Information Processing and Management 46 (2010) 448–469

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

# Conceptual language models for domain-specific retrieval[☆]

Edgar Meij[a,*], Dolf Trieschnigg[b], Maarten de Rijke[a], Wessel Kraaij[c]

[a] *ISLA, University of Amsterdam, Science Park 107, 1098 XG, Amsterdam, The Netherlands*
[b] *HMI group, Faculty of EEMCS, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands*
[c] *IFL, Radboud University Nijmegen and TNO ICT, P.O. Box 9010, 6500 GL Nijmegen, The Netherlands*

ARTICLE INFO

*Article history:*
Received 31 October 2008
Received in revised form 1 July 2009
Accepted 3 September 2009
Available online 13 October 2009

*Keywords:*
Information retrieval
Meta-language
Language modeling
Query modeling

ABSTRACT

Over the years, various meta-languages have been used to manually enrich documents with conceptual knowledge of some kind. Examples include keyword assignment to citations or, more recently, tags to websites. In this paper we propose generative concept models as an extension to query modeling within the language modeling framework, which leverages these conceptual annotations to improve retrieval. By means of relevance feedback the original query is translated into a conceptual representation, which is subsequently used to update the query model.

Extensive experimental work on five test collections in two domains shows that our approach gives significant improvements in terms of recall, initial precision and mean average precision with respect to a baseline without relevance feedback. On one test collection, it is also able to outperform a text-based pseudo-relevance feedback approach based on relevance models. On the other test collections it performs similarly to relevance models. Overall, conceptual language models have the added advantage of offering query and browsing suggestions in the form of conceptual annotations. In addition, the internal structure of the meta-language can be exploited to add related terms.

Our contributions are threefold. First, an extensive study is conducted on how to effectively translate a textual query into a conceptual representation. Second, we propose a method for updating a textual query model using the concepts in conceptual representation. Finally, we provide an extensive analysis of when and how this conceptual feedback improves retrieval.

## 1. Introduction

Explicit and often manually curated knowledge is frequently being added to documents for a variety of reasons, e.g., to increase their findability or to aid navigation of the collection to which they belong. Such knowledge is typically expressed in a meta-language and can be either formal (e.g., in the form of a thesaurus or ontology) or more informal (e.g., in the form of user-generated tags). Annotations of this kind may be found in a broad range of domains and a variety of document types. News articles, for example, can be annotated with concepts from the NewsCodes taxonomies provided by the IPTC. Another example is the annotation of bibliographic records with indexing terms from a controlled vocabulary. In the biomedical domain citations in the MEDLINE database are manually indexed with concepts from the Medical Subject Headings (MeSH) thesaurus. We will refer to this broad range of meta-languages as *concept languages* and to their vocabulary terms as *concepts*. Tables 1 and 2 show two examples of document–concept annotations from the two test collections we describe later.

[☆] This work is a revised and substantially expanded version of Meij, Trieschnigg, de Rijke, and Kraaij (2008) and Meij and de Rijke (2008).
[*] Corresponding author. Tel.: +31 205257565; fax: +31 205257490.
*E-mail address:* edgar.meij@uva.nl (E. Meij).

0306-4573/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2009.09.005

**Table 1**
Example of a CSA document from the CLEF domain specific test collection, annotated with SA concepts.

| Document text [CSASA-1-EN-9600048] | Concept annotations |
|---|---|
| Immigration and Economic Dependence in the US: approaches to presenting logistic regression results<br>Logistic regression models are found increasingly in the social science literature, but the coefficients can be difficult to interpret for novice users. Strategies are discussed that can enhance the substantive interpretation of logistic regression results. . . | United States of America<br>Immigrants<br>Citizens<br>Benefits<br>Social security<br>Regression analysis |

**Table 2**
Example of a MEDLINE document (title and part of abstract) annotated with MeSH concepts.

| Document text [PMID: 10077651] | Concept annotations |
|---|---|
| Mechanism of increased iron absorption in murine model of hereditary hemochromatosis: increased duodenal expression of the iron transporter DMT1<br>Hereditary hemochromatosis (HH) is a common autosomal recessive disorder characterized by tissue iron deposition secondary to excessive dietary iron absorption. We recently reported that HFE, the protein defective in HH, was physically associated with the transferrin receptor (TfR) in duodenal crypt cells and proposed that mutations in HFE attenuate the uptake of transferrin-bound iron from plasma by duodenal crypt cells, leading to up-regulation of transporters for dietary iron. . . | Animals<br>Carrier proteins<br>Cation transport proteins<br>Duodenum<br>Hemochromatosis<br>Iron<br>Iron-binding proteins<br>Mice<br>Mutation |

The introduction of concept languages was initially driven by a need to facilitate search and navigation of the collection (Roberts, 1984; Joyce & Needham, 1958). Concepts were defined to unambiguously and precisely represent the content of documents. Today, most of these early retrieval systems have been replaced by full-text search systems which have been shown to be at least as effective (Cleverdon, Mills, & Keen, 1966). Since full-text search systems do not require a manually curated concept language, they are far less labour-intensive. Despite the effectiveness of full-text search, full-text indexing terms (which typically comprise of all the terms used in the documents in a given collection) can be more ambiguous or less expressive than concepts. Not surprisingly then, information retrieval (IR) researchers continue to study ways of incorporating information from concept languages to address problems in textual query representations. For example, a textual query may be mapped to one or more concepts in a thesaurus and expanded with their synonymous terms (Voorhees, 1994). Results of such approaches, however, have been mixed at best.

In this paper we show that a concept language can be effectively used to improve full-text retrieval. In a two step process that extends on relevance feedback and uses a conceptual representation as a pivot language we improve the query model representing the information need of the user.

In the first step, the textual information need is translated into a conceptual representation. In a process we call *conceptual query modeling*, feedback documents from an initial retrieval run are used for obtaining a conceptual query model. This model represents the user's information need at a different, higher conceptual level than the original query. The intuition behind this step is that this conceptual representation gives an unambiguous representation of the information need. In contrast to traditional textual relevance feedback, where the query refinement is biased towards terms occurring in the initial query, this intermediate conceptual representation is less dependent on the original query words. On its own, this explicit conceptual representation can be used to aid retrieval, for example by suggesting relevant concepts to the user (Meij & de Rijke, 2007), or by matching it to a conceptual representation of the documents (Trieschnigg et al., 2009). In the second step, however, we translate the conceptual query model back into a contribution to the textual query model. We hypothesize that, since the textual representation of documents is more detailed than its conceptual representation,[1] retrieving information with a textual query representation translated from a conceptual form, results in better retrieval performance than strictly matching with only concepts. Essential to these two translation steps is the estimation of a query model, both for terms and for concepts. The textual query should be captured by a small set of specific concepts and the conceptual query model should be translated to specific textual terms. To achieve this, we employ an expectation maximization algorithm inspired by parsimonious language models (Hiemstra, Robertson, & Zaragoza, 2004).

The paper is organized around a number of research questions that aim to investigate the effectiveness of our proposed conceptual language models and place it in the context of state-of-the-art full-text retrieval systems. These questions are defined as follows:

[1] A document is typically represented by far more terms than concepts.

1. To estimate a conceptual query model we propose a method that looks at the top-ranked documents in an initially retrieved set (Section 4.1). In order to assess the effectiveness of this step, we compare the results of using these concepts with a standard language modeling approach. Moreover, since this method relies on pseudo-relevant documents from an initial retrieval run, we also compare the results of our conceptual query models to another, established pseudo-relevance feedback algorithm based on relevance models. We ask: What is the relative retrieval effectiveness of this method with respect to the standard language modeling and conventional pseudo-relevance feedback approach?
2. For the estimation of both the conceptual query model and generative concept model we apply an iterative EM algorithm which emphasizes more informative terms. We ask: What is the impact of applying this algorithm compared to conventional estimates in terms of retrieval effectiveness?
3. The proposed method based on conceptual language models is dependent on a number of parameters. We ask: What is the sensitivity of the method to its parameter settings? How robust are the results across different collections and test sets?
4. By definition, curated knowledge is domain specific. So we ask the question: How portable is our conceptual language model? What are the results of the model across multiple test collections? Can we say anything about which evaluation measures are helped most using our model? Is it mainly a recall or precision-enhancing device?

We make the following contributions in this article:

- We propose a method for determining the concepts that are most likely to be associated with a given query, which allows effective conceptual (blind) relevance feedback. Moreover, this explicit conceptual query representation may be used as a means of suggesting query-related concepts to the user.
- We propose generative concept models, that are used to generate terms for concepts related to the query. Besides this particular application, they may also be employed to determine semantic relatedness.
- Finally, we provide an empirical comparison of our proposed method to existing relevance feedback models.

The remainder of this paper is organized as follows: We discuss related work in Section 2. We then describe our retrieval framework and our conceptual language models are introduced next. We describe our experimental setup in Section 5 and report on the outcomes of our experimental evaluation and discuss our findings in Section 6. We end with a concluding section.

## 2. Related work

Work related to our proposed conceptual language models may be found in overlapping areas, viz. query expansion, conceptual retrieval, and cluster-based retrieval. These will be discussed in this section.

Query expansion aims at bridging the vocabulary gap between queries and documents by adding and reweighing terms in the original query (Voorhees, 1994). Query expansion approaches can be local or global (Xu & Croft, 1996). Local query expansion methods try to take into account the context of a query; one might, for example, consider a user's history or profile, in order to automatically enrich queries (Korfhage, 1984). Much later, similar notions were adopted in a language modeling setting (Bai et al., 2008). Finkelstein et al. (2002) propose to use the local context of query terms as they appear in documents to locate additional query terms.

Relevance feedback is a form of local query expansion that relies on the analysis of documents from an initial retrieval run. The retrieved documents serve as examples to select additional query terms (Rocchio, 1971). Pseudo-relevance feedback methods assume the top-ranked documents to be relevant, but explicit or implicit relevance judgements from users may also be used (Anick, 2003; Keskustalo, Järvelin, & Pirkola, 2008; Vakkari, Jones, Macfarlane, & Sormunen, 2004; Xu & Croft, 1996). The recent interest of the semantic web community regarding models and methods related to ontologies have also sparked a renewed interest in using ontological information for relevance feedback (Bhogal, Macfarlane, & Smith, 2007; Rocha, Schwabe, & Aragao, 2004). In a language modeling setting, local query expansion has been applied to estimate query language models (Lafferty & Zhai, 2003; Tao & Zhai, 2006) or relevance models (Lavrenko & Croft, 2001); we elaborate on the latter in Section 3. Our method is related to these approaches in that it also looks at the results of an initial retrieval run. Instead of looking at the terms in these documents, however, we consider the concepts associated with the documents.

Global query expansion uses global collection statistics or "external" knowledge sources such as concept languages to enhance the query. For example, concepts and lexical-syntactic relations as defined in a thesaurus have been used for query expansion, with varying degrees of effectiveness (Bai, Song, Bruza, Nie, & Cao, 2005; Gao, Nie, & Bai, 2005; Meij & de Rijke, 2007; Roberts et al., 1984; Voorhees, 1994).

Our method can be viewed as a combination of a local and global expansion method; a local expansion method is used to obtain a conceptual representation of a query, whereas a global method is used to translate the conceptual representation to a textual query contribution.

Using a conceptual representation obtained from pseudo-relevance feedback has been investigated by different researchers in the biomedical domain. Srinivasan (1996) proposes adding concepts directly to an initial query and reports the largest improvement in retrieval effectiveness when another round of blind relevance feedback on vocabulary terms is applied afterwards. This method is similar to ours, although there are distinct differences in her approach and evaluation. For one, Srinivasan (1996) creates a separate "concept index" in which tokenized concept labels are used as terms. In this way, searching using a concept labeled "Stomach cancer" also matches the related, but clearly different concept "Breast cancer" because they share the word "cancer". In our opinion, this obfuscates the added value of using clearly defined concepts; searching with a textual representation containing the word "cancer" will already result in matching related concepts. Therefore, we decide to use unique concept identifiers in our conceptual representation. Srinivasan (1996) concludes that concepts are beneficial for retrieval, but remarks that the OHSUMED collection used for evaluation was quite small. Our research uses the larger TREC Genomics test collections and, additionally, investigates the use of document level annotations in another domain using the CLEF domain specific test collections. Finally, we remark that our proposed model is an extension of the language modeling retrieval framework, whereas Srinivasan (1996) extends a vector space retrieval model. Camous, Blott, and Smeaton (2006) also use the annotations of the top-5 retrieved documents to obtain a conceptual query representation, but incorporate them in a different fashion. The authors use them to create a new ranked list of documents, which is subsequently combined with the initially retrieved documents. In contrast, we explicitly update the original query model.

All of the methods based on concept languages need a way of mapping between the concepts and their textual representation. Where the described approaches look for exact occurrences of the concepts in the text, we use the vocabulary terms *associated* with concepts to make this connection, as detailed in Section 4.

Taking a step back from query expansion, many different ways of directly improving text-based retrieval by incorporating concepts or a concept language have been proposed. For example, the entries from a concept language may be used to define the indexing terms employed by the retrieval system. In the absence of a concept language, similar information might be derived from statistical methods (Joyce et al., 1958; Salton, 1971; Sparck-Jones & Jackson, 1970). For instance, a co-occurrence analysis of the entire collection might be applied to estimate dependencies between vocabulary terms (Bai et al., 2005; Chung, 2004). Alternatively, term dependencies may be determined on a query-dependent subset of the collection, such as a set of initially retrieved documents (Metzler & Croft, 2005; Mitra, Singhal, & Buckley, 1998; Xu & Croft, 1996). These dependencies may then be employed to locate terms related to the initial query.

One of the first attempts at automatically relating concepts with text was introduced in the 1980s. Giger (1988) incorporated a mapping between concepts from a thesaurus and words as they appear in the collection. The main motivation was to move beyond text-based retrieval and bridge the semantic gap between the user and the information retrieval system, a motivation closely related to ours. His algorithm first defines *atomic concepts*, which are string-based concept to term mappings. Then, documents are placed in disjoint groups based on so-called elementary logical conjuncts, which are defined through the atomic concepts. At retrieval time, the query is parsed and the sets of documents with the lowest distance to the requested concepts are returned. His ideas relate to recent work done by Zhou, Hu, Zhang, Lin, and Song (2006) and Zhou, Hu, and Zhang (2007), who use so-called *topic signatures* to index and retrieve documents. These signatures are comprised of recognizing named entities within each document and query; when named entities are not available, term pairs are used. Their named entity recognition step is automated and might not be completely accurate; we suspect that the errors in this concept detection process do not strongly affect retrieval performance because *pairs* of concepts (topic signatures) are used for retrieval. In our method, we rely on manually curated concept annotations, making the topic signatures superfluous.

Trieschnigg, Kraaij, and Schuemie (2007) also use named entity recognition to obtain a conceptual representation of queries and documents. They conclude that searching only with an automatically obtained conceptual representation seriously degrades retrieval when searching for short documents (citations). Interestingly, the same approach performs on par with text-only search when larger documents (full-text articles) are retrieved.

Instead of using named entity recognition, Gabrilovich and Markovitch (2007) employ document-level annotations, in the form of Wikipedia categories. They represent the categories as term vectors, where the individual term weights are determined using TF.IDF scores from the documents that are labeled with the concept at hand. In this way, the strength between vocabulary terms and concepts can be quantified, which can subsequently be used to generate vectors of concepts for a piece of text—either a document or query. This approach is similar to the topic modeling approach described by Wei (2007), which uses Open Directory Project (ODP) concepts in conjunction with generative language models. Instead of using concept–document associations, however, she uses an ad hoc approach based on the descriptions of the concepts in the concept language (in this case, ODP categories). Our conceptual language models are related to these approaches in that they also bridge between concepts and terms. We, however, use an iterative EM algorithm in tandem with a statistical translation model to establish the association between terms and concepts. Interestingly, all of these approaches open up the door to providing conceptual relevance feedback to users. Instead of suggesting vocabulary terms that are related to the query, we can now suggest related concepts that can, for example, be used for navigational purposes (Keskustalo et al., 2008; Meij & de Rijke, 2007; Silveira & Ribeiro-Neto, 2004; Vakkari et al., 2004). Trajkova and Gauch (2004) describe another possible application; their system keeps track of a user's history by classifying visited web pages onto the concepts from the ODP.

Further examples of mapping queries to conceptual representations can be found in the area of web query classification. Broder et al. (2007) use a pseudo-relevance feedback technique to classify rare queries into a commercial taxonomy of web queries, with the goal to improve web advertisements. A classifier is used to classify the highest ranked results, and these classifications are subsequently used to classify the query by means of voting. We use a similar method to obtain the conceptual representation of our query described in Section 4.1, with the important difference that all our documents have been manually classified. Mishne et al. (2006) classify queries into taxonomies using category-based web services. Shen, Sun, Yang, and Chen (2006) improve web query classification by mapping the query to concepts in an intermediate taxonomy which in turn are linked to concepts in the target taxonomy. In our work, we use a single concept taxonomy which is used as a pivot language to improve the textual query model. Chen, Xue, and Yu (2008) use a taxonomy to suggest keywords. After mapping the seed keywords to a concept hierarchy, content phrases related to the found concepts are suggested. In our approach the concepts are used to update the query model, i.e., to update the probabilities of terms based on the found concepts rather than the addition of related discrete terms or phrases.

Concepts can be recognized at different levels of granularity, either at the term level, by recognizing concepts in the text, or at the document level, by using document-level annotations or categories. While the former can be described as a form of *concept-based indexing* (Lancaster, 1982), the latter is more related to text classification. Indeed, the mapping of vocabulary terms to concepts as described above is in fact a text (or concept) classification algorithm (Sparck-Jones & Needham, 1968).

Work done on cluster-based retrieval can be viewed as a variation on the same theme; in our case the clusters are defined by the concepts that are associated with the documents in the collection. Kurland et al. (2004), for example, determine overlapping clusters of documents in a collection, which are considered *facets* of the collection. They use a language modeling framework in which their aspect-x algorithm smoothes documents based on the information from the clusters and the strength of the connection between each document and cluster. Liu and Croft (2004) evaluate both the direct retrieval of clusters and cluster-based smoothing. Their CBDM model is a mixture between a document model, a collection model, and the cluster each document belongs to, which is able to significantly outperform a standard query-likelihood baseline. Instead of smoothing documents, Minker, Wilson, and Zimmerman (1972) use cluster-based information for query expansion. The authors evaluate their algorithm on several small test collections, without achieving any improvements over the unexpanded queries. More recently, Lee, Croft, and Allan (2008) have shown that detecting clusters in a set of (pseudo-)relevant documents is helpful for identifying dominant documents for a query and, thus, for subsequent query expansion, a finding which was corroborated on different test collections by Kurland (2008). These approaches all exploit the notion that "associations between documents convey information about the relevance of documents to requests" (Jardine & van Rijsbergen, 1971). Indeed, if we have evidence that a given concept is relevant for a particular query, it is natural to assume that all documents labeled with this concept have a higher prior probability of being relevant to the query. This is the main motivating idea for our current work.

## 3. The KL-divergence retrieval framework

The success of generative language models in statistical machine translation and automatic speech recognition inspired several IR researchers to re-cast IR in a generative probabilistic framework, by representing documents as generative probabilistic models. Such models can be used to compute the probability of observing a sequence of terms, by computing the product of the probabilities of observing the individual terms. The first published application of generative models for IR was based on the multiple Bernoulli distribution (Ponte & Croft, 1998), but the simpler multinomial unigram model became the mainstream model (Hiemstra, 1998; Miller, Leek, & Schwartz, 2000). Recent work has addressed some of the shortcomings of the multinomial model for modeling text and considers the Dirichlet compound multinomial distribution instead (Xu & Akella, 2008). This distribution provides a better model of the 'burstiness' of language and the authors show significant improvements over the standard multinomial model. Whether it is a better candidate for representing text in our current context remains a subject for future work.

In the multinomial unigram model, each document $D$ is represented as a multinomial probability distribution $P(t|\theta_D)$ over all the terms $t$ in the vocabulary. At retrieval time, each document is ranked according to the likelihood of having generated the query, i.e., the probability that the query terms $(t \in Q)$ are sampled independently and identically from the document language model (Hiemstra, 1998):

$$\text{Score}\ (Q,\ D) \propto P(Q|D) = \prod_{t\in Q} P(t|\theta_D)^{n(t,Q)}, \tag{1}$$

where $n(t,Q)$ denotes the count of term $t$ in query $Q$. This model was generalized soon after, by realizing that an information need can also be modeled by a language model. In this way, a more general and flexible retrieval model can be obtained by using a comparison of two language models as the basis for ranking. Several authors proposed the use of the Kullback–Leibler (KL)-divergence for ranking, since it is a well established measure for the comparison of probability distributions with some intuitive properties—it always has a non-negative value and equal distributions receive a zero divergence value (Lafferty et al., 2001; Ng, 2001; Xu & Croft, 1999). Using KL-divergence, documents are scored by measuring the divergence between a query model $\theta_Q$ and each document model $\theta_D$. Since we want to assign a high score for high similarity and a low score for low similarity, the KL-divergence is negated for ranking purposes. More formally, the score for each query–document pair using the KL-divergence retrieval model is:

$$\text{Score}\ (Q,\ D) = -\mathrm{KL}(\theta_Q\|\theta_D) = -\sum_{t\in\mathscr{V}} P(t|\theta_Q)\log\frac{P(t|\theta_Q)}{P(t|\theta_D)} = -\sum_{t\in\mathscr{V}} P(t|\theta_Q)\log P(t|\theta_D) + \sum_{t\in\mathscr{V}} P(t|\theta_Q)\log P(t|\theta_Q), \tag{2}$$

where $\mathscr{V}$ denotes the set of all terms used in all documents in the collection. KL-divergence is also known as the relative entropy, which is defined as the cross-entropy of the observed distribution (in this case the query) as if it was generated by a reference distribution (in this case the document) minus the entropy of the observed distribution. KL-divergence can also be measured in the reverse direction (also known as document likelihood), but this leads to poorer results for ad-hoc search tasks (Lavrenko, 2004). The entropy of the query, $\sum_{t\in\mathscr{V}} P(t|\theta_Q)\log P(t|\theta_Q)$, is a query specific constant and can thus be ignored for ranking purposes. In fact, one could argue that ranking on just the cross-entropy term provides a more concise ranking formula and is a suitable distance measure for comparing probability distributions in its own right (Kraaij, 2004). When the query model is generated using the empirical, maximum-likelihood estimate (MLE) on the original query, i.e.,

$$P(t|\tilde{\theta}_Q) = P(t|Q) = \frac{n(t,Q)}{|Q|}, \tag{3}$$

where $|Q|$ indicates the length of the query, it can be shown that documents are ranked in the same order as using the query likelihood model from Eq. (1) (Zhai, 2002). In fact, a query is just a verbal expression of an underlying information need. The query model is therefore an estimate of the model for the underlying information need, sometimes called a relevance model (Lavrenko & Croft, 2001). This initial estimate can be improved by adding and reweighing terms, using external resources or relevance feedback techniques as described in Section 2. Next, we describe our baseline query modeling (Section 3.1) and document modeling (Section 3.2) approaches. In Section 4 we define our conceptual language models on top of these baseline approaches.

### 3.1. Query models

Relevance models (Lavrenko & Croft, 2001) are one of the baselines we employ. Here, it is assumed that for every information need there exists an underlying relevance model and that the query and relevant documents are random samples from this model. The query model, parametrized by $\theta_Q$, may be viewed as an approximation of this model. However, in a typical retrieval setting improving the estimation of $\theta_Q$ is problematic because we have no or only limited training data. The authors present two methods for estimating relevance models without training data by constructing models from the queries and a set of pseudo-relevant documents, using different independence assumptions. They determine the probability of observing $t$ after having observed $Q$ as:

$$P(t|\hat{\theta}_Q) \approx P(t|q_1,\ldots,q_k) = \frac{P(t,q_1,\ldots,q_k)}{P(q_1,\ldots,q_k)} = \frac{P(t,q_1,\ldots,q_k)}{\sum_{t'} P(t',q_1,\ldots,q_k)}, \tag{4}$$

where $q_1,\ldots,q_k$ are the individual query terms. Under their method 2, the query terms are independent of each other, but keep their dependence on $t$:

$$P(t,q_1,\ldots,q_k) = P(t)\prod_{i=1}^{k}\sum_{D\in\mathscr{D}_Q} P(q_i|\theta_D)P(\theta_D|t), \tag{5}$$

where $\mathscr{D}_Q$ is a set of pseudo-relevant documents and

$$P(\theta_D|t) = \frac{P(t|\theta_D)P(D)}{P(t)}. \tag{6}$$

Then, in order to obtain a query model that is a better estimate of the information need, the initial query $P(t|\tilde{\theta}_Q)$ may be interpolated with the expanded part $P(t|\hat{\theta}_Q)$ (Balog, Weerkamp, & de Rijke, 2008; Kurland, Lee, & Domshlak, 2005; Rocchio, 1971; Zhai & Lafferty, 2001). Effectively, this reweighs the initial query terms and provides smoothing for the relatively sparse initial sample:

$$P(t|\theta_Q) = (1-\lambda_Q)P(t|\tilde{\theta}_Q) + \lambda_Q P(t|\hat{\theta}_Q). \tag{7}$$

In the next section, we will describe how we extend this work by leveraging conceptual knowledge in the form of document annotations to improve the estimation of $P(t|\hat{\theta}_Q)$. We discuss the issue of setting the smoothing parameter $\lambda_Q$ in Section 5.3.

### 3.2. Document models

It is an essential condition for retrieval models that are based on measuring the probability or cross-entropy of observed data given a reference generative model, that the reference model is adequately smoothed. Smoothing is applied both to avoid zero-frequency problems occurring with a MLE approach and to account for general and document-specific language use. We adopt Jelinek–Mercer smoothing by considering each document to be a mixture of a document-specific model and a more general background model. Thus, each document model is estimated as the MLE of each term in the document $P(t|D)$, linearly interpolated with a background language model $P(t)$, which in turn is calculated as the likelihood of observing $t$ in a sufficiently large corpus, such as the document collection (Jelinek & Mercer, 1980; Zhai & Lafferty, 2004):

$$P(t|\theta_D) = \lambda_D P(t|D) + (1-\lambda_D)P(t). \tag{8}$$

We address the parameter setting procedure for $\lambda_D$ in Section 5.5. Now that we have described the main components of our framework, we will zoom in on our proposed methods.

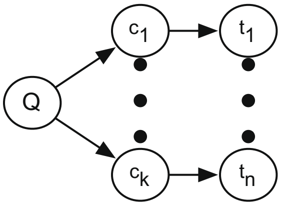


**Fig. 1.** Dependence network for our conceptual language models.

## 4. Conceptual language models

Our goal is to utilize the knowledge that is encapsulated in a concept language to enhance the estimation of the query model $\theta_Q$. To this end, we use the concepts as a pivot language (Kraaij & de Jong, 2004) in a double translation, similar to the method proposed by Berger and Lafferty (1999). Specifically, we utilize the concepts that are associated with a query to find terms related to these concepts in order to estimate the expanded part of the query model, $P(t|\hat{\theta}_Q)$. Fig. 1 shows a graphical representation of the dependencies of this process.

Put differently, first we translate the query into a set of relevant concepts. Next, the vocabulary terms associated with the concepts are considered as possible terms to include in the query model. More formally, for a query $Q$ and concepts $c \in \mathscr{C}$:

$$P(t|\hat{\theta}_Q) = \sum_{c \in \mathscr{C}} P(t|c)P(c|Q), \tag{9}$$

where we assume that the probability of selecting a term is only dependent on the concept once we have selected that concept for the query.

Two components need to be estimated: $P(t|c)$, to which we refer as a *generative concept model*, and $P(c|Q)$, to which we refer as a *conceptual query model*. As to the former, we will need to associate terms with concepts in the concept language. While the concepts may be directly usable for retrieving documents (Hersh, Hickam, Haynes, & McKibbon, 1994; Srinivasan, 1996; Trieschnigg et al., 2009), we *associate* each concept with a weighed set of most characteristic terms using a multinomial unigram model. To this end we consider the documents that are annotated using $c$ as bridges between the concept and terms, by representing concepts as multinomial distributions over terms, $P(t|c)$. Generative concept models will be detailed further in Section 4.2 below.

The second component—the conceptual query model $P(c|Q)$—is a distribution over concepts specific to the query. In some settings, concepts are provided with a query or as part of a query, see, e.g., the PubMed search interface (Herskovic, Tanaka, Hersh, & Bernstam, 2007), some early TREC ad-hoc tracks (6–8 in particular), and the INEX Entity Ranking track where Wikipedia categories are used (de Vries, Vercoustre, Thom, Craswell, & Lalmas, 2007). If this is not the case, however, we may leverage the document annotations to approximate this step: this is what we do in the next section.

### 4.1. Conceptual query modeling

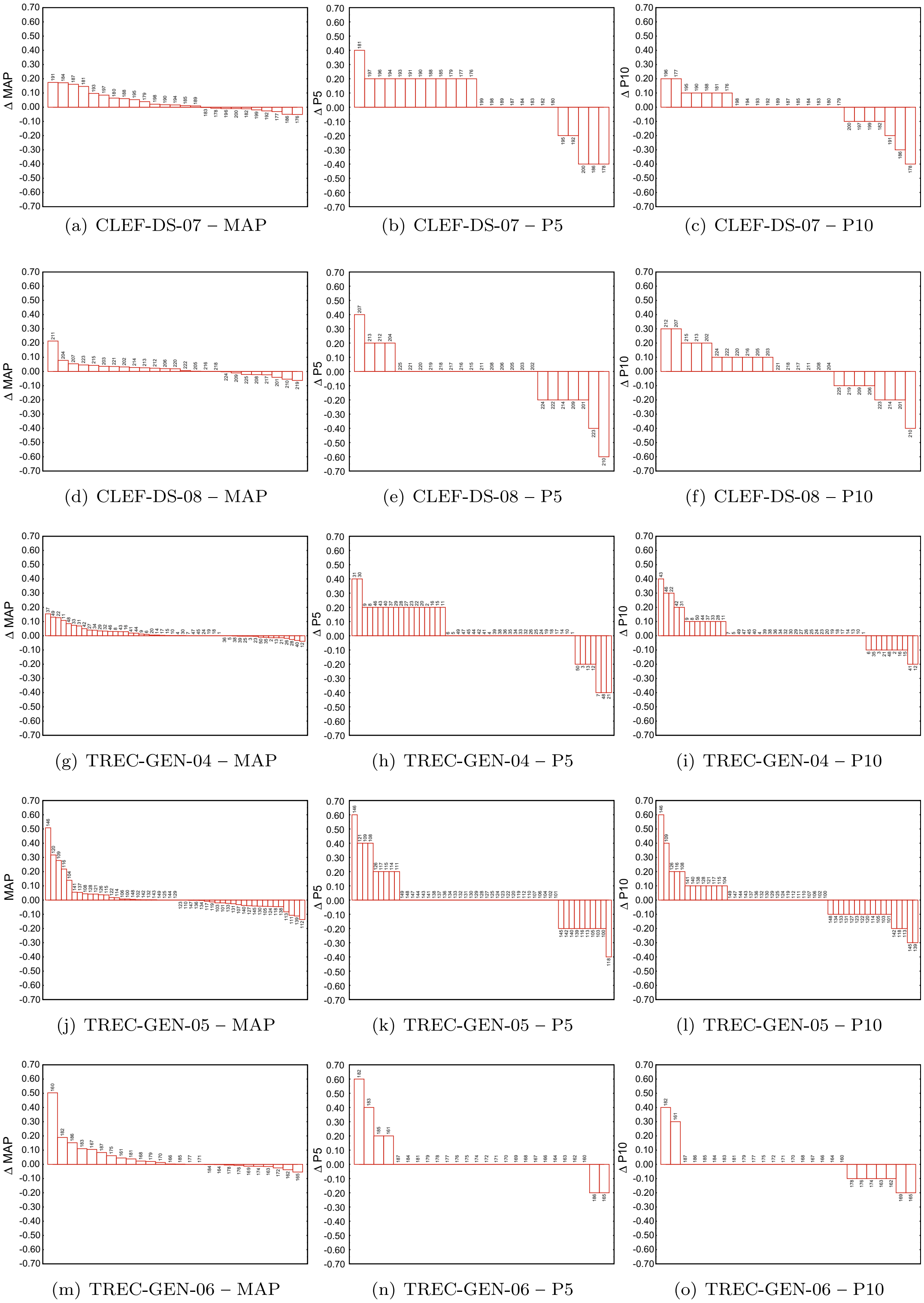
We now turn to defining $P(c|Q)$, the conceptual query model. Contrary to the alternatives mentioned at the end of the previous section, concepts are not provided with a query in a typical IR setting and need to be inferred, estimated, or recognized (Wei, 2007; Zhou et al., 2007). In this paper, we formulate the estimation of concepts relevant to a query in a standard language modeling manner, by determining which concepts are most likely given documents relevant to the query. Alternatively, we could involve the end user and ask which documents, associated concepts, or terms are relevant. Since we do not have access to such assessments, however, we resort to using pseudo-relevance methods. In recent work we studied different approaches of estimating a conceptual query model and concluded that using feedback documents is far more effective than using, e.g., string matching methods that try to recognize concepts in the query (Trieschnigg et al., 2009).

Like Lavrenko and Croft (2001), we view the process of obtaining a conceptual query model as a sampling process from a number of representative sources. The user has a notion of documents satisfying her information need, randomly selects one of these, and samples a concept from its representation. Hence, the conceptual query model is defined as follows:

$$P(c|Q) = \sum_{D \in \mathscr{D}_Q} P(c|D)P(D|Q). \tag{10}$$

Here, $\mathscr{D}_Q$ is a set of pseudo-relevant documents returned by an initial retrieval run using the textual query. $P(c|D)$ is the concept language model of the document, the estimation of which is discussed in the next paragraph. Note that we assume that the probability of observing a concept is independent of the query once we have selected a document given the query, i.e., $P(c|D,Q) = P(c|D)$. The term $P(D|Q)$ denotes the probability that document $D$ is chosen from $\mathscr{D}_Q$ given $Q$, which is obtained using the retrieval scores.

We assume that pseudo-relevant documents are a good source from which we can sample the conceptual query model. Indeed, manual inspection shows that they are annotated with many relevant concepts, but also that they contain a lot of



(a) CLEF-DS-07 – MAP (b) CLEF-DS-07 – P5 (c) CLEF-DS-07 – P10

(d) CLEF-DS-08 – MAP (e) CLEF-DS-08 – P5 (f) CLEF-DS-08 – P10

(g) TREC-GEN-04 – MAP (h) TREC-GEN-04 – P5 (i) TREC-GEN-04 – P10

(j) TREC-GEN-05 – MAP (k) TREC-GEN-05 – P5 (l) TREC-GEN-05 – P10

(m) TREC-GEN-06 – MAP (n) TREC-GEN-06 – P5 (o) TREC-GEN-06 – P10

**Fig. 2.** Per-topic breakdown of the improvement of conceptual language models over the query-likelihood baseline for all test collections, on various evaluation measures and sorted in decreasing order. A positive value indicates an improvement over the baseline. The vertical labels indicate the topic identifiers.

noise: some concepts are very frequent for all documents and, despite being related to the query, not very informative. Sampling from the maximum likelihood estimate for these documents would thus result in very general conceptual query models. Therefore, to re-estimate the probability mass of the concepts in the sampling process, we use a *parsimonious* language model. Table 3 illustrates the difference between a maximum likelihood estimation and a parsimonious estimation. It shows the concepts (in this case MeSH terms) with the highest probability for topic 186 from the TREC Genomics 2006 test collection. The conceptual query model based on the parsimonious document models contains more specific—and thus more useful—concepts, such as "Presenilin-1" and "Presenilin-2". The model based on maximum likelihood estimates includes more general concepts such as "Humans", which are relevant but too general to be useful for searching. In the next section we detail how re-estimation is performed.

### 4.2. Generative concept models

Given Eq. (9), our goal is to arrive at a probability distribution $P(t|c)$ over vocabulary terms for each concept in the concept language used for annotating the documents. We determine the level of association between a term and a concept by looking at the way trained annotators have labeled the documents. In the end, this method defines the parameters of a generative language model for each concept: a *generative concept model*. We determine the strength of association between a concept $c$ and a term $t$ by determining the probability of observing $t$ given $c$:

$$P(t|c) = \frac{P(t,\ c)}{P(c)}. \tag{11}$$

Concepts that are used to annotate documents may have different characteristics from other *parts* of a document, such as title and content. Annotations are selected by human indexers from a concept language while the remaining content consists of free text. Since the terms that make up the document are "generated" using a different process than the concepts, we may assume that $t$ and $c$ are independent and identical samples given a document $D$ in (or with) which they occur. So, the probability of observing both $t$ and $c$ is

$$P(t,c) = \sum_{D} P(D)P(c,t|D) = \sum_{D\in\mathscr{D}_C} P(D)P(t|D)P(c|D), \tag{12}$$

where $\mathscr{D}_C$ denotes the set of documents annotated with concept $c$. When we assume each document in this set to have a uniform prior probability of being selected, we obtain

$$P(t|c) = \frac{P(t,\ c)}{P(c)} = \frac{\sum_{D\in\mathscr{D}_C} P(D)P(t|D)P(c|D)}{P(c)} \propto \frac{1}{P(c)} \sum_{D\in\mathscr{D}_C} P(t|D)P(c|D). \tag{13}$$

Hence, it remains to define three terms: $P(c),\ P(t|D)$, and $P(c|D)$. First, the term $P(c)^{-1}$ functions as a penalty for frequently occurring and thus relatively non-informative concepts. We estimate this term using MLE on the document collection:

$$P(c) = \frac{\sum_D n(c,\ D)}{\sum_{c'}\sum_{D'} n(c', D')}, \tag{14}$$

where $n(c,\ D)$ is the number of times document $D$ is labeled with concept $c$.

Next we turn to $P(x|D)$, for $x \in \{t, c\}$. The size of these models (in terms of the number of words or the number of concepts that receive a non-zero probability) may be quite large, e.g., in the case of a large document collection or in the case of frequently occurring concepts. Moreover, as exemplified above, not all of the observed *events* (where events are either terms or concepts) are equally informative. Some may be common, whilst others may describe the general domain of the document. Earlier, we have assumed that each document is a mixture of document-specific and more general terms (Section 3.2, Eq. (8)); we now generalize this statement to also include concepts. Further, given this assumption, we may update each document model by reducing the amount and probability mass of non-specific events. We do so by iteratively adjusting the individual probabilities in each document, based on a comparison with a large reference corpus such as the collection. More formally, we maximize the posterior probability of $D$ after observing $x$:

$$P(D|x) = \frac{\lambda_x P(x|D)}{(1-\lambda_x)P(x) + \lambda_x P(x|D)}. \tag{15}$$

**Table 3**
A comparison of the concepts with the highest probability $P(c|Q)$ (cf. Eq. (10)) for the TREC Genomics topic: "How do mutations in the Presenilin-1 gene affect Alzheimer's disease". The two columns show the difference between using MLE on the concepts associated with the documents to determine $P(c|D)$, or the EM algorithm given in Eq. (19). Unique concepts are marked in boldface.

| $P(c\|D)$ estimated using MLE | $P(c\|D)$ estimated using Eq. (19) |
|---|---|
| Alzheimer disease | **Presenilin-1** |
| **Humans** | **Presenilin-2** |
| Membrane proteins | Alzheimer disease |
| **Amyloid beta-protein** | **Amyloid precursor, protein secretases** |
| Amyloid beta-protein, precursor | Membrane proteins |
| **Research support, US Gov't, P.H.S.** | Amyloid beta-protein, precursor |

Note that $\lambda_x$ may be set differently for $D$ (Eq. (8)) and $C$. For these estimations, we fix $\lambda_C = \lambda_D = 0.15$ (Hiemstra et al., 2004; Meij & de Rijke, 2008; Meij et al., 2008). We then apply the following EM algorithm until the estimates do not change significantly anymore:

$$\text{E-step}: \ e_x = P(D|x) = \frac{\lambda_C P(x|D)}{(1-\lambda_C)P(x) + \lambda_C P(x|D)}, \tag{16}$$

$$\text{M-step}: \ P_C(x|D) = \frac{n(x,\ D)e_x}{\sum_{x'} n(x',\ D)e_{x'}}. \tag{17}$$

This updating mechanism enables more specific events, i.e., events that are not well-explained by the background model, to receive more probability mass, making the resulting document model more specific. After the EM algorithm has converged, we remove those events with a probability lower than a certain threshold $\delta$. Thus, the resulting document model for terms, $P(t|\hat{\theta}_D)$, to be used in Eq. (13) is given by:

$$P(t|\hat{\theta}_D) = \begin{cases} Z_{D_t} \cdot P_C(t|D) & \text{if } t \in D \text{ and } P_C(t|D) > \delta_t \\ 0 & \text{otherwise}, \end{cases} \tag{18}$$

where $Z_{D_t}$ is a document-specific normalization factor: $Z_{D_t} = 1/\sum_t P_C(t|D)$. Table 4 provides an example of the effects of applying this algorithm on a document from the CLEF document collection (that will be introduced in Section 5). Similarly, the resulting document model for concepts, $P(c|\hat{\theta}_D)$, to be used for $P(c|D)$ in Eq. (13), is given by:

$$P(c|\hat{\theta}_D) = \begin{cases} Z_{D_c} \cdot P_C(c|D) & \text{if } c \in D \text{ and } P_C(c|D) > \delta_c \\ 0 & \text{otherwise}, \end{cases} \tag{19}$$

where $Z_{D_c}$ is a document-specific normalization factor: $Z_{D_c} = 1/\sum_c P_C(c|D)$. Table 3 provides an example of the effects of applying this algorithm on a topic from the TREC document collection (that will be introduced in Section 5). For the experiments in this paper we fix $\delta_t = \delta_c = 0.01$.

## 5. Experimental setup

To answer the research questions specified in the introduction, we set up a number of experiments in which we compare our conceptual language models with other retrieval approaches. Below, we first describe our test collections, the baseline approaches that we use for comparison, our experimental environment, estimation methods, and the method we use for significance testing. In Section 6, we turn to the results of our experiments.

### 5.1. Test collections

The test collections we employ were selected for several reasons. First, our retrieval model requires collections in which the documents have been manually annotated with an appropriate concept language. The TREC and CLEF test collections that we describe below both satisfy this requirement. Moreover, they have been used for evaluating well-defined IR tasks and have relevance assessments based on a sufficiently large pool. Tables 5 and 6 list key characteristics of the test collections we use. All documents (in all test collections) are stemmed using a Porter Stemmer and we do not remove stopwords.

#### 5.1.1. CLEF domain specific 2007–2008

The CLEF domain-specific track evaluates retrieval on structured scientific documents, using bibliographic databases from the social sciences domain as document collections (Petras, Baerisch, & Stempfhuber, 2007; Petras & Baerisch, 2008). The track emphasizes leveraging the structure of data in collections (defined by concept languages) to improve retrieval performance. The 2007 (CLEF-DS-07) and 2008 (CLEF-DS-08) tracks use the combined German Indexing and

**Table 4**
Top 10 stemmed terms for the document model belonging to document CSASA-1-EN-9706464 (entitled "American indian ethnic renewal: red power and the resurgence of identity and culture.") from the CLEF collection.

| $P(t\|D)$ | Estimated using MLE | $P(t\|D)$ | Estimated using Eq. (18) |
|---|---|---|---|
| 0.061 | The | 0.54 | Indian |
| 0.054 | Of | 0.46 | Ethnic |
| 0.045 | Indian | | |
| 0.038 | Ethnic | | |
| 0.028 | In | | |
| 0.028 | American | | |
| 0.021 | A | | |
| 0.021 | Renew | | |
| 0.019 | Cultur | | |
| 0.017 | Ident | | |

**Table 5**
Statistics of the document collections used in this paper. "Avg." indicates the average number of terms or concepts in a document, "Std. dev." the standard deviation, and "Med." the median. The CLEF-DS collection is the smallest in size, whereas the TREC Genomics 2004 collection has the smallest documents on average. Documents in the TREC Genomics 2006 collection have the most concepts assigned per document.

| | Documents | Size | Terms | | | Concepts | | |
|---|---|---|---|---|---|---|---|---|
| | | | Avg. | Std. dev. | Med. | Avg. | Std. dev. | Med. |
| CLEF-DS-07<br>CLEF-DS-08 | 171,319 | 232 MB | 62.3 | 42.3 | 51 | 10.1 | 4.2 | 10 |
| TREC-GEN-04<br>TREC-GEN-05 | 4,591,008 | 20 GB | 174.4 | 113.6 | 171 | 11.4 | 5.1 | 11 |
| TREC-GEN-06 | 162,169 | 12 GB | 4160.3 | 2750.2 | 4525 | 15.1 | 6.1 | 15 |

**Table 6**
Statistics of the topic sets used in this paper. The TREC Genomics 2004 (TREC-GEN-2004) topics are the longest queries, whereas the CLEF-DS-08 has the shortest. The CLEF-DS-07 topics retrieve the most relevant documents on average.

| | Topics | Queries | | | Relevant documents | | | |
|---|---|---|---|---|---|---|---|---|
| | | With rel. docs | Avg. length | Std. dev. length | Total | Avg. | Min. | Max. |
| CLEF-DS-07 | 25 | 25 | 4 | 1.6 | 4530 | 181 | 18 | 497 |
| CLEF-DS-08 | 25 | 25 | 3 | 1.7 | 2133 | 85 | 4 | 206 |
| TREC-GEN-04 | 50 | 50 | 7 | 4.7 | 8268 | 165 | 1 | 697 |
| TREC-GEN-05 | 50 | 49 | 5 | 2.5 | 4584 | 93 | 2 | 709 |
| TREC-GEN-06 | 28 | 26 | 4 | 2 | 1449 | 55 | 2 | 234 |

Retrieval Testdatabase (GIRT) and Cambridge Scientific Abstracts (CSA) databases as their document collection. The GIRT database contains extracts from two databases maintained by the German Social Science Information Centre from the years 1990–2000. The English GIRT collection is a pseudo-parallel corpus to the German GIRT collection, providing translated versions of the German documents (17% of these documents contain an abstract). For the 2007 domain-specific track, an extract from CSA's Sociological abstracts was added, covering the years 1994, 1995, and 1996. Besides the title and abstract, each CSA record also contains subject-describing keywords from the CSA Thesaurus of Sociological Indexing Terms and classification codes from the Sociological Abstracts classification. In this sub-collection, 94% of the records contains an abstract.

We only use the English mono-lingual topics and relevance assessments, which amounts to a total of 50 test topics. The documents in the collection contain three separate fields with concepts, `CLASSIFICATION-TEXT-EN`, `CONTROLLED-TERM-EN` and `CONTROLLED-TERM-EN-MINOR`; we only use the `CLASSIFICATION-TEXT-EN` annotations for the documents.

### 5.1.2. TREC Genomics 2004–2005

The document collection for the TREC 2004 and 2005 Genomics ad-hoc search task (TREC-GEN-04 and TREC-GEN-05) consists of a subset of the MEDLINE database (Hersh et al., 2005, 2006). MEDLINE is the bibliographic database maintained by the US National Library of Medicine (NLM). It currently contains over 16 million biomedical citations from around 5200 journals and several hundred thousand records are added each year. Despite the growing availability of full-text articles on the Web, MEDLINE remains a central access point for biomedical literature. Each Medline record contains free text fields (such as title and abstract), a number of fields containing other metadata (such as publication date and journal), and, most important for our current work, terms from the Medical Subject Headings (MeSH) thesaurus. We only use the main descriptors, without qualifiers. MeSH terms are manually assigned to citations by trained annotators from the NLM. The over 20,000 biomedical concepts in the MeSH thesaurus are organized hierarchically. Relationships between concepts in the MeSH thesaurus are primarily of the "broader/narrower than" type. The "narrower than" relationship is close to expressing hypernymy (is a), but can also include meronymy (part of) relations. One concept is narrower than another if the documents it is assigned to are contained in the set of documents assigned to the broader term. Each MEDLINE record is annotated with 10–12 MeSH terms on average.

It should be noted that the Medical Subject Headings thesaurus is not the most appropriate for Genomics information retrieval, since it covers general biomedical concepts rather than the specific genomics terminology used in the TREC topics (Stokes, Li, Cavedon, & Zobel, 2009). Despite this limited coverage, the thesaurus can still be used to improve retrieval effectiveness, as we will show later.

The document collection for TREC Genomics 2004 and 2005 contains 10 years of citations covering 1993–2004, which amounts to a total of 4,591,008 documents. All documents have a title, 75.8% contain an abstract and 99% are annotated with MeSH terms. For the 2004 track, 50 test topics are available, with an average length of seven terms, cf. Table 6. The 50 topics for 2005 (one of which has no relevant documents) follow pre-defined templates, so-called Generic Topic Types. An example

**Table 7**
Free parameters in the models described in the previous sections.

| Parameter | | Description |
|---|---|---|
| $\lambda_Q$ | Eq. (7) | Interpolation between initial query and expanded query part |
| $\lvert\mathscr{D}_Q\rvert$ | Eqs. (5) and (10) | The size of the set of pseudo-relevant documents |
| $\lvert\mathscr{V}_Q\rvert$ | Eqs. (5) and (13) | The number of terms to use, either for the expanded query part (Eq. (5)) or for each concept (Eq. (13)) |
| $\lvert\mathscr{C}\rvert$ | Eq. (9) | The number of concepts to use for the conceptual query representation |

of such a template is: "Find articles describing the role of [**gene**] in [**disease**]", where the topics instantiate the bold-faced terms. The topics in our experiments are derived from the original topic by only selecting the instantiated terms and discarding the remainder of the template.

### 5.1.3. TREC genomics 2006

The TREC 2006 Genomics track introduced a full-text document collection, replacing the bibliographical abstracts from the previous years (Hersh, Cohen, Roberts, & Rekapalli, 1994). The documents in the collection are full-text versions of scientific journal papers. The files themselves are provided as HTML, including all the journal-specific formatting. Most of the documents (99%) have a valid Pubmed identifier, through which the accompanying MEDLINE record can be retrieved. We use the MeSH terms assigned to the corresponding citation as the annotations of the full-text document.

The 2006 test topics are again based on topic templates and instantiated with specific genes, diseases or biological processes. Thus, we preprocess them in a similar fashion as the topics for the TREC Genomics 2005 track, by removing all the template-specific terms. This test collection has 28 topics, of which two do not have any relevant documents in the collection. The task put forward for this test collection is to first identify relevant documents and then extract the most relevant passage(s) from each document; relevance is measured at the document, passage, and aspect level. We do not perform any passage extraction and only use the judgments at the document level.[2]

## 5.2. Evaluation measures and significance testing

We report on the following evaluation measures, which are obtained with the trec_eval[3] program: mean average precision (MAP), recall, and early precision (at 5 and 10 retrieved documents). For significance testing, we use a Wilcoxon signed rank test and look for improvements at the $\alpha < 0.05$ level. We use a bold-faced font to indicate the best performing model in our result tables.

## 5.3. Parameter estimation

Given the models introduced in the previous sections, we have a number of parameters to estimate. Table 7 summarizes the parameters that we need to set.

There are various approaches that may be used to estimate these parameters. We choose to optimize the parameter values by determining the mean average precision for each set of parameters and show the results of the best performing settings. For $\lambda_Q$ we sweep in the interval $[0,1]$ with increments of 0.1. The other parameters are investigated in the range $[1,10]$ with increments of 1. We determine the MAP scores on the same topics that we present results for, similar to Liu and Croft (2004), Metzler and Croft (2005), Mitra et al. (1998), Lafferty et al. (2001) and Zhai and Lafferty (2004). While computationally expensive (exponential in the number of parameters), it does provide us with an upper bound on the retrieval performance that one might achieve using the described models.

## 5.4. Complexity and implementation

For all our experiments we use the Lemur Toolkit.[4] As to the complexity of our methods, we need to calculate two terms additional to the standard language modeling estimations (Lafferty et al., 2001): the generative concept models (offline) and the conceptual query model (online). The former is most time-consuming, with a maximum complexity per concept proportional to the number of terms in the vocabulary, the number of documents annotated with the concept, and the number of EM iterations. The advantage of this step, however, is that it can be performed offline. Determining a conceptual query model is, in terms of efficiency, comparable to standard pseudo-relevance feedback approaches except for the addition of the number of EM iterations. In general, the additional overhead of the online calculations scales well and its performance is acceptable for all test collections.

[2] 2007 was the final year of the TREC Genomics track and used the same document collection as 2006. However, in this edition a new task was introduced and because of the different nature of that task, we do not perform experiments using those topics.

[3] trec_eval is available from the TREC web site for registered participants at http://www.trec.nist.gov.

[4] See http://www.sourceforge.net/projects/lemur/.

### 5.5. Baselines

We use two baseline retrieval approaches for comparison purposes, viz. query likelihood and relevance models, which are described next. Table 8 shows an example of the generated query models for these baseline approaches and the CLEF 2008 query "Shrinking cities". As our first baseline, we employ a run based on the KL-divergence retrieval method and set $\lambda_Q = 1$ (cf. Section 3, Eq. (7)). This uses only the information from the initial, textual query and amounts to performing retrieval using query likelihood.

It has been shown that making the document interpolation parameter $\lambda_D$ (cf. Eq. (8)) dependent on the document length yields superior performance (Zhai & Lafferty, 2004). Thus, for our baseline experiments we set $\lambda_D = \frac{\mu}{|D|+\mu}$ and $(1 - \lambda_D) = \frac{|D|}{|D|+\mu}$, where $\mu$ is a hyperparameter that we set to the average document length (for each individual test collection). Effectively, this results in Bayesian smoothing using a Dirichlet prior (Chen & Goodman, 1996). All the results on which we report use this baseline as their initially retrieved document set.

Since our concept language models also rely on pseudo-relevance feedback (PRF), we use the text-based PRF method introduced by Lavrenko and Croft (2001) ("model 2") which was described in Section 3, Eq. (5) as another baseline. The functional form of our conceptual query model is reminiscent of Lavrenko and Croft's (2001) "model 1" and we also evaluated "model 1" as a text-based pseudo-relevance feedback baseline. We found that its performance was inferior to "model 2" on all test collections—a finding in line with results obtained by Lavrenko and Croft (2001) as well as other researchers (Balog, 2008). Consequently, we use "model 2" in our experiments and refrain from mentioning the results of "model 1".

## 6. Results and discussion

Now that we have detailed our conceptual language modeling approach (Section 4) and laid out the experimental environment (Section 5), we present the results of the experiments aimed at answering the research questions listed in the introduction. First, we look at the performance of the query likelihood model, which we use as our baseline. We emphasize that all the other models that we evaluate use the initial ranking from the query likelihood model as a set of pseudo-relevant documents. Whether improving upon this baseline will also improve the estimations based on it is a question for future work. We then look at the results of applying an established pseudo-relevance feedback algorithm based on relevance models. Next, we evaluate the results of using the conceptual language models as described in Section 4, using the conceptual query models and the generative concept models in conjunction.

We then perform an ablation study, by zooming in on the results after removing each component in the conceptual language models. First, we consider the generative concept models that we use to translate the conceptual query model to free-text terms. We look at the results of using MLE, i.e., without applying the EM algorithm described in Section 4.2. Second, since each document in our collections has associated concepts, we may use the conceptual query model in conjunction with the initial query for retrieval, as detailed in Section 6.2.3. Finally, we look at the sensitivity of our model with respect to the individual parameter settings and zoom out in order to see whether we can relate collection-specific properties with the reported results.

### 6.1. Baselines

Table 9 shows the results of the query likelihood model as well as the relevance model—which were introduced in Section 3.1—on the five test collections.

#### 6.1.1. Query likelihood

This model (abbreviated by QL) uses MLE on the initial query to build a query model, by distributing the probability mass evenly among the terms in the topic, cf. Eq. 3. First, we note that the results obtained for the query likelihood model are comparable to or better than the mean results of all the participating groups in the respective TREC Genomics (Hersh et al., 1994, 2005, 2006) and CLEF domain specific tracks (Petras et al., 2007; Petras & Baerisch, 2008). As to the TREC Genomics test

**Table 8**
Concepts or stemmed terms with the highest probability in the query models for the CLEF domain specific topic "Shrinking cities" generated by the query-likelihood baseline (QL; Eq. (3)), relevance models (RM; Eq. (5)), conceptual query model (EC; Eq. (10)), and the conceptual language models (GC; Eq. 9).

| QL | | RM | | EC | | GC | |
|---|---|---|---|---|---|---|---|
| 0.5000 | Citi | 0.2718 | Citi | 0.2500 | urban sociology | 0.2161 | Citi |
| 0.5000 | Shrink | 0.2500 | Shrink | 0.2500 | urban planning | 0.2000 | Shrink |
| | | 0.0241 | Of | 0.2500 | town planning | 0.1642 | Urban |
| | | 0.0235 | Develop | 0.2500 | town development | 0.0899 | Town |
| | | 0.0152 | Popul | | | 0.0890 | Develop |
| | | 0.0136 | Town | | | 0.0831 | Plan |
| | | 0.0099 | Economi | | | 0.0466 | Hous |
| | | 0.0094 | Sociolog | | | 0.0402 | Sociolog |

**Table 9**
Results of the baselines: query likelihood (QL) and the best performing run using relevance models, method 2 (RM). The right-most column indicates the relative difference between the query likelihood and relevance model scores.

| | | QL | RM | |
|---|---|---|---|---|
| CLEF-DS-07 | RelRet/TotalRel | 2289/4530 | **2430**/4530 | +6.2% |
| | P5 | 0.5120 | **0.5440** | +6.2% |
| | P10 | **0.5080** | 0.5040 | −0.8% |
| | MAP | 0.1952 | **0.2061** | +5.6% |
| CLEF-DS-08 | RelRet/TotalRel | 1468/2133 | **1473**/2133 | +0.3% |
| | P5 | 0.5280 | **0.5680** | +7.6% |
| | P10 | 0.4680 | **0.4800** | +2.6% |
| | MAP | 0.2819 | **0.2856** | +1.3% |
| TREC-GEN-04 | RelRet/TotalRel | 3847/8268 | **4205**/8268 | +9.3%$^{*}$ |
| | P5 | 0.5160 | **0.5680** | +10.1% |
| | P10 | 0.4800 | **0.5340** | +11.2%$^{*}$ |
| | MAP | 0.2856 | **0.3306** | +15.8%$^{*}$ |
| TREC-GEN-05 | RelRet/TotalRel | 2825/4584 | **3031**/4584 | +7.3%$^{*}$ |
| | P5 | 0.4122 | **0.4163** | +1.0% |
| | P10 | 0.3776 | **0.3857** | +2.1% |
| | MAP | 0.2153 | **0.2368** | +10.0% |
| TREC-GEN-06 | RelRet/TotalRel | 1078/1449 | **1160**/1449 | +7.6% |
| | P5 | 0.4154 | **0.4308** | +3.7% |
| | P10 | 0.4154 | **0.4346** | +4.6% |
| | MAP | 0.2731 | **0.2993** | +9.6%$^{*}$ |

collections, we do not perform any of the elaborate and knowledge-intensive preprocessing of the queries and/or documents that is common in this domain (Trieschnigg, Kraaij, & de Jong, 2007). Even without applying such explicit domain-specific knowledge, our baseline outperforms many systems that do.

#### 6.1.2. Relevance models

The runs based on relevance models (abbreviated by RM) use the retrieved documents from the query likelihood run to construct an improved query model which is subsequently used for retrieval. The optimal parameter settings for the relevance model, with which we obtain these results are determined in the same fashion as for our conceptual language models, i.e., we sweep over all possible values for $\lambda_Q$ (cf. Eq. (7)) and try varying numbers of documents and terms to find the optimal performance in terms of MAP.

Table 9 shows the results of the baseline QL model and the RM model. We observe that, on the CLEF collections, the RM runs show improvements over the baseline in terms of mean average precision (+6% and +1% for the 2007 and 2008 collection, respectively), average recall (+6% and +0.3%) and early precision (P@5: +6%, +8%). None of these differences is significant, however. Results on the individual CLEF-DS-07 topics show that three of the topics substantially increase average precision (a difference of more than 0.05), whereas only one topic decreases. The number of CLEF-DS-08 topics which improve in terms of average precision is about the same as the number which are hurt, causing the modest improvement.

The RM runs on the TREC Genomics collections do show significant differences compared to the QL baseline. For the 2004 query set, average precision (+17%), recall (+9%) and early precision (P@10: +12%) increase significantly. TREC-GEN-06 shows a larger significant improvement on mean average precision (10%). Recall and precision show improvements although they are not significant. Similar to the CLEF collections, TREC-GEN-05 shows a positive difference on average but, besides recall, none of the changes are significant. The increase in mean average precision on the TREC 2005 topics can be mainly attributed to a single topic which strongly benefits from using relevance models.

These findings regarding pseudo-relevance feedback using relevance models, i.e., where some topics are helped and some topics are hurt, are often found when applying pseudo-relevance feedback.

### 6.2. Conceptual language models

We now turn to the results of the conceptual language model presented in Section 4. Recall that this model consists of three steps. First, each query is mapped onto a conceptual query model, i.e., a distribution over concepts relevant to the query using Eq. (10). The concepts found are then translated back to terms using Eq. (13) in conjunction with the EM algorithm from Eq. (16).

In the first subsection, we discuss the results of applying all the steps in our conceptual language model (GC; Section 4). Then, in the following sections, we will perform an ablation study and discuss the results of not applying the EM algorithm (MLGC; Section 6.2.2) and not translating the found concepts using generative concept models (EC; Section 6.2.3). Example query models for GC and EC can be found in Table 8 for the CLEF topic “Shrinking cities”.

### 6.2.1. Results

In this section we present the results of using every step of the conceptual language model (abbreviated GC) we detailed in Section 4. Table 10 lists the results of the concept language models. The results for the two CLEF collections show that the GC model can result in a significant improvement in recall over the query likelihood approach: 13% and 9% more relevant documents are returned for CLEF-DS-07 and CLEF-DS-08, respectively. Fig. 3 shows the precision–recall graphs for our conceptual language model, versus the query-likelihood baseline and relevance models. The precision–recall curve of the CLEF-DS-07 query set shows improved precision over almost the whole recall range. The CLEF-DS-08 runs shows improved precision between recall levels 0.7 and 0.8, making up for the loss of initial precision. Overall, both CLEF test collections show improvements in mean average precision (19% and 6%, respectively), but only the results on CLEF-DS-07 are significantly different. We note that the RM approach was unable to achieve a significant difference against the query-likelihood baseline on these test collections and measures.

The three TREC Genomics test collections show a less consistent behavior. In terms of mean average precision, the TREC-GEN-04 and TREC-GEN-06 collections show significant improvements in favor of the GC model (+6.6% and +15.4% respectively). The TREC-GEN-05 topics also show substantial improvements between the query likelihood and GC model, although these changes are not significant. Fig. 2 shows a per-topic analysis of the difference of the GC model with respect to the QL baseline; a positive value in these graphs indicates the GC model outperformed the QL baseline. For TREC-GEN-05, it shows that half of the topics benefit from applying the GC model and the other half is actually hurt. This is what causes the difference to be non-significant. The overall increase in average precision measured over al the topics, however, is larger than its loss.

From a further look on the per-topic plots, we can observe that, in terms of MAP, more topics are helped than hurt for all the other test collections. The early precision plots show a less clear picture. The ratio between the number of topics that improve precision@5 (P5) versus topics that worsen is about 1.5, averaged over all test collections. The average number of topics which precision@10 (P10) scores increase is about the same as the number of topics for which it decreases.

A more in-depth analysis of the terms that are introduced provides more insight into when and where the GC model improves or hurts retrieval. We observe that when the initial textual query is not specific, the resulting set of feedback documents is unfocused. Hence, fairly general and uninformative words are added to the query model and it fails to achieve higher retrieval performance. Another reason for poor performance is that particular aspects in the original query are overemphasized in the updated query model, resulting in query drift. For example, the CLEF-DS-08 topic 210 entitled "Establishment of new businesses after the reunification" results in expansion terms related to the aspect "Establishment of new businesses", such as "entrepreneur" and "entrepreneurship", but fails to include words related to the "reunification" aspect. When the updated query model is a balanced expansion of the original query, i.e., when it does include expansion terms for all aspects of the query, the GC model show improved results.

Overall, we see that our conceptual language model mainly has a recall enhancing effect, indicated by the significant increases in MAP for the CLEF-DS-07 and TREC-GEN-06 test collections and the significant increases in recall on both CLEF topic sets.

Table 11 shows a comparison between the GC and the RM model. When comparing these results, we find significant improvements in terms of recall on the CLEF test collections. On the TREC-GEN-04 and TREC-GEN-06 topic set we find a

**Table 10**
Results of the baseline (QL) and the conceptual language model (GC).

| | | QL | GC | |
|---|---|---|---|---|
| CLEF-DS-07 | RelRet/TotalRel | 2289/4530 | **2596**/4530 | +13.4%$^{*}$ |
| | P5 | 0.5120 | **0.5520** | +7.8% |
| | P10 | **0.5080** | 0.4920 | −3.1% |
| | MAP | 0.1952 | **0.2315** | +18.6%$^{*}$ |
| CLEF-DS-08 | RelRet/TotalRel | 1468/2133 | **1602**/2133 | +9.1%$^{*}$ |
| | P5 | **0.5280** | 0.4880 | −7.6% |
| | P10 | 0.4680 | **0.4840** | +3.4% |
| | MAP | 0.2819 | **0.2991** | +6.1% |
| TREC-GEN-04 | RelRet/TotalRel | 3847/8268 | **4022**/8268 | +4.5% |
| | P5 | 0.5160 | **0.5560** | +7.8% |
| | P10 | 0.4800 | **0.5000** | +4.2% |
| | MAP | 0.2856 | **0.3045** | +6.6%$^{*}$ |
| TREC-GEN-05 | RelRet/TotalRel | 2825/4584 | **3330**/4584 | +17.9% |
| | P5 | 0.4122 | **0.4245** | +3.0% |
| | P10 | **0.3776** | **0.3776** | 0.0% |
| | MAP | 0.2153 | **0.2338** | +8.6% |
| TREC-GEN-06 | RelRet/TotalRel | 1078/1449 | **1244**/1449 | +15.4% |
| | P5 | 0.4154 | **0.4538** | +9.2% |
| | P10 | **0.4154** | 0.4077 | −1.9% |
| | MAP | 0.2731 | **0.3182** | +16.5%$^{*}$ |

**Table 11**
Results of the relevance model (RM) versus conceptual language models (GC).

| | | RM | GC | |
|---|---|---|---|---|
| CLEF-DS-07 | RelRet/TotalRel | 2430/4530 | **2596**/4530 | +6.8%$^{*}$ |
| | P5 | 0.5440 | **0.5520** | +1.5% |
| | P10 | **0.5040** | 0.4920 | −2.4% |
| | MAP | 0.2061 | **0.2315** | +12.3% |
| CLEF-DS-08 | RelRet/TotalRel | 1473/2133 | **1602**/2133 | +8.8%$^{*}$ |
| | P5 | **0.5680** | 0.4880 | −14.1% |
| | P10 | 0.4800 | **0.4840** | +0.8% |
| | MAP | 0.2856 | **0.2991** | +4.7% |
| TREC-GEN-04 | RelRet/TotalRel | **4205**/8268 | 4022/8268 | −4.4% |
| | P5 | **0.5680** | 0.5560 | −2.1% |
| | P10 | **0.5340** | 0.5000 | −6.4%$^{*}$ |
| | MAP | **0.3306** | 0.3045 | −7.9%$^{*}$ |
| TREC-GEN-05 | RelRet/TotalRel | 3031/4584 | **3330**/4584 | +9.9% |
| | P5 | 0.4163 | **0.4245** | +2.0% |
| | P10 | **0.3857** | 0.3776 | −2.1% |
| | MAP | **0.2368** | 0.2338 | −1.3% |
| TREC-GEN-06 | RelRet/TotalRel | 1160/1449 | **1244**/1449 | +7.2% |
| | P5 | 0.4308 | **0.4538** | +5.3% |
| | P10 | **0.4346** | 0.4077 | −6.2% |
| | MAP | 0.2993 | **0.3182** | +6.3%$^{*}$ |

significant improvement in terms of MAP. The results on the TREC Genomics 2004 and 2005 topic sets indicate that the GC model performs comparably (TREC-GEN-05) or slightly worse (TREC-GEN-04). We believe the latter result is caused by the fixed setting of $\delta_t$ in Eq. (18) in conjunction with the rather small average document length and the large number of documents in this particular document collection.

Unlike the relevance model, the GC model provides a weighted set of concepts in the form of a conceptual query model. Besides the possibility of suggesting these to the user, we hypothesize that the results of applying the remaining steps in our conceptual language models after a user has selected the concepts most relevant to his query would improve retrieval effectiveness. Since we do not have relevant concepts for our current topics, we consider the verification of this hypothesis a topic for future work.

In the following sections, we look at the results of not using the EM algorithm in the generative concept models and directly using the conceptual query models for retrieval.

### *6.2.2. Maximum likelihood-based generative concept models*

In this section, we investigate the added value of using the EM algorithm described in 4.2, by comparing a maximum likelihood based GC model (named *MLGC*) to the GC model shown in the previous section. Table 12 shows the results of this

**Table 12**
Results of the conceptual language models in conjunction with the EM algorithm (GC) described in Section 4 versus without (MLGC).

| | | MLGC | GC | |
|---|---|---|---|---|
| CLEF-DS-07 | RelRet/TotalRel | **2596**/4530 | **2596**/4530 | 0.0% |
| | P5 | **0.5520** | **0.5520** | 0.0% |
| | P10 | 0.4760 | **0.4920** | +3.4% |
| | MAP | 0.2311 | **0.2315** | +0.2% |
| CLEF-DS-08 | RelRet/TotalRel | 1566/2133 | **1602**/2133 | +2.3%$^{*}$ |
| | P5 | **0.5120** | 0.4880 | −4.7% |
| | P10 | **0.4960** | 0.4840 | −2.4% |
| | MAP | 0.2853 | **0.2991** | +4.8% |
| TREC-GEN-04 | RelRet/TotalRel | 3973/8268 | **4022**/8268 | +1.2% |
| | P5 | 0.5360 | **0.5560** | +3.7% |
| | P10 | 0.4960 | **0.5000** | +0.8% |
| | MAP | 0.2989 | **0.3045** | +1.9% |
| TREC-GEN-05 | RelRet/TotalRel | 2887/4584 | **3330**/4584 | +15.3% |
| | P5 | 0.4163 | **0.4245** | +2.0% |
| | P10 | 0.3571 | **0.3776** | +5.7% |
| | MAP | 0.2174 | **0.2338** | +7.5% |
| TREC-GEN-06 | RelRet/TotalRel | 1118/1449 | **1244**/1449 | +11.3% |
| | P5 | 0.4231 | **0.4538** | +7.3% |
| | P10 | **0.4192** | 0.4077 | −2.7% |
| | MAP | 0.2863 | **0.3182** | +11.1% |

**Table 13**
Results of the conceptual language models (GC) versus using the found concepts directly (EC).

| | | EC | GC | |
|---|---|---|---|---|
| CLEF-DS-07 | RelRet/TotalRel | 2448/4530 | **2596**/4530 | +6.0% |
| | P5 | 0.5040 | **0.5520** | +9.5% |
| | P10 | **0.5080** | 0.4920 | −3.1% |
| | MAP | 0.2104 | **0.2315** | +10.0% |
| CLEF-DS-08 | RelRet/TotalRel | 1485/2133 | **1602**/2133 | +7.9%$^{*}$ |
| | P5 | **0.5120** | 0.4880 | −4.7% |
| | P10 | **0.4880** | 0.4840 | −0.8% |
| | MAP | 0.2894 | **0.2991** | +3.4% |
| TREC-GEN-04 | RelRet/TotalRel | **4221**/8268 | 4022/8268 | −4.7% |
| | P5 | 0.5480 | **0.5560** | +1.5% |
| | P10 | **0.5240** | 0.5000 | −4.6% |
| | MAP | **0.3146** | 0.3045 | −3.2% |
| TREC-GEN-05 | RelRet/TotalRel | 2916/4584 | **3330**/4584 | +14.2% |
| | P5 | 0.4082 | **0.4245** | +4.0% |
| | P10 | **0.3776** | **0.3776** | 0.0% |
| | MAP | 0.2295 | **0.2338** | +1.9% |
| TREC-GEN-06 | RelRet/TotalRel | 1171/1449 | **1244**/1449 | +6.2% |
| | P5 | 0.4231 | **0.4538** | +7.3% |
| | P10 | 0.4000 | **0.4077** | +1.9% |
| | MAP | 0.2927 | **0.3182** | +8.7% |

method. We observe that applying the EM algorithm improves overall retrieval effectiveness compared to the MLGC model, although not significantly, and only in terms of recall and MAP. Only the number of relevant retrieved documents for the CLEF-DS-08 significantly improves when using the EM algorithm.

The topics that are helped most by the application of the EM algorithm—in terms of an absolute gain in MAP—include TREC-GEN-05 topic 146: “Provide information about Mutations of presenilin-1 gene and its/their biological impact in Alzheimer’s disease” (increased MAP by 0.51) and TREC-GEN-06 topic 160 “What is the role of PrnP in mad cow disease?” (increased MAP by 0.52). A closer look at the intermediate results for these topics reveals two things. In the first topic, the GC model introduces the term “PRP”, which is a synonym for “PrnP”. The second topic shows that the GC model introduces three new terms which do not seem directly relevant to the query, but are able to boost MAP substantially.

Besides having the potential of improving certain topics automatically we believe that, similar to our observation with regard to the GC model, the biggest improvements may be realized when a user selects the most relevant concepts. Future work should indicate if this is a valid assumption. Moreover, when one considers presenting the found concepts and/or terms to the user, the EM algorithm does provide a transparent function that helps filtering non-content-bearing terms and concepts.

### *6.2.3. Explicit conceptual query models*

In Section 4.1 we introduced a method for acquiring a weighted set of concepts for a query, by translating a textual query to a conceptual representation. In this section, we evaluate the results of using the conceptual query model (abbreviated *EC*) directly, i.e., using it in combination with the original textual representation to estimate the relevance of a document. Since all the documents in our current test collections have two representations (terms and concepts), we can use both disjunctively for retrieval. So, instead of interpolating the query model and using the result for retrieval, we interpolate the scores of each individual component as follows[5]:

$$\text{Score}\ (Q,\ D) = (1-\lambda_Q)\cdot -\mathrm{KL}(\tilde{\theta}_Q\|\theta_D) + \lambda_Q \cdot -\mathrm{KL}(\theta_C\|\theta_D). \tag{20}$$

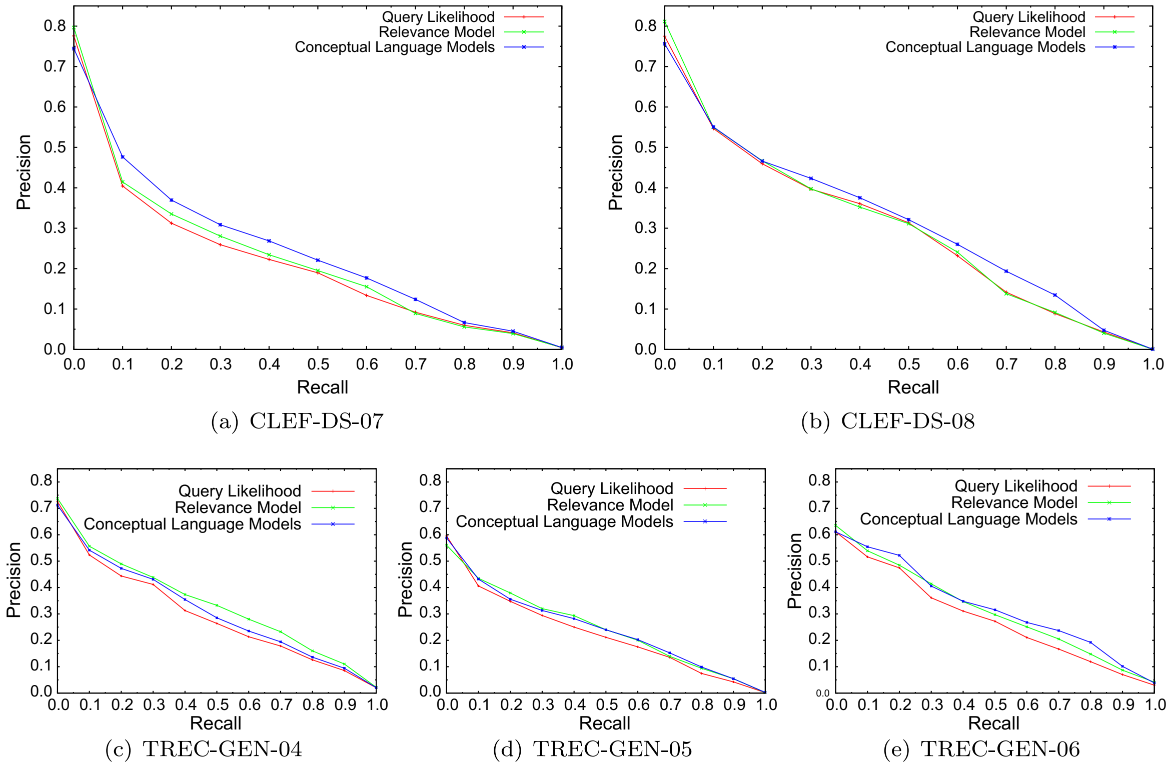
Here, the first term is the regular query-likelihood score. The second term is the score obtained from matching the conceptual query model with the conceptual representation of each document:

$$-\mathrm{KL}(\theta_C\|\theta_D) = -\sum_c P(c|\theta_C)\log\frac{P(c|\theta_C)}{P(c|\theta_D)} \propto -\sum_c P(c|\theta_C)\log P(c|\theta_D), \tag{21}$$

where $P(c|\theta_C) = P(c|Q)$ (Eq. (10) q.v.). In effect, this drops the dependence between $t$ and $c$ (see Fig. 1) and considers the concepts as regular indexing terms.

Thus, the EC model uses an explicit conceptual representation in combination with the textual representation for searching documents and, similar to the approaches described in the previous sections, the EC approach uses the same feedback documents for improving the query. However, instead of sampling terms from these documents, we now use their associated concepts.

[5] KL-divergence is a linear model and as such invariant under scaling.



(a) CLEF-DS-07

(b) CLEF-DS-08

(c) TREC-GEN-04

(d) TREC-GEN-05

(e) TREC-GEN-06

**Fig. 3.** Precision–recall plots for all evaluated test collections.

When we look at the results as compared to the GC model as depicted in Table 13, we find marginal differences. Only recall on the CLEF-DS-08 topic set is significantly different from the run based on conceptual language models. In comparison to the query-likelihood baseline (cf. Tables 9 and 13), the EC model shows similar improvements as the relevance models. The runs on the CLEF collections show small improvements in mean average precision, recall and initial precision. When tested, these differences are not statistically significant The EC model, when applied to the TREC Genomics collections, shows significant improvements for the 2004 and 2006 collection with respect to the QL baseline.

Before turning to the answers to our research questions based on the results in this section, we present a brief analysis of the parameter sensitivity of our conceptual language model.

### 6.3. Parameter sensitivity analysis

Both our conceptual language model and the relevance model have a number of parameters that need to be set, as introduced in Section 5.3. In this section we describe the optimal settings for each model and explore the sensitivity of the results. Similar to related work (Eguchi & Croft, 2006; Zhai & Lafferty, 2001), we did not evaluate $|\mathscr{D}_Q|, |\mathscr{V}_Q| > 10$. Even given this restriction, the obtained results are clear improvements and further improvements may be obtained with a larger set of terms or documents.

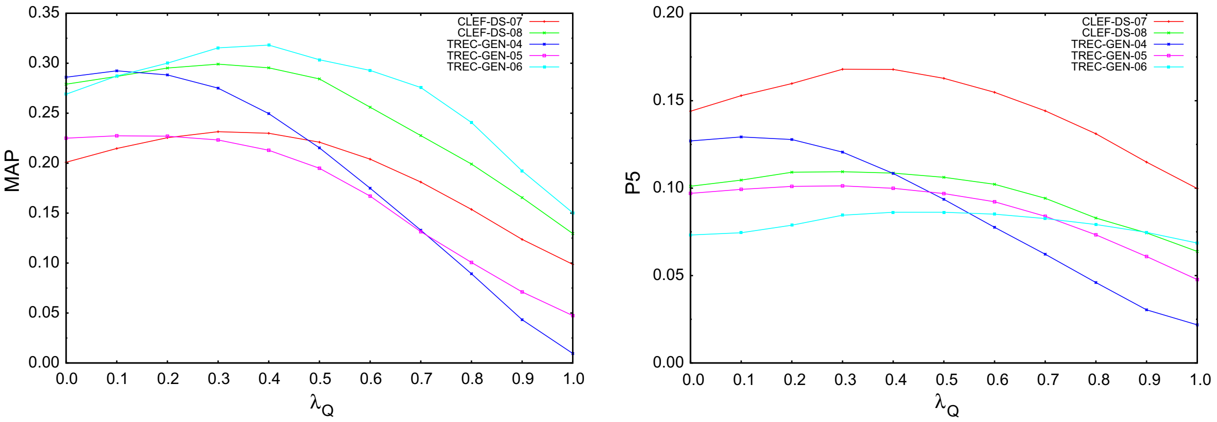
Table 14 lists the optimal parameter settings for the relevance model per test collection. We observe that the setting of $\lambda_Q$ for this model is roughly dependent on the document collection. Table 15 lists the optimal parameter values for the conceptual language model. Again we observe that the optimal value for $\lambda_Q$ is dependent on the document collection. We zoom in

**Table 14**
Free parameters in the relevance model described in Section 3.1. See Table 7 for a description of each parameter.

| | $\lambda_Q$ | $\|\mathscr{D}_Q\|$ | $\|\mathscr{V}_Q\|$ |
|---|---|---|---|
| CLEF-DS-07 | 0.5 | 7 | 8 |
| CLEF-DS-08 | 0.7 | 10 | 7 |
| TREC-GEN-04 | 0.5 | 7 | 10 |
| TREC-GEN-05 | 0.5 | 3 | 6 |
| TREC-GEN-06 | 0.4 | 4 | 10 |

**Table 15**
Free parameters for the conceptual language models. See Table 7 for a description of each parameter.

| | $\|\mathscr{C}\|$ | $\lambda_Q$ | $\|\mathscr{D}_Q\|$ | $\|\mathscr{V}_Q\|$ |
|---|---|---|---|---|
| CLEF-DS-07 | 8 | 0.3 | 7 | 4 |
| CLEF-DS-08 | 4 | 0.3 | 3 | 5 |
| TREC-GEN-04 | 9 | 0.1 | 10 | 10 |
| TREC-GEN-05 | 10 | 0.1 | 9 | 5 |
| TREC-GEN-06 | 3 | 0.4 | 6 | 2 |



(a) MAP (b) P5

**Fig. 4.** Results of varying $\lambda_Q$ on retrieval effectiveness on all test collections.

on the sensitivity of the results of the conceptual language model towards the setting of $\lambda_Q$, by displaying the effect of varying $\lambda_Q$ on MAP (Fig. 4a) and precision@5 (Fig. 4b). We observe that the curves follow a similar pattern for the CLEF document collection and for both measures, with both maxima lying around $\lambda_Q = 0.3$. The TREC-GEN-04 and TREC-GEN-05 topics—which both use the TREC 2004 document collection—follow a less similar pattern, although their maximum MAP scores have a similar corresponding $\lambda_Q$ value. The TREC-GEN-06 and the CLEF-DS-2007 topics show the largest relative improvement (both nearly 20% improvement over the query likelihood in terms of MAP, i.e., when $\lambda_Q = 0$). We also observe that selecting the best value for $\lambda_Q$ based on the highest MAP scores does not necessarily lead to the highest score in terms of early precision. Interestingly, the TREC-GEN-06 topics reach roughly the same precision@5 scores for the query likelihood model as when we would only use the terms suggested by the conceptual language model.

## 7. Conclusion

We have proposed and investigated conceptual language models for domain-specific document retrieval. The goal of conceptual language models is to leverage document-level concept annotations for improving full-text retrieval. In our method, the original textual query is translated to a *conceptual query model* and, by means of *generative concept models* this conceptual query model is used to update the original, textual query model. The motivation behind this dual translation is that an explicit conceptual representation of the information need can be used to derive related terms which are less dependent on the original query text. In both translation steps we have applied an EM algorithm to improve model estimation. Using an extensive set of experiments on five test collections from two domains, we have shown that conceptual language models can improve text-based retrieval, both with and without conventional pseudo-relevance feedback.

We now turn to answering the research questions posed in the introduction. First, we compared conceptual language models to a query-likelihood baseline and a model incorporating pseudo-relevance feedback. When evaluated on five test collections from two domains, we find that the conceptual language models yield significant improvements over a query-likelihood baseline on all the evaluated measures. In particular we have observed a significant improvement in terms of recall on all collections, which is in line with results obtained from relevance feedback methods in general. On the TREC collections, however, we have also observed a significant increase in early precision. As such, our method is both a recall and a precision-enhancing device.

When compared to relevance models and using the same pseudo-relevant documents, conceptual language models show a significant improvement in terms of MAP on two test collections, as well as a significant increase in recall on two other test collections. On the remaining measures, it gives similar improvements as relevance models. However, conceptual language models have the added advantage of offering query and browsing suggestions in the form of clearly understandable concepts. It should be noted that while each step in applying conceptual language models is not significantly different from each other or the steps combined, the full model is able to significantly outperform both a standard language modeling and a relevance modeling approach.

Our second research question concerns the use of an iterative EM algorithm to re-estimate textual and conceptual document models. These models are used in the process of determining a conceptual query model based on pseudo-relevant documents and for determining the translation probabilities from concepts to text. We have shown that this "parsimonisation" step is an essential component in order to achieve good performance, since it makes sure that the language models only generate content-bearing terms. Moreover, since the resulting terms and concepts are more specific (than without EM-based re-estimation), we believe they are more useful in case these were to be presented to a user. Third, we looked into the parameter sensitivity of the proposed approach. Similar to conventional pseudo-relevance feedback, the optimal parameter settings have to be determined on a per collection basis.

Our fourth and final research question concerned the portability of our models. The usefulness of the proposed approach has been evaluated in two domains, the social science and genomics domain, each with different types of documents and their own concept vocabularies. Despite these large differences, the concept-based feedback shows consistent improvements. It is interesting to note that while a thesaurus might be limited in representing specific information needs, it can still be used to improve retrieval effectiveness. The MeSH thesaurus can, be used to improve genomics information retrieval despite its general biomedical coverage. The annotations of the CLEF collections seems to fit the information needs better, resulting in even better retrieval performance in the social science domain.

As to future work, we envisage several directions. First, in this paper we have relied on manually curated concept annotations of documents. Future work should look into the robustness of the approach when working with automatic conceptual representations of documents, such as obtained through document classification.

Second, we want to look into the relationship between conceptual and (traditional) term-based relevance feedback. In our current work we have used relatively simple baseline results for the estimation of our models. We hypothesize that combining our generative concept models with well-performing methods such as relevance models may improve results even further and we will investigate this in future research. Further, for the results that we have presented we have utilized blind jrelevance feedback, i.e., we have assumed that the top-ranked documents and concepts were relevant. With the test collections currently available we are unable to confirm or refute whether and how explicit relevance assessments would influence the results. The same could be posited not only for documents, but also for the associated concepts. As such, we leave verifying whether user interaction influences the end results for future work. As an added value of this approach we noted that we obtain an explicit conceptual representation of the query. In future work we will look whether this conceptual representation is appreciated by and useful to an end user. Finally, although we have obtained significant improvements, we concede that the number of terms and documents employed in the estimations is of distinct influence on the end results. Whether increasing these numbers positively affect retrieval effectiveness remains a topic for future work.

## Acknowledgments

We thank the anonymous reviewers for their helpful comments and remarks. We also thank Wouter Weerkamp for his insightful suggestions. This research was supported by the BioRange programme of the Netherlands Bioinformatics Centre (supported by a BSIK grant through the Netherlands Genomics Initiative), by the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments (http://www.stevin-tst.org) under project number STE-09-12, and by 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, and by 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 and is part of the ICT innovation program of the Ministry of Economic Affairs.

## References

Anick, P. (2003). Using terminological feedback for web search refinement: A log-based study. In *SIGIR '03*.

Bai, J., Song, D., Bruza, P., Nie, J.-Y., & Cao, G. (2005). Query expansion using term relationships in language models for information retrieval. In *CIKM '05*.

Bai, J., & Nie, J.-Y. (2008). Adapting information retrieval to query contexts. *IPM, 44*(6), 1901–1922.

Balog, K. (2008). *People search in the enterprise*. PhD thesis, University of Amsterdam.

Balog, K., Weerkamp, W., & de Rijke, M. (2008). A few examples go a long way: Constructing query models from elaborate query formulations. In *SIGIR '08*.

Berger, A., & Lafferty, J. (1999). Information retrieval as statistical translation. In *SIGIR '99*.

Bhogal, J., Macfarlane, A., & Smith, P. (2007). A review of ontology based query expansion. *Information Processing & Management, 43*(4), 866–886.

Broder, A. Z., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., & Zhang, T. (2007). Robust classification of rare queries using web knowledge. In *SIGIR '07*.

Camous, F., Blott, S., & Smeaton, A. F. (2006). On combining MeSH and text searches to improve the retrieval of Medline documents. In *Proceedings of the third conference en recherche d'informations et applications (CORIA)*.

Chen, S. F., & Goodman, J. (1996). An empirical study of smoothing techniques for language modeling. In *ACL '96*.

Chen, Y., Xue, G.-R., & Yu, Y. (2008). Advertising keyword suggestion based on concept hierarchy. In *WSDM '08*.

Chung, Y. (2004). Optimization of some factors affecting the performance of query expansion. *Information Processing & Management, 40*(6), 891–917.

Cleverdon, C. W., Mills, J., & Keen, M. (1966). Aslib Cranfield research project – Factors determining the performance of indexing systems. *Test Results* (Vol. 2). Wharley End, Bedford, USA: Cranfield University.

Eguchi, K., & Croft, W. B. (2006). Boosting relevance model performance with query term dependence. In *CIKM '06*.

Finkelstein, L. E. V., Gabrilovich, E., Matias, Y., Rivlin, E. H. U. D., Solan, Z. A. C. H., Wolfman, G. A. D. I., et al (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems, 20*(1), 116–131.

Gabrilovich, E., & Markovitch, S. (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *IJCAI'07*.

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

Giger, H. P. (1988). Concept based retrieval in classical IR systems. In *SIGIR '88*.

Hersh, W., Cohen, A. M., Roberts, P., & Rekapalli, H. K. (2007). TREC 2006 genomics track overview. In *Proceedings of the 15th text retrieval conference (TREC 2006)*.

Hersh, W., Bhuptiraju, R., Ross, L., Johnson, P., Cohen, A., & Kraemer, D. (2005). TREC 2004 Genomics track overview. In *Proceedings of the 13th text retrieval conference (TREC 2004)*.

Hersh, W., Cohen, A., Yang, J., Bhupatiraju, R. T., Roberts, P., & Hearst, M. (2006). TREC 2005 genomics track overview. In *Proceedings of the 14th text retrieval conference (TREC 2005)*.

Hersh, W. R., Hickam, D. H., Haynes, R. B., & McKibbon, K. A. (1994). A performance and failure analysis of SAPHIRE with a MEDLINE test collection. *Journal of the American Medical Informatics Association: JAMIA, 1*(1), 51–60.

Herskovic, J. R., Tanaka, L. Y., Hersh, W., & Bernstam, E. V. (2007). A day in the life of PubMed: Analysis of a typical day's query log. *Journal of the American Medical Informatics Association: JAMIA, 14*(2), 212–220.

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

Hiemstra, D., Robertson, S., & Zaragoza, H. (2004). Parsimonious language models for information retrieval. In *SIGIR '04*.

Jardine, N., & van Rijsbergen, C. J. (1971). The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval, 7*(5), 217–240.

Jelinek, F., & Mercer, R. L. (1980). Interpolated estimation of markov source parameters from sparse data. In *Workshop pattern recognition in practice*.

Jing, Y., & Croft, W. B. (1994). An association thesaurus for information retrieval. In *Proceedings of RIAO '94*.

Joyce, T., & Needham, R. M. (1958). The thesaurus approach to information retrieval. *American Documentation, 9*(3), 192–197.

Keskustalo, H., Järvelin, K., & Pirkola, A. (2008). Evaluating the effectiveness of relevance feedback based on a user simulation model: Effects of a user scenario on cumulated gain value. *Information Retrieval, 11*(3), 209–228.

Korfhage, R. R. (1984). Query enhancement by user profiles. In *SIGIR '84*.

Kraaij, W. (2004). *Variations on language modeling for information retrieval*. PhD thesis, University of Twente.

Kraaij, W., & de Jong, F. (2004). Transitive probabilistic CLIR models. In *RIAO '04*.

Kurland, O. (2008). The opposite of smoothing: A language model approach to ranking query-specific document clusters. In *SIGIR '08*.

Kurland, O., & Lee, L. (2004). Corpus structure, language models, and ad hoc information retrieval. In *SIGIR '04*.

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

Lafferty, J., & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *SIGIR '01*.

Lafferty, J., & Zhai, C. (2003). Probabilistic relevance models based on document and query generation. In *Language modeling for information retrieval*. Springer.

Lancaster, W. F. (1982). *Information retrieval systems: Characteristics, testing and evaluation*. Wiley Interscience.

Lavrenko, V. (2004). *A generative theory of relevance*. PhD thesis, University of Massachusettes.

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

Lee, K. S., Croft, W. B., & Allan, J. (2008). A cluster-based resampling method for pseudo-relevance feedback. In *SIGIR '08*.

Liu, X., & Croft, B. W. (2004). Cluster-based retrieval using language models. In *SIGIR '04*.

Meij, E., & de Rijke, M. (2007). Thesaurus-based feedback to support mixed search and browsing environments. In *ECDL '07*.

Meij, E., & de Rijke, M. (2008). The University of Amsterdam at the CLEF 2008 domain specific track – Parsimonious relevance and concept models. In *Evaluating systems for multilingual and multimodal information access – 9th Workshop of the cross-language evaluation forum*. Revised selected papers.

Meij, E., Trieschnigg, D., de Rijke, M., & Kraaij, W. (2008). Parsimonious concept modeling. In *SIGIR '08*.

Metzler, D., & Croft, B. W. (2005). A markov random field model for term dependencies. In *SIGIR '05*.

Miller, D. R. H., Leek, T., & Schwartz, R. M. (2000). BBN at TREC-7: Using hidden markov models for information retrieval. In *Proceedings of the 7th text retrieval conference (TREC 1999)*.

Minker, J., Wilson, G. A., & Zimmerman, B. H. (1972). An evaluation of query expansion by the addition of clustered terms for a document retrieval system. *Information Storage and Retrieval, 8*(6), 329–348.

Mishne, G., & de Rijke, M. (2006). A study of blog search. In M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, & A. Yavlinsky (Eds.), *Advances in information retrieval: Proceedings 28th European conference on IR research (ECIR 2006). LNCS* (Vol. 3936, pp. 289–301). Springer.

Mitra, M., Singhal, A., & Buckley, C. (1998). Improving automatic query expansion. In *SIGIR '98*.

Ng, K. (2001). A maximum likelihood ratio information retrieval model. In *Proceedings of the 9th text retrieval conference (TREC 2000)*.

Petras, V., & Baerisch, S. (2008). The domain-specific track at CLEF 2008. In *Evaluating systems for multilingual and multimodal information access – 9th Workshop of the cross-language evaluation forum*.

Petras, V., Baerisch, S., & Stempfhuber, M. (2007). The domain-specific track at CLEF 2007. In *Evaluating systems for multilingual and multimodal information access – 8th Workshop of the cross-language evaluation forum*.

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

Roberts, N. (1984). The pre-history of the information retrieval thesaurus. *Journal of Documentation, 271–285*(15).

Qiu, Y., & Frei, H.-P. (1993). Concept based query expansion. In *SIGIR '93*.

Rocchio, J. (1971). Relevance feedback in information retrieval. In *The SMART retrieval system: Experiments in automatic document processing*. Prentice Hall.

Rocha, C., Schwabe, D., & Aragao, M. P. (2004). A hybrid approach for searching in the semantic web. In *WWW '04*.

Salton, G. (1971). Information analysis and dictionary construction. In *The SMART retrieval system: Experiments in automatic document processing*. Prentice Hall.

Shen, D., Sun, J.-T., Yang, Q., & Chen, Z. (2006). Building bridges for web query classification. In *SIGIR '06*.

Silveira, M. L., & Ribeiro-Neto, B. (2004). Concept-based ranking: A case study in the juridical domain. *Information Processing & Management, 40*(5), 791–805.

Sparck-Jones, K., & Jackson, D. M. (1970). The use of automatically-obtained keyword classifications for information retrieval. *Information Processing & Management, 5*(1), 175–201.

Sparck-Jones, K., & Needham, R. M. (1968). Automatic term classification and retrieval. *Information Processing & Management, 4*(1), 91–100.

Srinivasan, P. (1996). Query expansion and medline. *Information Processing & Management, 32*(4), 431–443.

Stokes, N., Li, Y., Cavedon, L., & Zobel, J. (2009). Exploring criteria for successful query expansion in the genomic domain. *Information Retrieval, 12*(1), 17–50.

Tao, T., & Zhai, C. (2006). Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR '06*.

Trajkova, J., & Gauch, S. (2004). Improving ontology-based user profiles. In *Proceedings of RIAO '04*.

Trieschnigg, D., Kraaij, W., & de Jong, F. (2007). The influence of basic tokenization on biomedical document retrieval. In *SIGIR '07*.

Trieschnigg, D., Kraaij, W., & Schuemie, M. (2007). Concept based passage retrieval for genomics literature. In *Proceedings of the 15th text retrieval conference (TREC 2006)*.

Trieschnigg, D., Pezik, P., Lee, V., Kraaij, W., de Jong, F., & Rebholz-Schuhmann, D. (2009). MeSH Up: Effective MeSH text classification and improved document retrieval. *Bioinformatics, 25*(11), 1412–1418.

Vakkari, P., Jones, S., Macfarlane, A., & Sormunen, E. (2004). Query exhaustivity, relevance feedback and search success in automatic and interactive query expansion. *Journal of Documentation, 60*(2), 109–127.

Voorhees, E. M. (1994). Query expansion using lexical-semantic relations. In *SIGIR '94*.

de Vries, A. P., Vercoustre, A.-M., Thom, J. A., Craswell, N., Lalmas, M. (2007). Overview of the INEX 2007 entity ranking track. In *INEX-6*.

Wei, X. (2007). *Topic models in information retrieval*. PhD thesis, University of Massachusetts.

Xu, Z., & Akella, R. (2008). A new probabilistic retrieval model based on the dirichlet compound multinomial distribution. In *SIGIR '08*.

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

Xu, J., & Croft, W. B. (1999). Cluster-based language models for distributed retrieval. In *SIGIR '99*.

Zhai, C. (2002). *Risk minimization and language modeling in text retrieval*. PhD thesis, Carnegie Mellon University.

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

Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems, 22*(2), 179–214.

Zhou, X., Hu, X., Zhang, X., Lin, X., & Song, I.-Y. (2006). Context-sensitive semantic smoothing for the language modeling approach to genomic IR. In *SIGIR '06*.

Zhou, X., Hu, X., & Zhang, X. (2007). Topic signature language models for ad hoc retrieval. *IEEE Transactions on Knowledge and Data Engineering, 19*(9), 1276–1287.