A Query Model Based on Normalized Log-Likelihood

Edgar Meij
edgar.meij@uva.nl

Wouter Weerkamp
w.weerkamp@uva.nl

Maarten de Rijke
mdr@science.uva.nl

ISLA, University of Amsterdam
Science Park 107, 1098 XG Amsterdam

ABSTRACT

Leveraging information from relevance assessments has been proposed as an effective means for improving retrieval. We introduce a novel language modeling method which uses information from each assessed document and their aggregate. While most previous approaches focus either on features of the entire set or on features of the individual relevant documents, our model exploits features of both the documents and the set as a whole. When evaluated, we show that our model is able to significantly improve over state-of-art feedback methods.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval—*Retrieval Models*

General Terms

Algorithms, Theory, Experimentation, Measurement

Keywords

Language modeling, Query models, Relevance feedback

1. INTRODUCTION

A query is usually a brief, sometimes imprecise expression of an underlying information need [19]. Examining how queries can be *transformed* to equivalent, potentially better queries is a theme of recurring interest to the information retrieval community. Such transformations include expansion of short queries to long queries, paraphrasing queries using an alternative vocabulary, mapping unstructured queries to structured ones [14], identifying key concepts in verbose queries [3], etc.

To inform the transformation process, multiple types of information sources have been considered. A recent one is search engine logs for query substitutions [20]. Another recent example is where users complement their traditional keyword query with additional information, such as example documents [2], tags [6], images [7], categories [21], or their search history [1]. The ultimate source of information for transforming a query, however, is the user, through relevance feedback [16, 17]: given a query and a set of judged documents for that query, how does a system take advantage of the judgments in order to transform the original query and retrieve more documents that will be useful to the user? As demonstrated by the recent launch of a dedicated relevance feedback track at TREC [4], we still lack the definitive answer to this question.

Let's consider an example to see what aspects play a role in transforming a query based on judgments for a set of initially retrieved documents. Suppose we have a set of documents which are judged to be relevant to a query. These documents may vary in length and, furthermore, they need not be completely on topic because they may discuss more topics than the ones that are relevant to the query. While the users' judgments are at the document level, not all of the documents' sections can be assumed to be equally relevant. Most relevance feedback models that are currently available do not model or capture this phenomenon; instead, they attempt to transform the original query based on the full content of the documents. Clearly this is not ideal and we would like to account for the possibly multi-faceted character of documents. We hypothesize that a relevance feedback model that attempts to capture the topical structure of individual judged documents ("For each judged document, what is important about it?") as well as of the set of all judged documents ("Which topics are shared by the entire set of judged documents?") will outperform relevance feedback models that capture only one of these types of information.

We are working in a language modeling (LM) setting and our aim in this paper is to present an LM-based relevance feedback model that uses both types of information—about the topical relevance of a document and about the general topic of the set of relevant documents— to transform the original query. The proposed model uses the whole set of relevance assessments to determine how much each document that has been judged relevant should contribute to the query transformation. We use the TREC relevance feedback track test collection to evaluate our model and compare it to other, established relevance feedback methods. We show that it is able to achieve superior performance over all evaluated models. We answer the following two research questions in this paper. (i) Can we develop a relevance feedback model that uses evidence from both the individual relevant documents and the set of relevant documents as a whole? (ii) Can our new model achieve state-of-the-art results and how do these results compare to related models?

CIKM'09, November 2–6, 2009, Hong Kong, China.

The remainder of this paper is organized as follows. In Section 2 we recall some basic facts from the language modeling approach to information retrieval. Building on this, we introduce our new feedback model in Section 3. In Section 4 we detail the set-up of the experiments. We report on the outcomes in Section 5 and we end with a concluding section.

2. BACKGROUND

In this section we recall basic notions from the language modeling approach to information retrieval. In the approach based on multinomial unigram models [8], each document D is represented as a multinomial probability distribution $P(t|\theta_D)$, assuming term independence. At retrieval time, each document is ranked according to the likelihood of having generated the query Q. Building on this basic set-up, several authors proposed the use of the Kullback-Leibler divergence measure for ranking [10, 15]. Using KL-divergence, documents are scored by measuring the divergence between a query model θ_Q and each document model θ_D. Since we want to assign scores proportional to their similarity, the KL-divergence is negated for ranking purposes:

$$\begin{aligned} \text{Score}(Q,D) &= -\text{KL}(\theta_Q||\theta_D) \\ &= -H(\theta_Q,\theta_D) + H(\theta_Q) \\ &\overset{rank}{=} -\sum_{t\in\mathcal{V}} P(t|\theta_Q)\log P(t|\theta_D). \end{aligned} \quad (1)$$

Note that the sum is over the vocabulary $\mathcal{V}$ although terms which do not appear in Q have $P(t|\theta_Q) = 0$. $H(\theta_Q,\theta_D)$ is the cross-entropy of the query model and the document model and $H(\theta_Q)$ is the entropy of the query, a query specific constant that can be ignored for ranking. When the query model θ_Q is estimated using the maximum-likelihood estimate, i.e., when $P(t|\theta_Q) = P(t|\tilde{\theta}_Q) = n(t,Q)/|Q|$, it can be shown that documents are ranked in the same order as using the query likelihood (QL). Thus, we will refer to this basic model as QL in the remainder of this paper.

3. RELEVANCE FEEDBACK MODEL

In this section we introduce our relevance feedback model based on normalized log-likelihood. The goal of a relevance feedback algorithm is, given a query and a set of judged documents, to transform the original query and return more documents that will be useful to the user. Most relevance feedback approaches for LM use an improved estimate or estimation method for the query model to incorporate relevance feedback information. Typically, the initial query is mixed with an expanded part θ_R, which is a distribution over terms that represents the outcome of a transformation of the initial query [2, 13, 16, 22]. This mixture is usually modeled as a linear interpolation, effectively reweighing the initial query terms and providing smoothing: $P(t|\theta_Q) = (1-\lambda_Q)P(t|\tilde{\theta}_Q) + \lambda_Q P(t|\theta_R)$.

If we were to have an infinite number of relevance judgments from the user and, hence, could fully enumerate the documents relevant to a query, we could simply use the empirical estimate of the terms in those documents to estimate θ_R. Furthermore, given all sources of information available to the system (query, assessments, and documents in the collection), the parameters of this model would fully describe the information need from the system's point of view. Assuming independence between terms, the joint likelihood of observing the terms given θ_R under this model is:

$$P(t_1,\ldots,t_{|\mathcal{V}|}|\theta_R) = \prod_{i=1}^{|\mathcal{V}|} P(t_i|R), \quad (2)$$

where R denotes the set of relevant documents. Then, we can use a simple maximum likelihood estimate to obtain

$$P(t|R) = \frac{\sum_{D\in R} n(t,D)}{\sum_{D\in R}\sum_{t'} n(t',D)}, \quad (3)$$

where $n(t,D)$ indicates the count of t in D. Later, we will refer to this model as MLE. In contrast with this approach, however, a typical search engine user would not provide an infinite amount of data and only arrive at judgements on the relevance status of a small number of documents [18]. Even in larger-scale TREC evaluations, the number of assessments per query is still a fraction of the total number of documents in the collection [5]. So in any realistic scenario, the relevance of all remaining, non-judged documents is unknown and this fact jeopardizes the confidence we can put in the MLE model to accurately estimate θ_R.

Moreover, as we pointed out in the introduction, not every document in R is necessarily entirely relevant to the information need. Ideally, we would like to weigh documents according to their "relative" level of relevance. As such, each relevant document should be considered as a separate piece of evidence towards the estimation of θ_R, instead of assuming full independence between documents as in Eq. 3.

Let's consider the following sampling process to substantiate this intuition. We pick a relevant document according to some probability and then select a term from that document. Assuming that each term is generated independently of θ_R once we pick a relevant document, the probability of randomly picking a document and then observing t is $P(t,D|\theta_R) = P(D|\theta_R)P(t|D)$. Then, the overall probability of observing all terms can be expressed as a sum of the marginals:

$$P(t_1,\ldots,t_{|\mathcal{V}|}|\theta_R) = \sum_{D\in R} P(D|\theta_R)\prod_{i=1}^{|\mathcal{V}|} P(t_i|\theta_D). \quad (4)$$

The key term here is $P(D|\theta_R)$; it conveys the level of relevance of D. While we know that every $D \in R$ is relevant, we posit that documents that are more similar to θ_R are more topically relevant and should thus receive a higher probability of being picked. We propose to base the estimate of $P(D|\theta_R)$ on the divergence between D and θ_R and we measure this divergence by determining the log-likelihood ratio between D and θ_R, normalized by the collection C:

$$\begin{aligned} P(D|\theta_R) &\propto H(\theta_D,\theta_C) - H(\theta_D,\theta_R) \\ &= Z_D \sum_{t\in\mathcal{V}} P(t|\theta_D)\log\frac{P(t|\theta_R)}{P(t|\theta_C)}. \end{aligned} \quad (5)$$

Interpreted loosely, this measure indicates the average surprise of observing document D when we have θ_R in mind, normalized using a background collection, C. The measure has the attractive property that it is high for documents for which $H(\theta_D,\theta_C)$ is high and $H(\theta_D,\theta_R)$ is low. So, in order to receive a high score, documents should contain specific terminology, i.e., they should be dissimilar from the collection model but similar to the topical model of relevance. Since we do not know the actual parameters of θ_R by which we could calculate this, we use R as a surrogate and linearly interpolate it with the collection model (viz. Eq. 3): $P(t|\hat{\theta}_R) = (1-\lambda_R)P(t|R) + \lambda_R P(t|\theta_C)$. This also ensures

that the sum in Eq. 5 is over the same event space for all language models involved and that zero-frequency issues are avoided. Then, in order to use this discriminative measure as a probability, we define a document-specific normalization factor $Z_D = 1/\sum_{D \in R} P(D|\theta_R)$.

As an aside, other ways of estimating $P(D|\theta_R)$ have been proposed, such as simply assuming a uniform distribution, the retrieval score of a document, the inverse thereof, or information from clustered documents [2, 9]. Our approach is equivalent to using document cluster information under the assumption that only a single cluster is used, namely that which contains all relevant documents. Using the retrieval scores or, in an LM setting, the likelihood that a document generated the query, is a much simpler implementation of the same idea, essentially replacing θ_R with the initial query. And, since the initial query is quite sparse compared to θ_R, we avoid overfitting.

Finally, by putting the earlier equations together, we obtain the estimate of our expanded query model:

$$P(t|\theta_R) = \sum_{D \in R} \left\{ Z_D \sum_{t' \in \mathcal{V}} P(t'|\theta_D) \log \frac{P(t'|\hat{\theta}_R)}{P(t'|\theta_C)} \right\} P(t|\theta_D). \quad (6)$$

This model, to which we refer as NLLR (normalized log-likelihood), effectively determines the expanded query model $P(t|\theta_R)$ based on information from each individual relevant document and the most representative sample we have of θ_R, namely R.

4. EXPERIMENTAL SETUP

We use the test data provided by the TREC Relevance Feedback track, where the task is to retrieve additional relevant documents given a query and an initial set of assessments [4]. Retrieval is done on the .GOV2 corpus, from which we remove stopwords and to which we apply Porter stemming. We use the titles of the 31 topics that received additional judgments. For each of these topics, a large set of relevance assessments is provided (159 relevant documents on average, with a minimum of 50 and a maximum of 338). Participating systems were to return 2500 documents, from which the initially provided relevant documents are removed. The resulting rankings were then pooled and re-assessed. This yielded 55 new relevant documents on average per topic, with a minimum of 4 and a maximum of 177. We follow the same setup and remove all initially judged documents from the final rankings in our experiments.

Below, we report on the following measures; precision at 5 (P5), precision at 10 (P10), mean average precision (MAP), and the number of retrieved relevant documents (relret).

4.1 Parameter Settings

We fix the estimation method of the document models and use Dirichlet smoothing which has been shown to achieve superior performance on a variety of tasks and collections [12, 23]. We set $\mu = 1600$ and we keep only the 10 terms with the highest probability for all models. We optimize the remaining parameter settings on MAP using a grid search.

4.2 Reference Models

In our experiments, we compare our model with two other established relevance feedback methods. In particular, we look at Lavrenko and Croft's relevance models [11] and Zhai and Lafferty's model-based feedback [22] which are indicated by RM and MBF respectively. For all our experiments we use the Lemur toolkit and use the provided implementations whenever possible.

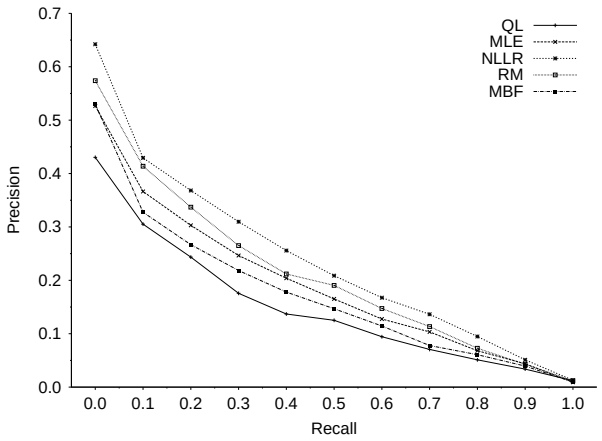

Figure 1: Precision-Recall graph.

5. RESULTS

In this section we present our main experimental results. Table 1 shows the experimental results of applying the various approaches for estimating $P(t|\theta_R)$. As indicated earlier, these results are obtained by using the full set of judged relevant documents for estimation and subsequently removing these from the rankings. Figure 1 further illustrates the differences using a precision-recall graph.

First, we observe that the query-likelihood results (QL) are on a par with the median of all submitted runs for the TREC Relevance Feedback track [4] and all models described improve over this baseline. If we would have submitted the results of the NLLR model, it would have ended up in the top-3 for this particular category. The RM run would have been placed at around rank 7.

Since these results are obtained by using the full set of relevance assessments, one might expect that the MLE achieves high scores since this set should be representative of the information need and, hence, of the distribution of θ_R. Contrary to this intuition, however, the MLE approach does not achieve the highest performance when new relevant documents are to be retrieved; a finding in line with observations made by Buckley et al. [5]. MBF—which re-estimates the MLE model—mainly has a precision-enhancing effect: recall and MAP are hurt using this approach when compared against MLE.

A precision enhancing effect is also visible when using NLLR and RM. Indeed, NLLR achieves the highest scores overall, except for the number of relevant retrieved documents (RM retrieves 2 relevant documents more). We further observe that NLLR obtains a significant 63.7% improvement in terms of MAP over the baseline. The MAP score is higher than the one obtained by RM and, moreover, NLLR obtains a statistically significant improvement with $\alpha = 0.001$.

Figure 1 shows that NLLR improves over all models on all recall levels. This figure also shows that the MBF approach does not help as compared to MLE. However, MBF should be equivalent to MLE when the collection element

	QL	MLE		NLLR		RM		MBF	
relret	1122	1279	+14.0%*	1349	+20.2%**	**1351**	+20.4%**	1254	+11.8%*
P5	0.2516	0.2968	+18.0%	**0.3935**	+56.4%*	0.3871	+53.9%*	0.3097	+23.1%
P10	0.2452	0.3065	+25.0%	**0.3871**	+57.9%*	0.3613	+47.3%*	0.2710	+10.5%
MAP	0.1364	0.1779	+30.4%*	**0.2233**	+63.7%***	0.1998	+46.5%**	0.1598	+17.2%*

Table 1: Results of the models contrasted in this paper (best scores in boldface). The percentages indicate the difference w.r.t. the baseline. The *, **, * indicate a statistically significant difference as compared to the baseline at the $p < 0.001$, $p < 0.01$, $p < 0.05$ level respectively, measured using a Wilcoxon signed rank test.**

is removed. It seems that under this particular model and the current experimental conditions, the introduction of the collection model deteriorates the results.

When we look at the individual topics, we find that topic 808 ('north korean counterfeiting') seems particularly difficult and the retrieval performance is worst on this topic for all employed query models (although there are 530 judged relevant and 330 new relevant documents available). We note that, in general, NLLR is able to improve over the baseline on a larger number of topics than the other methods. RM works best for topic 766, on which NLLR also performs very well. The other two models (MBF and MLE) improve most on topic 814. Interestingly, this topic is also helped a lot by NLLR, but not by RM. These observations provide evidence that NLLR is indeed able to reap the benefits both of the individual relevant documents (like RM) and of the set as a whole (like MBF).

6. CONCLUSION

Relevance assessments by a user are an important and valuable source of information for retrieval. In a language modeling setting, various methods have been proposed to estimate query models from these. Most of these models, however, attempt to update the initial query based on either the full contents of each assessed document or their aggregate. In this paper we have presented a novel query modeling method which incorporates both sources of evidence in a principled manner. It leverages the distance between each relevant document and the set of relevant documents to inform the query model estimates and, as such, it is more general than the methods proposed before.

We have evaluated our proposed model on the TREC Relevance Feedback test collection and found that it improves over a query-likelihood baseline as well as over other established methods.

7. ACKNOWLEDGEMENTS

This research was carried out in the context of the Virtual Laboratory for e-Science project and supported by the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments under project number STE-09-12 and the Netherlands Organisation for Scientific Research (NWO) under project numbers 017.001.190, 640.001.501, 640.002.501, 612.066.512, 612.061.814, 612.061.815, 640.004.802.

8. REFERENCES

[1] J. Bai and J.-Y. Nie. Adapting information retrieval to query contexts. *IPM*, 44(6):1901–1922, 2008.

[2] K. Balog, W. Weerkamp, and M. de Rijke. A few examples go a long way: constructing query models from elaborate query formulations. In *SIGIR '08*, 2008.

[3] M. Bendersky and B. W. Croft. Discovering key concepts in verbose queries. In *SIGIR '08*, 2008.

[4] C. Buckley and S. Robertson. Relevance feedback track overview: TREC 2008. In *TREC '08*, 2008.

[5] C. Buckley, D. Dimmick, I. Soboroff, and E. Voorhees. Bias and the limits of pooling for large collections. *Information Retrieval*, 10(6):491–508, 2007.

[6] M. Clements, A. de Vries, and M. Reinders. The influence of personalization on tag query length in social media search. *Information Processing & Management*, In Press, Corrected Proof:–, 2009.

[7] P. Clough, H. Müller, T. Deselaers, M. Grubinger, T. Lehmann, J. Jensen, and W. Hersh. The CLEF 2005 Cross-Language Image Retrieval Track. In *CLEF 2005 Working Notes*, 2005.

[8] D. Hiemstra. A linguistically motivated probabilistic model of information retrieval. In *ECDL '98*, 1998.

[9] O. Kurland. The opposite of smoothing: a language model approach to ranking query-specific document clusters. In *SIGIR '08*, 2008.

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

[11] V. Lavrenko and B. W. Croft. Relevance models in information retrieval. In B. W. Croft and J. Lafferty, editors, *Language Modeling for Information Retrieval*, pages 11–54. Kluwer, 2003.

[12] D. Losada and L. Azzopardi. An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11(2):109–138, 2008.

[13] E. Meij, W. Weerkamp, K. Balog, and M. de Rijke. Parsimonious relevance models. In *SIGIR '08*, 2008.

[14] D. Metzler and B. W. Croft. A markov random field model for term dependencies. In *SIGIR '05*, 2005.

[15] K. Ng. A maximum likelihood ratio information retrieval model. In *TREC 2000*, 2000.

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

[17] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *JASIST*, 41(4):288–297, 1990.

[18] A. Spink, B. J. Jansen, and C. H. Ozmultu. Use of query reformulation and relevance feedback by excite users. *Internet Research: Electronic Networking Applications and Policy*, 10(4):317–328, 2000.

[19] A. Spink, B. J. Jansen, D. Wolfram, and T. Saracevic. From e-sex to e-commerce: Web search changes. *IEEE Computer*, 35(3):107–109, 2002.

[20] X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In *CIKM '08*, 2008.

[21] W. Weerkamp, K. Balog, and E. J. Meij. A generative language modeling approach for ranking entities. In *Advances in Focused Retrieval*, 2009.

[22] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01*, 2001.

[23] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM TOIS*, 22(2):179–214, 2004.