Combining Candidate and Document Models for Expert Search

Krisztian Balog **Maarten de Rijke**

ISLA, University of Amsterdam
http://ilps.science.uva.nl/

Abstract: We describe our participation in the TREC 2008 Enterprise track and detail our language modeling-based approaches. For document search, our focus was on query expansion using profiles of top ranked experts and on document priors. We found that these techniques result in small, but noticeable improvements over our baseline method. For expert search, we combine candidate- and document-based models, and also bring in web evidence. We found that the combined models significantly and consistently outperformed our very competitive baseline models.

1 Introduction

Similarly to last year, the TREC 2008 enterprise track featured two separate tasks: *document search* and *expert finding*. For both tasks, we experiment with a query expansion technique using profiles of top ranked experts and with encoding query-independent features as (document and candidate) priors. Further, concerning the expert search task we consider both candidate- and document-based models, as well as their combination.

Our main findings are that for document search our attempts at query modeling and the use of document priors meet with limited success, although noticeable improvements in average precision can be observed. For expert finding, we arrive at more interesting findings. First, in contrast with the literature and with our previous studies [3, 7] we find that candidate models (introduced as "Model 1" in [3]) can outperform document-based models (a.k.a. "Model 2" from [3]). Specifically, we compare a proximity-based version of the candidate-based model ("Model 1B"), complemented with a fine-grained method for estimating the strength of the association between documents and candidates, based on global statistics and semantic relatedness [2] with the document-based model employed on top of our best performing document search run. Second, we find that a combination of the two strategies (Model 1B and Model 2) outperforms both. Third, query modeling, using blind feedback both from documents and experts, helps improve retrieval performance. Fourth, bringing in web evidence boosts performance even further.

The paper is organized as follows. We discuss our work on the document search task (Section 2) and on the expert search task (Section 3) in two largely independent sections. We conclude our findings and put forward suggestions for future work in Section 4.

2 Document Search

The aim of the document search task is to retrieve documents that help a science communicator within an organization (in this case CSIRO) create an overview page for a given topical area. Relevant documents are therefore documents that discuss the given topic in detail and not the ones that only touch on the topic. Last year the usual TREC-style topic definitions were expanded with a number of examples of key pages. These example documents could then be used to construct rich query models [5, 6]. One of our major aims this year is to devise ways of constructing rich query models when such elaborate specifications of information needs are not available. In addition, we experiment with using a document prior.

2.1 Modeling

We employ a standard language modeling approach to IR and rank documents by their log-likelihood of being relevant given a query. Without presenting details here we only provide our final formula for ranking documents, and refer the reader to [6] for a derivation of this equation:

$$\log P(D|Q) \propto \log P(D) + \textstyle\sum_{t \in Q} P(t|\theta_Q) \cdot \log P(t|\theta_D). \quad (1)$$

Here, both documents and queries are represented as multinomial distributions over terms in the vocabulary. We estimate each document model (θ_D) by:

$$P(t|\theta_D) = (1-\lambda_D) \cdot P(t|D) + \lambda_D \cdot P(t), \quad (2)$$

where $P(t|D)$ and $P(t)$ are maximum likelihood estimates of the term t on the document and on the collection, respectively, and λ_D is a smoothing parameter.

Next, we address the estimation of the other two components of our modeling: the query model θ_Q in Section 2.1.1 and document priors $P(D)$ in Section 2.1.2.

2.1.1 Query models

We consider constructing the query model from three components according to the following equation:

$$\begin{aligned} P(t|\theta_Q) &= \lambda_Q \cdot P(t|\hat{\theta}_Q) \\ &+ \mu \cdot P(t|\check{\theta}_Q) \\ &+ (1-\lambda_Q-\mu) \cdot P(t|Q). \end{aligned} \tag{3}$$

Here, $P(t|\hat{\theta}_Q)$ is estimated using relevance models (method 2) of Lavrenko and Croft [11], $P(t|\check{\theta}_Q)$ is constructed from profiles of candidate experts, and $P(t|Q)$ is the initial query.

Sampling expansion terms from expert profiles is performed using the following algorithm. First, we rank experts using expert finding Model 1B described in Section 3.1.1. Then, we obtain $P(t|S)$ by taking terms from the profiles of the top ranked M experts:

$$P(t|S) = \sum_{ca \in M} P(t|\theta_{ca}) \cdot P(ca|S), \tag{4}$$

where $P(t|\theta_{ca})$ is the probability of term t given the candidate's language model, and $P(ca|S)$ is proportional to how likely candidate ca is an expert, given the top M experts:

$$P(ca|S) = \frac{P(ca|Q)}{\sum_{ca' \in M} P(ca'|Q)}. \tag{5}$$

Calculating the sampling distribution $P(t|S)$, therefore, can be viewed as the following generative process:

1. Let the set of candidate experts $\{ca \in M\}$ be given
2. Select a candidate ca from this set with probability $P(ca|S)$.
3. From this candidate, generate the term t with probability $P(t|\theta_{ca})$

Finally, we take the top K terms from $P(t|S)$ to form $P(t|\check{\theta}_Q)$.

2.1.2 Document priors

Since we are looking for key pages, our intuition is that these pages have shorter URLs than non-key pages. This heuristic has already proved useful for web document search and can effectively be encoded as a document prior [9, 10]. We set $P(D)$ in Eq. 1 as follows:

$$P(D) \propto C - URL\_LENGTH(D), \tag{6}$$

where C is a constant (here set to 255), and $URL\_LENGTH(D)$ denotes the length of the URL (number of characters) of document D.

2.2 Runs

We submitted the runs listed below, all of which were automatic. To estimate the parameters of our models, such as the number of feedback documents and terms, and the interpolation weights in Eq. 3 we use the 2007 topic set.

UvA08DSbl the baseline run; uses only the initial query without expansion ($\lambda_Q = \mu = 0$) and document priors are set to be uniform.

UvA08DSbfb blind feedback run; query model uses the relevance model component ($\lambda_Q = 0.5$, top 10 terms from top 5 documents) but not the expert profiles component ($\mu = 0$). Document priors are set to be uniform.

UvA08DSexp query expansion using expert profiles; same as UvA08DSbfb but with $\lambda_Q = 0.4$ and also using candidate profiles for expansion ($\mu = 0.2$, top 10 terms from top 5 experts). Document priors are set to be uniform.

UvA08DSall all features; query model is constructed as in UvA08DSexp and document priors are set based on URL character length.

For the estimation of the document language model (θ_D) we employ Bayes smoothing with Dirichlet priors, i.e., put $\lambda_D = \beta/(|d|+\beta)$ in Eq. 2, and set β to be the average document length ($\beta = 260$).

2.3 Results

Our results for the 2008 document search task are listed in Table 2. In terms of infAP, UvA08DSall outperforms the other runs, but in terms of infNDCG, no run beats the baseline run UvA08DSbl. For comparison, we have included the results of runs produced on last year's data; see Table 1. Although the official metrics used in 2007 were different from those used in 2008, we can observe similar patterns: UvA07DSall beating the other approaches on all metrics except MRR, where the baseline beats the other approaches.

Run	MAP	P5	P10	P20	MRR
UvA07DSbl	.3853	.6520	.5940	.4870	**.8675**
UvA07DSbfb	.3953	.6560	.6100	**.4930**	.8030
UvA07DSexp	.4002	.6640	.6040	.4920	.7981
UvA07DSall	**.4056**	**.6800**	**.6140**	**.4930**	.8098

Table 1: Results for the document search task: 2007 topic set. Best scores for each metric are in boldface.

Run	infAP	infNDCG
UvA08DSbl	.3103	**.4938**
UvA08DSbfb	.3209	.4889
UvA08DSexp	.3242	.4854
UvA08DSall	**.3306**	.4909

Table 2: Results for the document search task: 2008 topic set. Best scores for each metric are in boldface.

3 Expert Search

For the expert search task, our aim was to experiment with a proximity-based version of the candidate model that we have introduced before [2], to combine it with document-based models, to determine the effectiveness of query modeling, and to bring in web evidence.

3.1 Modeling

Our approach to ranking candidates is as follows:

$$P(ca|Q) \propto P(ca) \cdot P(Q|ca), \qquad (7)$$

where $P(ca)$ is the *a priori* probability of the candidate ca being an expert, and $P(Q|ca)$ is the probability of ca generating the query Q. Our choice of setting $P(ca)$ is presented in Section 3.1.3. For estimating $P(Q|ca)$ we consider both candidate (Section 3.1.1) and document (Section 3.1.2) models.

3.1.1 Candidate model (Model 1B)

We use a proximity-based version of the candidate model, referred to as *Model 1B* [7]. Here, a language model θ_{ca} is inferred for each candidate and the log-query-likelihood of a candidate producing the query is obtained as follows:

$$\log P(Q|ca) = \sum_{t \in Q} P(t|\theta_Q) \cdot \log P(t|\theta_{ca}), \qquad (8)$$

where $P(t|\theta_{ca})$ is a linear interpolation between an empirical candidate model ($P(t|ca)$) and the background (collection) language model ($P(t)$):

$$P(t|\theta_{ca}) = (1 - \lambda_{ca}) \cdot P(t|ca) + \lambda_{ca} \cdot P(t). \qquad (9)$$

The probability $P(t|ca)$ is estimated based on the co-occurrance of the term t and candidate ca in a particular window size w (which was set to 125 based on empirical exploration). The model we use corresponds to Model 1B with semantic document-candidate associations (SEM) described in [6].

Recent work on expertise retrieval has indicated the usefulness of web evidence [8, 12]. In these studies Model 2 is applied on top of search engine results (either snippets or full documents). We also used web evidence, but in a candidate-based fashion. A web-based variation of Model 1B was employed, where the candidate's name was used as a query, issued to a web search engine API (in our case: Yahoo!). Then, text from the top 100 result snippets was used to construct $P(t|ca)$.

3.1.2 Document model (Model 2)

Using a document-based model the estimation of $P(Q|ca)$ is goes as follows:

$$P(Q|ca) = \sum_{D} P(Q|D) \cdot P(D|ca). \qquad (10)$$

We use the approach developed for ranking documents to estimate $P(Q|D)$ (see Section 2.1). As to $P(D|ca)$, we use the semantic relatedness of document D and candidate ca (the same settings that for the candidate model); see [1, Section 6.3.5] for details.

3.1.3 Candidate priors

We use candidate priors to filter out science communicators (SC) (often called *communication officer/manager/advisor* or *manager public affairs communication*). Following [2], we first extracted names and positions from contact boxes of CSIRO pages. Then, SCs were assigned the value 0, while all other people were assigned the value 1 as a candidate prior:

$$P(ca) = \begin{cases} 1, & ca \notin SC, \\ 0, & ca \in SC. \end{cases} \qquad (11)$$

3.1.4 Runs

We submitted the following 4 runs:

UvA08ESm1b Model 1B using the initial query (without expansion).

UvA08ESm2all Model 2 using expanded query models and all document search features (on top of document search run UvA08DSall)

UvA08EScomb linear combination of Model 1B (with weight 0.7) and Model 2 (with weight 0.3). Both models use the initial query (without expansion).

UvA08ESweb linear combination of the run UvA08EScomb (with weight 0.75) and the Web-based variation of Model 1B (with weight 0.25). The web run uses the query model from UvA08DSexp.

We employed candidate priors as described in Section 3.1.3 for all runs.

3.2 Results

Table 4 shows that the most successful strategy is to put everything together: UvA08ESweb outperforms our other runs. Interestingly, Model 1B outperforms Model 2; note that the run labeled UvA08ESm1b does not employ query expansion, while UvA08ESm2all uses features that improved performance on the document search task (see Section 2.3), including query expansion. Furthermore, we see that a combination of the two methods outperforms both models on all metrics. And finally, bringing in web evidence helps improve retrieval comparison even further (see the run labeled UvA08ESwb). Looking at the corresponding scores on the 2007 topic set (Table 3), we observe very similar behavior.

Run	#rel_ret	MAP	P@5	P@10	MRR
UvA07ESm1b	124	.4838	.2800	.1740	.6334
UvA07ESm2all	**126**	.4799	.2600	.1800	.6268
UvA07EScomb	121	.5267	.2880	**.1820**	**.6828**
UvA07ESweb	122	**.5405**	**.3080**	.1780	.6468

Table 3: Results for the Expert Search task: 2007 topic set. Best scores for each metric are in boldface.

Run	#rel_ret	MAP	P@5	P@10	MRR
UvA08ESm1b	394	.3935	.4836	.3473	.8223
UvA08ESm2all	395	.3679	.4473	.3436	.6831
UvA08EScomb	419	.4331	.4982	.3836	.8547
UvA08ESweb	**425**	**.4490**	**.5527**	**.3982**	**.8721**

Table 4: Results for the Expert Search task: 2008 topic set. Best scores for each metric are in boldface.

4 Conclusions

We described our participation in the TREC 2008 Enterprise track. Building on our earlier work [1–7], we employed a standard language modeling setting for both the document and expert tasks. Our aim for the document search task was to experiment with query expansions and with document priors. While we observed improvements, our overall conclusion is that these techniques resulted in limited success.

As to the expert search task, our experiments concerned the combination of candidate- and document-based methods, and bringing in web evidence. We found that these models captured different experts, and therefore, combining them resulted in substantial improvements for all metrics.

These results suggest that possible improvements might be pursued in the combination of methods, as well as in further use of web evidence.

Acknowledgments

This research was supported 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, and 640.004.-802.

5 References

[1] K. Balog. *People Search in the Enterprise*. PhD thesis, University of Amsterdam, June 2008.

[2] K. Balog and M. de Rijke. Non-local evidence for expert finding. In *ACM 17th Conference on Information and Knowledge Managment (CIKM 2008)*, pages 489–498. ACM, October 2008.

[3] 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*, pages 43–50, New York, NY, USA, 2006. ACM Press. ISBN 1-59593-369-7. doi: http://doi.acm.org/10.1145/1148170.1148181.

[4] K. Balog, E. Meij, and M. de Rijke. Language models for enterprise search: Query expansion and combination of evidence. In *The Fourteenth Text Retrieval Conference (TREC 2006)*. NIST, 2007. Special Publication.

[5] K. Balog, K. Hofmann, W. Weerkamp, and M. de Rijke. Query and document models for enterprise search. In *The Sixteenth Text Retrieval Conference (TREC 2007)*. NIST, 2008. Special Publication.

[6] K. Balog, W. Weerkamp, and M. de Rijke. A few examples go a long way: constructing query models from elaborate query formulations. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 371–378, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-164-4. doi: http://doi.acm.org/10.1145/1390334.1390399.

[7] K. Balog, L. Azzopardi, and M. de Rijke. A language modeling framework for expert finding. *Information Processing and Management*, 45(1):1–19, January 2009. doi:10.1016/j.ipm.2008.06.003.

[8] J. Jiang, S. Han, and W. Lu. Expertise retrieval using search engine results. In *Proceedings of the SIGIR 2008 Workshop on Future Challenges in Expertise Retrieval (fCHER)*, pages 11–16, 2008.

[9] J. Kamps, G. Mishne, and M. de Rijke. Language models for searching in web corpora. In *The Thirteenth Text REtrieval Conference (TREC 2004)*, 2004.

[10] W. Kraaij, T. Westerveld, and D. Hiemstra. The importance of prior probabilities for entry page search. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 27–34. ACM Press, 2002.

[11] V. Lavrenko and W. Croft. Relevance-based language models. In *SIGIR '01*, pages 120–127, 2001.

[12] P. Serdyukov and D. Hiemstra. Being omnipresent to be almighty: The importance of global web evidence for organizational expert finding. In *Proceedings of the SIGIR 2008 Workshop on Future Challenges in Expertise Retrieval (fCHER)*, pages 17–24, 2008.