

Enhancing Expert Search through Query Modeling

Pavel Serdyukov¹, Sergey Chernov², and Wolfgang Nejdl²

¹ Database Group, University of Twente, PO Box 217, 7500 AE Enschede, The Netherlands

² L3S / University of Hannover, Appelstr. 9a D-30167 Hannover, Germany
serdyukovpv@cs.utwente.nl, {chernov, nejdl}@l3s.de

Abstract. An expert finding is a very common task among enterprise search activities, while its usual retrieval performance is far from the quality of the Web search. Query modeling helps to improve traditional document retrieval, so we propose to apply it in a new setting. We adopt a general framework of language modeling for expert finding. We show how expert language models can be used for advanced query modeling. A preliminary experimental evaluation on TREC Enterprise Track 2006 collection shows that our method improves the retrieval precision on the expert finding task.

1 The Expert Finding Task

New challenges for the information retrieval research community are posed by the emerging field of Enterprise Search [2]. The diversity of complex information needs within a typical enterprise together with heterogeneity of Intranet data make it difficult to improve the quality of search in general. Instead, researchers concentrate on several important search tasks. One important example of such a task is finding a relevant expert within an organization. This problem implies that user needs to find the most knowledgeable expert to answer her query personally. User submits several keywords to a local Intranet search engine and receives a set of experts, ranked by their likelihood to be an expert for the query. The current developments in expert search are driven by the Expert Finding task within the TREC 2006 Enterprise Track initiative³. So far, one of the most comprehensive descriptions of the problem and possible solutions using language modeling approach is presented in [1]. We also adopt a theoretically-sound language modeling method, while using different techniques for the model estimation and ranking.

Numerous ad-hoc query expansion and language model based query modeling methods operate on the top- k ranked documents. At the same time, nobody applied these methods in the scope of expert finding task, what appears to be an omission in our opinion. Our algorithm allows performing a query modeling which consists of pseudo-relevance feedback and query expansion. To the best of our knowledge, this is the first study of query modeling applied to the expert search task. The preliminary evaluation on the official TREC Enterprise Track 2006 test collection shows that our method improves the retrieval performance.

³ <http://www.ins.cwi.nl/projects/trec-ent/wiki/index.php>

2 Expert Finding as a 2-Step Process

A comprehensive description of a language modeling approach to expert finding task is presented in [1]. We adopt the notation from this work and omit some details of model estimation; an interested reader can refer to the original paper. The Step-1 of our expert finding method is similar to Model 1 approach from [1], while The Step-2 contains the actual refinement and is essentially the core of our proposal.

2.1 Step 1: Using Language Model for Expert Ranking

The basic idea of language modeling is to estimate a language model for each expert, and then to rank experts by cross-entropy of estimated query model w.r.t. expert language model [3]. In our setup, each document d in the collection is associated with a candidate ca , the association is defined as $a(d, ca)$. Expert finding problem according to a probability ranking principle in IR is rephrased as: “What is the probability of a candidate ca to be an expert given the query q ?” Each candidate ca is represented by a multinomial probability distribution $p(t|ca)$ over a term vocabulary. Expert language model θ_{ca} is computed as the maximum likelihood estimate of a term generation probability, smoothed by the background language model. The query q is also represented by the probability distribution $p(t|q)$, and a query language model is denoted as θ_q . So, the system output should contain the ranking of candidates in descending order of cross-entropy between language models θ_q and θ_{ca} . A cross-entropy of query w.r.t. expert models is computed as shown in Eq.1:

$$ExpertScore_{ca}(q) = - \sum_{t \in q} p(t|\theta_q) \log p(t|\theta_{ca}) \quad (1)$$

The top- k experts with the highest scores are returned to the system (not to the user) as a result of a Step 1, where k is set empirically. So far we described the state-of-art approach, while Step 2 contains our enhancement for the expert search.

2.2 Step 2: Expert Ranking Refinement Using Query Modeling

In order to model a user query more precisely we need a source of additional knowledge about her information need. Traditionally, top- k documents for the query served in IR as such a source and were used to build an expanded and detailed query model. Expert search is a task which differs noticeably from a standard document retrieval. Users search not for the specific pieces of information, but for people who are actually generators and/or collectors of the information. It means that despite the query can be very specific, the experts in this topic can have an expertise in related topics too. Moreover, the broader their expertise, the higher are chances that they can consult on a more specific question. Therefore, we need to utilize two evidences about user information need in the context of expert finding task:

1. The top- k documents retrieved from the whole collection (using classic LM approach to document retrieval)

2. The top- k persons which we could consider relevant experts (retrieved on a Step 1).

The first source enriches our knowledge about the initial user information need. Whereas second one makes it less specific and relaxes a query towards a broader topic. So, as a new query model we use a mixture of two query models: document-based ($DocumentBasedNew\theta_q$) built on top- k documents and expert-based ($ExpertBasedNew\theta_q$) built on top- k experts:

$$p(t|New\theta_q) = \lambda p(t|DocumentBasedNew\theta_q) + (1 - \lambda)p(t|ExpertBasedNew\theta_q) \quad (2)$$

For the both query models estimation, instead of the methods proposed in [1], we use a principled and theoretically-sound method by Zhai and Lafferty from [3], which in our previous experiments for distributed IR outperformed other similar algorithms.

Once it is computed, we mix the new query model with an initial query to prevent a topic drift. As a result, we build a new expert ranking using expanded query and term generation probabilities. In Eq.3 we again measure a cross-entropy, but using a new query model:

$$NewExpertScore_{ca}(q) = - \sum_{t \in q} p(t|New\theta_q) \log p(t|\theta_{ca}) \quad (3)$$

3 Preliminary Results and Discussion

In our experiments we used the W3C collection, provided by the TREC 2006 Enterprise Track, and the Lucene⁴ open source information retrieval library. We indexed the mailing lists of W3C dataset⁵ and searched for the Title query part of the official topics of the Expert Finding task 2006. The comparison between precision at first 10 results (P@10) of baseline method (Step 1 only) and our method (Step 1 and Step 2) is presented on the Fig. 1 and Fig. 2.

We observe that the improvement of our method is promising, while not significant in the current experiment. Our method is effective when an average precision is high already at the step 1, and fails where average precision is below median. This is explainable since our method uses best top- k experts and documents from the Step 1 for the following query modeling. If the initial ranking is poor, the query modeling is poor too. But the precision for the best queries was improved by 10-20%, so this method is suitable to apply on top of already effective retrieval systems. It appears that a prediction of query performance could be crucial for a query modeling. The further study of the expert-search-specific query modeling and predicting of a query performance is the main focus of our future research.

⁴ <http://lucene.apache.org/>

⁵ For a rapid experimental setup we used only the mailing list part, while we are planning to evaluate our method on the whole collection later.

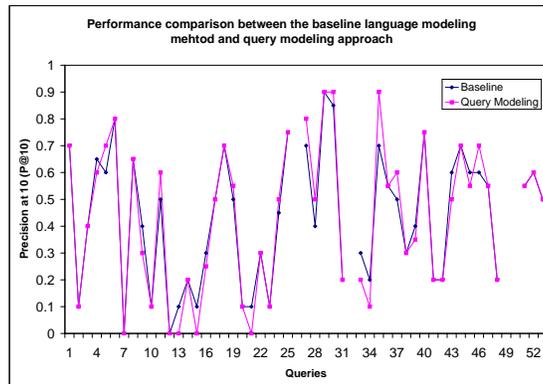


Fig. 1. Performance of the baseline language modeling ranking and query modeling approach.

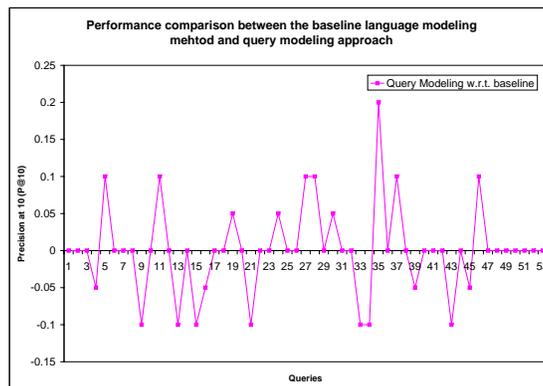


Fig. 2. Difference in performance of the baseline language modeling ranking and query modeling approach.

References

1. K. Balog, L. Azzopardi, and M. de Rijke. Formal models for expert finding in enterprise corpora. In *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, USA, pages 43–50. ACM Press, 2006.
2. D. Hawking. Challenges in enterprise search. In *Proceedings of the Australasian Database Conference ADC2004*, pages 15–26, Dunedin, New Zealand, 2004.
3. C. Zhai and J. D. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01: Proceedings of the 2001 ACM CIKM International Conference on Information and Knowledge Management*, Atlanta, Georgia, USA, November 5-10, 2001, pages 403–410, 2001.