation underlying most of these approaches is to improve the estimation of the query representation, often trying to make up for the unedited nature of the corpus from which posts or blogs need to be retrieved. Elsas et al. [2008b] go a step further and develop an interesting query expansion technique using the links in Wikipedia.

Another approach to using external evidence for query expansion is explored by Yin et al. [2009]. They use evidence found in Web search snippets, query logs, and Web search documents to expand the original query and show that especially the snippets (generated by Web search engines) are very useful for this type of query expansion. Xu et al. [2009] apply query expansion on Wikipedia after classifying queries into entity, ambiguous, and broader queries and find that this external expansion works well on various TREC collections. This work shows some resemblance to our work in this article, but it also shows large differences. The method proposed by Xu et al. [2009] is a two-step approach and makes a binary decision on how to expand the query. Our model is a one-step approach and is more general in that it can mix various external collections based on the query without making a binary decision of whether or not to expand the query on a certain collection.

3. GENERAL RETRIEVAL FRAMEWORK

We work in the setting of generative language models. Here, one usually assumes that a document's relevance is correlated with query likelihood [Hiemstra 2001; Miller et al. 1999; Ponte and Croft 1998]. Within the language modeling approach, one builds a language model from each document, and ranks documents based on the probability of the document model generating the query, that is $P(D|Q)$. Instead of calculating this probability directly, we apply Bayes' theorem 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 for query Q, since it will be the same for all documents. This leaves us with

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

Assuming that query terms are independent from each other, $P(Q|D)$ is estimated by taking the product over each term t in query Q. 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. To prevent numerical underflows, we perform the computation in the log domain (thus compute the log-likelihood of the document being relevant to the query). This leads to the following equation.

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

Finally, 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 the *query model*. We also generalize $P(t|D)$ to a *document model*, $P(t|\theta_D)$, and arrive at our final formula for ranking documents.

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

Here, we see the prior probability of a document being relevant, $P(D)$ (which is independent of the query Q), the probability of observing the term t given the document model, θ_D, and the probability of a term t for a given query model, θ_Q.

We assume $P(D)$ to be uniformly distributed, that is, each document is assigned the same prior probability. We briefly touch on the choice of the document prior in Section 5.4. The ranking produced by this model is equivalent to ranking by the negative KL-divergence between query Q and document D [Balog et al. 2008]. As to $P(t|\theta_D)$, we follow a common approach and smooth the document probability with the collection probability: $P(t|\theta_D) = \lambda P(t|D) + (1 - \lambda)P(t|C)$ and we take (an empirically chosen value) $\lambda = 0.6$. For our experiments we use the implementation as provided by Indri.[2] The main interest of this article lies in improving the estimation of the query model, which is discussed in the next section.

4. QUERY MODELING USING EXTERNAL COLLECTIONS

To improve the estimation of the query model and close the vocabulary gap between the information need and the query we take the query model to be a linear combination of the maximum-likelihood query estimate $P(t|Q)$ and an expanded query model $P(t|\hat{Q})$.

$$P(t|\theta_Q) = \lambda_Q \cdot P(t|Q) + (1 - \lambda_Q) \cdot P(t|\hat{Q}) \tag{6}$$

We use the maximum-likelihood estimate for $P(t|Q)$, that is, $P(t|Q) = n(t, Q) \cdot |Q|^{-1}$, where $|Q|$ is the query length. We focus on the expanded query, $\hat{Q}$, where our goal is to build this expanded query model by *combining evidence from multiple external collections*, as explained in Section 1.

We estimate the probability of a term t in the expanded query $\hat{Q}$ using a mixture of collection-specific query expansion models

$$P(t|\hat{Q}) = \sum_{C \in \mathcal{C}} P(t|Q, C) \cdot P(C|Q), \tag{7}$$

where $\mathcal{C}$ is the set of external collections that we want to use for query expansion (see Section 6.4 for a discussion on our external collections). In the remainder of this section we work our way through the general model of Eq. (7) to end up with a final implementation of the model.

First, we look at $P(C|Q)$, the probability of a collection for the given query. To account for the sparseness of query Q compared to collection C, we apply Bayes' theorem to $P(C|Q)$, and rewrite it

$$P(C|Q) = \frac{P(Q|C) \cdot P(C)}{P(Q)}, \tag{8}$$

where $P(Q|C)$ is the probability of collection C generating query Q, $P(C)$ is the prior probability of the collection, and $P(Q)$ is the probability of observing the query.

[2]We used Lemur version 4.10, `http://www.lemurproject.com`.

Next, we shift focus to the first component of Eq. (7), the probability of observing a term t given a query and collection jointly (i.e., $P(t|Q, C)$). To estimate this probability we bring in the documents in collection C as latent variable

$$P(t|Q, C) = \sum_{D \in C} P(t|Q, C, D) \cdot P(D|Q, C), \tag{9}$$

where we again have the problem of the sparseness of query Q compared to document D. We apply Bayes' theorem to the probability of observing document D given a query and collection (i.e., $P(D|Q, C)$), resulting in

$$P(t|Q, C) = \sum_{D \in C} P(t|Q, C, D) \cdot \frac{P(Q|D, C) \cdot P(D|C)}{P(Q|C)}. \tag{10}$$

We now substitute Eqs. (8) and (10) back into Eq. (7), leading to the following set of equations.

$$\begin{aligned} P(t|\hat{Q}) &= \sum_{C \in \mathcal{C}} P(t|Q, C) \cdot P(C|Q) \\ &= \sum_{C \in \mathcal{C}} \frac{P(Q|C) \cdot P(C)}{P(Q)} \sum_{D \in C} P(t|Q, C, D) \cdot \frac{P(Q|D, C) \cdot P(D|C)}{P(Q|C)} \\ &\propto \sum_{C \in \mathcal{C}} P(C) \sum_{D \in C} P(t|Q, C, D) \cdot P(Q|D, C) \cdot P(D|C) \end{aligned} \tag{11}$$

Since $P(Q)$, the probability of the query, is equal for all terms and therefore does not influence the "ranking" of terms, we can safely ignore it.

The model in Eq. (11) is our final model for generating query expansion terms from a set of external collections. We refer to this model as *External Expansion Model*, and it includes the following four components.

Collection prior. This is the a priori probability of selecting collection C for term generation (i.e., $P(C)$).
Term generator. This is the probability of a term t being generated by the combination of a query Q, collection C, and document D (i.e., $P(t|Q, C, D)$).
Query generator. This is the probability of a query Q being generated by a document D and collection C jointly (i.e., $P(Q|D, C)$).
Document generator. This is the probability of a document D being generated by a collection C (i.e., $P(D|C)$).

For two of the components, the term generator and the query generator, we need further details on how to estimate them. The next section discusses how we can instantiate our External Expansion Model. The other two components, the collection prior and document generator, are briefly discussed in Sections 5.1 and 5.4.

4.1. Instantiating the External Expansion Model

We first look at the term generator, that is, $P(t|Q, C, D)$. We make the assumption that expansion term t and both collection C and original query Q are independent given document D. Hence,

$$P(t|Q, C, D) = P(t|D). \tag{12}$$

The independence between t and Q is assumed by design; we want to be able to sample expansion terms that do not necessarily co-occur with the original query [Balog et al. 2008]. In other words, both term t and query Q are sampled from document D, and

they are sampled independently. The dependence between t and C is implicitly present, since document D is conditioned on the collection C (see Eq. (11)).

For estimating the probability of a query being generated given a document and collection, we make the assumption that the document and collection are independent (this can be done here, since the dependence between document D and collection C is captured in the document generator component) and we ignore $P(Q)$ for ranking purposes.

$$\begin{aligned} P(Q|D,C) &= P(D,C|Q) \cdot \frac{P(Q)}{P(D,C)} \\ &= P(D|Q) \cdot P(C|Q) \cdot \frac{P(Q)}{P(D,C)} \\ &= \frac{P(Q|D) \cdot P(D)}{P(Q)} \cdot \frac{P(Q|C) \cdot P(C)}{P(Q)} \cdot \frac{P(Q)}{P(D) \cdot P(C)} \\ &\propto \frac{P(Q|C) \cdot P(Q|D)}{P(Q)} \\ &\propto P(Q|C) \cdot P(Q|D) \end{aligned} \tag{13}$$

Although the independence between the document and the collection is a strong assumption to make, the resulting model is plausible: the probability of a query being generated jointly by the document and the collection depends on the probability of the query being generated by the collection (i.e., $P(Q|C)$) and the probability of the query being generated by the document (i.e., $(P(Q|D))$).

Substituting Eqs. (12) and (13) into Eq. (11) we obtain the following instance of our External Expansion Model.

$$P(t|\hat{Q}) \propto \sum_{C \in \mathcal{C}} P(Q|C) \cdot P(C) \sum_{D \in C} P(t|D) \cdot P(Q|D) \cdot P(D|C) \tag{14}$$

The model in Eq. (14) is the instance of our External Expansion Model that we use in the remainder of this article. It takes into account the prior probability of a collection (i.e., $P(C)$), the query-dependent collection importance (i.e., $P(Q|C)$), the term probability (i.e., $P(t|D)$), the document relevance (i.e., $P(Q|D)$), and the importance of a document in a given collection (i.e., $P(D|C)$).

4.2. Relation to the Mixture of Relevance Models

We obtain a special instance of our External Expansion Model when we assume $P(Q|C)$ to be uniformly distributed, that is, all collections are equally likely to generate a query. Using this assumption, we get

$$P(t|\hat{Q}) \propto \sum_{C \in \mathcal{C}} P(C) \sum_{D \in C} P(t|D) \cdot P(Q|D) \cdot P(D|C). \tag{15}$$

Following Lavrenko and Croft [2001] and assuming that $P(D|C) = \frac{1}{|\mathcal{R}_C|}$, the size of the set of top ranked documents in C (denoted by $\mathcal{R}_C$) we arrive at

$$P(t|\hat{Q}) \propto \sum_{C \in \mathcal{C}} \frac{P(C)}{|\mathcal{R}_C|} \sum_{D \in \mathcal{R}_C} P(t|D) \cdot P(Q|D). \tag{16}$$

The resulting model in Eq. (16) is in fact the Mixture of Relevance Models (MoRM) proposed by Diaz and Metzler [2006]. The main difference between this MoRM and

our External Expansion Model is the query-dependent collection importance, which is introduced in our model. In Section 8.2 we analyze the differences in performance between these two models.

Now that we have described our choices for the final components of our query expansion model, we proceed by looking for ways to estimate these components in the next section.

5. ESTIMATING MODEL COMPONENTS

Our External Expansion Model consists of five components that we need to estimate. In this section we discuss each of the components and introduce ways of estimating them.

5.1. Prior Collection Probability

In a Web setting, prior probabilities of documents are often assigned based on "authoritativeness," with PageRank and HITS [Manning et al. 2008] being well-known ways of computing authoritativeness scores. For collections it seems harder to come up with a proper estimate of a prior probability, as they usually exist completely separated from each other. The most straightforward solution is to ignore the prior probability and assign a uniform probability to all collections: $P(C) = |\mathcal{C}|^{-1}$, where $|\mathcal{C}|$ is the size of $\mathcal{C}$.

In this article we do not explore other ways of estimating the collection prior, but we briefly touch on two options: (i) Based on the ideas of Weerkamp and de Rijke [2012] we could turn credibility into a collection-wide feature. We determine the credibility of a sample of documents from the collection and take the average credibility score to reflect the collection's credibility. (ii) A second option would be to make the prior probability task dependent. Consider the following three examples: (a) A time-sensitive (real-time) search task could benefit more from real-time collections, like microblogs and news sources. (b) A technical search task could benefit from a collection of manuals. (c) A filtering task, which mostly asks for general topics, could benefit from a general knowledge source (e.g., an encyclopedia). In-depth knowledge of the character of the task could be used to predefine the collection probabilities.

We examine the effects of estimating the collection prior in Section 8.2.

5.2. Document Relevance

We need to estimate the relevance of a document D for a given query Q. The goal of our models is to bring in high-quality expansion terms and we therefore take two precision enhancing steps in determining document relevance: (1) Only documents that contain all query terms can be considered relevant; (2) we use the Markov Random Field Model as introduced by Metzler and Croft [2005] to search for individual query terms and phrases constructed from the query. In Section 3 we introduced our general retrieval framework, including $P(Q|D)$. We take

$$P(Q|D) \propto \sum_{c \in T} \lambda_T f_T(c) + \sum_{c \in O} \lambda_O f_O(c) + \sum_{c \in O \sqcup U} \lambda_U f_U(c), \tag{17}$$

where T are individual query terms, O are the sequences of two or more contiguously appearing query terms, and U are the sequences of two or more noncontiguously appearing query terms. For setting the parameters we use the values proposed by Metzler and Croft [2005], that is, $\lambda_T = 0.8$, $\lambda_O = 0.1$, and $\lambda_U = 0.1$. Remember that documents need to contain *all* query terms to be considered relevant.

5.3. Collection Relevance

We already discussed the prior probability of a collection, which is independent of the query at hand. Here, however, we need an estimate of the likelihood that collection C generated query Q. We can also look at this as the relevance of the collection to the given query. We try to determine the average relevance of documents in the collection and use that as indication of how well this collection will be able to answer the query. We have

$$P(Q|C) = \sum_{D \in C} P(Q|D) \cdot P(D|C)$$
$$\propto \frac{1}{|\mathcal{R}_C|} \cdot \sum_{D \in \mathcal{R}_C} P(Q|D), \tag{18}$$

where we assume all documents to be equally important, that is, $P(D|C)$ is uniform. The query likelihood, $P(Q|D)$, is calculated the same way as we did in Eq. (17). Instead of iterating over all documents in a collection ($D \in C$ in Eq. (18)), we estimate this value using the top $\mathcal{R}_C$ documents according to Eq. (17), where $\mathcal{R}_C$ depends on the collection size. We refer to this estimation method as "relevance."

The second approach we try here for estimating the collection relevance is using the ratio of documents containing *all* query terms to the total number of documents in the collection. For example, we have the query *state of the union*, for which we find that 5,427 documents in a (news) collection contain all the query terms. Given that the collection has 135,763 documents in total, we estimate $P(Q|C) = \frac{5,427}{135,763} = 0.040$. We refer to this method as "boolean."

There are other indicators of collection relevance that we could take into account, for example, the individual query term frequency in the collection, popularity of query terms in query logs related to the collection, or the frequency of query terms in special (important) fields in the collection (e.g., anchor text, article title). Indeed, there is previous work on predicting whether expansion terms are helpful or not [Cao et al. 2008; Cartright et al. 2009], which could be translated to our estimation of collection importance. We investigate the effects of estimating the collection relevance in Section 8.2.

5.4. Document Importance

Not all documents in a collection are equally important. Document importance estimators allow us to create a ranking of documents independent of a query. PageRank has proven itself beneficial to retrieval performance in a Web setting, although this is hard to confirm in smaller collections [Hawking et al. 1999]. Other beneficial ways of estimating document importance include credibility [Weerkamp and de Rijke 2008, 2012], recency [Dong et al. 2010], URL length [Westerveld et al. 2002], and document length [Kamps et al. 2004].

Although various options for estimating document importance are available, it is not the focus of this article. Document importance and query modeling are orthogonal, in the sense that the former is independent of the query and the latter is all about the query. And, although both techniques can be used in one system, we want to investigate the impact of our external query modeling and we therefore assume this probability to be uniformly distributed, giving all documents in collection C the same probability. We leave it as future work to implement other features like the ones mentioned before.

Table II. Collection Statistics

Period	12/06/2005 – 02/21/2006
After boilerplate removal	
Number of blogs	100,649
Number of posts	3,215,171
Index size	12.0 GB
After boilerplate removal and language detection	
Number of blogs	76,358
Number of posts	2,574,356
Index size	9.3 GB

5.5. Term Probability

The term probabilities $P(t|D)$ indicate how likely it is that we observe a term t given a document D. For this probability we use the maximum-likelihood estimate

$$P(t|D) = \frac{n(t, D)}{|D|}, \tag{19}$$

where $n(t, D)$ is the number of times term t occurs in document D and $|D|$ is the length of D in words.

We have now finalized our modeling sections and discussed how to estimate the various components of our External Expansion Model. Next, we put our model to the test using the experimental setup detailed in the next section.

6. EXPERIMENTAL SETUP

To test our External Expansion Model we apply it to the task of blog post retrieval. Details of the task, document collection, and test topics we use are given in Section 6.1. We introduce the metrics and significance test on which we report in Sections 6.2 and 6.3. The collections that we deploy as potential external sources for our query expansion terms are introduced in Section 6.4. Finally we discuss the parameter setting of our model in Section 6.5.

6.1. Task, Collection, and Topics

We apply our model to the task of retrieving topically relevant blog posts. This task ran at the Text REtrieval Conference (TREC), as part of the Blog track, in 2006–2008 [Macdonald et al. 2008; Ounis et al. 2007, 2009]. Given a set of blog posts and a query, we are asked to return relevant blog posts for that query. We apply our model to the TREC Blog06 corpus [Macdonald and Ounis 2006], which has been constructed by monitoring around 100,000 blog feeds for a period of 11 weeks in early 2006, downloading all posts created in this period. In our experiments we use only the HTML documents (permalinks), and ignore syndicated (RSS) data. We perform two preprocessing steps: (i) keep only long sentences [Hofmann and Weerkamp 2008] and (ii) apply language identification using TextCat,[3] to select English posts. The collection statistics are displayed in Table II. As an additional (post-)processing step we ignore terms shorter than 3 characters. The reason for this is that due to encoding issues in the crawl of some of the collections, we observe frequently occurring strange characters and we use this postprocessing step to get rid of these encoding errors.

The TREC 2006, 2007, and 2008 Blog tracks each offer 50 topics and corresponding relevance assessments, adding up to 150 topics in total. For topical relevancy, assessment was done using a standard two-level scale: the content of the post was judged to

[3]http://odur.let.rug.nl/~vannoord/TextCat/

be topically relevant or not. For all our retrieval tasks we only use the title field (T) of the topic statement as query; this boils down to the use of keyword queries.

6.2. Metrics

We report on four standard IR metrics [Manning et al. 2008]. Three of these metrics are precision oriented: precision at ranks 5 and 10 (P5 and P10) and Mean Reciprocal Rank (MRR). We also report on Mean Average Precision (MAP), which captures both precision and recall and therefore is our most important metric. In case of optimization, we do so for MAP and P5. Next, we briefly introduce the four metrics.

Mean Average Precision (MAP). This metric is used most commonly in research in the field of information retrieval. For each relevant document in the returned document list we take the precision at the position of that document. We sum over these precision values and divide it by the total number of relevant documents. This gives us the Average Precision (AP) for a query. When we take the mean of AP values over a set of test queries, we get the Mean Average Precision (MAP) for a system on that set of queries.

Precision at Rank r (Pr). The precision at rank r metrics (P5 and P10) indicates the percentage of relevant documents within the top r returned documents. In Web-search-related tasks this metric is often considered important, because users tend to look only at the top results of a ranking.

Mean Reciprocal Rank (MRR). The final precision-oriented metric we report on is mean reciprocal rank. This metric indicates how good a system is in returning the first relevant document as high up in the ranking as possible. To measure this we take the reciprocal of the position of the first relevant document (RR). After taking the average over the RR values of a set of queries we get the mean reciprocal rank for a system on that set of queries.

6.3. Significance

We test for statistical significant differences using a two-tailed paired t-test. Significant improvements over the baseline are marked with $^{\triangle}$ ($\alpha = 0.05$) or $^{\blacktriangle}$ ($\alpha = 0.01$), and we use $^{\triangledown}$ and $^{\blacktriangledown}$ for a drop in performance (for $\alpha = 0.05$ and $\alpha = 0.01$, respectively).

6.4. Collections

We need to decide on the set of external collections that we use in our experiments. The most important criterion for deciding which collections to use is the task one is trying to solve. In our case, we are looking at blog post retrieval, which leads us to the following (external) collections. For each collection we briefly explain why this collection is suitable. Note that all four collections introduced next are generally available, ensuring reproducibility of the experiments.

News articles. Based on observations by Mishne and de Rijke [2006] and the relation between news and social media [Kwak et al. 2010; Leskovec et al. 2009], we hypothesize that news articles are an important part of the bloggers' environment. We use AQUAINT-2 [Aquaint-2 2007], a collection of news articles from six sources covering the same period as the blog post collection. This collection gives us 135,763 English news articles, mostly of high text quality (i.e., formal text).

Encyclopedia. In the Introduction we already showed an example of a concept query (*jihad*). Many of these concept queries Mishne and de Rijke [2006] are quite

Table III. Baseline Scores for All Three Topic Sets and the Combination of All 150 Topics

Year	MAP	P5	P10	MRR
2006	0.3365	0.6880	0.6720	0.7339
2007	0.4514	0.7200	0.7240	0.8200
2008	0.3800	0.6760	0.6920	0.7629
all	0.3893	0.6947	0.6960	0.7722

generic and are part of people's general interests. To represent this part of the environment we use a general knowledge source (i.e., encyclopedia). We use a Wikipedia dump of August 2007 as encyclopedia, which contains 2,571,462 English Wikipedia articles. The articles are preprocessed to contain only the articles' actual content.

User-generated content. Social media like blogs and microblogs allow people to report and comment on anything they come across in (near) real time. Much of what is reported by (micro)bloggers ends up in other blog posts and the content in the (micro)blogosphere is therefore part of the environment. Ideally, we would like to have a Twitter collection from the same period as our blog collection. However, since this is not available, we use the blog post collection itself as a near-real-time user-generated content source. Details of this collection are listed earlier.

Web content. Finally, bloggers are influenced by what they read online, that is, their virtual environment. To represent this virtual environment, we use a general Web collection. Here, we use the category B part of Clueweb [ClueWeb09 2009], minus Wikipedia. This gives us 44,262,894 (English) Web pages. All pages are preprocessed to eliminate HTML code and scripts. We use category B, and not category A, so as to avoid the need for elaborate spam filtering.

6.5. Parameters

Our model has two parameters. First, the main query model (viz. Eq. (6)) has a parameter λ_Q, indicating the influence of the expanded query. Second, we have an implicit parameter K indicating the number of expansion terms to be included in the new, expanded query. We determine the parameter values by training on two topic sets and testing on the third topic set (e.g., train on 2006 and 2007 topics, test on 2008 topics). We find that for all three years the same parameter values are optimal: $K = 20$ and $\lambda_Q = 0.5$. We revisit the influence of these parameters on the performance of our model in Section 8.3.

7. RESULTS

We first assess the performance of our baseline system (i.e., before applying query expansion). Table III lists the scores on each of the three topic sets. Compared to TREC participants in these years our scores are (far) above the median, indicating that our baseline is already strong without any additional techniques.

To limit the number of tables and make results easier to interpret, we report on the performance of our system on the combination of all 150 topics in the remainder of the results and analysis sections. We first explore the impact of using each of the four collections individually in Section 7.1 and we continue by looking at the combination of the collections using our External Expansion Model in Section 7.2.

7.1. Individual Collections

We apply our External Expansion Model to each of the external collections individually. By doing so, we ignore the prior collection probability (i.e., $P(C)$) and the probability of

Table IV. Performance of Query Expansion on the Individual External Collections for All 150 Topics

	MAP	P5	P10	MRR
baseline	0.3893	0.6947	0.6960	0.7722
news	0.4035	0.7173	0.7080	0.7955
web	0.4023$^{\blacktriangle}$	0.7160	0.6980	0.8062$^{\triangle}$
Wikipedia	0.4034$^{\triangle}$	**0.7360**$^{\blacktriangle}$	**0.7273**$^{\triangle}$	**0.8105**
blog posts	**0.4121**$^{\blacktriangle}$	0.7160	0.7073	0.7933

Significance tested against the baseline.

Table V. Performance of Query Expansion Using the External Expansion Model on All External Collections for All 150 Topics Using the Relevance and Boolean Method for Estimating Collection Importance

		MAP	P5	P10	MRR
baseline		0.3893	0.6947	0.6960	0.7722
EEM	relevance	**0.4117**$^{\blacktriangle}$	**0.7427**$^{\blacktriangle}$	**0.7133**	0.8005
EEM	boolean	0.4110$^{\blacktriangle}$	0.7360$^{\blacktriangle}$	0.7113	**0.8053**
MoRM		0.4102$^{\blacktriangle}$	0.7293$^{\blacktriangle}$	0.7120	0.7985

Significance tested against the baseline.

observing the query given a collection (i.e., $P(Q|C)$). The results of expansion on the individual collections are listed in Table IV.

The first thing we notice is that expansion on each of the individual collections is beneficial and performance on each of the metrics goes up for every external collection. Unlike in previous work [Arguello et al. 2008; Jijkoun et al. 2010], though, expansion on the blog post collection itself seems to work very well, especially for MAP (+6%). For purely precision-oriented metrics Wikipedia seems to be a good source for query expansion terms, resulting in significant improvements on precision at ranks 5 and 10, and a large increase in MRR (although not significant). The Web collection shows significant improvements for MAP and MRR, which is an interesting combination of recall-and precision-oriented metrics. Finally, the news collection does improve on all metrics and it achieves the second highest score on most metrics, but improvements are not significant.

An interesting observation regarding the performance of the news collection is the fact that it only expands 139 out of 150 topics. For the remaining 11 topics we could not find relevant documents in this collection (i.e., none of the documents contains all query terms). The other three collections have more topics with relevant documents: 147 for Wikipedia, 149 for blog posts, and 150 for the Web collection.

7.2. Combination of Collections

We now focus on the actual implementation of our External Expansion Model, which can take on board the per-topic importance of collections. We use the methods detailed in Section 5.3 to estimate this importance (i.e., $P(Q|C)$) and compare these runs to the model when this probability is assumed to be uniformly distributed. As mentioned before, this boils down to the Mixture of Relevance Models [Diaz and Metzler 2006]. The results of both methods and the baseline without expansion are listed in Table V.

The results show that our External Expansion Model with two rather simple estimations of $P(Q|C)$ performs at least as good as the Mixture of Relevance Models on all metrics, and significantly improves over it on precision at 5. Although the differences between the two methods are small, they indicate that weighing the collections on a per-topic basis can be beneficial.

Table VI. Overview of the Analyses Presented in Section 8

Section 8.1 (page 16)	Section 8.2 (page 22)	Section 8.3 (page 25)
Individual collections: - per-topic changes - interesting topics - actual query models EEM: - per-topic changes - easy and hard topics - new query models	Collection importance: - priors - query-dependent - combined - compared to MoRM	Parameters: - λ_Q - K

Comparing the results of our EEM with the performance on individual collections, we observe that the highest scores on each metric are obtained by different runs (MAP on blog posts, P5 on all four, P10 and MRR on Wikipedia), but that EEM is most stable across metrics. Another interesting observation is that, although query expansion is usually referred to as a recall-enhancing method, here, it shows performance improvements on all metrics, recall-oriented (MAP) and precision-oriented (P5, P10, and MRR). To explain what really happens, we perform an extensive analysis of the runs in the next section.

8. ANALYSIS AND DISCUSSION

We perform an extensive analysis of our results: Table VI lists the analyses presented in this section. In Section 8.1 we look at the per-topic performance of query expansion on individual collections and of our External Expansion Model on all collections. We give examples of query models that are generated by different collections and by EEM. In Section 8.2 we explore the influence of both the collection prior and the query-dependent collection importance. We use the (per-topic) performance of individual collections as oracle weights and we compare EEM to the Mixture of Relevance Models (MoRM). Finally, in Section 8.3 we look at the impact of parameters λ_Q (the weight of the original query compared to the expanded query) and K (the number of terms in the expanded query model) on the retrieval performance of EEM.

8.1. Per-Topic Analysis

Looking at the overall performances is useful for obtaining a general understanding and assessment of a new model, but it also hides a lot of detail. In this section we perform a per-topic analysis of the runs using individual collections and our External Expansion Model and we show how performance changes across different topics.

8.1.1. Individual Collections. We start our analysis by exploring the per-topic influence of query expansion using the various external collections. To this end we plot the difference in AP between the nonexpanded baseline and the expanded runs using each of the four external collections. For presentational reasons we present the results per topic set (i.e., 2006, 2007, and 2008 topics separated). The plots in Figures 1, 2, and 3 show stacked bars, which makes it easy to see which of the collections works best or worst for each topic.

We can draw several conclusions from the plots: (i) there is a large difference between topics as to how much improvement can be obtained from (external) query expansion. For some topics we achieve 0.4, 0.5, or even 0.6 improvement in AP, whereas in other cases, we see a decrease in AP up to 0.4. (ii) The collection that works best differs per topic, as we expected. In some cases (e.g., topic 924, *Mark Driscoll*) we see a clear difference between collections, where one or more collections hurt performance and the others help the topic (in case of topic 924, news and Wikipedia hurt the topic,

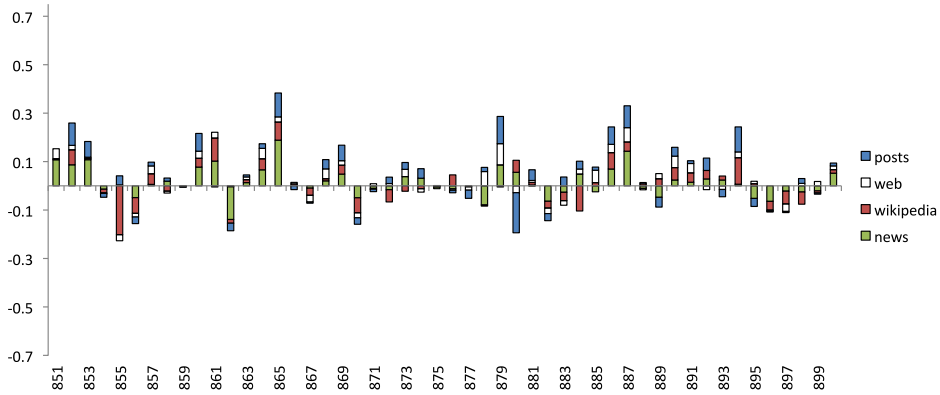

Fig. 1. Change in AP between the nonexpanded baseline and expansion on each of the individual collections for 2006 topics.

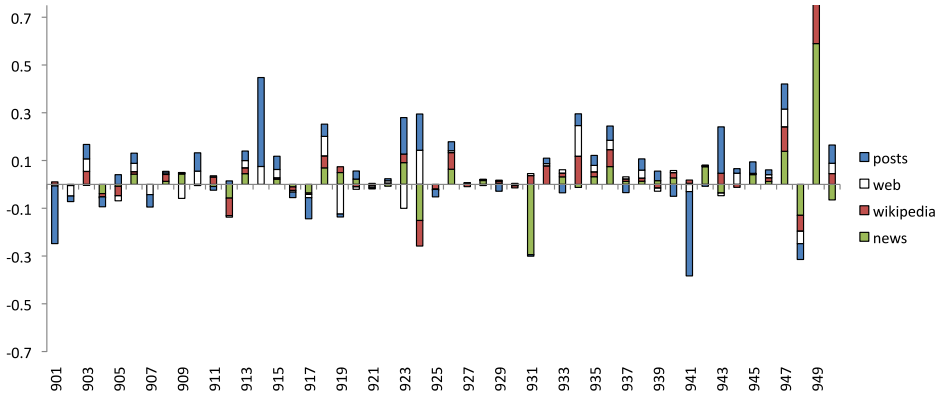

Fig. 2. Change in AP between the nonexpanded baseline and expansion on each of the individual collections for 2007 topics (note that the total difference for topic 949 is 1.67).

whereas Web and blog posts help). For other topics, however, it seems it does not matter much which collection is chosen, as they all improve effectiveness.

Looking at the total number of topics that benefit from using each external collection for query expansion, we obtain the numbers listed in Table VII. Here, we observe that the Web collection helps most topics and hurts relatively few (compared to the other collections). The news collection helps the least topics, but that is partially due to the fact that for 11 topics it does not have any results, which also explains the large number of equal topics.

We zoom in on individual topics and list seven "interesting" topics in Table VIII. The first two topics show large improvements in AP for all collections compared to the nonexpanded baseline, although some collections help more than others. The last two topics are particularly hard and show no improvement after expanding the query, regardless of the external collection that is used. The middle three topics are interesting in that they improve for some collections, but are hurt by others. It is these topics for which we included the query-dependent collection weight in our model.

Why do certain topics improve on, say, the news collection, but are hurt by the Web collection? We look at the actual query models generated for the collections on the three topics in Table VIII (i.e, topics 924, 1049, and 1031). First we look at two

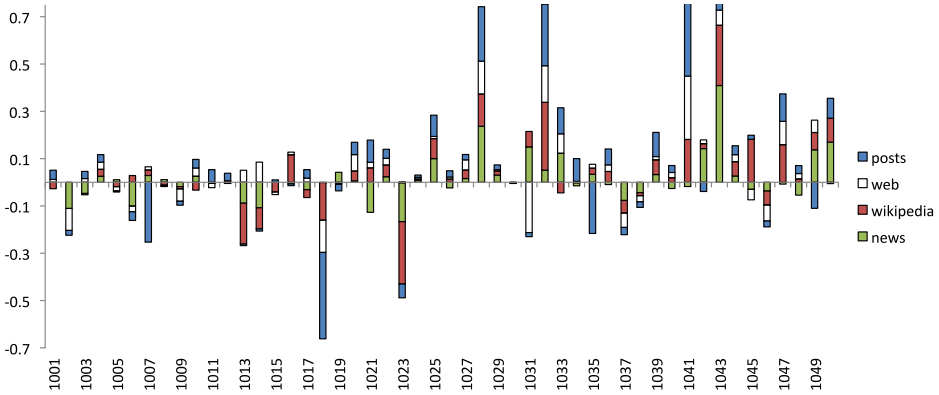

Fig. 3. Change in AP between the nonexpanded baseline and expansion on each of the individual collections for 2008 topics (note that the total difference for topic 1043 is 1.13).

Table VII. Number of Topics Each Collection Helps or Hurts Compared to the Nonexpanded Baseline

Collection	Number of topics		
	up	equal	down
news	80	13	57
Wikipedia	90	3	57
Web	97	2	51
blog posts	91	1	58

Table VIII. Topics that Show Interesting Behavior

Topic ID	query	change in AP compared to baseline			
		news	Wikipedia	Web	blog posts
949	ford bell	0.5897	0.2584	0.2259	0.5919
1043	a million little pieces	0.4090	0.2546	0.0645	0.4062
924	mark driscoll	−0.1511	−0.1070	0.1430	0.1518
1049	youtube	0.1373	0.0731	0.0523	−0.1101
1031	sew fast sew easy	0.1496	0.0652	−0.2126	−0.0173
1023	yojimbo	−0.1667	−0.2635	0.0017	−0.0583
1018	mythbusters	−0.0016	−0.1586	−0.1364	−0.3653

query models generated for topic 924, *mark driscoll* using the news collection (left) and the blog post collection (right) in the left part of Table IX. The news collection hurts the topic, dropping AP by 0.1511, while the post collection helps (AP improvement of 0.1518). Mark Driscoll is an evangelist. Looking at the query models generated by the two collections, we find relevant terms like *church*, *god*, and *McLaren* (one of his friends) in the blog post query model, whereas the news query model not only lacks these terms, but also introduces very unrelated terms like *bowl*, *athletic*, and *sports*. We find that there is another Mark Driscoll (an athletics director at CSU), which accounts for the terms in the news collection.

The second example is topic 1049, *youtube*. Here, we see an opposite effect: the news collection helps the topic (+0.1373 AP) and the blog post collection hurts (−0.1101 AP). The two query models are displayed in the center of Table IX, with news on the left and blog posts on the right. The terms extracted from the news collection are fairly "clean," all pointing to YouTube in some way, leading to an improvement in AP. The terms from the blog posts on the other hand, are more general (e.g., *www*, *download*,

Table IX. Query Models for Topics that Show Interesting Behavior

Topic 924 *mark driscoll*		Topic 1049 *youtube*		Topic 1031 *sew fast sew easy*	
News	Blog posts	News	Blog posts	News	Web
bowl	driscoll	youtube	youtube	sew	sew
athletic	mark	video	openfb	knitting	sewing
audit	church	music	video	group	knitting
families	people	site	download	trademark	easy
sports	posted	clips	written	meyrich	fast
director	god	nbc	www	stoller	machine
coaches	emerging	clip	javascript	stitch	stitch
college	dont	web	programming	bitch	projects
games	mclaren	television	bookmarklet	fast	home
state	emergent	copyright	videos	knitters	book

We only show the top 10 terms.

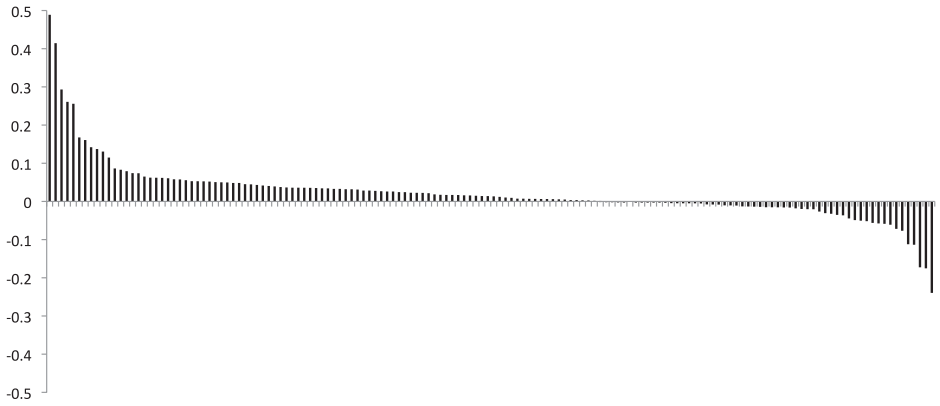

Fig. 4. Change in AP between the nonexpanded baseline and EEM. Topics ordered by their improvement in AP over the baseline.

programming, *javascript*) or seem to be unrelated (*openfb*, *written*, *bookmarklet*), causing the query to shift focus from YouTube to more general, unrelated topics.

The final example is topic 1031, *sew fast sew easy*. This company delivers sewing and knitting classes, patterns, and books. In the original topic description, relevant documents are said to be about this company, but also about its objections against the use of a trademarked statement. Interestingly, the news collection (which improves AP by 0.1496) generates the term *trademark*, besides other relevant terms like *Meyrick* (founder), *stitch* and *bitch* (Stitch & Bitch Café, the online forum), and *knitters* and *knitting*. The terms from the Web collection (leading to a drop in AP of 0.2126) include very general terms like *machine*, *projects*, *home*, and *book*, causing the query to drift away from its original focus.

8.1.2. External Expansion Model. Zooming in on the performance of our External Expansion Model we can perform similar analyses as before. First, we look at the per-topic performance by plotting the differences in AP between the EEM run and the nonexpanded baseline in Figure 4. We order the topics by decreasing AP improvement to make the plot easier to interpret.

The plot shows that the majority of topics improve over the baseline in terms of AP. Besides that, we also observe that the improvements are larger than the decreases (the height of the columns). Adding numbers to this plot, we find that 98 topics improve in AP over the baseline and 51 topics show a drop (1 topic stays the same). Looking

Table X. Topics that Are Helped or Hurt Most in Terms of AP by EEM Compared to the Nonexpanded Baseline

Topic ID	query	AP change	
949	ford bell	+0.4888	+327%
1043	a million little pieces	+0.4145	+219%
1041	federal shield law	+0.2932	+190%
914	northernvoice	+0.2608	+125%
1032	i walk the line	+0.2558	+179%
1007	women in saudi arabia	−0.2395	−75%
1018	mythbusters	−0.1752	−42%
1013	iceland european union	−0.1725	−34%
919	pfizer	−0.1134	−21%
1023	yojimbo	−0.1121	−19%

Table XI. Topics that Are Helped or Hurt Most in Terms of Precision at 5 by EEM Compared to the Nonexpanded Baseline

Topic ID	query	AP change	
1041	federal shield law	+0.8000	+400%
851	march of the penguins	+0.6000	+150%
949	ford bell	+0.6000	+150%
943	censure	+0.6000	+150%
1007	women in saudi arabia	−0.4000	−67%

at the differences in precision at 5, we have 34 improved topics compared to 11 topics with a drop. The remaining 105 topics do not change. Comparing these numbers to the previous numbers in Table VII, we find that the numbers here are slightly better, giving an indication of the strength of the model. Exploring the plot in Figure 4 we ask ourselves which topics are located on the rightmost and leftmost parts of the plot, that is, which topics are helped or hurt most by EEM? Table X shows these topics and their (relative) change in AP compared to the baseline.

As we already concluded from the plot, the increase in AP is much higher than the decrease, with improvements as high as 327% for topic 949. Topic 1007 seems particularly hard for our method, as it also features in Table XI. This table lists the topics that are helped or hurt in terms of precision at 5. The only topic showing a rather large decrease is topic 1007. All the topics that improve most on precision at 5, reported in Table XI, have a precision of 1.0000.

Why do some of these topics perform well after expanding and why are others hurt? We take a closer look at the query models of three topics: topic 949 (*ford bell*), topic 1041 (*federal shield law*), and topic 1007 (*women in saudi arabia*). To start with the first topic, Table XII (left) shows the expansion terms EEM selects for topic 949. Ford Bell was a U.S. Senate candidate from the DFL party. The query model shows terms related to his candidacy (*senate*, *candidate*, *race*), his political environment (*democrats*, *Amy Klobuchar*), and himself (*Minnesota*, *Minneapolis*, *DFL*).

The second example topic, 1041, is about the Federal Shield Law, which should protect sources of journalists. The terms extracted by EEM show relevant terms on the journalist side (*journalists*, *media*, *press*, *reporters*, *journalism*, *SPJ* (society of professional journalists)), on the source side (*sources*), and on the topic of the law (*free*, *freedom*, *information*). The terms *times* and *miller* are related to a case in which New York Times reporter Judith Miller was sent to jail for not giving up her source. She became an advocacy of the Federal Shield Law.

Finally, we look at a topic that proves difficult, topic 1007. Relevant documents for this topic should be about treatment of women in Saudi Arabia, but this is not clear from the extracted terms in Table XII. Although some terms could be related to this

Table XII. Query Models Constructed by EEM for Three Example Topics

Topic 949 *ford bell*		Topic 1041 *federal shield law*		Topic 1007 *women in saudi arabia*	
bell	ford	law	shield	university	women
minnesota	library	federal	journalists	saudi	arab
james	university	media	information	east	mother
minneapolis	klobuchar	press	reporters	chapter	teresa
senate	associates	sources	free	arabia	islam
kennedy	democrats	spj	court	middle	war
candidate	amy	government	public	angry	served
history	mark	journalism	freedom	lebanon	state
dfl	maps	times	laws	arabic	service
race	party	miller	national	college	washington

We show all 20 terms.

Table XIII. Performance of EEM on Three Example Topics, with the $P(Q|C)$ for Each Collection

	Topic 924	Topic 1031	Topic 1049
Individual collections			
Collection	blog posts	news	news
AP change	+0.1518	+0.1496	+0.1373
External Expansion Model			
AP change	+0.1372	−0.0265	+0.0526
$P(Q\|news)$	0.0383	0.2955	0.0099
$P(Q\|Wikipedia)$	0.1960	0.1503	0.2169
$P(Q\|web)$	0.3085	0.0409	0.7401
$P(Q\|blogs)$	0.4572	0.5133	0.0331

topic, for example, *islam*, *middle east*, and *arabic*, most of them are too general to improve the representation of the topic, leading to a decrease in AP and P5 for this topic.

We go back to the three examples we have shown in Table IX. The reason for focusing on these topics was that they show a mixed performance depending on the external collection used. Since our model is supposed to take into account the suitability of a collection for a given query, we hope to find that these topics show an improvement over the baseline. Table XIII shows the three topics, the performance of the best collection, followed by the performance of EEM and the $P(Q|C)$ our model assigned to each of the collections.

The table shows different behavior for each of the three topics. For topic 924 it is clear our model "got it right." It assigns the highest query likelihood (i.e., $P(Q|C)$) to the blog posts and Web collections, both of them very strong individual collections as well, which is reflected by the improvement in AP. For topic 1031 we see a drop in AP, whereas the best individual collection achieves a strong increase. We observe that, for this topic, the news collection is assigned a probability of 0.3, giving it a reasonable influence. Its influence is, however, marginalized by the blog post collection. The blog post collection is by far the worst performing expansion collection for this topic (viz. Table VIII). Finally, topic 1049 shows an increase in AP, although it assigns a low probability to the best individual collection (again, news). This is true for the worst collection (blog posts) too, however, leaving the Web and Wikipedia collections to achieve an increase in AP, just as they did individually.

We have shown that our External Expansion Model is capable of capturing the per-topic importance of a collection and improves over individual collections and the Mixture of Relevance Models. Next, we explore the query-dependent collection

Table XIV. Weights of External Collections ($P(C)$) in EEM, Optimized for MAP and P5

Optimization metric	news	Wikipedia	web	blog posts
MAP	0.221	0.220	0.203	0.356
P5	0.212	0.388	0.200	0.200

Table XV. Performance of EEM on All Collections Using "Oracle" Settings for $P(C)$ Based on the Performances of the Individual Collections on MAP and P5 and Uniform $P(Q|C)$

	MAP	P5	P10	MRR
baseline	0.3893	0.6947	0.6960	0.7722
EEM	**0.4117**	**0.7427**	0.7133	0.8005
MoRM	0.4102	0.7293	0.7120	0.7985
EEM oracle (MAP)	**0.4117**	0.7293	0.7133	0.7979
EEM oracle (P5)	0.4110	0.7373^{Δ}	**0.7153**	**0.8024**

Significance is tested against MoRM.

importance, as well as the prior probability of a collection, which will show the full potential of EEM.

8.2. Influence of (Query-Dependent) Collection Importance

In the previous section we have shown that, as expected, the best collection to use for query expansion is dependent on the original query. Besides that, we also saw, in Section 7.1, that certain collections show a better overall performance when used to extract new query terms (e.g., blog posts for MAP and Wikipedia for precision at 5). In this section we use these results to construct "oracle" runs.

Instead of assuming a uniform probability distribution over collections (i.e., $P(C) = |C|^{-1}$) we take the performances of the individual collections and weigh their importance based on the improvement they show over the baseline. We look at optimizing $P(C)$ this way for MAP and for precision at 5. Table XIV shows the actual weights for the collections in our External Expansion Model. For MAP we favor the blog post collection most, while for P5 we rely mostly on the Wikipedia collection.

For this experiment, we take a uniform distribution for $P(Q|C)$, making the run comparable to the Mixture of Relevance Models (MoRM) run. The results of our oracle runs are listed in Table XV. We check for significant differences against the MoRM run and observe that optimizing collection importance this way is only marginally beneficial. The MAP-optimized run does improve on MAP, but not significantly. We do get a significant improvement on precision at 5 for the P5-optimized oracle run and this run also improves on the other precision metrics, as well as on MAP. Compared to the EEM run, where $P(Q|C)$ is not uniform, but $P(C)$ is, we see hardly any improvements. Even more so, the performance on precision at 5 for the MAP-optimized run is significantly worse than the EEM run.

We now shift to the estimation of $P(Q|C)$. Our results in Section 7.2 show that even a rather simple way of estimating this probability leads to performance improvements. Here, we take the performance of each of the individual collections on each topic, similarly to Section 8.1, and use their improvement over the baseline as an estimate for $P(Q|C)$. The results of this optimization are listed in Table XVI.

We test for significant differences with the EEM run, for which we also kept $P(C)$ uniform and used different $P(Q|C)$ depending on the query and collection. Results of the oracle runs are very good and show significant improvements on most metrics. Especially optimizing for precision at 5 seems very beneficial, with all metrics showing a significant improvement.

Table XVI. Performance of EEM on All Collections Using "Oracle" Settings for $P(Q|C)$ Based on the Performances of the Individual Collections on MAP and P5 and Uniform $P(C)$

	MAP	P5	P10	MRR
baseline	0.3893	0.6947	0.6960	0.7722
EEM	0.4117	0.7427	0.7133	0.8005
MoRM	0.4102	0.7293	0.7120	0.7985
EEM oracle (MAP)	**0.4275**▲	0.7547	0.7427▲	0.8156
EEM oracle (P5)	0.4227▲	**0.7947**▲	**0.7527**▲	**0.8434**▲

Significance is tested against EEM.

Table XVII. Performance of EEM on All Collections Using "Oracle" Settings for $P(Q|C)$ and $P(C)$ Based on the Performances of the Individual Collections on MAP and P5

	MAP	P5	P10	MRR
baseline	0.3893	0.6947	0.6960	0.7722
EEM	0.4117	0.7427	0.7133	0.8005
MoRM	0.4102	0.7293	0.7120	0.7985
EEM oracle (MAP)	**0.4304**▲	0.7627	0.7493▲	0.8214
EEM oracle (P5)	0.4226▲	**0.7933**▲	**0.7533**▲	**0.8467**▲

Significance is tested against EEM.

Finally, we can combine the two oracle runs, that is, we apply the oracle weights for $P(C)$ (see Table XIV) and the query-dependent oracle weights for $P(Q|C)$. The results for this oracle run are listed in Table XVII. Here, we observe similar results as for the previous experiment: most metrics show a significant improvement compared to the EEM run and the P5-optimized run performs best on all metrics except MAP. It is interesting to compare results from Tables XVI and XVII. We observe that for the MAP-optimized run adding the oracle $P(C)$ to the External Expansion Model on top of the oracle $P(Q|C)$ helps, although differences are small. For the P5-optimized run, however, adding $P(C)$ does not help for all metrics, as it only shows marginal improvements on precision at 10 and MRR.

To get an idea of the per-topic performance of the oracle EEM runs, we plot the differences in AP between the nonexpanded baseline and the oracle EEM run with $P(Q|C)$ optimized for P5. The resulting plot is depicted in Figure 5. By far, most topics are helped by this run (110 topics) and far fewer are hurt (40 topics). Not only that, but the absolute numbers are much higher for improving topics than they are for decreasing topics. If we look at which collection is most often picked as most important expansion source, we find that the news collections is most important for 15 topics, followed by the Web collection (8 topics), Wikipedia (7 topics), and the blog posts (6 topics). For all other topics we have two or more collections being equally important.

Since our model is a generalization of the Mixture of Relevance Models (MoRM), we want to compare the performance of our model to the MoRM. Focusing on the EEM oracle run (for P5), we find that the results in Table XVII show that this run significantly outperforms the MoRM on all metrics ($p < 0.001$ for all metrics). This is confirmed by the plot in Figure 6, which shows the AP difference per topic between MoRM and our External Expansion Model. We find that EEM works better than MoRM for most topics (92 of 150), whereas MoRM works better for 46 topics.

It is interesting to look at the topics for which the MoRM actually works substantially better, that is, the bars on the far right of Figure 6. There is only one topic with a decrease in AP larger than 0.1 and that is topic 1023 (*Yojimbo*). Although this topic should be about an information organization software for Mac OS, there is also

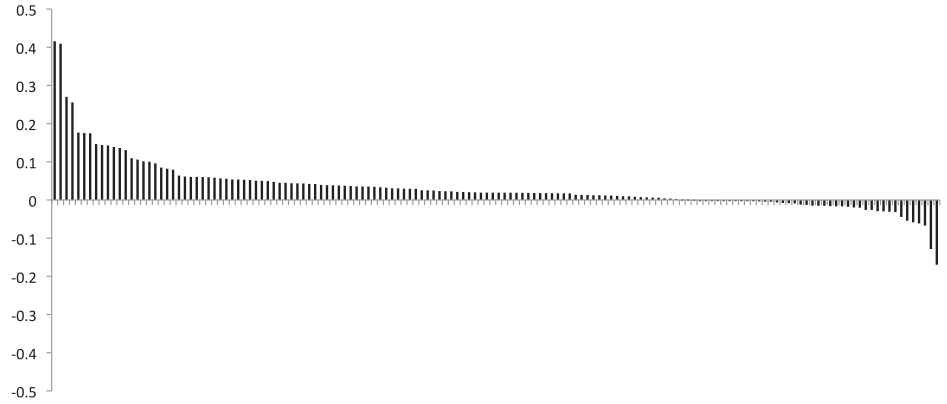

Fig. 5. Change in AP between the nonexpanded baseline and oracle EEM (P5-optimized). Topics ordered by their improvement in AP.

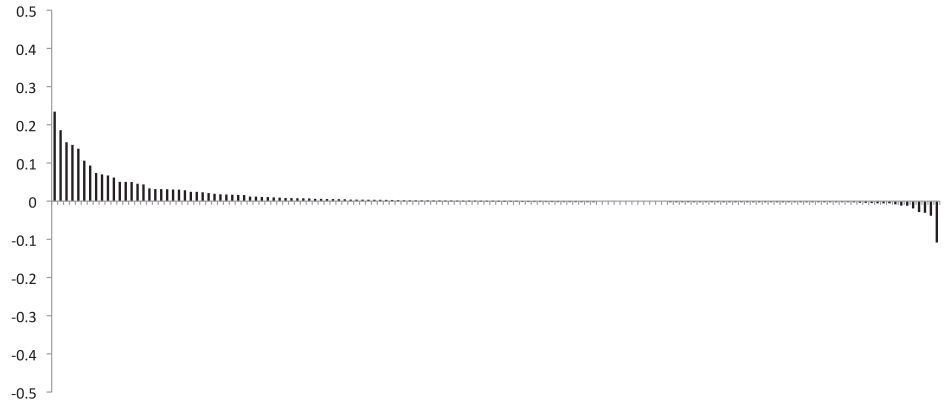

Fig. 6. Change in AP between MoRM and oracle EEM (P5-optimized). Topics ordered by their improvement in AP.

a movie with the same name. Topic drift causes this topic to drop for both MoRM and EEM when compared to the nonexpanded baseline.

On the other end of the plot, we find six topics for which the AP improvement of EEM over MoRM is more than 0.1. Most improvement is obtained for topics 1045 (*women on numb3rs*), 1031 (*sew fast sew easy*), and 923 (*challenger*). Topic 1045 benefits greatly from Wikipedia as external collection, as one could expect from a tv show topic. Terms introduced by EEM include relevant character names like *charlie*, *don*, and *eppes*, and terms that in general are related to tv shows, like *episode*, *season*, and *series*. The MoRM, on the other hand, suffers many general terms introduced by less-suitable collections, like the Web collection. Examples of these terms include *www*, *video*, and *movies*. Similar patterns can be found for the other improving topics: EEM is capable of excluding nonrelevant terms by limiting the importance of certain collections, whereas MoRM is incapable of doing so, leading to noisy query models.

Summarizing, we show that conditioning the external collection on the query is very beneficial, with large, significant improvements on all metrics. The influence of the prior probability is less significant, but can help to achieve even better performances. The final scores show not only a good performance on MAP, but also on early precision (P5 and MRR). Comparing EEM to MoRM, we find that conditioning the collection on

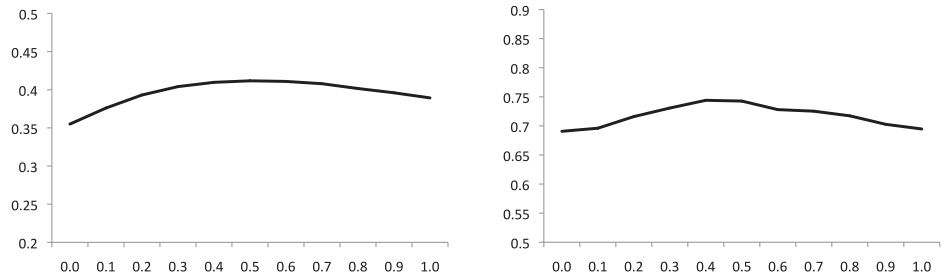

Fig. 7. Impact of parameter λ_Q (x-axis) on (left) MAP and (right) precision at 5.

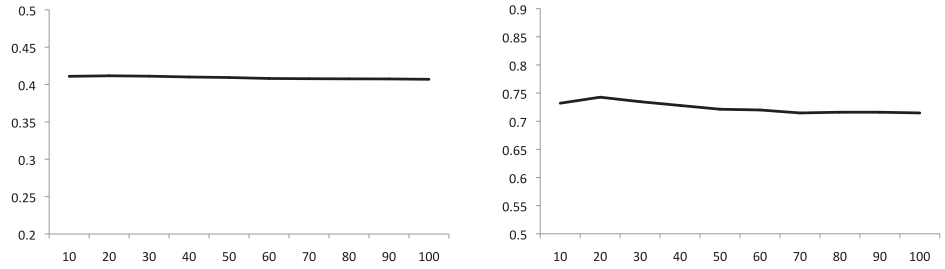

Fig. 8. Impact of parameter K, that is, the number of terms (x-axis) on (left) MAP and (right) precision at 5.

the query is beneficial. EEM: (i) limits the number of noisy terms from nonrelevant collections and (ii) gives higher weights to highly relevant terms.

8.3. Impact of Parameter Settings

In this section we touch on the impact of our model's parameters on the final results. For the experiments in this section we use our External Expansion Model run from Table V (we do not use the oracle run). First, we explore the impact of λ_Q on the performance of our model. From Eq. (6) we know that this parameter balances the original query and the expanded query and so far we used a value that gives equal weights to both parts of the query (i.e., $\lambda_Q = 0.5$). The plots in Figure 7 show how MAP and P5 performance changes when varying the value of λ_Q.

We observe that we need to mix in the original query with the expanded query to maintain good performance on MAP, since performance using low λ_Q values (e.g., 0.0 and 0.1) is worse than when we completely ignore the expanded query (i.e., $\lambda_Q = 1.0$). For precision at 5 this effect seems to have less impact, with performances for $\lambda_Q = 0.0$ and $\lambda_Q = 1.0$ being almost the same.

Moving on to the number of terms we use to expand the original query, that is, K, we explore how performances change when we use more (or fewer) terms in our expanded query model. So far we always used 20 terms in our expended query model and in this experiment we look at values for K between 10 and 100. Results are plotted in Figure 8. For this parameter we find that performance decreases in terms of retrieval effectiveness when we add more terms, although the decrease is marginal. MAP drops from 0.4117 for 20 terms to 0.4070 for 100 terms, while precision at 5 drops from 0.7427 (20 terms) to 0.7147 (100 terms). Besides that, adding more expansion terms leads to a less efficient retrieval process.

Although the overall performance is insensitive to the number of terms we use, it is likely that the optimal setting of this parameter varies per topic. Similarly, the weight of the original query (λ_Q) is dependent on the quality of the feedback terms that we

generate. Previous work in this direction explored the impact of query difficulty on query expansion [Amati et al. 2004] and on other ways to make these parameters query dependent (e.g., Lv and Zhai [2009], Sheldon et al. [2011], and He and Ounis [2009]). Although outside of the scope of this article, it is possible to implement these approaches on top of our proposed model, combining the strengths of both.

9. CONCLUSIONS

A major problem in information retrieval is the vocabulary gap between the user's information need and the relevant documents. Query modeling is a way to address these problems. In this article we have proposed a general generative query expansion model that uses external document collections for query expansion: the External Expansion Model (EEM). The main rationale behind our model is our hypothesis that each query requires its own mixture of external collections for expansion and that an expansion model should account for this. Our EEM allows for query-dependent weighing of the external collections.

We have put our model to the test on the task of blog post retrieval. This task and data are known to be a difficult environment for query expansion and we believe it makes a good setting to test our EEM. We used four external collections that represent the environment of the bloggers: (i) a news collection, (ii) Wikipedia, (iii) a Web collection, and (iv) a collection of blog posts. Following a set of experiments and an extensive analysis of the results, we have found the following.

(i) The blog post collection does perform well as the expansion corpus, especially in terms of MAP. When we require high early precision (e.g., precision at 5) Wikipedia seems to be a better choice. In general, all external collections improve over the baseline and mostly significantly so.
(ii) Our External Expansion Model works better than individual collections, especially on precision-oriented metrics, and it also outperforms the special case in which EEM boils down to Diaz and Metzler [2006]'s Mixture of Relevance Models.
(iii) The EEM does not only improve on recall-oriented metrics like MAP (which is usually the case for query expansion), but it also significantly improves early precision, which is an important metric in Web-search-related tasks.
(iv) Experiments using "oracle" runs show the full potential of our EEM, achieving very good performance on most topics. We observe that the query-dependent collection importance has more influence than the collection prior, which strengthens our belief in our model.
(v) Finally, we show that the original and expanded queries should be mixed with equal weights and that using about 20 terms in the expanded query model leads to best retrieval performance. Adding more terms, however, hardly hurts MAP, but it does hurt early precision.

We briefly go back to the Introduction and we revisit the examples of vocabulary mismatch in Table I. What do the representations of these information needs look like after applying our EEM? Here, we do not use the oracle settings, but the estimated probabilities. Table XVIII shows the new query models for these information needs.

Do the new query models close the vocabulary gap? In case of *state of the union* we find terms that point to the event (*speech*, *address*, *congress*, *united states*), to the person giving it (*president*, *george bush*, *bushs*), and to topics of the speech (*war*, *iraq*). In the second case, *shimano*, we find terms related to the cycling department of Shimano (*dura*, *ace*, *ultegra*, *deore*, *bike*, *mountain*, ...) and the fishing department (*fishing*, *reel*, *baitrunner*). In both cases the new representation of the query matches the user information need better than the original query.

Table XVIII. Query Models Constructed by Our EEM

Topic 851 *state of the union*		Topic 885 *shimano*	
union	state	shimano	dura
bush	credit	ace	road
president	address	bike	ultegra
speech	bull	mountain	deore
states	federal	faqs	fishing
united	people	speed	cycling
house	university	mtb	wheels
george	congress	xtr	tech
bushs	iraq	coasting	baitrunner
war	american	rear	reel

Note that the general model we have presented in this article is not limited to being used in social media retrieval tasks alone. Other retrieval tasks that could benefit from using external collections for query expansion can apply the same models and estimation methods. Future work is aimed at doing this, applying the External Expansion Model in other domains and tasks, like microblog search. We plan to investigate other ways of estimating the query-dependent collection importance (e.g., using supervised learning approaches), the prior collection probability, and document priors to improve the results of our model. We also plan to look at the combination of selective query expansion (deciding whether to expand the query or not) and our model to further improve retrieval performance. Finally, we aim at incorporating query-dependent ways of estimating our parameters: the number of expansion terms, the number of documents to extract terms from, and the weight of the original query.

ACKNOWLEDGMENTS

We are grateful to our reviewers and the editors of the journal for providing valuable comments and feedback, helping us to improve the quality of this article.

REFERENCES

AMATI, G., CARPINETO, C., AND ROMANO, G. 2004. Query difficulty, robustness, and selective application of query expansion. In *Proceedings of the 26th European Conference on IR Research (ECIR'04)*. Lecture Notes in Computer Science Series, vol. 2997. Springer, 127–137.

AQUAINT-2. 2007. `http://trec.nist.gov/data/qa/2007_qadata/qa.07.guidelines.html#documents`.

ARGUELLO, J., ELSAS, J., CALLAN, J., AND CARBONELL, J. 2008. Document representation and query expansion models for blog recommendation. In *Proceedings of the 2nd International Conference on Weblogs and Social Media (ICWSM'08)*. AAAI Press.

BAEZA-YATES, R. AND RIBEIRO-NETO, B. 2011. *Modern Information Retrieval: The Concepts and Technology behind Search*. Addison-Wesley Professional.

BALOG, K., WEERKAMP, W., AND DE RIJKE, M. 2008. A few examples go a long way: Constructing query models from elaborate query formulations. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*. ACM, New York, 371–378.

BALOG, K., MEIJ, E., WEERKAMP, W., HE, J., AND DE RIJKE, M. 2009. The University of Amsterdam at TREC 2008: Blog, enterprise, and relevance feedback. In *Proceedings of the 17th Text REtrieval Conference (TREC'08)*. NIST.

CAO, G., NIE, J.-Y., GAO, J., AND ROBERTSON, S. 2008. 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)*. ACM, New York, 243–250.

CARTRIGHT, M.-A., SEO, J., AND LEASE, M. 2009. UMass amherst and UT austin @ the TREC 2009 relevance feedback track. In *Proceedings of the 18th REtrieval Conference (TREC'09)*. NIST.

CLUEWEB09. 2009. http://www.lemurproject.org/clueweb09.php/.

CRONEN-TOWNSEND, S., ZHOU, Y., AND CROFT, W. B. 2004. A framework for selective query expansion. In *Proceeding of the 13th ACM International Conference on Information and Knowledge Management (CIKM'04)*. ACM, New York, 236–237.

DIAZ, F. AND METZLER, D. 2006. Improving the estimation of relevance models using large external corpora. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*. ACM, New York, 154–161.

DONG, A., CHANG, Y., ZHENG, Z., MISHNE, G., BAI, J., ZHANG, R., BUCHNER, K., LIAO, C., AND DIAZ, F. 2010. Towards recency ranking in web search. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM'10)*. ACM, New York, 11–20.

ELSAS, J., ARGUELLO, J., CALLAN, J., AND CARBONELL, J. 2008a. Retrieval and feedback models for blog distillation. In *Proceedings of the 16th Text REtrieval Conference (TREC'07)*. NIST.

ELSAS, J. L., ARGUELLO, J., CALLAN, J., AND CARBONELL, J. G. 2008b. Retrieval and feedback models for blog feed search. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*. ACM, New York, 347–354.

ERNSTING, B. J., WEERKAMP, W., AND DE RIJKE, M. 2008. The university of amsterdam at the TREC 2007 blog track. In *Proceedings of the 16th Text REtrieval Conference (TREC'07)*. NIST.

FAUTSCH, C. AND SAVOY, J. 2009. UniNE at TREC 2008: Fact and opinion retrieval in the blogsphere. In *Proceedings of the 17th Text REtrieval Conference (TREC'08)*. NIST.

HAWKING, D., BAILEY, P., AND CRASWELL, N. 1999. ACSys TREC-8 experiments. In *Proceedings of the 8th Text REtrieval Conference (TREC'99)*. NIST.

HE, B. AND OUNIS, I. 2007. Combining fields for query expansion and adaptive query expansion. *Inf. Process. Manag. 43,* 5, 1294–1307.

HE, B. AND OUNIS, I. 2009. Finding good feedback documents. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM'09)*. ACM, New York, 2011–2014.

HIEMSTRA, D. 2001. Using language models for information retrieval. Ph.D. thesis, University of Twente.

HOFMANN, K. AND WEERKAMP, W. 2008. Content extraction for information retrieval in blogs and intranets. Tech. rep., University of Amsterdam, ISLA.

JAVA, A., KOLARI, P., FININ, T., JOSHI, A., AND MARTINEAU, J. 2007. The blogvox opinion retrieval system. In *Proceedings of the 15th Text REtrieval Conference (TREC'06)*. NIST.

JIJKOUN, V., DE RIJKE, M., AND WEERKAMP, W. 2010. Generating focused topic-specific sentiment lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 585–594.

KAMPS, J., DE RIJKE, M., AND SIGURBJÖRNSSON, B. 2004. Length normalization in XML retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*. ACM, New York, 80–87.

KURLAND, O., LEE, L., AND DOMSHLAK, C. 2005. Better than the real thing?: Iterative pseudo-query processing using cluster-based language models. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*. ACM, New York, 19–26.

KWAK, H., LEE, C., PARK, H., AND MOON, S. 2010. What is twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web (WWW'10)*. ACM, New York, 591–600.

KWOK, K. L., GRUNFELD, L., DINSTL, N., AND CHAN, M. 2001. TREC-9 cross language, web and question-answering track experiments using PIRCS. In *Proceedings of the 9th Text REtrieval Conference (TREC-9)*. NIST.

LAFFERTY, J. AND ZHAI, C. 2003. Probabilistic relevance models based on document and query generation. In *Language Modeling for Information Retrieval*, Kluwer International Series on Information Retrieval, Springer.

LANCASTER, F. 1968. *Information Retrieval Systems: Characteristics, Testing and Evaluation*. John Wiley & Sons, New York.

LAVRENKO, V. AND CROFT, W. B. 2001. Relevance based language models. In *Proceedings of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*. ACM, New York, 120–127.

LESKOVEC, J., BACKSTROM, L., AND KLEINBERG, J. 2009. Meme-Tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09)*. ACM, New York, 497–506.

LV, Y. AND ZHAI, C. 2009. Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM'09)*. ACM, New York, 255–264.

MACDONALD, C. AND OUNIS, I. 2006. The TREC blogs06 collection: Creating and analyzing a blog test collection. Tech. rep. TR-2006-224, Department of Computer Science, University of Glasgow.

MACDONALD, C., OUNIS, I., AND SOBOROFF, I. 2008. Overview of the TREC 2007 blog track. In *Proceedings of the 16th Text REtrieval Conference (TREC'07)*. NIST.

MANNING, C. D., RAGHAVAN, P., AND SCHÜTZE, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

MEIJ, E., WEERKAMP, W., AND DE RIJKE, M. 2009. A query model based on normalized log-likelihood. In *Proceeding of the 18th ACM Conference on Information and Knowledge Managemnt (CIKM'09)*. ACM, New York, 1903–1906.

METZLER, D. AND CROFT, W. B. 2005. A markov random field model for term dependencies. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*. ACM, New York, 472–479.

MILLER, D., LEEK, T., AND SCHWARTZ, R. 1999. A hidden markov model information retrieval system. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)*. ACM, New York, 214–221.

MISHNE, G. AND DE RIJKE, M. 2006. A study of blog search. In *Proceedings of the 28th European Conference on IR Research (ECIR'06)*. Lecture Notes in Computer Science, vol. 3936, Springer, 289–301.

OUNIS, I., DE RIJKE, M., MACDONALD, C., MISHNE, G., AND SOBOROFF, I. 2007. Overview of the TREC-2006 blog track. In *Proceedings of the 15th Text REtrieval Conference (TREC'06)*. NIST.

OUNIS, I., MACDONALD, C., AND SOBOROFF, I. 2009. Overview of the TREC-2008 blog track. In *Proceedings of the 17th Text REtrieval Conference (TREC'08)*. NIST.

PONTE, J. M. AND CROFT, W. B. 1998. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*. ACM, New York, 275–281.

QIU, Y. AND FREI, H.-P. 1993. Concept based query expansion. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93)*. ACM, New York, 160–169.

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

SAKAI, T. 2002. The use of external text data in cross-language information retrieval based on machine translation. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC'02)*. IEEE, 6–9.

SHELDON, D., SHOKOUHI, M., SZUMMER, M., AND CRASWELL, N. 2011. Lambdamerge: Merging the results of query reformulations. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM'11)*. ACM, New York, 795–804.

TAO, T. AND ZHAI, C. 2006. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*. ACM, New York, 162–169.

WEERKAMP, W. 2011. Finding people and their utterances in social media. Ph.D. thesis, University of Amsterdam.

WEERKAMP, W. AND DE RIJKE, M. 2008. Credibility improves topical blog post retrieval. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 923–931.

WEERKAMP, W. AND DE RIJKE, M. 2012. Credibility-Inspired ranking for blog post retrieval. *Inf. Retriev. J. 15,* 3–4, 243–277.

WESTERVELD, T., KRAAIJ, W., AND HIEMSTRA, D. 2002. Retrieving web pages using content, links, urls and anchors. In *Proceedings of the 10th REtrieval Conference (TREC'01)*. NIST.

XU, Y., JONES, G. J., AND WANG, B. 2009. Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09)*. ACM, New York, 59–66.

YAN, R. AND HAUPTMANN, A. 2007. Query expansion using probabilistic local feedback with application to multimedia retrieval. In *Proceeding of the 16th ACM International Conference on Information and Knowledge Management (CIKM'07)*. ACM, New York, 361–370.

YIN, Z., SHOKOUHI, M., AND CRASWELL, N. 2009. Query expansion using external evidence. In *Proceedings of the 31st European Conference on IR Research (ECIR'09)*. Lecture Notes in Computer Science, vol. 5478. Springer, 362–374.

ZHANG, W. AND YU, C. 2007. UIC at TREC 2006 blog track. In *Proceeding of the 15th Text REtrieval Conference (TREC'06)*. NIST.

Received June 2011; revised March 2012; accepted July 2012