Concept Models for Domain-Specific Search

Edgar Meij and Maarten de Rijke

ISLA, University of Amsterdam
Science Park 107, 1098 XG Amsterdam, The Netherlands
{emeij,mdr}@science.uva.nl

Abstract. We describe our participation in the 2008 CLEF Domain-specific track. We evaluate blind relevance feedback models and concept models on the CLEF domain-specific test collection. Applying relevance modeling techniques is found to have a positive effect on the 2008 topic set, in terms of mean average precision and precision@10. Applying concept models for blind relevance feedback, results in even bigger improvements over a query-likelihood baseline, in terms of mean average precision and early precision.

Keywords: Language modeling, Blind relevance feedback, Concept models.

1 Introduction

Our approach to retrieving documents that are annotated with thesaurus terms is to model the language use associated with concepts from a thesaurus or ontology. To this end we use the document annotations as a "bridge" between vocabulary terms and the concepts in the knowledge source at hand. We model the language use associated with concepts using a generative language modeling framework, which provides theoretically sound estimation methods and builds upon a solid statistical background.

Our concept models may be used to determine semantic relatedness or to generate navigational suggestions, either in the form of concepts or vocabulary terms. These can then be used as suggestions for the user or for blind relevance feedback [8,9,14]. In order to apply blind relevance feedback using our models, we perform a double translation. First, we estimate the most likely concepts given a query and then we use the most distinguishing terms from these concepts to formulate a new query. To find the most distinguishing terms given a concept, we apply a technique based on expectation-maximization (EM) [4] to re-estimate probabilities of one model with respect to another. Events that are well-predicted by the latter model will lose probability mass, which in turn will be given to the remaining events. Recently, we have successfully applied this technique to the estimation of relevance models on a variety of tasks and collections [9,10].

We address two research questions: (i) What are the effects of estimating and applying relevance models to the collection used at the CLEF 2008 Domain-specific track [7]? And (ii) what are the results of applying our concept models for blind relevance feedback? We find that applying relevance models helps for the CLEF 2008 Domain-specific test collection in terms of both mean average precision and early precision, although not

C. Peters et al. (Eds.): CLEF 2008, LNCS 5706, pp. 207–214, 2009.

significantly. Our concept models are able to significantly outperform a baseline query-likelihood run, both in terms of mean average precision and early precision. Moreover, we even improve over relevance models in terms of MAP.

The remainder of this paper is organized as follows. In Section 2 we introduce our retrieval framework. In Section 3 we introduce the details of our models. In Section 4 we describe our experimental setup, parameter settings, and document preprocessing steps. In Section 5 we discuss our results and we end with a concluding section.

2 Language Modeling

In the area of information retrieval, language modeling-based methods have been around for about a decade now [5,12,16]. Such methods are centered around the assumption that a query as issued by a user is a sample generated from an underlying term distribution—the information need. The documents in the collection are modeled in a similar fashion and are usually considered to be a mixture of a document-specific model and a more general background model. At retrieval time, each document is ranked according to the likelihood of having generated the query (query-likelihood).

Lafferty and Zhai [6] propose to generalize the query likelihood model to the KL-divergence scoring method, in which the query is modeled separately. Scoring documents then comes down to measuring the divergence between a query model $P(t|\theta_Q)$ and each document model $P(t|\theta_D)$, in which the divergence is negated for ranking purposes. The query model can be defined using the empirical maximum-likelihood estimate (MLE) on the original query as follows:

$$P(t|\tilde{\theta}_Q) = P(t|Q) = n(t;Q) \cdot |Q|^{-1}, \tag{1}$$

where $n(t;Q)$ is the number of occurrences of term t in query Q and $|Q|$ the length of the query. Under this definition, KL-divergence produces the same document ranking as the query likelihood model [16]. More formally, the score for each document given a query using the KL-divergence retrieval model is:

$$\begin{aligned} \mathrm{Score}(Q,D) &= -\mathrm{KL}(\theta_Q||\theta_D) \\ &= -\sum_{t\in\mathcal{V}} P(t|\theta_Q)\log P(t|\theta_D) + \sum_{t\in\mathcal{V}} P(t|\theta_Q)\log P(t|\theta_Q), \end{aligned} \tag{2}$$

where $\mathcal{V}$ denotes the vocabulary. The expression $\sum_{t\in\mathcal{V}} P(t|\theta_Q)\log P(t|\theta_Q)$—i.e., the entropy of the query—is constant per query and can be ignored for ranking purposes.

2.1 Document Modeling

Each document model $P(t|\theta_D)$ 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 entire document collection:

$$P(t|\theta_D) = \lambda_D P(t|D) + (1-\lambda_D)P(t). \tag{3}$$

This may be interpreted as a way of accounting for the fact that the (pseudo-)relevant documents contain terms related to the information need as well as terms from a more general model. We smooth using Bayesian smoothing with a Dirichlet prior and set $\lambda_D = \frac{\mu}{|D|+\mu}$ and $(1-\lambda_D) = \frac{|D|}{|D|+\mu}$, where μ is the Dirichlet prior that controls the influence of smoothing [3,18].

2.2 Query Modeling

Relevance feedback can be applied to better capture a user's information need [1,7,15]. In a language modeling context, this can be performed by re-estimating the query model, i.e., $P(t|\theta_Q)$ in Eq. 2 [12,17]. For blind relevance feedback one considers terms in a set of (pseudo-)relevant documents and selects the most informative ones. These terms may then be reweighed and used to estimate a query model.

Relevance modeling is one specific technique for estimating a query model given a set of (pseudo-)relevant documents $\mathcal{D}_Q$. The query and documents are both taken to be samples of an underlying generative model—the relevance model. There are several ways to estimate the parameters of this model given the observed data, each following a different independence assumption [7]. We use method 2, which is formulated as:

$$P(t|\hat{\theta}_Q) \propto P(t) \prod_{q_i \in Q} \sum_{D_i \in \mathcal{D}_Q} P(q_i|\theta_{D_i}) P(\theta_{D_i}|t), \tag{4}$$

where $q_1, \ldots, q_k$ are the query terms, D a document, and t a term. Bayes' rule is used to estimate the term $P(\theta_D|t)$:

$$P(\theta_D|t) = P(t|\theta_D)P(\theta_D) \cdot P(t)^{-1}, \tag{5}$$

where we assume the document prior $P(\theta_D)$ to be uniform. The initial query is interpolated with the expanded part [2,13,17], thus reweighing the initial query terms and providing smoothing for the relatively sparse initial sample $P(t|\tilde{\theta}_Q)$:

$$P(t|\theta_Q) = \lambda_Q P(t|\tilde{\theta}_Q) + (1-\lambda_Q) P(t|\hat{\theta}_Q) \tag{6}$$

3 Concept Models

In order to leverage the explicit knowledge encapsulated in the GIRT/CSASA thesauri used in the CLEF Domain-specific track, we perform blind relevance feedback using the concepts defined therein. To incorporate concepts in the retrieval process, we propose to leverage the conceptual knowledge in the estimation of a query model, which is obtained from a double translation. In this translation, concepts are used as a pivot language; the initial query is translated to concepts and back to expanded query terms:

$$P(t|\hat{\theta}_Q) = \sum_{c \in C} P(t|c) P(c|Q). \tag{7}$$

We assume that the probability of selecting a term is no longer dependent on the query once we have selected a concept given that query. Two components need to be estimated here: $P(t|c)$, to which we refer as a *generative concept model*, and $P(c|Q)$, to which we will refer as *conceptual query model*. These will be detailed in the following sections.

Table 1. Top 6 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 Domain Specific collection.

$P(t\|D)$ estimated using MLE	$P(t\|D)$ estimated using Eq. 13
0.061 the	0.54 indian
0.054 of	0.46 ethnic
0.045 indian	
0.038 ethnic	
0.028 in	
0.028 american	

3.1 Conceptual Query Modeling

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. If this is not the case, however, we may leverage the document annotations to approximate this step. We formulate the estimation of concepts relevant to a query by determining which concepts are most likely given the query. To estimate this probability, we consider the top-ranked documents returned by an initial retrieval run, denoted $\mathcal{D}_Q$, and look at the annotations associated with these documents. So, in order to determine the probability of a concept given a query, we look for concepts with the highest posterior probability:

$$P(c|Q) = \sum_{D \in \mathcal{D}_Q} P(c|D) P(D|Q). \tag{8}$$

Here, $P(D|Q)$ is determined by applying Bayes' rule on the initial retrieval scores, similar to Eq. 5. We assume that the probability of observing a concept is independent of the query, once we have selected a document given the query; the estimation of this term is addressed below (viz. Eq. 15). As an example, Table 1 shows the top six terms from a (term) document model, before and after parsimonization; clearly, the parsimonious document model is much more specific.

3.2 Generative Concept Models

As to the first component in Eq. 7—the concept model $P(t|c)$—we associate each GIRT/CSASA thesaurus concept with a language model. We determine the level of association between a term t and a concept c by looking at the way annotators have labeled the documents and determine the probability of observing t given c: $P(t|c) = P(t,c) \cdot P(c)^{-1}$. The concepts used to annotate documents may have different characteristics from other parts of a document, such as title and content. The annotations are selected by trained indexers from a concept language while the actual content consists of free text. Since the terms that make up the document are "generated" using a different process than the concepts, we assume that t and c are independent and identical samples given a document D in which they occur. So, the probability of observing both t and c is

$$P(t,c) = \sum_D P(D) P(c,t|D) = \sum_{D \in \mathcal{D}_C} P(D) P(t|D) P(c|D), \tag{9}$$

where $\mathcal{D}_C$ 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

$$P(t|c) = \frac{P(t,c)}{P(c)} \propto \frac{1}{P(c)} \sum_{D \in \mathcal{D}_C} P(t|D)P(c|D). \tag{10}$$

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

$$P(c) = \frac{\sum_D n(c;D)}{\sum_{c'} \sum_{D'} n(c';D')}. \tag{11}$$

Next we turn to $P(x|D)$, where $x \in \{t,c\}$. The size of these models (in terms of the number of words or concepts that receive a non-zero probability) may be large, e.g., in the case of a large document collection or of frequently occurring concepts. Not all observed *events* (i.e., terms or concepts) are equally informative. We have assumed that each document is a mixture of document-specific and more general terms (Eq. 3); we generalize this to also include concepts. We update each document model by reducing the probability mass of non-specific events by iteratively adjusting the individual probabilities in each document, based on a comparison with a large reference corpus (the collection). Formally, we maximize the posterior probability of D after observing x:

$$P(D|x) = \frac{\lambda_C P(x|D)}{(1-\lambda_C)P(x) + \lambda_C P(x|D)}. \tag{12}$$

Note that λ_C may be set differently from λ_D (Eq. 3) and differently for either terms or concepts. In this paper, we fix $\lambda_C = 0.15$ [9]. We then apply the following EM algorithm until the estimates no longer change significantly:

$$\begin{aligned} &\text{E-step:} \qquad && e_x = P(D|x) \qquad\qquad (13)\\ &\text{M-step:} && P_C(x|D) = \frac{n(x;D)e_x}{\sum_{x'} n(x';D)e_{x'}}. \end{aligned}$$

After the EM algorithm converges, we remove those events with a probability lower than a threshold δ. Thus, the resulting document model for terms, $P(t|\hat{\theta}_D)$, to be used in Eq. 10 is given by:

$$P(t|\hat{\theta}_D) = \begin{cases} Z_{D_t} \cdot P_C(t|D) & \text{if } t \in D \text{ and } P_C(t|D) > \delta_t \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

where Z_{D_t} is a document-specific normalization factor: $Z_{D_t} = 1/\sum_t P_C(t|D)$. Table 1 gives an example of the effects of applying this algorithm to a document from the current document collection. Similarly, the resulting document model for concepts, $P(c|\hat{\theta}_D)$, to be used for $P(c|D)$ in Eq. 10, is given by:

$$P(c|\hat{\theta}_D) = \begin{cases} Z_{D_c} \cdot P_C(c|D) & \text{if } c \in D \text{ and } P_C(c|D) > \delta_c \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

where Z_{D_c} is a document-specific normalization factor: $Z_{D_c} = 1/\sum_c P_C(c|D)$. We fix $\delta_t = \delta_c = 0.01$.

Table 2. Statistics of the CLEF 2008 Domain-specific test collection

Documents					Topics			Relevant Documents			
Total	Avg. length	Std. dev. length	Avg. concepts	Std. dev. concepts	Total	Avg. length	Std. dev. length	Total	Avg.	Min.	Max.
171319	198.3	42.3	10.1	4.2	25	3	1.7	2133	85	4	206

4 Experimental Setup

Other than replacing HTML entities we did not apply any preprocessing to the document collection. To estimate our concept models, we used the `CONTROLLED-TERM-EN` field in the documents. Given the models introduced in the previous sections, we need to estimate a number of parameters, viz. λ_Q (Eq. 6), $|\mathcal{D}_Q|$ (Eq. 4), $|\mathcal{V}_Q|$ (Eq. 4), and $|\mathcal{C}|$ (Eq. 7). 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 λ_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 [11,18]. While computationally expensive (exponential in the number of parameters), this approach provides us with an upper bound on the performance one might achieve using the described models.

As our baseline, we employ a run based on the KL-divergence retrieval method and set $\lambda_Q = 1$ (viz. Section 2, Eq. 6). As to μ (Eq. 3), we set this parameter to the average document length. All the results that we report on use this baseline as their initially retrieved document set. Since our concept language models also rely on pseudo-relevance feedback, we use the method introduced by [7] (Eq. 4) as another baseline.

5 Results and Discussion

Table 3 lists the results of our runs. We see that our conceptual language model (CM) has a significant positive effect on the number of relevant documents retrieved. Compared with QL and RM, CM loses in very early precision (P5), but not significantly. It already makes up for this later in the top 10 (P10) and even more so further down the ranking. The differences in P5, P10 and MAP between the three runs are not significant; given the relatively small number of topics (25), it is hard to achieve statistically significant differences.

Table 3. Results of the query likelihood (QL), relevance (RM) and conceptual language model (CM). Percentages indicate relative difference with QL. Significance is tested using a Wilcoxon sign rank test; * indicates a statistically significant difference against QL ($p < 0.05$).

	QL	RM		CM	
Relevant retrieved	1468	1473	+0.3%	**1602**	+9.1%*
P5	0.5280	**0.5680**	+7.6%	0.4880	-7.6%
P10	0.4680	0.4800	+2.6%	**0.4840**	+3.4%
MAP	0.2819	0.2856	+1.3%	**0.2991**	+6.1%

Next we turn to the precision-recall plot for our three runs, QL, RM and CM; see Figure 1. As can be expected, given the numbers in Table 3, at very low recall levels RM and QL both outperform CM; at high recall levels (between 0.5 and 0.9) CM outperforms QL and RM, that perform at very comparable levels.

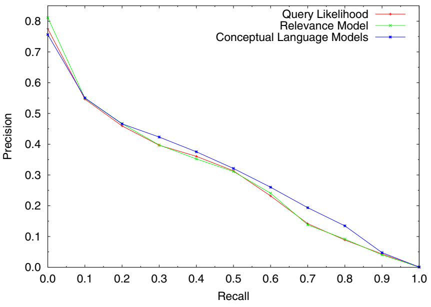
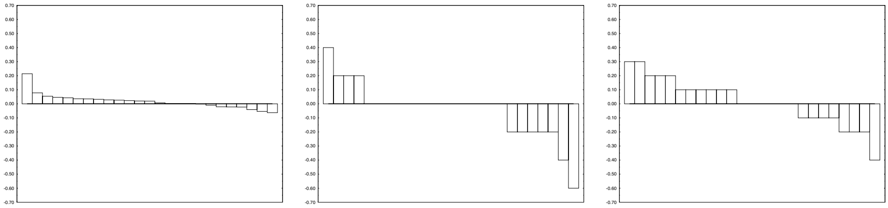


Fig. 1. Precision recall graph

Finally, we turn to a topic level comparison of CM and the baseline run QL; see Figure 2. First, in terms of MAP, CM outperforms QL on 14 out 25 topics, while QL beats CM on 8; there is a large gain for one topic (211: *Shrinking cities*). In terms of P5, CM outperforms QL on only 4 topics, while QL beats CM on 7; here, topics 223 (*Media in the preschool age*) and 210 (*Establishment of new businesses after the reunification*) are especially hard for CM (-0.40 and -0.70, respectively). In terms of P10, CM beats QL on 11 topics, but loses on 8: topics 223 and 210 are still amongst the topics on which CM loses, but the losses are not as dramatic as they were for P5 (-0.20 and -0.40, respectively).



(a) MAP (b) P5 (c) P10

Fig. 2. Per-topic breakdown of the improvement of CM over the QL baseline on various evaluation measures. A positive value indicates an improvement over the baseline.

6 Conclusion

We described our participation in the 2008 edition of the CLEF Domain Specific track. Specifically, we examined blind relevance feedback models and concept models. Applying relevance modeling techniques was found to have a positive effect on the current topics, in terms of mean average precision and precision@10. When applying concept models for blind relevance feedback, we observed an even bigger as well as significant improvement over the query-likelihood baseline, also in terms of mean average precision and early precision. The most noticable effect of our concept models was on recall; in future work, on larger topic sets, we aim to analyze these effects further.

Acknowledgements

This work was carried out in the context of the Virtual Laboratory for e-Science project, 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. The work was also supported by the DuOMAn project carried out within the STEVIN programme 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.

References

1. Anick, P.: Using terminological feedback for web search refinement: a log-based study. In: SIGIR 2003 (2003)
2. Balog, K., Weerkamp, W., de Rijke, M.: A few examples go a long way: constructing query models from elaborate query formulations. In: SIGIR 2008 (2008)
3. Chen, S.F., Goodman, J.: An empirical study of smoothing techniques for language modeling. In: ACL 1996 (1996)
4. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Royal Statistical Society. Series B 39(1), 1–38 (1977)
5. Hiemstra, D.: A linguistically motivated probabilistic model of information retrieval. In: Nikolaou, C., Stephanidis, C. (eds.) ECDL 1998. LNCS, vol. 1513, p. 569. Springer, Heidelberg (1998)
6. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In: SIGIR 2001 (2001)
7. Lavrenko, V., Croft, B.W.: Relevance based language models. In: SIGIR 2001 (2001)
8. Meij, E., de Rijke, M.: Thesaurus-based feedback to support mixed search and browsing environments. In: Kovács, L., Fuhr, N., Meghini, C. (eds.) ECDL 2007. LNCS, vol. 4675, pp. 247–258. Springer, Heidelberg (2007)
9. Meij, E., Trieschnigg, D., de Rijke, M., Kraaij, W.: Parsimonious concept modeling. In: SIGIR 2008 (2008)
10. Meij, E., Weerkamp, W., Balog, K., de Rijke, M.: Parsimonious relevance models. In: SIGIR 2008 (2008)
11. Mitra, M., Singhal, A., Buckley, C.: Improving automatic query expansion. In: SIGIR 1998 (1998)
12. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR 1998 (1998)
13. Rocchio, J.: Relevance feedback in information retrieval. In: The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice Hall, Englewood Cliffs (1971)
14. Trieschnigg, D., Meij, E., de Rijke, M., Kraaij, W.: Measuring concept relatedness using language models. In: SIGIR 2008 (2008)
15. Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: SIGIR 1996 (1996)
16. Zhai, C.: Risk Minimization and Language Modeling in Text Retrieval. PhD thesis, Carnegie Mellon University (2002)
17. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: CIKM 2001 (2001)
18. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems 22(2), 179–214 (2004)