Document Expansion with External Collections for Information Retrieval

Garrick Sherman

September 15, 2016

Abstract

1 Introduction

When using language models (LMs) to perform document retrieval, we treat the query and document texts as observations from unseen (and unobservable) probabilistic processes. In the query likelihood (QL) retrieval model, we rank each document by the likelihood that its language model would "generate" the query text. However, if the query is a sample from an underlying stochastic process, it is generally too sparse to fully represent an informtion need.

Relevance modeling is an extremely influential pseudo-relevance feedback technique that helps to allievate the problem of vocabulary mismatch. Instead of assuming that queries are observations sampled from *document* models, we assume that both queries and documents are observations sampled from a *relevance* model (RM) [5]. We can use this idea to expand the query representation into an estimate of the relevance model (see Section 3.1.2 for more details) and rank documents by the similarity of their estimated language model to the estimated RM.

As a form of pseudo-relevance feedback, part of the relevance modeling process involves assuming that top-ranked documents are relevant to the query. Although this assumption is not uniformly true, it is true of enough top-ranked documents that the estimated RM will be of good quality. That is, initially retrieved documents provide a good estimate of the underlying relevance model *in their aggregate*; taken individually, documents are not assumed to accurately represent the underlying RM.

Relevance modeling makes use of a query likelihood weight, $P(Q|D)$, which helps to quantify the degree to which each document should contribute to the final RM. However, given that individual documents are not considered accurate representations of the relevance model R, we may reasonably be concerned that our estimate of $P(Q|D)$ is the result of chance. This concern applies to both high and low estimates of $P(Q|D)$, and, if true, we would expect this problem to affect the quality of our relevance models. We suggest that this may be solved by expanding documents: adding terms to a document should improve our estimates of the underlying language model and in turn improve the quality of retrieval results and relevance models based on these expanded documents.

In this paper, we propose a method of document expansion that closely mirrors the process of relevance modeling. This document expansion process serves many of the same roles as traditional relevance modeling without excluding the use of RMs; indeed, our document expansion process is complementary to query expansion. However, the concerns that lead us to expand documents in the first place—potentially inaccurate term probability estimates drawn from top-ranked documents—also apply to our document expansion procedure.

To help alleviate this, we introduce a general method of incorporating external document collections into the document expansion process. Our idea is that, while documents in individual collections may suffer from fairly high term frequency variance and other forms of bias, documents from multiple collections will likely average out to create good general purpose estimates. Specifically, we use Wikipedia as our external collection. We believe that, as a general encyclopedia, Wikipedia is most likely to provide unbiased language

model estimates; that is, estimates of $P(Q|D)$ calculated where D is from Wikipedia seem likely to exhibit minimal variance.

2 Related Work

2.1 Document Expansion in IR

Document expansion has been well studied in information retrieval literature, e.g. [6, 8, 3, 7]. For example, Liu & Croft propose a method of retrieval that uses document clusters to smooth document language models [6]. Wei & Croft propose a similar style of document expansion by smoothing term probabilities in documents with those in latent Dirichlet allocation topics [9]. Even more pertinent to our work is that of Tao et al., who propose a similar approach to that of Liu & Croft, but make each document the center of its own cluster, which helps to ensure that the expansion documents are as closely as related to the target document as possible [8].

Most similar to our approach is that of Efron, Organisciak & Fenlon [2], who issue very short microblog documents as pseudo-queries and use the retrieved documents for expansion. Their approach relates closely to that of relevance modeling [5]. We take this work as the starting point for our own and explore its application and adaptation to different scenarios. Firstly, Efron, Organisciak & Fenlon are concerned with microblog retrieval, in which documents are extremely short—perhaps as small as a keyword query. In contrast, we are interested in performing document expansion with more typical full-length documents, such as those found in news and web corpora. Secondly, while their work used only the target document collection, we propose an expansion of their method that allows for multiple expansion corpora. Finally, we investigate pairing document expansion with query expansion.

2.2 Incorporating External Collections

The incorporation of external collections into document retrieval is a similarly common trope in IR literature. One prior study with relevance to ours is that of Diaz & Metzler, whose mixture of relevance models bears a resemblance to our Eq. 7 [1]; of course, our approach is qualitatively different since Diaz & Metzler were interested in query expansion while we apply the technique to document expansion.

3 Document Expansion Procedure

3.1 Underlying Retrieval Model

Throughout this paper we rely on the language modeling retrieval framework [4], though this is not strictly necessary and imposes no particular mathematical constraints on our approach. More specifically, we employ the query likelihood (QL) and relevance modeling ranking methods.

3.1.1 Query Likelihood

Given a query Q and a document D, we rank documents on $P(Q|\theta_D)$, where θ_D is the language model (typically a multinomial over the vocabulary V) that generated the text of document D. Assuming independence among terms and a uniform distribution over documents, each document is scored by

$$\log P(Q|D) = \prod_{w \in Q} P(w|Q) \cdot \log P(w|\theta_D). \tag{1}$$

We follow standard procedures for estimating the probabilities in Eq. 1. We simply use the maximum likelihood estimate of $\hat{P}(w|Q) = \frac{c(w,Q)}{|Q|}$ where $c(w, Q)$ is the frequency of word w in Q. For $P(w|\theta_D)$ we estimate a smoothed language model by assuming that document language models in a given collection have a Dirichlet prior distribution:

$$\hat{P}(w|\theta_D) = \frac{c(w, D) + \mu \hat{P}(w|C)}{|D| + \mu} \tag{2}$$

where $\hat{P}(w|C)$ is the maximum likelihood estimate of the probability of seeing word w in a "background" collection C (typically C is the corpus from which D is drawn), and $\mu \geq 0$ is the smoothing hyper-parameter.

3.1.2 Relevance Modeling

Relevance modeling is a form of pseudo-relevance feedback that uses top ranked documents to estimate a language model representing documents relevant to a query [5]. This language model, known as a relevance model, acts as a form of query expansion and is generally used with the KL divergence retrieval model [10] to score documents.

A relevance model takes the form of

$$P(w|R) = \sum_{D \in C} P(D)P(w|D)P(Q|D) \tag{3}$$

where $P(Q|D)$ is calculated as in Equation 1 and essentially weights word w in document D by the query likelihood of the document. Though theoretically calculated over all words and documents, relevance models are more efficient and robust when calculated over only the top terms in only the top ranked documents. These parameters are referred to as $fbTerms$ and $fbDocs$ respectively in Table 1.

Because relevance models are prone to query drift, research has shown that linearly interpolating a relevance model with the original query model to improve performance:

$$P(w|Q') = (1 - \alpha)P(w|R) + \alpha P(w|Q). \tag{4}$$

α is a mixing parameter controlling the influence of the original query. This form of relevance model is known as "RM3" and is the baseline used throughout this paper.

3.2 Expanding with Document Pseudo-Queries

To expand a document D, we begin by treating the text of D as a pseudo-query which we pose against a collection of documents C_E. To transform a document into a pseudo-query we apply two transformations. First we remove all terms from D that appear in the standard Indri stoplist[1]. Next, we prune our pseudo-query by retaining only the $0 < k \leq |D|$ most frequent words in the stopped text of D. The integer variable k is a parameter that we choose empirically. Let Q_D be the pseudo-query for D, consisting of the text of D after our two transformations are applied.

We obtain a ranking of related documents, which we call expansion documents, by running Q_D over an index C_E. More formally, we rank the documents in C_E against D using Eq. 1, substituting Q_D for the query and E_i—the text of the i^{th} expansion document—for the document. Let π_i be the log-probability for expansion document E_i with respect to D given by Eq. 1.

We now have a ranked list of tuples $\{(E_1, \pi_1), (E_2, \pi_2), ..., (E_N, \pi_N)\}$ relating expansion document E_i to D with log-probability π_i. We take the top n documents where $0 \leq n \leq N$. We call these top documents $\mathcal{E}_D$ and designate them as our expansion documents for D. Finally, we exponentiate each π_i and normalize our retrieval scores so they sum to 1 over the n retained documents. Assuming a uniform prior over documents, we now have a probability distribution over our n retained documents: $P(E|D)$.

Since this procedure does not depend on the query, we may compute $\mathcal{E}_D$ once at indexing time and reuse our expansion documents across queries.

[1] http://www.lemurproject.org/stopwords/stoplist.dft

4 Document Expansion Retrieval Model

We would now like to incorporate our expansion documents into a retrieval model over documents. We assume that a query is generated by a mixture of the original document language model θ_D and a language model θ_E representing the expansion documents. We assume that θ_E can be estimated using the text of the expansion documents $\mathcal{E}_D$. This mixture model may be expressed as:

$$\hat{P}^{\lambda}(Q|D) = \prod_{i=1}^{|Q|}(1-\lambda)P(q_i|D) + \lambda P(q_i|\mathcal{E}_D) \tag{5}$$

with $0 \leq \lambda \leq 1$. The larger λ is, the more we believe that the expansion documents are a good representation of Q and the less we believe that the original document represents the "true" language model of Q. We estimate $P(q_i|\mathcal{E}_D)$ in expectation:

$$P(q_i|\mathcal{E}_D) = \sum_{E \in \mathcal{E}_D} P(q_i|E)P(E|D). \tag{6}$$

Like $P(q_i|D)$, we estimate the probability of q_i given expansion document E, $P(q_i|E)$, as a Dirichlet-smoothed query likelihood. By virtue of our expansion document scoring and normalization, we also have $P(E|D)$.

4.1 Combining Evidence

We investigate the use of multiple collections for document expansion. We expect a more diverse set of expansion documents to result in more robust (less biased) language models at the conclusion of the expansion process.

To expand a document D given a set of collections $C = \{C_1, C_2, ..., C_n\}$, we first construct a set of expansion documents $\mathcal{E}_{Di}$ using C_i in the manner described in Section 3. We may incorporate these multiple sets of expansion documents by simply altering Eq. 5 to sum over $\mathcal{E}_D$ terms:

$$\hat{P}^{\lambda}(Q|D) = \prod_{i=1}^{|Q|}(1 - \sum_{j=1}^{n}\lambda_{\mathcal{E}_{Dj}})P(q_i|D) + \sum_{j=1}^{n}\lambda_{\mathcal{E}_{Dj}}P(q_i|\mathcal{E}_{Dj}) \tag{7}$$

where $0 \leq \sum_{j=1}^{n}\lambda_{\mathcal{E}_{Dj}} \leq 1$. As before, each value of $\lambda_{\mathcal{E}_{Dj}}$ reflects our confidence in that set of expansion documents accurately representing the query language model.

5 Evaluation

5.1 Data

Although Eq. 7 allows for an arbitrary number of collections, for now we limit ourselves to two: the collection that the document appears in and Wikipedia. To this end, we make use of the September 1, 2015 dump of English Wikipedia. We build an Indri[2] index over the Wikipedia page text. The text of each page serves as the "description text" used in Eq. 6.

We test our approach using several TREC datasets:

- The **AP** newswire collection from TREC disks 1 and 2 with topics 101-200.
- The **robust** 2004 topics, numbering 250, from TREC disks 4 and 5.
- The **wt10g** collection with the 100 topics from the 2000 and 2001 TREC Web tracks.
- Topics 851-900 with the **blogs06** dataset.
- The **clueweb09** category B dataset with likely spam documents removed, topics 1-200.

These datasets were chosen because they provide a good range of collection types, from relatively homogeneous and small with well-formed documents (AP) to heterogeneous and large with varied document quality (clueweb09).

[2]http://www.lemurproject.org/indri/

5.2 Runs

We produce four runs per collection:

- *baseline-ql*, a baseline query likelihood run
- *baseline-rm*, a baseline RM3 run with optimal α
- *baseline-rm-cv*, a baseline RM3 run with 10-fold cross validation
- *exp-ql*, incorporating expansion documents using Eq. 5

We remove stop words in documents and entity descriptions for all runs. For the *exp* runs, we retrieve the top 1000 documents per query using the default Indri query likelihood implementation. We then re-rank these documents by incorporating their knowledge base links as described in Section 4.

5.3 Parameters

Param	Meaning	Value
k	The maximum number of document terms to use in constructing Q_D.	20
n	The maximum number of expansion documents in $\mathcal{E}_D$.	10
$\lambda_{\mathcal{E}_D}$	One of several related mixing parameters controlling the weights of $P(q\|D)$ and $P(q\|\mathcal{E}_D)$	0.0-1.0
μ	Used for Dirichlet smoothing of both $P(q\|D)$ and $P(q\|E)$.	2500
$fbDocs$	The number of feedback documents to use for RM3 runs.	20
$fbTerms$	The number of terms per document to use for RM3 runs.	20
α	Mixing parameter controlling the weights of the original query and relevance model for RM3 runs.	0.0-1.0

Table 1: Parameter settings for the document expansion procedure and retrieval model

The various parameters required for our approach, along with their meanings and the values used in our experiments, are shown in Table 1.

For this work, we set k heuristically. In principle, this parameter need not be limited beyond the length of the document; however, this would increase computation time significantly, so we have opted to set it to the specified value. The parameter n is similarly set heuristically to 10 expansion documents; see Section 6.1 for a discussion of the performance sensitivity to the setting of n.

To set the two $\lambda_{\mathcal{E}_D}$ values, we use 10-fold cross validation. This entails splitting each set of topics into ten random groups (folds) and using nine of them in concert with relevance judgments to determine optimal parameter settings, which we test using the tenth. By using each of the ten folds as the test fold once, we produce results that fully cover the topic set; we combine these results in order to perform batch evaluation.

6 Results

Retrieval performance of the baselines and top-scoring runs are shown in Table 2. Mean average precision (MAP) and normalized discounted cumulative gain at 20 (nDCG@20) scores marked with $\uparrow$ are greater than the optimal RM3 run with statistical significance at $p < 0.05$ using a paired one-tailed t-test. We also run a more realistic RM3 run using 10-fold cross validation to set the mixing parameter α. Scores marked with $\Uparrow$

Collection	Run	MAP	nDCG@20
AP	*baseline-ql*	0.2337	0.4170
	baseline-rm	0.3298 ($\alpha = 0.2$)	0.4852 ($\alpha = 0.3$)
	baseline-rm-cv	0.3291	0.4858
	exp-ql	0.2799	0.4687
Robust	*baseline-ql*	0.2185	0.3825
	baseline-rm	0.2349 ($\alpha = 0.2$)	0.3893 ($\alpha = 0.6$)
	baseline-rm-cv	0.2663	0.3940
	exp-ql	0.2420	$0.4066^{\uparrow}$
wt10g	*baseline-ql*	0.1687	0.2753
	baseline-rm	0.1735 ($\alpha = 0.7, 0.8$)	0.2738 ($\alpha = 1.0$)
	baseline-rm-cv	0.1683	0.2633
	exp-ql	0.1836	$0.3181^{\uparrow\Uparrow}$
blogs06	*baseline-ql*	0.3064	0.2415
	baseline-rm	0.3103 ($\alpha = 0.7$)	0.2431 ($\alpha = 1.0$)
	baseline-rm-cv	0.3064	0.2330
	exp-ql	0.2990	0.2343
clueweb09	*baseline-ql*	0.1117	0.1944
	baseline-rm	0.1163 ($\alpha = 0.2$)	0.2034 ($\alpha = 0.7$)
	baseline-rm-cv	??	??
	exp-ql	$0.1183^{\uparrow}$	0.2100

Table 2: The top-scoring runs and baselines by MAP. $\uparrow$ indicates statistically significant improvements over the oracle RM3 baseline, *baseline-rm*, while $\Uparrow$ indicates statistically significant improvement and decrease compared to the cross validated RM3 baseline, *baseline-rm-cv*. Significance at $p < 0.05$.

signal statistically significant increases in performance relative to this cross validated RM3 run. Note that baselines correspond to $\sum_{j=1}^{n} \lambda_{\mathcal{E}_{D\,j}} = 0.0$.

6.1 Sensitivity to n

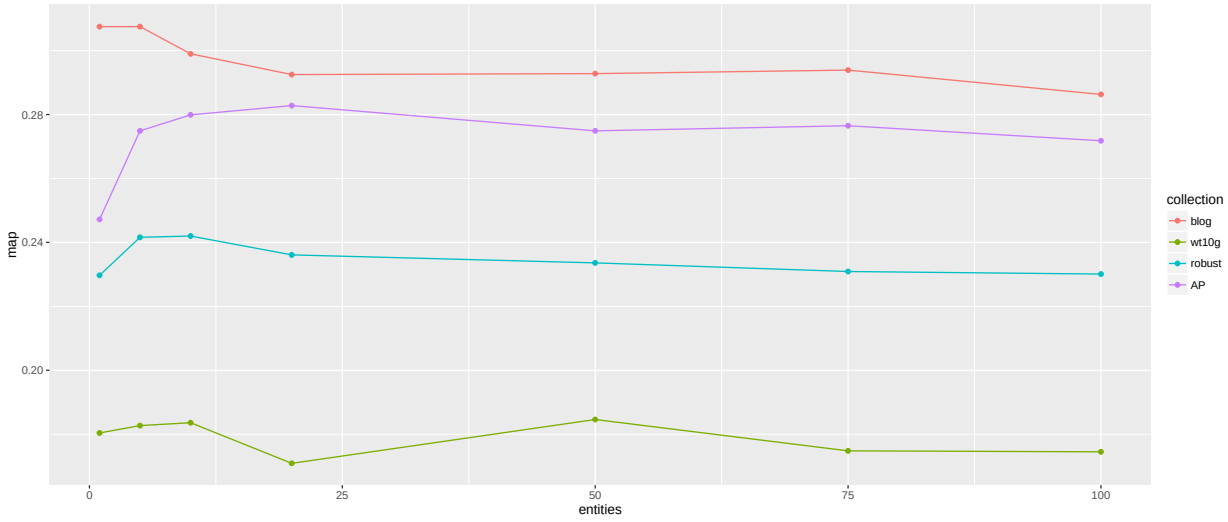

Figure 1: Sweeps over values of n, the number of entities, for each *exp-ql* run.

Similar to k, the value of n may theoretically equal the number of expansion documents available, i.e. $|C_E|$; in practice, of course, this is infeasible, and n will be substantially smaller. We must be careful in setting the value of n for computational efficiency reasons: if n is set high, we must process more documents. Figure 1 shows sweeps over several values of n for each target collection using our established 10-fold cross validation procedure with identical folds. Happily, the performance of our approach is not sensitive to the setting of n, and lower values generally perform as well or better than higher values. We therefore choose the value of 10 as an apparently safe value; this is a convenient result since it allows for more efficient document expansion.

6.2 Is Expansion Worthwhile?

	Self		Wikipedia	
Collection	λ	**MAP**	λ	**MAP**
AP	?	?	0.2	0.2662
Robust	?	?	0.2	0.2354
wt10g	?	?	?	?
blogs06	?	?	?	?
clueweb09	?	?	?	?

Table 3: The top-scoring runs by MAP using only one collection of expansion documents. "Self" indicates expansion using the document's original collection while "Wikipedia" indicates expansion using Wikipedia.

7 Conclusions

The results indicate that our approach for expanding documents using Wikipedia and the document's original collection produces useful data for document retrieval purposes. Our simple document expansion model performs well compared to both a query likelihood and RM3 baseline.

In this paper, we have limited ourselves to using only two collections. However, future work may benefit from incorporating more collections to improve our language model estimates. Since our retrieval model performs *document* expansion, we also plan to investigate its utility when paired with *query* expansion techniques that employ knowledge base links.

8 Acknowledgments

This work was supported in part by the US National Science Foundation under Grant No. [blind]. Any opinions, findings, conclusions, or recommendations expressed are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] F. Diaz and D. Metzler. Improving the estimation of relevance models using large external corpora. In *SIGIR '06*, pages 154–161, 2006.

[2] M. Efron, P. Organisciak, and K. Fenlon. Improving retrieval of short texts through document expansion. In *SIGIR '12*, pages 911–920, 2012.

[3] G. Hubert and J. Mothe. An adaptable search engine for multimodal information retrieval. *JASIS*, 60(8):1625–1634, 2009.

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

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

[6] X. Liu and W. B. Croft. Cluster-based retrieval using language models. In *SIGIR '04*, pages 186–193, 2004.

[7] A. Singhal and F. Pereira. Document expansion for speech retrieval. In *SIGIR '99*, pages 34–41, 1999.

[8] T. Tao, X. Wang, Q. Mei, and C. Zhai. Language model information retrieval with document expansion. In *NAACL '06*, pages 407–414, 2006.

[9] X. Wei and W. B. Croft. LDA-based document models for ad-hoc retrieval. In *SIGIR '06*, pages 178–185, 2006.

[10] C. Zhai and J. D. Lafferty. A risk minimization framework for information retrieval. *Inf. Process. Manage.*, 42(1):31–55, 2006.