INTEGRATING CONCEPTUAL KNOWLEDGE INTO RELEVANCE MODELS
A Model and an Estimation Method

Edgar Meij and Maarten de Rijke
ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
emeij, mdr@science.uva.nl

Keywords: Statistical language modeling, Relevance models, Query expansion, Pseudo-relevance feedback

Abstract: We address the issue of combining explicit background knowledge with pseudo-relevance feedback from within a document collection. To this end, we use document-level annotations in tandem with generative language models to generate terms from pseudo-relevant documents and bias the probability estimates of expansion terms in a principled manner. By applying the knowledge inherent in document annotations, we aim to control query drift and reap the benefits of automatic query expansion in terms of recall without losing precision. We consider the parameters which are associated with our modeling and describe ways of estimating these automatically. We then evaluate our modeling and estimation methods on two test collections, both provided by the TREC Genomics track.

1 INTRODUCTION

When formulating queries, users of a search engine "translate" their information need into terms. These terms are generally a combination of background knowledge and terms that the user associates with relevant documents. E.g., as part of her background knowledge a user knows what kind of synonyms there are for particular terms, or which terms are related to her information need. When a user cannot find what she wants, she may reformulate her query in an iterative manner, a process known as query expansion or relevance feedback. Automatic query expansion (or pseudo-relevance feedback) methods are designed to take this burden from the user and address one of the issues at stake—looking at an initial result set or applying background knowledge—automatically. One of the best known and oldest query expansion methods is probably Rocchio's method, in which a set of top-ranked documents is used to locate additional terms to add to the query (Rocchio and Salton, 1965). More recent examples do not only look at documents, but also at local context for additional query terms (Xu and Croft, 1996, 2000). A different approach is taken by Zhai and Lafferty (2001a), who consider generating terms through pseudo-relevance feedback as sampling from a generative feedback query model.

Most automatic methods for using background knowledge in query (re)formulation, utilize structured databases, thesauri, or ontologies. In these (extensively researched) methods, a query term is mapped onto a concept and, subsequently, related concepts are added to the query and possibly reweighed (Voorhees, 1993; Wollersheim and Rahayu, 2005; Zhou et al., 2007). The relation of the expansion concepts to the initial query term(s) may be defined by synonymy, hypernymy or any other kind of relation. Most of these automatic methods report an increase in recall at a loss of precision, mostly due to a common artifact of the methods known as *query drift*.

As of yet there are few methods that combine background knowledge and pseudo-relevance feedback in a principled and transparent manner. Collins-Thompson and Callan (2005) describe an elaborate way of combining multiple sources of evidence to predict relationships between query and vocabulary terms. The semantic features they investigate are general word associations and synonymy relations as defined in WordNet. Cao et al. (2005) describe a more principled way of integrating WordNet term relationships into statistical language models but, in the end, they rely mostly on co-occurence data for their esti-

mations.

The statistical language modeling approach to information retrieval has attracted significant attention over the last years. The underlying ideas are intuitive and transparent, and empirical results show that the approach is competitive. However, pseudo-relevance feedback in a language modeling framework has thus far been incorporated mostly in a way that seems contradictory to the essence of the approach: the estimation of models. Building upon the work laid out by Robertson and Sparck Jones (1976), Lavrenko and Croft (2001) suggested a way to converge the binary independence model and a language modeling approach through a generative model of relevance.

Rather than looking directly at knowledge structures, we address the issue of combining background knowledge with pseudo-relevance feedback from the document collection itself. To this end, we use document-level annotations in tandem with generative language models—relevance models, to be precise—to generate terms from pseudo-relevant documents and bias the probability estimates of expansion terms in a principled manner. By applying the knowledge inherent in document annotations, we aim to control query drift and reap the benefits of automatic query expansion in terms of recall, without losing precision.

Our contributions in this paper are three-fold. First, we address the question how we can integrate concepts as found in a thesaurus into a relevance model framework. Next, we look at the various parameters which are associated with our model and describe ways of estimating these automatically. We then evaluate our model and estimation methods on two test collections, both provided by the TREC Genomics track (Hersh et al., 2006).

We build upon the foundations laid out in previous work (Meij and de Rijke, 2007). Our focus in the present paper is on automatically estimating one of the free parameters in our model, thus removing the need for computationally intensive parameter sweeps. Additionally, our empirical evaluation goes beyond earlier work in that we establish an increase in retrieval effectiveness using multiple test collections (instead of a single one.)

The remainder of the paper is organized as follows. In the next section, we describe our model and the generative language modeling context in which it is situated. Then, we look at an elegant way of automatically estimating one of the free parameters using the EM algorithm. In Section 4, we turn to the empirical results of our model and contrast its performance with regular, state-of-the-art approaches. Next, we look at the influence of the number of documents used for the estimations and we end with a concluding section.

2 LANGUAGE MODELING

Within information retrieval, language modeling is a relatively novel approach (Hiemstra, 2001; Kraaij, 2004; Miller et al., 1999; Ponte and Croft, 1998; Zhai, 2002). Generative language modeling originates from speech recognition, where the modeling of speech utterances is mapped onto textual representations. The ideas behind it are intuitive and theoretically well-motivated and the approach provides us with an easily extendible setting for incorporating the information captured in document annotations. Before introducing our novel feedback model, we recall some general facts about statistical language modeling for IR.

2.1 Generative Language Modeling

Language modeling for IR is centered around the assumption that a query, as issued by a user, is a sample generated from some underlying term distribution. The documents in the collection are modeled in a similar fashion, and are also regarded as samples from an unseen term distribution—a generative language model.

At retrieval time, the language use in documents is compared with that of the query and the documents are ranked according to the likelihood of generating the query. Assuming independence between query terms, the probability of a document given a query can be more formally stated using Bayes' rule:

$$P(d|Q) \propto P(d) \cdot \prod_{q \in Q} P(q|\theta_d), \qquad (1)$$

where θ_d is a language model of document d, and the q_i the individual query terms in query Q. The term $P(d)$ captures the prior belief in a document being relevant, which is usually assumed to be uniform. The term $P(\cdot|\theta_d)$ is estimated using maximum-likelihood estimates which, in this case, means using the frequency of a query term in a document: $P(q|\theta_d) = c(q,d)/|d|$. Here, $c(q,d)$ indicates the count of term q in document d and $|d|$ the length of the particular document. This captures the notion that $P(q|\theta_d)$ is the relative frequency with which we expect to see the term q when we repeatedly and randomly sample terms from this document. The higher this frequency, the more likely it is that this document will be relevant to the query.

2.2 Smoothing

It is clear from Eq. 1 that taking the product of term frequencies has a risk of resulting in a probability of zero: "unseen" terms will produce a probability of zero for that particular document. To tackle this problem, *smoothing* is usually applied, which assigns a very small, non-zero probability to unseen words. Dirichlet smoothing (Chen and Goodman, 1996; Zhai and Lafferty, 2001b) is formulated as follows:

$$P(t|\theta_d) = \frac{c(t,d)+\mu P(t|\theta_C)}{|d|+\mu}, \quad (2)$$

where t is a vocabulary term, θ_C the language model of a large reference corpus C (e.g., the collection) and μ a constant by which to tune the influence of the reference model. When comparing the language modeling framework for IR with more well-known TF.IDF schemes, the application of smoothing has an IDF like effect (Hiemstra, 1998; Zhai and Lafferty, 2001b).

2.3 Query Models

One deficiency of the query likelihood model is that it is difficult to naturally incorporate relevance feedback information to improve ranking accuracy. In particular, since we model our query in a similar fashion as the document, it is unclear how the likelihood of an "expanded query" is to be computed and it is even harder to allow different query terms to have different weights. One solution to this problem is to generalize the query likelihood model to a measure of difference between two probability distributions, such as the Kullback-Leibler divergence (Zhai, 2002; Zhai and Lafferty, 2001a). Taking this approach, a second language model (i.e., the query language model) is introduced and documents are ranked according to the difference between the query model and the document model. It is easy to show that when the query language model is estimated with the empirical query term distribution, documents are ranked in the same order as the original query likelihood model. The advantage of this model, though, is the possibility of casting (pseudo-)relevance feedback as estimating the query language model differently, viz. based on both the query and some feedback documents, thus treating feedback as updating the query model.

2.4 Relevance Models

Relevance models are a special class of language models, which are used to estimate a distribution θ_Q over terms in a query's vocabulary (Lavrenko and Croft, 2001). The intuition is that the query and the set of relevant documents are both samples from the same (relevant) term distribution. Generative language models and relevance models differ in the way how these distributions are modeled. While generative language modeling assumes that queries are generated from documents or vice versa, relevance models assume that both are generated from an unseen source—the relevance model.

How is a relevance model created? A set of documents R, which has been judged relevant to a specific query, can be used as a model from which terms are sampled. In the absence of such relevance information, an initial retrieval run can be performed and the top-ranked documents are assumed to be relevant. Then, the probability of a term being generated from the relevance model is related to the conditional probability of observing the term, given that the query terms $q_1,\ldots,q_n$ have just been observed (Lavrenko and Croft, 2001):

$$\begin{aligned} P(t|\hat{\theta}_Q) &\approx P(t|Q) \quad (3)\\ &= P(t|q_1,\ldots,q_n)\\ &= \frac{P(t,q_1,\ldots,q_n)}{P(q_1,\ldots,q_n)}. \end{aligned}$$

We follow Lavrenko and Croft's Method 2, which assumes that the query terms $q_1,\ldots,q_n$ are independent of each other, but keep their dependence on t:

$$P(t,q_1,\ldots,q_n) = P(t)\cdot\prod_i P(q_i|t). \quad (4)$$

To estimate the rightmost conditional probability, the expectation over R is computed:

$$P(q_i|t) = \sum_{d\in R} P(\theta_d|t)\cdot P(q_i|\theta_d). \quad (5)$$

Note that their approach assumes that q_i is independent of t given θ_d (Lavrenko and Croft, 2001). The query prior in Eq. 3 is set to:

$$P(q_1,\ldots,q_n) = \sum_t P(t,q_1,\ldots,q_n),$$

and the word prior in Eq. 4 is set to:

$$P(t) = \sum_{d\in R} P(t|\theta_d)\cdot P(\theta_d).$$

Combining these equations we obtain

$$P(t|\hat{\theta}_Q) \propto P(t)\cdot\prod_i\sum_{d\in R} P(q_i|\theta_d)\cdot P(\theta_d|t), \quad (6)$$

in which the conditional probability of picking a document model θ_d, given t is defined as:

$$P(\theta_d|t) = \frac{P(t|\theta_d)P(\theta_d)}{P(t)}. \quad (7)$$

To obtain the estimates for the terms $P(t|\theta_d)$ and $P(Q|\theta_d)$, smoothed maximum-likelihood techniques are used, as described earlier.

2.5 Biasing Relevance Models

Our approach extends the relevance modeling approach, by not only looking at the document models to estimate a relevance model, but also at thesaurus terms that are associated with the documents. Suppose we have a collection in which each document is annotated using terms from a thesaurus or controlled vocabulary; we then explicitly state that, although q_i is independent of t given θ_d, the probability of observing a document model θ_d is also dependent on the thesaurus terms $m_1,\ldots,m_l$ that are associated with that document. More formally, let M be a set of thesaurus terms. Then, we define the posterior probability of selecting a document model, given t and $m_1,\ldots,m_l \in M$ (cf. Eq. 7) as:

$$P(\theta_d|t,M) = \frac{P(t|\theta_d)P(M|\theta_d)P(\theta_d)}{\sum_{\theta_{d'}} P(t|\theta_{d'})P(M|\theta_{d'})P(\theta_{d'})}, \qquad (8)$$

which yields the following estimation of a term t, given the relevance model:

$$P(t|\hat{\theta}_Q) \propto \qquad (9)$$
$$P(t)\cdot\prod_i \sum_{d\in R} P(q_i|\theta_d)\cdot P(t|\theta_d)P(m_1,\ldots,m_l|\theta_d).$$

For reasons of efficiency, we limit the number of thesaurus terms $m_1, \ldots, m_l$ used in these estimations. We select the 20 top-ranked thesaurus terms according to the probability of observing a thesaurus term, given the query: $P(m_k|Q)$. To obtain this ranking, we created a smoothed language model per thesaurus term by aggregating the documents they are associated with.

We additionally assume the thesaurus terms to be independent, so we can express their joint probability $P(m_1,\ldots,m_l|\theta_d)$ as the product of the marginals: $\prod_{k=1}^{l} P(m_k|\theta_d)$. Each term $P(m_k|\theta_d)$ can be estimated using Bayes' rule, by determining the following posterior distribution, based on documents annotated with that particular term:

$$P(m_k|\theta_d) = \frac{P(\theta_d|m_k)\cdot P(m_k)}{P(\theta_d)}. \qquad (10)$$

We estimate the prior probability $P(m_k)$ of seeing a thesaurus term as: $P(m_k) = (|M|\cdot c(m_k))^{-1}$ for any given thesaurus term m_k, where $c(m_k)$ is the total number of times this thesaurus term is used to categorize a document and $|M| = \sum_{m\in M} c(m)$. Doing so ensures that frequently occuring, more general (and thus less discriminative thesaurus terms) receive a relatively lower probability mass. The term $P(\theta_d|m_k)$ is estimated in a similar fashion: it is 0 if m_k is not associated with d, and the reciprocal of the number of documents associated with thesaurus term m_k otherwise.

Table 1: Comparison of terms with the highest probability for topic 173: "How do alpha7 nicotinic receptor subunits affect ethanol metabolism?" Terms specific to a model are marked in boldface.

Relevance models (RM)	Thesaurus-biased models (MM)
receptor	receptor
nicotin	nicotin
subunit	of
of	the
acetylcholin	subunit
the	**humans**
alpha7	acetylcholin
abstract	**animals**
alpha	**nicotinic**
medlin	**study**
2003	alpha7

Table 1 shows the difference in terms generated by standard relevance models versus thesaurus-biased models on a random topic from the TREC Genomics 2006 test set. This table suggests that the thesaurus-biased model is able to allocate more probability mass to more topic-specific terms.

2.6 Clipped Relevance Model

Relevance models are shown to perform better when they are linearly interpolated with the original query—a so-called "clipped relevance model" (Kurland et al., 2005)—using a mixing weight λ:

$$P(t|\theta_Q) = \lambda\cdot P(t|\tilde{\theta}_Q) + (1-\lambda)\cdot P(t|\hat{\theta}_Q) \qquad (11)$$
$$= \lambda\cdot\frac{c(t,Q)}{|Q|} + (1-\lambda)\cdot P(t|\hat{\theta}_Q).$$

Our final model is thus composed of an original part $P(t|\tilde{\theta}_Q)$ and an expanded part $P(t|\hat{\theta}_Q)$. When λ is set to 1, the ranking function reduces to the query-likelihood ranking algorithm described earlier in Section 2.1.

3 ESTIMATIONS

Following the previous section, we assume that the final query model is generated from a mixture of the terms found in the (pseudo-)relevant documents and the initial query Q. In order to estimate lambda in Eq. 11, we approximate the query model space by the probability estimates of the terms, as they are found in the (pseudo-)relevant documents. The log of the

Table 2: Ad-hoc retrieval results of the query-likelihood baseline (QL), Relevance model (RM), and MeSH-biased model (MM), in terms of mean average precision (MAP) and precision@10 (P10). Changes in scores are given relative to the baseline and the best scores are marked in boldface.

	P10					MAP				
Coll.	QL	RM		MM		QL	RM		MM	
trecgen05	0.369	**0.374**	1.36%	0.360	-2.44%	0.218	0.220	2.33%	**0.241**	12.09%
trecgen06	0.450	0.454	0.89%	**0.465**	3.33%	0.359	0.360	0.28%	**0.416**	15.88%

likelihood of observing these terms is:

$$\log P(t_1,\ldots,t_n|\theta_Q,\lambda) = \sum_i \pi_i \prod_{j=1}^{n} \log\left((\lambda P(t_j|\tilde{\theta}_{q_i}) + (1-\lambda)P(t_j|\hat{\theta}_{q_i})\right), \quad (12)$$

where $\pi_i = P(\hat{\theta}_{q_j}|\tilde{\theta}_{q_j})$. When π_i is left free to be estimated, it will allocate higher weights to the query terms which contributed most to the prediction of the terms found in the relevant documents. The EM algorithm can then be used to find the λ which maximizes the (log of the) likelihood (Dempster et al., 1977; McLachlan and Krishnam, 1997):

$$\lambda^* = \arg\max_{\lambda} \log P(t_1,\ldots,t_n|\theta_Q). \quad (13)$$

The following iterative steps can be used to find λ:

$$\pi_i^{k+1} = \frac{\pi_i^k \prod_{j=1}^{n} \left(\lambda^k P(t_j|\tilde{\theta}_{q_i}) + (1-\lambda^k)P(t_j|\hat{\theta}_{q_i})\right)}{\sum_{i'=1}^{n} \pi_i^k \prod_{j=1}^{n} (\lambda^k P(t_j|\tilde{\theta}_{q_{i'}}) + (1-\lambda^k)P(t_j|\hat{\theta}_{q_{i'}})} \quad (14)$$

and

$$\lambda^{k+1} = \frac{1}{n}\sum_i \pi_i^{k+1} \sum_{j=1}^{n} \frac{\lambda^k P(t_j|\tilde{\theta}_{q_i})}{\lambda^k P(t_j|\tilde{\theta}_{q_i}) + (1-\lambda^k)P(t_j|\hat{\theta}_{q_i}}.$$

4 EXPERIMENTAL SETUP

We now turn to the setup of the experiments used to determine the effectiveness of our thesaurus-biased language model and of our estimation methods.

Our experiments are based on the test collections from the TREC Genomics track 2005 and 2006 (Hersh et al., 2005, 2006). All of the documents used in our experiments are accessible through PubMed, a bibliographic database maintained by the National Library of Medicine (NLM). It contains bibliographical records of all documents available in MedLine. MedLine in turn contains almost all publications from the major biomedical research areas, conferences, and journals.

PubMed uses controlled vocabulary terms to catalog and index the documents. This vocabulary, called MeSH (Medical Subject Headings), is a thesaurus containing 22,997 hierarchically structured concepts, and is used by trained annotators from the NLM to assign one or more MeSH terms to every document, with an average of around 10 MeSH terms per document. These MeSH terms are used for the estimations of our thesaurus-biased models, hence we will further refer to our model as the *MeSH-biased model* (MM). The TREC Genomics 2005 ad-hoc retrieval task focused on retrieving abstracts from a 10-year subset of MedLine, given 50 topics. A novel task and collection were put forward in the 2006 track. First, given 28 topics, relevant documents needed to be identified from a full-text document collection and, then, the relevant passages from these documents were to be returned. Relevance for the latter year's track was measured at three levels: the document, passage and aspect level. For our current experiments we only use the judgments at the document level. Table 4 provides an overview of the characteristics of both collections.

Table 4: Test collections.

Coll.	Size	Vocab. size
trecgen05	4,591,008 abstracts	800,477,879
trecgen06	162,259 full- docs	1,090,232,994

All runs are morphologically normalized as described by Huang et al. (2005) and stemmed using a Porter stemmer. No stopwords were removed; the smoothing parameter μ is set to 100 and we use the top-10 ranked documents for all estimations, as detailed in the previous section.

5 RESULTS AND DISCUSSION

We discuss three sets of results. First, we show that our proposed model outperforms a query likelihood, as well a relevance model approach on two different collections. For these we use the models as found in Eq. 1 and Eq. 11, respectively. We then take a closer look at the estimation methods, and show how the results differ by varying the number of pseudo-relevant documents used for estimations. We conclude with a per-topic analysis of our model.

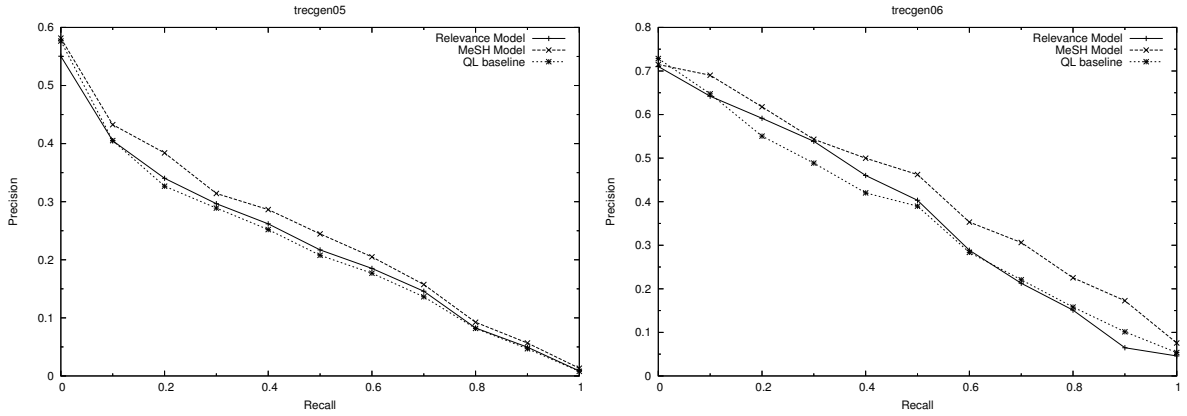

Figure 1: Precision-recall graphs comparing relevance models and MeSH-biased models for both collections.

Table 3: Detailed ad-hoc retrieval results of the query-likelihood baseline (QL) versus the MeSH-biased model (MM), in terms of precision at varying levels of recall. Changes in scores are given relative to the baseline. Statistical significance is tested using a one-tailed Wilcoxon signed-rank test (***indicates an improvement at the $p < 0.001$ level, **at the $p < 0.01$ level, and *at the $p < 0.05$ level.)

trecgen05					trecgen06				
recall level	QL	MM	change		recall level	QL	MM	change	
0.00	0.577	0.582	0.88%	***	0.00	0.728	0.715	-1.86%	***
0.10	0.405	0.433	6.77%		0.10	0.648	0.690	6.57%	***
0.20	0.327	0.384	17.55%	*	0.20	0.550	0.618	12.25%	**
0.30	0.289	0.314	8.70%		0.30	0.488	0.543	11.20%	
0.40	0.252	0.287	13.68%	*	0.40	0.420	0.500	18.96%	
0.50	0.208	0.245	17.78%		0.50	0.390	0.462	18.62%	
0.60	0.177	0.205	15.87%		0.60	0.284	0.353	24.48%	
0.70	0.136	0.157	15.50%		0.70	0.221	0.306	38.78%	
0.80	0.082	0.092	12.91%	**	0.80	0.158	0.225	42.61%	
0.90	0.047	0.057	20.96%	***	0.90	0.101	0.173	70.26%	
1.00	0.008	0.013	66.15%	***	1.00	0.054	0.076	40.66%	***
rel.ret.	3010	3205	6.48%	*	rel.ret.	1005	1212	20.60%	
rel.	4584	4584			rel.	1449	1449		

5.1 Retrieval Results

In Table 2 we list the results of our model (MM) and compare it against a query-likelihood baseline (QL) and a relevance model (RM). We see that MM outperforms the baseline, as well as state-of-the-art relevance models, when measured in terms of mean average precision (MAP). Figure 1 further illustrates the differences between the three approaches for both collections using precision-recall graphs. Table 3 shows the improvements of our model over the baseline in terms of precision over varying levels of recall. We observe that our MeSH-biased model improves precision at all recall levels, except one. This effect is especially visible on the 2006 collection, where our model is able to retrieve over 20% more relevant documents, while also slightly increasing early precision.

A few comments. First, our baseline performance (QL) beats the median scores achieved at the 2005 and 2006 editions of the TREC Genomics track; 2005: 0.216, 2006: 0.308 (Hersh et al., 2005, 2006). Also, the MAP scores of our MeSH-biased model MM are not among the highest, when compared to all participating TREC Genomics systems. However, our results originate from purely automatic methods, without any elaborate tuning of parameters. Additionally, we do not make use of extensive query expansion techniques for gene names, proteins, and/or diseases, which is common for this application field. The strength of our model lies in the fact that it uses the data that is already present in the collection, either explicitly (the MeSH terms) or implicitly (the associated term distributions over the vocabulary.)

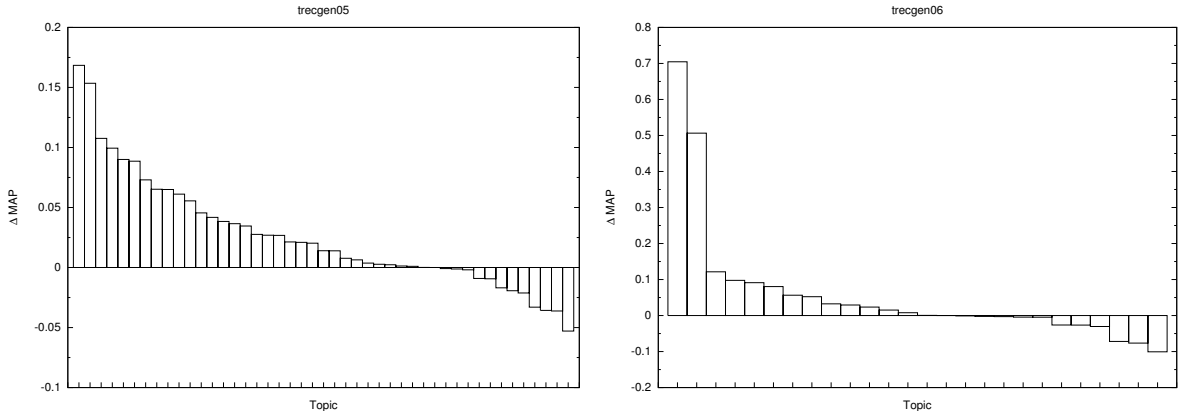

Figure 2: Per-topic comparison of the MeSH-biased model with the query likelihood baseline for both collections.

5.2 Estimation

When we look at the best results of our model established using a full parameter sweep, we find that the current, automatic results are slightly worse (Meij and de Rijke, 2007). The difference is minimal however, proving the effectiveness of the current estimation approach. The amount of documents used for the automatic estimations has a distinct impact on retrieval effectiveness, as can be seen from Figure 3. Two observations can be made here. First, our model outperforms relevance models in general—on the 2005 collection for the full range of documents used, whilst on the 2006 collection for a subset. This leads to the next observation: the optimal number of documents to use is dependent on the collection at hand.

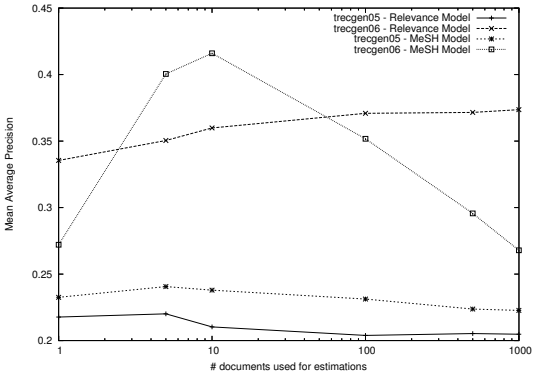

Figure 3: Influence of varying the number of documents used for estimations.

5.3 Per-topic Analysis

Figure 2 shows a graphical representation of the per-topic difference between retrieval scores (in terms of MAP) of our approach and the baseline, ordered decreasingly. Two topics seem to be helped most for both sets; typical for these particular queries is that they have synonymous terms in the vocabulary, which are found through the document annotations. This leads to the view of our model as a natural, semantic "bridge" between an initial query and semantically related terms. For a more qualitative evaluation of the found annotations and generated terms, we refer the interested reader to (Meij and de Rijke, 2007).

6 CONCLUSION

We have presented a novel way of incorporating background information in relevance models. In our model, the estimation of term probabilities are biased towards document-level annotations for a given query. Thus, we are able to naturally leverage the information that is "encoded" in a document collection through these annotations. Additionally, we provide a way of automatically estimating an important free parameter, one that controls the influence of the initial query in the resulting query representation. We have provided the results of an empirical evaluation on two distinct test collections and show the consistent improvements over a query-likelihood, as well as a relevance model baseline. When compared to the baseline, our model is able to increase recall, whilst increasing early precision at the same time on a collection of full-text documents. On a collection of abstracts, early precision is hurt only slightly.

Future work includes modeling the structure of the thesaurus that is used to annotate the documents. The knowledge encapsulated in this structure might provide an additional performance increase. Additionally, we have assumed that thesaurus terms are inde-

pendent of each other, given a document, while in fact they may not be. In our future work, we aim to address this issue.

ACKNOWLEDGMENTS

We thank our reviewers for their valuable comments. This work was carried out in the context of the Virtual Laboratory for e-Science project (http://www.vl-e.nl), which is supported by a BSIK grant from the Dutch Ministry of Education, Culture and Science (OC&W) and is part of the ICT innovation program of the Ministry of Economic Affairs (EZ). De Rijke was supported by the Netherlands Organization for Scientific Research (NWO) and by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104.

REFERENCES

Cao, G., Nie, J.-Y., and Bai, J. (2005). Integrating word relationships into language models. In *SIGIR '05*.

Chen, S. F. and Goodman, J. (1996). An empirical study of smoothing techniques for language modeling. In *ACL*, pages 310–318.

Collins-Thompson, K. and Callan, J. (2005). Query expansion using random walk models. In *CIKM '05*.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Hersh, W., Cohen, A., Yang, J., Bhupatiraju, R. T., Roberts, P., and Hearst, M. (2005). TREC 2005 Genomics track overview. In *Proceedings of the 14th Text Retrieval Conference*. NIST.

Hersh, W., Cohen, A. M., Roberts, P., and Rekapalli, H. K. (2006). TREC 2006 Genomics track overview. In *Proceedings of the 15th Text Retrieval Conference*. NIST.

Hiemstra, D. (1998). A linguistically motivated probabilistic model of information retrieval. In *ECDL '98*.

Hiemstra, D. (2001). *Using Language Models for Information Retrieval*. PhD thesis, University of Twente.

Huang, X., Ming, Z., and Si, L. (2005). York University at TREC 2005 Genomics track. In *Proceedings of the 14th Text Retrieval Conference*.

Kraaij, W. (2004). *Variations on Language Modeling for Information Retrieval*. PhD thesis, University of Twente.

Kurland, O., Lee, L., and Domshlak, C. (2005). Better than the real thing?: Iterative pseudo-query processing using cluster-based language models. In *SIGIR '05*.

Lavrenko, V. and Croft, W. B. (2001). Relevance based language models. In *SIGIR '01*.

McLachlan, G. and Krishnam, T. (1997). *The EM algorithm and Extensions*. Wiley, New York.

Meij, E. and de Rijke, M. (2007). Thesaurus-based feedback to support mixed search and browsing environments. In *ECDL 2007*.

Miller, D. R. H., Leek, T., and Schwartz, R. M. (1999). A hidden markov model information retrieval system. In *SIGIR '99*.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *SIGIR '98*.

Robertson, S. E. and Sparck Jones, K. (1976). Relevance weighting of search terms. *JASIS*, 27:129–146.

Rocchio, J. and Salton, G. (1965). Information search optimization and interactive retrieval techniques. In *Proceedings AFIPS*, volume 27.

Voorhees, E. M. (1993). Using wordnet to disambiguate word senses for text retrieval. In *SIGIR '93*.

Wollersheim, D. and Rahayu, J. W. (2005). Ontology based query expansion framework for use in medical information systems. *IJWIS*, 1(2):101–115.

Xu, J. and Croft, W. B. (1996). Query expansion using local and global document analysis. In *SIGIR '96*.

Xu, J. and Croft, W. B. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18(1):79–112.

Zhai, C. (2002). *Risk Minimization and Language Modeling in Text Retrieval*. PhD thesis, Carnegie Mellon University.

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

Zhai, C. and Lafferty, J. (2001b). A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR '01*.

Zhou, W., Yu, C., Smalheiser, N., Torvik, V., and Hong, J. (2007). Knowledge-intensive conceptual retrieval and passage extraction of biomedical literature. In *SIGIR '07*.