High Quality Expertise Evidence for Expert Search

Craig Macdonald, David Hannah, and Iadh Ounis

University of Glasgow, G12 8QQ, UK
{craigm,hannahd,ounis}@dcs.gla.ac.uk

Abstract. In an Enterprise setting, an expert search system can assist users with their "expertise need" by suggesting people with relevant expertise to the topic of interest. These systems typically work by associating documentary evidence of expertise to each candidate expert, and then ranking the candidates by the extent to which the documents in their profile are about the query. There are three important factors that affect the retrieval performance of an expert search system - firstly, the selection of the candidate profiles (the documents associated with each candidate), secondly, how the topicality of the documents is measured, and thirdly how the evidence of expertise from the associated documents is combined. In this work, we investigate a new dimension to expert finding, namely whether some documents are better indicators of expertise than others in each candidate's profile. We apply five techniques to predict the quality documents in candidate profiles, which are likely to be good indicators of expertise. The techniques applied include the identification of possible candidate homepages, and of clustering the documents in each profile to determine the candidate's main areas of expertise. The proposed approaches are evaluated on three expert search task from recent TREC Enterprise tracks and provide conclusions.

1 Introduction

Modern expert search systems in Enterprise settings work by using documents to form the profile textual evidence of expertise for each candidate. The profiles represent the system's knowledge of the expertise of each candidate, and on receiving a user query, they are ranked by how well the documents in their profile are related to the query [1,2]. For example, the Voting Model for expert search [3] sees this as a voting process: documents in the collection are ranked in response to the query, and then each document retrieved that is associated with a candidate is seen as a vote for that candidate to be retrieved for the query.

The retrieval performance of an expert search system is very important. Indeed, expert search has been a retrieval task in the Enterprise tracks of the Text REtrieval Conferences (TREC) since 2005 [4], aiming to evaluate state-of-the-art expert search approaches. This effort has generated two test collections for expert search, namely the W3C collection, and the CERC collection.

Several important factors have been investigated that can impact the retrieval performance of an expert search system. Firstly, the manner in which the

C. Macdonald et al. (Eds.): ECIR 2008, LNCS 4956, pp. 283–295, 2008.

evidence of expertise in the associated documents of each candidate are combined has an impact on the retrieval performance of the expert search system [3,5]. Secondly, it has been shown several times that the retrieval performance of an expert search system can be improved if the means by which the topicality of a document to a query is improved [6,7,8,9]. The better the expert search system is able to identify only on-topic documents in the corpus, the more likely it is that the inferences of expertise that can be drawn from the documents will be correct - i.e. off-topic documents will not give erroneous votes to non-relevant candidates. Moreover, various past research has applied query expansion [6,8,9], document structure [6,7] and proximity of query terms in documents to improve the underlying document retrieval system [6,10]. Thirdly, various research in expert search has observed that the quality of the candidate profiles has a major impact on the retrieval performance of the expert search system [11,12]. In particular, if one or more documents about the query topic which should be associated to a relevant candidate are omitted, then retrieval performance can be impaired. Indeed, the principle of accumulation of evidence suggests it is better to obtain as much expertise evidence as possible for a candidate.

In the area of Web IR, documents usually have a notion of quality associated with them. For example, a document that is linked to by many other documents is considered to be more authoritative about a topic than another less linked document, or a document that has a short URL is likely to be a homepage which users prefer. Web IR retrieval systems often take such sources of evidence into account when ranking Web documents, to improve the retrieval performance of the search engine [13,14].

In a similar vein, the aim of this work is to investigate a new aspect of the expert search system, which is the identification of high-quality evidence in the candidate profiles. We believe that if a notion of high-quality expertise evidence for a candidate can be defined, then this evidence can be successfully taken into account when ranking candidate experts. For instance, a document which is the homepage of a candidate is more likely to contain useful evidence of expertise than the minutes of a meeting that the candidate attended. However, it is not necessarily safe to remove all meeting minutes from all the candidate profiles, as this could prevent a relevant candidate from being retrieved for a difficult query. Instead, it is safer to weight higher (i.e. give stronger votes) the documents in a profile that we believe bring more expertise evidence about the candidate.

In this paper, we propose five techniques to predict the quality documents in candidate profiles, which are likely to be good indicators of expertise. We carry out the experiments with integrating these technique using the Voting Model for expert search, because the voting paradigm provides a natural and flexible mechanism to incorporate such additional evidence into an expert search system. This paper is structured as follows: Section 2 reviews models for expert search, and defines the voting technique we apply in this work; Section 3 proposes the five techniques to determine the quality expertise evidence in candidate profiles; We detail the experimental setting, including the test collections used in Section 4;

Section 5 provides results and analysis of the proposed techniques; We make concluding remarks in Section 6.

2 Expert Search

There are two requirements for an expert search system: a list of candidate persons that can be retrieved by the system, and some textual evidence of the expertise of each candidate to include in their profile. In most Enterprise settings, a staff list is available and this list defines the candidate persons that can be retrieved by the system. Candidate profiles can be created either explicitly or implicitly: candidates may explicitly update their profile with an abstract or list of their skills and expertise; or alternatively, the expert search system can implicitly and automatically generate each profile from a corpus of documents. This documentary evidence can take many forms, such as intranet documents, documents or emails authored by the candidates, or even emails sent by the candidate or web pages visited by the candidate (see [3] for an overview). In this work, the profile of a candidate is considered to be the set of documents associated with the candidate.

Once a profile of evidence has been identified for every expert, these can then be used to rank candidates automatically in response to a query. Various expert search approaches were proposed by participants of the TREC 2005 and TREC 2006 Enterprise tracks. These include that of Balog et al., who proposed the use of language models in expert search [5]. They proposed two models for expert search, however the approach is limited to the use of language modelling to provide the estimates for the relevance of a document to the query. Similarly to Balog et al., Fang and Zhai [15] applied relevance language models to the expert search task. In contrast, the probabilistic approach proposed by Cao et al. [16] and the hierarchical language models proposed by Petkova and Croft [6] do not consider expertise evidence on a document level, but instead work on a more fine-grained approach using windowing.

Instead, this work uses the Voting Model for expert search proposed by Macdonald & Ounis in [3], which considers the problem of expert search as a voting process. Instead of directly ranking candidates, it considers the *ranking of documents*, with respect to the query Q, denoted by $R(Q)$. The ranking of candidates can then be modelled as a voting process, from the retrieved documents in $R(Q)$ to the profiles of candidates: every time a document is retrieved and is associated with a candidate, then this is a vote for that candidate to have relevant expertise to Q. The ranking of the candidate profiles can then be determined by aggregating the votes of the documents. Twelve voting techniques for ranking experts were defined in [3], each employing various sources of evidence that can be derived from the ranking of documents with respect to the query topic.

In this work, we only use the expCombMNZ voting technique [3], because it provides effective and robust results across several expert search test collections and document weighting models - for example experiments applying various voting techniques combined with BM25, PL2 and DLH13 showed that expCombMNZ is not only one of the best voting techniques, but that it is stable across different document weighting model [7]. expCombMNZ ranks candidates by considering the sum of the exponential of the relevance scores of the documents associated with each candidate's profile. Moreover, it includes a component which takes into account the number of documents in $R(Q)$ associated to each candidate, hence explicitly modelling the number of votes made by the documents for each candidate. Hence the relevance score of a candidate expert C with respect to a query Q, $score\_cand(C, Q)$, is:

$$score\_cand(C, Q) = \|R(Q) \cap profile(C)\| \cdot \sum_{d \in R(Q) \cap profile(C)} exp(score(d, Q)) \quad (1)$$

where $profile(C)$ is the set of documents associated with candidate C, and $score(d, Q)$ is the relevance score of the document in the document ranking $R(Q)$. $\|R(Q) \cap profile(C)\|$ is the number of documents from the profile of candidate C that are in the ranking $R(Q)$, and $exp()$ is the exponential function. The exponential function boosts candidates that are associated to highly scored documents (strong votes).

Documents are ranked using the DLH13 document weighting model [17] from the Divergence from Randomness (DFR) framework. We chose to experiment using DLH13 because it has no term frequency normalisation parameter that requires tuning, as this is assumed to be inherent to the model. Hence, by applying DLH13, we remove the presence of any term frequency normalisation parameter in our experiments. Moreover, as mentioned above, it performs comparably to BM25 and PL2 when combined with expCombMNZ on this task [3,7].

3 Quality Evidence in Candidate Profiles

As described in the introduction, there are three factors that can have a major impact on the retrieval performance of an expert search system. Firstly, the technique used to generate the initial ranking of documents $R(Q)$ has an impact on the retrieval performance of the expert search system. Previous work has shown that applying various document retrieval enhancing techniques (such as query expansion) results in a better ranking of candidates [6,7,8,9].

Secondly, the technique used to aggregate the document votes into a ranking of candidates also has a bearing on the retrieval performance. Of the twelve voting techniques described in [3], only some techniques produce a good retrieval performance, of which we use expCombMNZ in this work for the reasons detailed in Section 2.

Lastly, the quality of the candidate profiles used in an expert search system can have a major impact on the retrieval performance of the system. Due to the ambiguity of names, obfuscation of email addresses etc., the authorship of a document is difficult to generically identify in a heterogeneous corpus. Hence, if an on-topic document is not associated with its author (say), then that candidate will not receive a vote from that document.

In [11], Balog et al. investigated how expertise evidence should be identified from the emails of the W3C corpus. Interestingly, it was found that being included in the CC field on an email was more important than being the author of an email, for use as expertise evidence. Similarly, in [12], the authors investigated the impact on retrieval performance of the method of identifying expertise evidence for each candidate. For instance, they compared the effectiveness of an expert search system when candidates were identified by their full names, by their emails or by their last-name alone in the documents. They found that the choice of identification method had a major impact on the performance of the expert search system, and that the most exact form of identification (full name) gave the best retrieval performance.

For this work, we aim not to investigate the identification of profile evidence for candidates, but instead to determine which part of the candidates profiles should be considered as quality expertise evidence. This is similar to the notion of quality documents that exists in the Web IR field, where techniques such as, to name but a few, link analysis and URL length can be used as measures of the quality of a document. As mentioned in Section 1, the central idea of this paper is to take into account a quality measure in assessing the documents within a candidate profile. In particular, we propose measures that predict the high quality expertise evidence in a candidate's profile. The central hypothesis of this paper is that by identifying and weighting quality expertise evidence in the candidate profiles, the retrieval performance of the expert search system will be improved. In this work we propose five different techniques for identifying high quality expertise evidence within a candidate profile. While all techniques depend on the document, some techniques take into account the query, and/or the name of the candidate. The techniques include Web IR techniques such as URL Length and document Inlinks, as well as techniques that examine the proximity of the query to occurrences of the candidate's name, attempt to identify each candidate's home page, and lastly determine if a document is about a central interest of a candidate by using clustering. These are detailed in Sections 3.1-3.4 below.

We can compute a score for each of the above sources of evidence of a quality document in a candidate profile, which is denoted as $Qscore(d, C, Q)$, and integrate it with the expCombMNZ voting technique as follows:

$$score\_cand(C, Q) = \|R(Q) \cap profile(C)\| \cdot \sum_{d \,\in\, R(Q) \cap profile(C)} exp(score(d, Q) + \omega \cdot Qscore(d, C, Q)) \quad (2)$$

where ω is a parameter. Note that if $Qscore$ is 0, then the candidate still receives a vote equivalent to the relevance score of the document. In this way, no expertise evidence is removed and the principle of accumulation of evidence is upheld. Note also that Equation (2) is only one way in which the measures of quality could be integrated - other ways may exist that might improve the overall effectiveness of the expert search system, but for the purpose of this paper our main objective is to ascertain to which extent taking into account the quality evidence within a profile is important. In the remainder of this section, we detail each proposed technique for identifying quality documents, and explain how they can be weighted and the resultant *Qscore* integrated into the applied voting technique.

3.1 Candidate Homepages

Usually, the homepage of a person contains personalised information, particularly about professional interests and role in the organisation, while in a research environment, it may also contain the titles of their publications. If the corpus contains webpages that could be seen as the candidate's homepage, then we can assume that this page has good evidence of the candidate's expertise. We believe that this is a form of high quality evidence of expertise, which should be weighted higher if it matches an expert search query.

Both the TREC W3C and CERC collections pose a problem for the identification of candidate homepages, for various reasons. In the W3C collection, not all candidates are employed by the W3C and hence only some candidate have homepages within the w3c.org domain, even though the URL location of the homepages of the candidates that have them is fairly predictable. For the CERC collection, not all staff have homepages, and the form of the URL of these vary from person to person. Some employees have personal homepages that they maintain, while others have just database-managed pages detailing their research interests. However, the problem here is that these are difficult to identify from the URL structure, due to the compartmentalised nature of the CSIRO organisation (e.g. different research divisions), which is mirrored in the different URL hosts with different directory layouts in the corpus.

In this paper, we propose a general technique to identify homepages in both of the test collections used. It is based on the assumption that pages such as a candidate's homepage (or the candidate's research interests page) will often have anchor text linking to that page containing predominantly the candidate's name. To identify these homepages, we firstly build an index for all documents that consists only of the anchor text of the incoming hyperlinks to each document. Then, for each candidate, we construct a phrasal search query using the exact full name of the candidate. This query is then run on the anchor-text index, giving a ranking of predicted homepages for each candidate, and a score for the document as calculated by a document weighting model. For efficiency, this procedure can be done offline, before retrieval. During expert search, votes from the predicted homepage documents are strengthened.

We integrate this homepage evidence into the expCombMNZ voting technique (Equation (2)) by calculating *Qscore* as follows:

$$Qscore_{HP}(d, C, Q) = score_{Anchor}(name(C), d) \tag{3}$$

where $score_{Anchor}(name(C), d)$ is the score calculated by the document weighting model on the anchor text only index, for document d and the query being the name of the candidate. To remain consistent with $score(d, Q)$, we use the DLH13 document weighting model to generate $score_{Anchor}(name(C), d)$.

3.2 Candidate-Name and Query Proximity

Some types of documents can have many topic areas and many occurrences of candidate names (for instance, the minutes of a meeting). In such documents, the closer a candidate's name occurrence is to the query terms, the more likely that the document is a high quality indicator of expertise for that candidate [6,16].

We define $Qscore_{prox}(d, C, Q)$ in terms of the DFR term proximity document weighting model [10]. The term proximity model is designed to measure the informativeness in a document of a pair of query terms occurring in close proximity. We adapt this to the expert search task and into the expCombMNZ voting technique (Equation (2)), by measuring the informativeness of a query term occurring in close proximity to a candidate's name, as follows:

$$Qscore_{prox}(d, C, Q) = \sum_{p=name(C) \times t \in Q} score(d, p) \tag{4}$$

Here p is a tuple of a term t from the query and the full name of candidate C. $score(d, p)$ can be calculated using any DFR weighting model [10], however, for efficiency reasons, we use a model that does not consider the frequency of tuple p in the collection but only in the document:

$$\begin{aligned} score(d, p) = \frac{1}{pfn + 1} \cdot \Big(& -\log_2 (avg\_w - 1)! + \log_2 pfn! \\ & + \log_2(avg\_w - 1 - pfn)! \\ & - pfn \log_2(p_p) \\ & - (avg\_w - 1 - pfn) \log_2(p'_p) \Big) \end{aligned} \tag{5}$$

where $avg\_w = \frac{T - N(ws-1)}{N}$ is the average number of windows of size ws tokens in each document in the collection, N is the number of documents in the collection, and T is the total number of tokens in the collection. $p_p = \frac{1}{avg\_w - 1}$, $p'_p = 1 - p_p$, and pfn is the normalised frequency of the tuple p, as obtained using Normalisation 2 [10]: $pfn = pf \cdot \log_2(1 + c_p \cdot \frac{avg\_w - 1}{l - ws})$. In Normalisation 2, pf is the number of windows of size ws in document d in which the tuple p occurs. l is the length of the document in tokens and $c_p > 0$ is a hyper-parameter that controls the normalisation applied to pfn frequency against the number of windows in the document.

3.3 URL Length and Inlinks

In order to ascertain the high quality documents within a candidate profile, we apply sources of evidence inspired by work in the Web IR field about measuring the quality of a web page. In a Web IR setting, a document with many incoming links is likely to be of good quality, and indeed, link information within Enterprise settings has previously been found to be useful in intranet search [18,19].

In adapting this evidence to expert search, we assume that documents with shorter URLs are of higher importance and quality in the organisation, and that evidence of expertise obtained from them is of more importance. Similarly, documents with more inlinks are likely to be of good quality, and of more use in an expert search system. Note that most link analysis techniques (e.g. PageRank and Absorbing Model) have been shown to be strongly correlated to a simple count of the number of incoming hyperlinks (inlinks) to each document [20]. For this reason, in this paper we only use inlinks.

We follow Craswell et al. [14] by integrating URL path length and inlinks into the expCombMNZ voting technique (Equation (2)) using two saturation functions, respectively:

$$Qscore_{URL}(d,C,Q) = \frac{\kappa}{\kappa + URLPathLength(d)} \tag{6}$$

$$Qscore_{Inlinks}(d,C,Q) = \frac{\kappa \cdot \beta \cdot Inlinks(d)}{\kappa + \beta \cdot Inlinks(d)} \tag{7}$$

where $URLPathLength(d)$ is the number of characters in the path component of the URL of document d, κ is a parameter, $Inlinks(d)$ is the number of incoming hyperlinks to document d, and $\beta = \frac{N}{\sum_d Inlinks(d)}$, in which N is the number of documents in the collection. The purpose of β is to ensure that the mean of the inlinks distribution is 1.

3.4 Clustering of Candidate Profiles

Candidates can have many areas of expertise over the timespan of the organisation, and this can be measured as topic drift in their candidate profiles [9]. In this work, we use clustering to identify the main interests of each candidate, particularly for these prolific candidates. By clustering a candidate profile, the main expertise areas of the candidate should be reflected as the largest clusters. Votes for the candidate to be retrieved by documents that are about one of the candidate's main interests (i.e. one of the larger clusters) should be higher weighted.

In particular, in this paper we use a single-pass clustering algorithm to cluster the profiles of candidates who have more than θ documents in their profile. In the clustering, the cluster distance is defined as the Cosine between the average of each clusters. The clusters obtained are then ranked by the number of documents they contain, and we select the largest K clusters as representatives of the central interests of the expert. We integrate this evidence into the expCombMNZ voting technique (Equation (2)):

$$Qscore_{Cluster}(d,C,Q) = \begin{cases} \frac{1}{cluster(d,C)} & if\ cluster(d,C) \leq K \\ 0 & otherwise \end{cases} \tag{8}$$

where $cluster(d,C)$ is the rank of the cluster in which document d occurred for candidate C (largest cluster has rank 1). The above integration of cluster expertise evidence into the voting technique strengthens votes from documents which are found in larger clusters in the profile of candidate c, because the largest clusters are assumed to be the candidate's strongest expertise area. Note that if a document d does not occur in the top K clusters, then $Qscore_{Cluster}(d,C,Q) = 0$, i.e. its vote is not strengthened further. Moreover, if no clustering has been applied for the candidate (i.e. they have less than θ documents in their profile), then $Qscore_{Cluster}(d, C, Q) = 0$.

In the remainder of this paper, we experiment with the proposed techniques for identifying quality evidence in the candidate profiles. In particular, we define the experimental setup of our experiments in the next section. Results and conclusions follow in Sections 5 and 6, respectively.

4 Experimental Setup

Our experiments are carried out in the setting of the Expert Search task of the TREC Enterprise tracks, namely 2005, 2006 and 2007. For TREC 2005 and 2006, the document collection used was a crawl of the World Wide Web Consortium (W3C), a virtual Internet organisation responsible for HTML, XML standards and the like. For TREC 2007, a different and more realistic corpus, known as CERC, was introduced, which is a crawl of the website of Commonwealth Scientific and Industrial Research Organisation (CSIRO). CSIRO is the national government body for scientific research in Australia. In terms of measuring retrieval performance, we use the Mean Average Precision (MAP) measure for all tasks. Moreover, for TREC 2005 and TREC 2006 for which there are generally more than 10 relevant candidates per-topic, we measure for Precision at 10 (P@10). In the CERC collection, in which there are typically less than 10 relevant candidates per topic, we measure Mean Reciprocal Rank (MRR).

Table 1. Statistics of the TREC W3C and CERC Enterprise research test collections

Statistic	W3C	CERC
# of Documents	331,037	370,715
# of Topics	99	50
# of Candidates	1,092	3,490
Average Profile Size (# of Documents)	913.2	217.7
Largest Profile Size (# of Documents)	88,080	62,285

The TREC W3C and CERC collections are indexed using Terrier [21], removing standard stopwords and applying the first two steps of Porters stemming algorithm. Moreover, we add onto each document, the anchor text of the incoming hyperlinks from other documents in the corpus. For the calculation of the clustering *Qscore*, we apply $K = 10$ and $\theta = 30$, because for prolific persons, 10 areas of expertise would seem intuitive for most people. The setting of all other *Qscore* parameters is described in the following section. To identify the profile of documents to represent each candidate, we search for each candidate's full name in the corpus. For the CERC test collection, where no initial list of candidates is provided, candidates are initially identified by the presence of an email address in the form `firstname.lastname@csiro.au` in the corpus. Statistics of the W3C and CERC test collections are given in Table 1.

5 Experimental Results

In our experiments, we are not focused on the particular integration of the *Qscore* with expCombMNZ. Instead, we wish to see if any benefit is possible in applying that evidence. For this reason, we firstly train to maximise MAP on the set of topics being tested. Secondly, we use a more realistic setting, where for TREC 2006 we train using the TREC 2005 topics, and for TREC 2007, we train using the TREC 2005 and 2006 topics combined (even though it is not the same corpus). Appendix 1 details the obtained parameters for all settings. Table 2 presents the results of our experiments. On the first row, the median MAP is shown. Our baseline is the retrieval performance achieved by applying DLH13 with expCombMNZ. It can be seen that this baseline is markedly above the median performance of all participating groups (except MRR for TREC 2007). In particular, for TREC 2005 and TREC 2006, this baseline would have been ranked in the top three automatic title-only runs, and in the top four for TREC 2007 automatic title-only runs. The remainder of the table presents the retrieval performance of each proposed technique for identifying quality expertise. For the columns denoted '/test', the parameters have been trained on the test set, while '/train' denotes when the parameters were trained using a separate test set of topics, as detailed above.

Table 2. Results for TREC 2005, 2006 and 2007 expert search tasks, when trained on the test set. Significant increases over the baseline are denoted $>$ ($p < 0.05$) and $\gg$ ($p < 0.01$) respectively. '/test' and '/train' denote whether the parameters for the quality evidence techniques were trained using the test set or a separate training set.

TREC Year	2005/test		2006/test		2006/train		2007/test		2007/train	
	MAP	P@10	MAP	P@10	MAP	P@10	MAP	MRR	MAP	MRR
Median	0.1402	-	0.3412	-	0.3412	-	0.2468	0.5011	0.2468	0.5011
Baseline	0.2040	0.3100	0.5502	0.6837	0.5502	0.6837	0.3519	0.4730	0.3519	0.4730
+ Prox	0.2155	0.3200	**0.5621**$>$	0.6878	0.5427	0.6551	**0.4319**$\gg$	**0.5742**$\gg$	**0.3688**	0.4891
+ URL	0.2232$\gg$	0.3300	0.5565	**0.7020**	**0.5657**	**0.7000**	0.3779$>$	0.5309$>$	0.3683	**0.5015**
+ Inlinks	0.2212$\gg$	**0.3540**$\gg$	0.5600	0.6857	0.5522	0.6755	0.3654	0.4847	0.3474	0.4778
+ Clusters	**0.2324**$>$	0.3420	0.5517	0.6816	0.4830	0.6020	0.3915$>$	0.5400	0.3584	0.4726
+ Homepage	0.2040	0.3100	0.5530	0.6837	0.5501	0.6837	0.3885	0.5334	0.3463	0.4569

On the optimal setting ('/test'), the Proximity quality evidence performs well, particularly on the CERC collection. URL and Inlinks evidence also appear to be reliable at discriminating between high and low quality expertise evidence in the candidate profiles. For the homepage, the results are mixed: it improves retrieval performance on the TREC 2007 collection (suggesting that many of the CSIRO experts do have homepages); for TREC 2005 and 2006, there are only minor differences in performance. By further examination of the W3C corpus, there are only 58 candidates from the 1092 in the collection are staff members of the W3C, therefore this evidence does not apply well in this case. Lastly, the clustering provides significant improvements for MAP on the TREC 2005 and TREC 2007 topic sets, while for TREC 2006 there is little change. For the plausible training ('/train'), Table 2 shows the performance is slightly less than the optimal training, the results are still similar. In particular, proximity and URL are the best indicators, followed by clustering. Again, the homepages and inlinks did not bring much difference in retrieval performance. The slightly lower performance of the clustering on TREC 2007 is explained by the fact that the combined TREC 2005 + 2006 topics are not a good training set for this quality evidence.

Overall, as mentioned above, our main aim was not to propose how to combine the quality evidences with the proposed voting technique. However, given that the retrieval performance could be improved in the future by better combinations and further training of parameters, some of the proposed quality evidences, such as proximity and clustering, seem very promising. In particular, the best setting for proximity on the TREC 2007 topics would have been ranked 2nd out of the submitted automatic title-only runs that year.

6 Conclusions

In this paper we have proposed five techniques to predict the quality of documents within a candidate's profile in the expert search task. We have thoroughly tested these techniques using two test collections and three TREC topic sets. The experiments show that among them, the novel clustering and proximity techniques seem very promising. However, in contrast to Web search settings, various Web IR features such as URL and Inlinks did not exhibit large increases in performance.

It is of interest that in the field of Web IR, it is natural to learn document features based on their occurrence in a set of relevance assessments. However, in the expert search task only the final outcome of the expert search system is evaluated. None of the three important performance-affecting factors described in this paper (see abstract, Sections 1 & 3) can be directly evaluated, making it particular difficult to have a complete overview of the performance of the system. While the initial steps taken in [22] work towards a more complete evaluation, perhaps in the future, the evaluation methodology can evolve to provide enough details such that a thorough failure analysis can be conducted and conclusions can be drawn about all components of an expert search system.

References

1. Craswell, N., Hawking, D., Vercoustre, A.M., Wilkins, P.: Panoptic expert: Searching for experts not just for documents. In: AusWeb Poster Proceedings, Queensland, Australia (2001)
2. Liu, X., Croft, W.B., Koll, M.: Finding experts in community-based question-answering services. In: Proceedings of CIKM 2005, pp. 315–316 (2005)
3. Macdonald, C., Ounis, I.: Voting for candidates: Adapting data fusion techniques for an expert search task. In: Proceedings of CIKM 2006 (2006)

4. Craswell, N., de Vries, A.P., Soboroff, I.: Overview of the TREC-2005 Enterprise Track. In: Proceedings of TREC-2005, pp. 199–204 (2006)
5. Balog, K., Azzopardi, L., de Rijke, M.: Formal models for expert finding in enterprise corpora. In: Proceedings of SIGIR 2006, pp. 43–50 (2006)
6. Petkova, D., Croft, W.B.: Hierarchical language models for expert finding in enterprise corpora. In: Proceedings of ICTAI 2006, pp. 599–608 (2006)
7. Macdonald, C., Ounis, I.: Voting Techniques for Expert Search. J. Knowledge and Information Systems (in press, 2008), DOI:10.1007/s10115-007-0105-3
8. Macdonald, C., Ounis, I.: Using Relevance Feedback in Expert Search. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 431–443. Springer, Heidelberg (2007)
9. Macdonald, C., Ounis, I.: Expertise drift and query expansion in expert search. In: Proceedings of CIKM 2007 (in press, 2007)
10. Lioma, C., Macdonald, C., Plachouras, V., Peng, J., He, B., Ounis, I.: University of Glasgow at TREC 2006: Experiments in Terabyte and Enterprise Tracks with Terrier. In: Proceedings of TREC 2006 (2007)
11. Balog, K., de Rijke, M.: Finding experts and their details in e-mail corpora. In: Proceedings of WWW 2006 (2006)
12. Macdonald, C., Ounis, I.: Voting for Experts: The Voting Model for Expert Search. Special issue of the Computer Journal on Expertise Profiling (to appear, 2008)
13. Kraaij, W., Westerveld, T., Hiemstra, D.: The importance of prior probabilities for entry page search. In: Proceedings of SIGIR 2002, pp. 27–34 (2002)
14. Craswell, N., Robertson, S., Zaragoza, H., Taylor, M.: Relevance weighting for query independent evidence. In: Proceedings of SIGIR 2005, pp. 416–423 (2005)
15. Fang, H., Zhai, C.: Probabilistic models for Expert Finding. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 418–430. Springer, Heidelberg (2007)
16. Cao, Y., Li, H., Liu, J., Bao, S.: Research on expert search at enterprise track of TREC 2005. In: Proceedings of TREC-2005 (2006)
17. Amati, G.: Frequentist and Bayesian approach to information retrieval. In: Lalmas, M., MacFarlane, A., Rüger, S.M., Tombros, A., Tsikrika, T., Yavlinsky, A. (eds.) ECIR 2006. LNCS, vol. 3936, pp. 13–24. Springer, Heidelberg (2006)
18. Hawking, D., Craswell, N., Crimmins, F., Upstill, T.: How valuable is external link evidence when searching enterprise webs? In: Proceedings of ADC 2004, pp. 77–84 (2004)
19. Fagin, R., Kumar, R., McCurley, K.S., Novak, J., Sivakumar, D., Tomlin, J.A., Williamson, D.P.: Searching the workplace web. In: Proceedings of WWW 2003, pp. 366–375 (2003)
20. Peng, J., Macdonald, C., He, B., Ounis, I.: Combination of Document Priors in Web Information Retrieval. In: Proceedings of RIAO 2007 (2007)
21. Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., Lioma, C.: Terrier: A high performance and scalable information retrieval platform. In: Proceedings of the OSIR Workshop 2006, pp. 18–25 (2006)
22. Macdonald, C., Ounis, I.: Expert Search Evaluation by Supporting Documents. In: Macdonald, C., et al. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 555–563. Springer, Heidelberg (2008)

Appendix 1: Parameters

Table 3. Trained parameters, headings are as in Table 2: Proximity trained using manual scanning; other techniques trained using simulated annealing process for MAP

TREC Year	2005/test 2006/train	2006/test	2007/train	2007/test
Prox	$\omega = 1$ $ws = 20$ $c_p = 0.1$	$\omega = 1$ $ws = 10$ $c_p = 0.01$	$\omega = 1$ $ws = 20$ $c_p = 0.0001$	$\omega = 0.5$ $ws = 200$ $c_p = 1$
URL	$\omega = 14.12$ $\kappa = 99.78$	$\omega = 12.22$ $\kappa = 70.03$	$\omega = 8.27$ $\kappa = 9.82$	$\omega = 18.41$ $\kappa = 85.44$
Inlinks	$\omega = 5.88$ $\kappa = 0.39$	$\omega = 3.04$ $\kappa = 3.31$	$\omega = 4.55$ $\kappa = 0.59$	$\omega = 5.74$ $\kappa = 2.13$
Clusters	$\omega = 6.50$	$\omega = 0.80$	$\omega = 3.87$	$\omega = 1.74$
Homepage	$\omega = 0.004$	$\omega = 0.067$	$\omega = 0.03$	$\omega = 0.25$