Voting for Candidates: Adapting Data Fusion Techniques for an Expert Search Task

Craig Macdonald, Iadh Ounis
Department of Computing Science
University of Glasgow Scotland, UK
{craigm,ounis}@dcs.gla.ac.uk

ABSTRACT

In an expert search task, the users' need is to identify people who have relevant expertise to a topic of interest. An expert search system predicts and ranks the expertise of a set of candidate persons with respect to the users' query. In this paper, we propose a novel approach for predicting and ranking candidate expertise with respect to a query. We see the problem of ranking experts as a voting problem, which we model by adapting eleven data fusion techniques.

We investigate the effectiveness of the voting approach and the associated data fusion techniques across a range of document weighting models, in the context of the TREC 2005 Enterprise track. The evaluation results show that the voting paradigm is very effective, without using any collection specific heuristics. Moreover, we show that improving the quality of the underlying document representation can significantly improve the retrieval performance of the data fusion techniques on an expert search task. In particular, we demonstrate that applying field-based weighting models improves the ranking of candidates. Finally, we demonstrate that the relative performance of the adapted data fusion techniques for the proposed approach is stable regardless of the used weighting models.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*search process*; H.3.4 [**Information Storage and Retrieval**]: Systems and software—*User profiles and alert services*

General Terms

Experimentation, Measurement

Keywords

Voting, Expert Finding, Expertise Modelling, Expert Search Information Retrieval, Ranking, Data fusion

1. INTRODUCTION

With the advent of the vast pools of information and documents in large enterprise organisations, collaborative users regularly have the need to find not only documents, but also people with whom they share common interests, or who have specific knowledge in a required area.

Mertzum & Pejtersen [11] found that engineers in product-development organisations often intertwine looking for informative documents with looking for informed people. People are a critical source of information because they can explain and provide arguments about why specific decisions were made.

Yimam-Seid & Kobsa [36] identified five scenarios when people may seek an expert as a source of information to complement other sources:

1. *Access to non-documented information* - e.g. in an organisation where not all relevant information is documented.
2. *Specification need* - the user is unable to formulate a plan to solve a problem, and resorts to seeking experts to assist them in formulating the plan.
3. *Leveraging on another's expertise (group efficiency)* - e.g. finding a piece of information that a relevant expert would know/find with less effort than the seeker.
4. *Interpretation need* - e.g. deriving the implications of, or understanding, a piece of information.
5. *Socialisation need* - the user may prefer that the human dimension be involved, as opposed to interacting with documents and computers.

An *expert search* system is an Information Retrieval (IR) system that can aid users with their "expertise need" in the above scenarios. In contrast with classical document retrieval where documents are retrieved, an expert search system supports users in identifying informed people: The user formulates a query to represent their topic of interest to the system; the system then ranks *candidate* persons with respect to their predicted expertise about the query, using available documentary evidence.

The retrieval performance of an expert search system is an important issue. An expert search system should aim to rank candidate experts while maximising the traditional evaluation measures in IR: *precision*, the accuracy of suggested candidates expertise; and *recall*, the number of candidates with relevant expertise retrieved.

The creation of the expert search task in the recent TREC 2005 Enterprise track [6] has increased interest in this area. An active research problem is how best to generate a ranking of candidates from a collection of documents. Systems typically use a *profile* of evidence for each candidate that indicates their expertise. These profiles can be generated manually by the candidate, or automatically by the system using documentary evidence.

In this paper, we consider a *ranking of documents* with respect to the expert search query. We see each document retrieved as an implicit vote for the candidate whose profile contains that document. We propose several ways to aggregate document votes into a ranking of candidates, based on appropriate data fusion techniques.

The techniques are evaluated across a range of probabilistic weighting models, using the TREC W3C test collection and the TREC 2005 Enterprise expert search task. The obtained results show that applying the voting approach to expert search is very effective compared with the TREC 2005 results, while making no use of collection-specific heuristics.

In order to improve the underlying ranking of documents, we further refine the representation of documents used by the retrieval system, to take the structure of documents into account. We use content, title and anchor text of incoming hyperlinks as separate fields during retrieval. Each document is represented by these fields. We demonstrate that applying a weighting model that uses these fields significantly improves the performance of the proposed voting approach.

The structure of this paper is as follows: We give an overview of expert search systems and previous related work in Section 2. In Section 3, we describe how expert search can be modelled as a voting problem. We propose the use of data fusion techniques to convert document rankings into candidate rankings, and present the data fusion techniques adapted. We describe our experimental setup in Section 4, and evaluate the voting approach across a selection of document weighting models in Section 5. In Section 6, we use field-based weighting models in an expert search context, and show how this significantly improves the performance of the expert search data fusion techniques adapted for the approach. In Section 7, we demonstrate that the performance of the adapted data fusion techniques is stable across various weighting models and settings. Finally, we provide concluding remarks and suggestions for future work in Section 8.

2. EXPERT SEARCH SYSTEMS

Expert search systems make use of textual evidence of expertise to rank candidates. Predominantly, these systems work by generating a *profile* of textual evidence for each candidate. The profiles represent the system's knowledge of the expertise of each candidate, and they are ranked in response to a user query [7, 9, 15, 34].

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 [9]; or alternatively, the expert search system can implicitly and automatically generate each profile from a corpus of documents. There are several strategies for associating documents to candidates, to generate a profile of their expertise:

- Documents containing the candidate's name: exact or partial match [7]
- Emails sent or received by the candidate [3, 5, 8]
- The candidate's homepage on the Internet or intranet and their C.V. [20]
- Documents written by the candidate [20]
- Team, group or department-level evidence [21]
- Web pages visited by the candidate [35]

Having defined profiles of expertise for each candidate, an expert search system needs to accurately rank the candidate profiles in response to a user query. There is some previous work on ranking candidate profiles for expert search. Craswell et al. proposed concatenating the terms of all documents in each profile into "virtual documents", and ranking these using a traditional IR system [7].

Liu et al. [15] addressed the expert search problem in the context of a community-based question-answering service. They applied three different language models, and experimented with varying the size of the candidate profiles. They concluded that retrieval performance can be enhanced by including more evidence in the profiles.

Social network analysis also features in some related work to expert search. Graph-based techniques are used to infer connections between candidates, and are particularly useful on corpora of email communications [8, 20, 34]. Two approaches make use of the HITS algorithm [13] to calculate "repute" and "resourcefulness" scores for each candidate [5, 35]. In [21], McLean et al. use a graph structure to propagate expertise evidence between members of a project team. Finally, various expert search approaches were proposed by participants in the TREC 2005 Enterprise track [6], and techniques such as document structure and clustering were applied.

In this work, we consider a different and novel approach to ranking expertise. In particular, we consider that expert search is a voting process. Using the ranked list of retrieved documents for the expert search query, we propose that the ranking of candidates can be modelled as a voting process using the retrieved document ranking and the set of documents in each candidate profile. The problem is how to aggregate the votes for each candidate so as to produce the final ranking of experts. In Section 3, we show how existing data fusion techniques can be appropriately adapted to combining votes for candidates.

3. EXPERT SEARCH AS A VOTING PROBLEM

Data fusion techniques - also known as metasearch techniques - are used to combine separate rankings of documents into a single ranking, with the aim of improving over the performance of any constituent ranking. Each time a document is retrieved by a ranking, an implicit vote has been made for that document to be included higher in the combined ranking. Fox & Shaw [10] defined several data fusion techniques

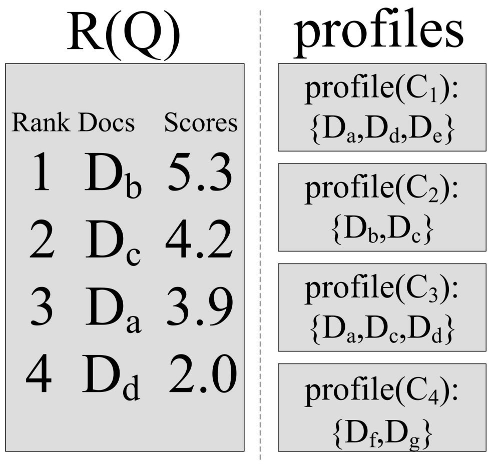

Figure 1: A simple example from expert search: the ranking $R(Q)$ of documents (each with a rank and a score), must be transformed into a ranking of candidates using the documentary evidence in the profile of each candidate ($profile(C)$).

(CombSUM, CombMNZ, etc.), and these have been the object of much research since. (For examples, see [14, 23, 33]).

Two main classes of data fusion techniques exist: those that combine rankings using the ranks of the retrieved documents, and those that combine rankings using the scores of the retrieved documents.

As introduced in Section 2, we see expert search as a voting problem: In this work, the profile of each candidate is a set of documents associated to them to represent their expertise. We then consider a *ranking of documents* by an IR system with respect to the query. Each document retrieved by the IR system that is associated with the profile of a candidate, can be seen as an implicit vote for that candidate to have relevant expertise to the query. The ranking of the candidate profiles can then be determined from the votes. In this work, we choose to adapt well-known data fusion techniques in IR, to aggregate the votes for candidates by the retrieved documents.

Let $R(Q)$ be the set of documents retrieved for query Q, and the set of documents belonging to the profile of candidate C be denoted $profile(C)$. In expert search, we need to find a ranking of candidates, given $R(Q)$. Consider the simple example in Figure 1 above. The ranking of documents with respect to the query has retrieved documents $\{D_b, D_c, D_a, D_d\}$. Using the candidate profiles, candidate C_1 has then accumulated 2 votes, C_2 2 votes, C_3 3 votes and C_4 no votes. Hence, if all votes are counted as equal, and each document in a candidate's profile is equally weighted, a possible ranking of candidates to this query could be $\{C_3, C_1, C_2\}$. By using appropriate vote aggregation techniques, we can have different rankings of candidates. In the remainder of this paper, we introduce eleven adapted data fusion techniques, and evaluate them to establish how well they model the proposed voting paradigm.

We determine the score of the candidate with respect to the query, $score\_cand(C, Q)$, as the aggregation of votes of all documents d that are retrieved, but which also belong to the profile of the candidate (i.e. $d \in R(Q) \cap profile(C)$). We consider three forms of evidence when aggregating the votes to each candidate: (i) the number of retrieved documents voting for each candidate; (ii) the scores of the retrieved documents voting for each candidate; and (iii) the ranks of the retrieved documents voting for each candidate. We adapt data fusion techniques to aggregate the votes from the single ranking of documents into a ranking of candidates, using the appropriate forms of evidence. We examine and evaluate eleven data fusion techniques as they can be adapted to expert search.

Notice, however, in the normal application of data fusion techniques, several ranking of documents are combined into a single ranking of documents. In contrast, our novel approach aggregates votes from a single ranking of documents into a single ranking of candidates, using the document-to-candidate associations of the candidate profiles.

3.1 Adapting Data Fusion Techniques

We now show how some established data fusion techniques can be adapted for expert search. Firstly, we adapt the Reciprocal Rank (RR) data fusion technique [39] for expert search. In this data fusion technique, the rank of a document in the combined ranking is determined by the sum of the reciprocal rank received by the document in each of the individual rankings. Adapting the Reciprocal Rank technique to our approach, we define the score of a candidate's expertise as:

$$score\_cand_{RR}(C, Q) = \sum_{d \,\in\, R(Q) \cap \, profile(C)} \frac{1}{rank_d} \qquad (1)$$

where $rank_d$ is the rank of document d in the document ranking R(Q). RR is an example of a rank aggregation data fusion technique.

In CombSUM [10] - a score aggregation technique - the score of a document is the sum of the normalised scores received by the document in each individual ranking. CombSUM can also be used in expert search. In this case, the score of a candidate's expertise is:

$$score\_cand_{CombSUM}(C, Q) = \sum_{d \,\in\, R(Q) \cap \, profile(C)} score_d \qquad (2)$$

where $score_d$ is the score of the document d in the document ranking $R(Q)$. Similarly, CombMNZ [10] can be adapted for expert search:

$$score\_cand_{CombMNZ}(C, Q) = \|R(Q) \cap \ profile(C)\| \cdot \sum_{d \,\in\, R(Q) \cap \, profile(C)} score_d \qquad (3)$$

where $\|R(Q) \cap profile(C)\|$ is the number of documents from the profile of candidate C that are in the ranking $R(Q)$.

Normally, in the CombSUM and CombMNZ data fusion techniques, it is necessary to normalise the scores of documents across all the rankings [23]. However, in Equations (2) and (3), no score normalisation is necessary: Indeed, in our case, as stressed above, only one ranking of documents is involved, and hence the scores are all comparable.

Table 1 summarises all the data fusion techniques that we adapt and evaluate in this work. In addition to the three techniques described above, we also adapt and evaluate: a technique that we call Votes, which simply counts the number of retrieved documents of each candidate profile; BordaFuse [4] - a rank aggregation technique; and several

Name	Relevance score of candidate is:
Votes	$\|D(C,Q)\|$
RR	sum of inverse of ranks of docs in $D(C,Q)$
BordaFuse	sum of ($\|R(Q)\|$ - ranks of docs in $D(C,Q)$)
CombMED	median of scores of docs in $D(C,Q)$
CombMIN	minimum of scores of docs in $D(C,Q)$
CombMAX	maximum of scores of docs in $D(C,Q)$
CombSUM	sum of scores of docs in $D(C,Q)$
CombANZ	CombSUM $\div \|D(C,Q)\|$
CombMNZ	$\|D(C,Q)\| \times$ CombSUM
expCombSUM	sum of exp of scores of docs in $D(C,Q)$
expCombANZ	expCombSUM $\div \|D(C,Q)\|$
expCombMNZ	$\|D(C,Q)\| \times$ expCombSUM

Table 1: Summary of expert search data fusion techniques used in this paper. $D(C,Q)$ is the set of documents $R(Q) \cap profile(C)$. $\|\cdot\|$ is the size of the described set.

other score aggregation techniques first defined by Fox & Shaw in [10]. The final three adapted data fusion techniques listed in the table, namely expCombSUM, expCombANZ and expCombMNZ, are slight variants of CombSUM, CombANZ and CombMNZ respectively. In these variants, the score of each document is transformed by applying the exponential function (e^{score}), as suggested by Ogilvie & Callan in [25]. Applying the exponential function boosts the scores of highly ranked documents.

Other data fusion techniques could also have been considered in this work, including one based on Condorcet voting-theory [24], a technique that models score distributions [19], and a logical regression model [32]. However, in this work, due to the more complex nature of these techniques, we focus the evaluation of our proposed voting approach on the techniques in Table 1.

4. EXPERIMENTAL SETTING

In the following, we aim to demonstrate that voting is an effective approach for expert search and that the data fusion techniques adapted are suitable to implement the proposed approach. We use three different statistical document weighting models to assess the extent to which the performance of the adapted data fusion techniques is affected by the choice of weighting model.

To evaluate our approach, we use the Expert Search task of the TREC 2005 Enterprise track. The TREC 2005 Enterprise test collection consists of 331,037 documents collected from the World Wide Web Consortium (W3C) website in 2005 [6]. For research purposes, the W3C is a useful example of an enterprise organisation, as it operates almost entirely over the Internet. Moreover, its documents are freely available online. This allows research on an enterprise-level corpus, without the intellectual property issues normally associated with obtaining such a corpus. The corpus is also wide-ranging, containing the main W3C Web presence, personal homepages, standards documents, email discussion list archives, a wiki, and a source code repository.

The W3C test collection includes a list of 1,092 candidate experts. We use the 50 topics from the TREC 2005 Expert Search task. The retrieval performance is evaluated using Mean Average Precision (MAP) - to assess the overall quality of the ranking - and Precision @ 10 (P@10), to assess the accuracy of the top-ranked candidates retrieved by the system.

We index the W3C collection using Terrier [26, 27]. During indexing, each document is represented by its textual content and the anchor text of its incoming hyperlinks. Stopwords are removed, and as we would like to favour high precision, we use a weak stemming algorithm, which only applies the first two steps of Porter's stemming algorithm.

To generate the profile for each candidate, we generate queries to identify documents in which variations of each candidate's name or email address occur in the entire collection. The set of documents identified for each candidate C form their $profile(C)$.

We test our proposed voting approach using the eleven adapted data fusion techniques listed in Table 1 with three statistically different document weighting models. The first of these weighting models is the well-established BM25 [31], where the relevance score of a document d for a query Q is given by:

$$score(d,Q) = \sum_{t \in Q} w^{(1)} \frac{(k_1+1)tfn}{k+1+tfn} \frac{(k_3+1)qtf}{k_3+qtf} \quad (4)$$

where qtf is the frequency of the query term t in the query; k_1 and k_3 are parameters, for which the default setting is $k_1 = 1.2$ and $k_3 = 1000$ [30]; $w^{(1)}$ is the *idf* factor, which is given by:

$$w^{(1)} = \log_2 \frac{N - N_t + 0.5}{N_t + 0.5}$$

N is the number of documents in the whole collection. N_t is the document frequency of term t.

The normalised term frequency tfn is given by:

$$tfn = \frac{tf}{(1+b) + b \cdot \frac{l}{avg\_l}}, (0 \le b \le 1) \quad (5)$$

where tf is the term frequency of the term t in document d. b is the term frequency normalisation hyper-parameter, for which the default setting is $b = 0.75$ [30]. l is the document length and avg_l is the average document length in the collection.

The remaining two weighting models tested are from the Divergence from Randomness (DFR) framework [1]. The first of these, PL2, is robust and performs particularly well for tasks requiring high early-precision [28]. For the PL2 model, the relevance score of a document d for a query Q is given by:

$$\begin{aligned} score(d,Q) \quad = \quad & \sum_{t \in Q} qtw \cdot \frac{1}{tfn+1} \big(tfn \cdot \log_2 \frac{tfn}{\lambda} \quad (6) \\ & +(\lambda - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn) \big) \end{aligned}$$

where λ is the mean and variance of a Poisson distribution. It is given by $\lambda = F/N$. F is the frequency of the query term in the collection and N is the number of documents in the whole collection. The query term weight qtw is given by qtf/qtf_{max}. qtf is the query term frequency. qtf_{max} is the maximum query term frequency among the query terms.

The normalised term frequency tfn is given by the so-called Normalisation 2 from the DFR framework:

$$tfn = tf \cdot \log_2(1 + c \cdot \frac{avg\_l}{l}), (c > 0) \quad (7)$$

where tf is the term frequency of the term t in document d and l is the length of the document. avg_l is the average document length in the whole collection. c is the hyper-parameter that controls the normalisation applied to the term frequency with respect to the document length. The default value is $c = 1.0$ [1].

The DLH13 document weighting model is a generalisation of the parameter-free hypergeometric DFR model in a binomial case [2, 16]. The hypergeometric model assumes that the document is a sample, and the population is from the collection. For the DLH13 document weighting model, the relevance score of a document d for a query Q is given by:

$$score(d, Q) = \sum_{t \in Q} \frac{qtw}{tf + 0.5} \cdot \Big(\log_2 (\frac{tf \cdot avg\_l}{l} \cdot \frac{N}{F}) + 0.5 \log_2 \left(2\pi tf (1 - \frac{tf}{l}) \right) \Big) \quad (8)$$

Note that the DLH13 weighting model has no term frequency normalisation component, as this is assumed to be inherent to the model. Hence, DLH13 has no parameters that require tuning. Indeed, all variables are automatically computed from the collection and query statistics.

The BM25 and PL2 document weighting models include parameters, and require tuning using relevance assessments. In our experiments, we assess the performance of the data fusion techniques, both using the default parameter settings for each weighting model, and when, for each fusion technique, the parameters of the weighting model have been empirically set to maximise MAP. This allows assessment of the maximum potential in the proposed approach. Note again that the DLH13 model has no parameters that need to be tuned, and therefore is deployed directly in the expert search task.

5. EXPERIMENTAL RESULTS

Table 2 shows the retrieval performance of the proposed voting approach, using the eleven adapted data fusion techniques, across three weighting models, namely BM25, PL2, and DLH13[1]. In these results, the default setting is used for BM25 and PL2 (see Section 4). Table 3 shows the retrieval performance when the term frequency hyper-parameters of the weighting models are empirically set, to enable the assessment of the maximum potential of each combination of the adapted data fusion techniques and the weighting models. In this table, the effectiveness of the data fusion techniques has improved. The relative performance of each data fusion technique remains roughly consistent across all weighting models in both settings.

Examining Tables 2 and 3 in detail, we make the following observations. Firstly, the Votes technique, which simply counts the number of document votes for each candidate, shows good performance (MAP ≥ 0.1650). The rank-based techniques, RR and BordaFuse, both perform well across the three weighting models. Note the good performance of RR on P@10 in both used settings. RR highly scores candidate profiles that have documents occurring at the very top of the ranking, suggesting that the highly ranked documents contribute more to the expertise of a candidate, and should be considered as stronger votes. In contrast, the BordaFuse technique assigns linearly scaled votes across the document ranking to candidates, without emphasising the strength of votes by top ranked documents, which slightly hinders the retrieval performance compared to RR.

On the other hand, the score-based data fusion techniques have varying effectiveness, depending on the technique used. CombSUM and CombMNZ and their exponential variants are the strongest of the score-based techniques. These both take into account the strength of the document votes, i.e. the magnitude of the score for each retrieved document of the candidate's profile. Moreover, CombMNZ adds a second component, the number of votes for each candidate, explaining its slight overall performance edge over CombSUM. The good effectiveness of these techniques mirrors previous studies of their use in classical data fusion [22, 23].

The high performance of the exponential variants of CombSUM and CombMNZ, expCombSUM and expCombMNZ, can be explained in that the exponential function increases the scores of the highly-scored documents more than the low-scored documents, increasing the strength of their votes. Hence a candidate with many weak votes will be lower ranked, while a candidate with fewer stronger votes will be higher ranked. In terms of MAP, expCombSUM and expCombMNZ outperform all other techniques across all weighting models and settings. Moreover, expCombMNZ always outperforms expCombSUM on MAP when the parameters have been empirically set. However, expCombSUM gives better P@10 (e.g. 0.3720 vs. 0.3520 and 0.3260 vs. 0.3220 in Table 3).

CombMAX almost performs as well as CombSUM and CombMNZ, even though it does not take into account the number of votes for a candidate profile. This shows that the most highly ranked document for each candidate is still a good indicator of its expertise.

The CombANZ, CombMIN, and expCombANZ techniques do not perform well, because they focus too much on the low scoring documents of each profile, which, intuitively, are not good indicators of expertise. Interestingly, taking the median of the scored documents in a profile (CombMED) outperforms taking the average (CombANZ). This finding is inconsistent with previous experiments using these techniques for classical data fusion [10]. A possible interpretation is that the denominator component of CombANZ impairs the evidence in the distribution of the candidate scores.

Tables 2 and 3 also present the statistical significance of results, when compared to the median run of all participants of TREC 2005 (MAP 0.1402), using the Wilcoxon Matched-Pairs Signed-Rank test[2]. Most of the adapted data fusion techniques lead to a clear increase in performance over this median run. In particular, applying expCombSUM or expCombMNZ always results in a statistically significant increase in MAP from the baseline.

From the results, we can surmise that good indicators of expertise of a candidate seem to be the number of documents in the candidate's profile retrieved for a query (number of votes), and the relative magnitude of the retrieval scores in the candidate's profile (strength of votes). The strongly performing CombMNZ and expCombMNZ techniques exemplify both these indicators.

Overall, we have shown that the proposed voting approach using adapted data fusion techniques can be effectively applied to expert search, Indeed, the best performing runs in Table 2 would rank as high as the top three participants in TREC 2005 Enterprise track runs, without using any

[1] Equivalent experiments performed using full Porter stemming showed little differences to the results in Table 2.

[2] We do not have access to the P@10 of the median run.

	BM25			PL2			DLH13		
Fusion	MAP	ΔMAP	P@10	MAP	ΔMAP	P@10	MAP	ΔMAP	P@10
Votes	0.1691	(+21%)	0.3180	0.1661	(+18%)	0.3100	0.1650	(+17%)	0.3080
RR	$0.1940^{>}$	(+39%)	**0.3560**	0.1758	(+25%)	0.3120	$0.1849^{>}$	(+32%)	**0.3500**
BordaFuse	0.1774	(+27%)	0.3360	0.1691	(+21%)	0.3160	$0.1738^{>}$	(+24%)	0.3280
CombANZ	$0.0316^{\ll}$	(-77%)	0.0380	$0.0344^{\ll}$	(-75%)	0.0420	$0.0313^{\ll}$	(-78%)	0.0240
CombMED	$0.1055^{<}$	(-25%)	0.1900	$0.1022^{\ll}$	(-27%)	0.1720	$0.1089^{\ll}$	(-22%)	0.1880
CombMIN	$0.0654^{\ll}$	(-53%)	0.1380	$0.0637^{\ll}$	(-55%)	0.1380	$0.0728^{\ll}$	(-48%)	0.1500
CombMAX	0.1756	(+25%)	0.3120	0.1630	(+16%)	0.2960	0.1632	(+16%)	0.3080
CombSUM	0.1769	(+26%)	0.3280	0.1736	(+26%)	**0.3240**	$0.1743^{>}$	(+24%)	0.3180
CombMNZ	0.1747	(+25%)	0.3280	0.1733	(+25%)	0.3220	0.1715	(+22%)	0.3220
expCombANZ	$0.0333^{\ll}$	(-76%)	0.0340	$0.0300^{\ll}$	(-79%)	0.0380	$0.0333^{\ll}$	(-76%)	0.0420
expCombSUM	$\mathbf{0.1980}^{>}$	(+41%)	0.3420	$0.1757^{>}$	(+25%)	0.3120	$0.1792^{>}$	(+28%)	0.3380
expCombMNZ	$0.1970^{>}$	(+40%)	0.3420	$\mathbf{0.1816}^{>}$	(+30%)	0.3220	$\mathbf{0.1873}^{\gg}$	(+34%)	0.3440

Table 2: Performance of the 11 data fusion techniques for expert search. We use the default settings for the weighting models (see Section 4). Relative MAP differences from the median run of the participating groups of the TREC 2005 Enterprise track are shown (MAP 0.1402): statistically significant improvements at $p <= 0.05$ are denoted $>$; significant improvements at $p <= 0.01$ are denoted $\gg$. Similarly, statistically significant degradations in MAP are denoted $<$ and $\ll$, respectively. The best data fusion technique for each weighting model is highlighted in bold.

	BM25			PL2		
Fusion	MAP	ΔMAP	P@10	MAP	ΔMAP	P@10
Votes	0.1767	(+26%)	0.3160	0.1674	(+19%)	0.3100
RR	$0.2002^{\gg}$	(+43%)	0.3640	$0.1788^{>}$	(+28%)	**0.3440**
BordaFuse	$0.1877^{>}$	(+34%)	0.3420	0.1692	(+21%)	0.3140
CombANZ	$0.0335^{\ll}$	(-75%)	0.0340	$0.0345^{\ll}$	(-75%)	0.0420
CombMED	$0.1085^{<}$	(-23%)	0.1880	$0.1029^{\ll}$	(-27%)	0.1780
CombMIN	$0.0673^{\ll}$	(-52%)	0.1400	$0.0701^{\ll}$	(-50%)	0.1460
CombMAX	0.1790	(+28%)	0.3200	0.1632	(+16%)	0.2980
CombSUM	$0.1886^{>}$	(+35%	0.3300	0.1740	(+24%)	0.3180
CombMNZ	$0.1846^{>}$	(+32%)	0.3260	0.1741	(+24%)	0.3220
expCombANZ	$0.0347^{\ll}$	(-75%)	0.0460	$0.0333^{\ll}$	(-76%)	0.0420
expCombSUM	$0.1987^{>}$	(+42%)	**0.3720**	$0.1793^{>}$	(+28%)	0.3260
expCombMNZ	$\mathbf{0.2044}^{\gg}$	(+46%)	0.3520	$\mathbf{0.1824}^{>}$	(+30%)	0.3220

Table 3: Performance of the 11 data fusion techniques for expert search. The parameter of each weighting model (b or c) is empirically tuned to maximise MAP (see Section 4). Relative MAP differences from the median run of the participating groups of the TREC 2005 Enterprise track are shown (MAP 0.1402): the notations are the same as in Table 2. DLH13 is not included as it does not have any parameters that need tuning.

collection-specific heuristics, nor any parameter tuning. In addition, these techniques are low-cost, and are easy to deploy in an operational enterprise setting.

In the next section, we will show that we can significantly improve on the performance of the proposed approach by improving the quality of the underlying document ranking.

6. USE OF DOCUMENT STRUCTURE

The quality of the underlying document ranking returned by the IR system in response to the expert search query is important to the success of the proposed voting approach. If the quality of the document ranking is improved, then the ranking of candidates will also likely improve.

Experiments in the Web and Enterprise track have shown that when the structure of documents (fields) is taken into account by a retrieval system, then the retrieval performance can be improved [16, 38]. For example, a Web document can be represented by three fields: the body, the title, and the anchor text of its incoming hyperlinks. Robertson et al. [29] showed improved retrieval effectiveness in Web tasks when the contribution of each field to the document ranking was controlled by the use of weights. We hypothesise that the retrieval performance of our proposed voting approach for expert search could be improved if the quality of the underlying document ranking is increased. In this section, we use field-based document weighting models that account for the document structure to improve the quality of document ranking, and assess the effect on the proposed voting approach for expert search.

In the previous section, the best performing adapted data fusion techniques used BM25 and PL2 in an empirical setting (see Table 3). In the remainder of this section, we use variations of these two models that take fields into account. Next, we apply these models in our voting approach for expert search.

6.1 Field-based Document Weighting Models

The BM25 weighting model can be extended into a field-based document weighting model, BM25F [38], by replacing Equation (5) with:

field	#tokens	avg_l
body	302,447,426	913.64
anchor text	25,048,853	75.67
title	4,037,394	12.20
total	331,533,673	1001.50

Table 4: The number of tokens (#tokens) & average document length (avg_l) of each field of the W3C collection.

$$tfn = \sum_f w_f \cdot \frac{tf_f}{(1 - b_f) + b_f \cdot \frac{l_f}{avg\_l_f}}, (0 \leq b_f \leq 1) \quad (9)$$

where tf_f is the term frequency of term t in field f of document d, l_f is the length of field f in d, and avg_l_f is the average length of documents in f. The normalisation applied to terms from field f can be controlled by the field hyper-parameter, b_f, while the contribution of the field is controlled by the weight w_f.

Similarly, we can extend the PL2 document weighting model to handle fields. The so-called Normalisation 2 (Equation (7)) is replaced with *Normalisation 2F* [16, 18], so that the normalised term frequency tfn corresponds to the weighted sum of the normalised term frequencies tf_f for each used field f:

$$tfn = \sum_f \left(w_f \cdot tf_f \cdot \log_2(1 + c_f \cdot \frac{avg\_l_f}{l_f}) \right), (c_f > 0) \quad (10)$$

where c_f is a hyper-parameter for each field controlling the term frequency normalisation, and the contribution of the field is controlled by the weight w_f. Having defined Normalisation 2F, the PL2 model (Equation (6)) can be extended to PL2F by using Normalisation 2F.

In the following experiments, we index the body, anchor text and titles of documents as separate fields using Terrier. Table 4 shows the breakdown of the statistics of each field on the W3C collection. As in Section 4, we remove stopwords and apply the first two steps of Porter's stemming algorithm. We again use the 50 expert search task topics from TREC 2005 Enterprise track.

We follow [38] to optimise the involved hyper-parameter values and the field weights as follows. Firstly, for each data fusion technique, the hyper-parameter for each field is tuned using a simulated annealing. During this, the w_f of that field is set to 1, and the weights of the other fields are set to 0. Once good hyper-parameter values have been found, a 3-dimensional simulated annealing is used to find the optimal w_f values. Contrary to Zaragoza et al. in [38] who assumed that the body field should have a weight of 1, we do not assume any constraints on the weights of any fields.

6.2 Experiments and Results

Table 5 shows the retrieval performance of the proposed voting approach using the eleven adapted data fusion techniques, and the BM25F and PL2F field-based weighting models. From the obtained results, we can see that the use of fields has led to a marked improvement in effectiveness compared to Table 3, for both MAP and P@10. In particular, for a large number of cases, there is a statistically significant improvement over the corresponding entry in Table 3. For example, expCombMNZ shows a MAP of 0.2254 for BM25F, compared to only 0.2044 for BM25 - a statistically significant improvement of 10%. The relative performance of the data fusion techniques remains mostly consistent with the previous experiments.

A further inspection of the results shows that while BM25F has higher effectiveness, the average relative improvement in MAP of BM25F compared to BM25 and PL2F compared to PL2 are identical (both +13%). Notice that for the weighting scheme PL2F, the average improvement in P@10 was clearly more marked than for BM25F (+17% vs. +1%), even though both weighting models were tuned for MAP.

By introducing fields into the underlying document ranking technique, we are able to rank highly more documents that are good indicators of expertise for the query. This increased quality of the document ranking leads to an increased performance by the proposed voting approach, using all data fusion techniques evaluated. The obtained results are in the top three most effective techniques reported in [6], while not using any collection-dependent means, such as focusing on the pages containing many of the answers. This is very encouraging, as our approach could be extended to include other factors, such as document and candidate priors, degree of association between documents and profiles etc. Our voting approach is general, and can easily be applied in an enterprise setting independent of the collection and its structure.

7. STABILITY OF ADAPTED DATA FUSION TECHNIQUES

Our voting approach relies on the adapted data fusion techniques to provide a suitable aggregation of candidate votes. In this section, we test the robustness and stability of the adapted data fusion techniques used in the proposed approach, across the seven weighting schemes and settings applied in Tables 2, 3 and 5. Using MAP as the evaluation measure, for each weighting model, we order the adapted data fusion techniques from the best to the worst performing. For example, from Table 5, the ordering for BM25F has expCombMNZ as the best and CombANZ as the worst data fusion technique.

Figure 2 shows the performance of each adapted data fusion technique across all weighting schemes and settings. From the figure, we can observe that the lines joining the performance of each fusion technique on the different weighting models are mostly parallel. This suggests that the relative performance of the data fusion techniques is stable, since their ordering remains unchanged regardless of the used weighting model.

To check that the data fusion techniques are indeed stable, we can use a statistical concordance measure. Kendall's W of concordance [12] measures the concordance of n items over a set of m rankings. W is in the range $W \in$[0,1], where $W = 1$ are identical rankings, and $W = 0$ are completely disagreeing rankings. We use Kendall's W to measure the concordance of the seven rankings of the eleven adapted data fusion techniques. The calculated value $W = 0.9603$ is very close to complete concordance. Moreover, using Table 8 in [12], we see that this value is significant at $p <= 0.01$.

Hence we can see that there is a statistically significant concordance between the rankings of the data fusion techniques, showing that the relative performance of the vari-

	BM25F				PL2F			
Fusion	MAP	ΔMAP	P@10	ΔP@10	MAP	ΔMAP	P@10	ΔP@10
Votes	0.1989$^{\gg}$	(+12%)	0.3260	(+1%)	0.1733	(+4%)	0.3140	(+1%)
RR	0.2235	(+16%)	**0.3780**	(+4%)	0.2030$^{\gg}$	(+14%)	0.3600$^{>}$	(+5%)
BordaFuse	0.2006$^{\gg}$	(+7%)	0.3540	(+4%)	0.1856$^{\gg}$	(+10%)	0.3240	(+3%)
CombANZ	0.0308	(+9%)	0.0220	(-35%)	0.0378	(+10%)	0.0540	(+29%)
CombMED	0.1246	(+15%)	0.2180	(+16%)	0.1276$^{>}$	(+24%)	0.2360$^{>}$	(+33%)
CombMIN	0.0880$^{\gg}$	(+31%)	0.1520	(+9%)	0.1122$^{\gg}$	(+60%)	0.2220$^{\gg}$	(+52%)
CombMAX	0.2051	(+15%)	0.3520	(+10%)	0.1983$^{\gg}$	(+22%)	0.3580$^{>}$	(+20%)
CombSUM	0.2123$^{\gg}$	(+13%)	0.3480	(+5%)	0.1864$^{\gg}$	(+7%)	0.3200	(0%)
CombMNZ	0.2073$^{\gg}$	(+12%)	0.3440	(+6%)	0.1761	(+1%)	0.3220	(0%)
expCombANZ	0.0348	(0%)	0.0420	(-9%)	0.0374	(+12%)	0.0500	(+19%)
expCombSUM	0.2130$^{\gg}$	(+7%)	0.3660	(-2%)	0.2047	(+14%)	**0.3700**	(+13%)
expCombMNZ	**0.2254**$^{\gg}$	(+10%)	**0.3780**	(+7%)	**0.2072**	(+14%)	0.3580	(+11%)
Mean Δ		(+13%)		(+1%)		(+13%)		(+17%)

Table 5: Performance of the 11 data fusion techniques for expert search, when used with two field-based document weighting models. Relative improvements over the equivalent entry in Table 3 are shown. Statistically significant improvements at $p <= 0.05$ are denoted $>$; and significant improvements at $p <= 0.01$ are denoted $\gg$.

ous adapted data fusion techniques are indeed very stable, regardless of the weighting model used. Although we cannot predict the absolute performance of each adapted data fusion technique on an arbitrary weighting model, we can conclude that some data fusion techniques are always more likely to perform better than others. These are the ones which model the important sources of expertise evidence, as surmised in Section 5.

8. CONCLUSIONS & FUTURE WORK

In this paper, we proposed that expert search can be seen as a voting problem, where documents vote for the candidates with relevant expertise. We adapted eleven data fusion techniques to our proposed approach. Three statistically different document weighting models were tested, to assess the effectiveness and stability of the data fusion techniques in our approach. The evaluation was conducted in the context of the expert search task of the TREC 2005 Enterprise track.

The results show that our proposed approach is effective when using appropriate adapted data fusion techniques. While the techniques have varying degrees of performance, some of them consistently outperform others, regardless of the applied document weighting model. The most successful techniques usually integrate the most highly ranked documents of the profile (strong votes), and the number of retrieved documents from the profile (number of votes). Our experimental results also suggest that the quality of the underlying ranking of documents is important in enhancing the retrieval performance of the expert search system. Indeed, we showed that a recent Web IR technique - i.e. the use of fields to represent the document structure - constantly leads to marked performance improvements, which are very often significant.

We also demonstrate that the relative performance of the data fusion techniques is stable across the various weighting models and settings applied. Indeed, when the data fusion techniques are compared across various weighting models, the concordance of their relative performance rankings shows that some of the data fusion techniques are always more likely to outperform others.

The approach proposed in this paper is general in the sense that it is not dependent on heuristics from the used enterprise collection, and can be easily operationally deployed with little computational overhead. Moreover, we have successfully deployed an expert search system based on these techniques [17].

In the future, we will investigate the use of query expansion to enhance the underlying document ranking. However, query expansion has to be used with caution. Indeed, if the performance of the query is predicted to be poor, applying query expansion can lead to a further degradation of retrieval performance [37].

Moreover, this work can be naturally extended to integrate prior knowledge. For example, we believe that not all documents are likely to be good indicators of expertise, and furthermore that not all candidates are likely to be experts. Designing and integrating document and candidate priors with our approach could increase the retrieval effectiveness of the expert search system. Finally, we are keen to evaluate our proposed approach on another expert search test collection.

9. REFERENCES

[1] G. Amati. *Probabilistic Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, University of Glasgow, 2003.

[2] G. Amati. Frequentist and Bayesian Approach to Information Retrieval. In *Proceedings of ECIR 2006*, volume 3936 *Lecture Notes in Computer Science*, pages 13–24, Springer, 2006.

[3] K. Balog and M. de Rijke. Finding experts and their details in e-mail corpora. In *15th International World Wide Web Conference (WWW2006)*, Edinburgh, Scotland, 2006.

[4] J. A. Aslam and M. Montague. Models for metasearch. In *Proceedings of ACM SIGIR 2001*, pages 276–284, New Orleans LA, 2001.

[5] C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom. Expertise identification using email communications. In *Proceedings of ACM CIKM 2003*, pages 528–531, New Orleans, LA, 2003.

[6] N. Craswell, A. P. de Vries, and I. Soboroff. Overview

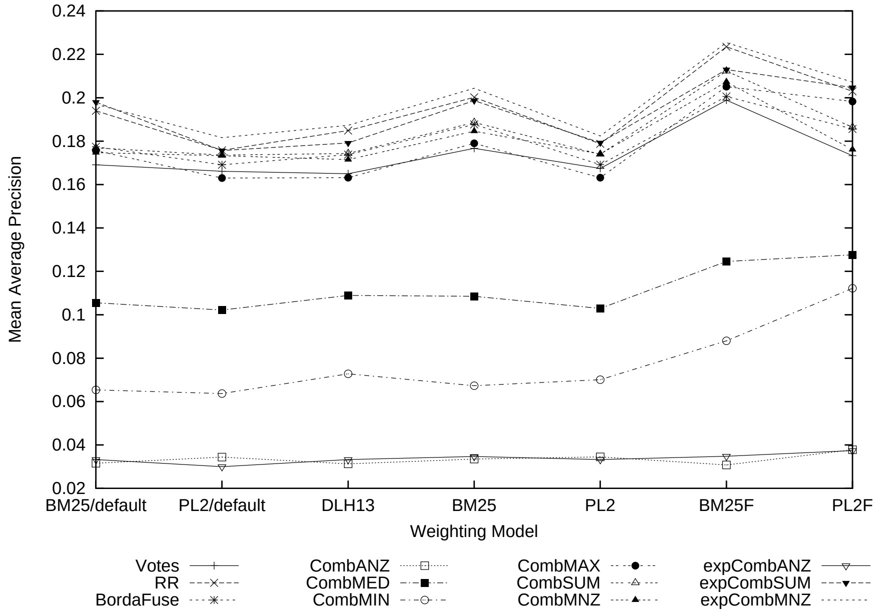

Figure 2: The performance of adapted data fusion techniques plotted across various weighting models. The notation "/default" denotes the default settings of BM25 and PL2 document weighting models (see Table 2).

of the TREC-2005 Enterprise Track. In *Proceedings of TREC-2005*, Gaithersburg, MD, 2005.

[7] N. Craswell, D. Hawking, A.-M. Vercoustre, and P. Wilkins. Panoptic expert: Searching for experts not just for documents. In *Ausweb Poster Proceedings*, Queensland, Australia, 2001.

[8] B. Dom, I. Eiron, A. Cozzi, and Y. Zhang. Graph-based ranking algorithms for e-mail expertise analysis. In *Proceedings of ACM SIGMOD DMKD Workshop 2003*, pages 42–48, San Diego, CA, 2003.

[9] S. T. Dumais and J. Nielsen. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of ACM SIGIR 1992*, pages 233–244, Copenhagen, Denmark, 1992.

[10] E. A. Fox and J. A. Shaw. Combination of multiple searches. In *Proceedings of TREC-2*, Gaithersburg, MD, 1994.

[11] M. Hertzum and A. M. Pejtersen. The information-seeking practises of engineers: searching for documents as well as for people. *Inf. Process. Manage.*, 36(5):761–778, 2000.

[12] M. G. Kendall. *Rank Correlation Methods, 2nd ed.* Charles Griffin & Co. Ltd., London WC2, 1955.

[13] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[14] J. H. Lee. Analyses of multiple evidence combination. In *Proceedings of ACM SIGIR 1997*, pages 267–276, Philadelphia, PA, 1997.

[15] X. Liu, W. B. Croft, and M. Koll. Finding experts in community-based question-answering services. In *Proceedings of ACM CIKM 2005*, pages 315–316, Bremen, Germany, 2005.

[16] C. Macdonald, B. He, V. Plachouras, and I. Ounis. University of Glasgow at TREC 2005: Experiments in Terabyte and Enterprise tracks with Terrier. In *Proceedings of TREC-2004*, Gaithersburg, MD, 2004.

[17] C. Macdonald and I. Ounis. Searching for expertise using the Terrier platform. In *Proceedings of SIGIR 2006*, Seattle, WA, 2006.

[18] C. Macdonald, V. Plachouras, B. He, C. Lioma, and I. Ounis. University of Glasgow at WebCLEF 2005: Experiments in per-field normalisation and language specific stemming. In *Proceedings of CLEF Workshop 2005*, volume 4022 *Lecture Notes in Computer Science*, 2006.

[19] R. Manmatha, T. Rath, and F. Feng. Modelling score distributions for combining the outputs of search engines. In *Proceedings of ACM SIGIR 2001*, pages 267–275, New Orleans, LA, 2001.

[20] M. Maybury, R. D'Amore, and D. House. Expert finding for collaborative virtual environments. *Commun. ACM*, 44(12):55–56, 2001.

[21] A. McLean, A.-M. Vercoustre, and M. Wu. Enterprise PeopleFinder: Combining Evidence from Web Pages and Corporate Data. In *The 8th Australasian Document Computing Conference (ADCS'03)*, 2003.

[22] M. Montague and J. A. Aslam. Metasearch consistency. In *Proceedings of ACM SIGIR 2001*, pages 386–387, New Orleans, LA, 2001.

[23] M. Montague and J. A. Aslam. Relevance score normalization for metasearch. In *Proceedings of ACM CIKM 2001*, pages 427–433, Atlanta, GA, 2001.

[24] M. Montague and J. A. Aslam. Condorcet fusion for improved retrieval. In *Proceedings of ACM CIKM 2002*, pages 538–548, McLean, VA, 2002.

[25] P. Ogilvie and J. Callan. Combining document representations for known-item search. In *Proceedings of ACM SIGIR 2003*, pages 143–150, Toronto, Canada, 2003.

[26] I. Ounis, G. Amati, Plachouras V., B. He, C. Macdonald, and D. Johnson. Terrier Information Retrieval Platform. In *Proceedings of ECIR 2005*, volume 3408 *Lecture Notes in Computer Science*, pages 517–519, Springer, 2005.

[27] I. Ounis, G. Amati, Plachouras V., B. He, C. Macdonald, and C. Lioma. Terrier: A High Performance and Scalable Information Retrieval Platform. In *Proceedings of the OSIR Workshop 2006*, pages 18–25, Seattle, WA, 2006.

[28] V. Plachouras, B. He, and I. Ounis. University of Glasgow at TREC2004: Experiments in Web, Robust and Terabyte tracks with Terrier. In *Proceedings of TREC-2004*, Gaithersburg, MD, 2004.

[29] S. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In *Proceedings of ACM CIKM 2004*, pages 42–49, Washington, DC, 2004.

[30] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, M. Gatford, and A. Payne. Okapi at TREC-4. In *Proceedings of TREC-4*. Gaithersburg, MD, 1995

[31] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at TREC. In *Proceedings of TREC-1*, Gaithersburg, MD, 1992.

[32] J. Savoy, A. L. Calvé, and D. Vrajitoru. Report on the TREC-5 experiment: data fusion and collection fusion. In *Proceedings of TREC-5*, Gaithersburg, MD, 1997.

[33] J. A. Shaw and E. A. Fox. Combination of multiple searches. In *Proceedings of TREC-3*, Gaithersburg, MD, 1994.

[34] W. Sihn and F. Heeren. Xpertfinder - Expert finding within specified subject areas through analysis of E-mail communication. In *Proceedings of Euromedia 2001*, pages 279–283, 2001.

[35] J. Wang, Z. Chen, L. Tao, W.-Y. Ma, and L. Wenyin. Ranking user's relevance to a topic through link analysis on web logs. In *Proceedings of WIDM 2002 workshop*, pages 49–54, McLean, VA, 2002.

[36] D. Yimam-Seid and A. Kobsa. Expert finding systems for organizations: Problem and domain analysis and the DEMOIR approach. *Journal of Organizational Computing and Electronic Commerce* 13(1):1-24, 2003.

[37] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *Proceedings of ACM SIGIR 2005*, pages 512–519, Salvador, Brazil, 2005.

[38] H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson. Microsoft Cambridge at TREC-13: Web and HARD tracks. In *Proceedings of TREC-2004*, Gaithersburg, MD, 2004.

[39] M. Zhang, R. Song, C. Lin, S. Ma, Z. Jang, Y. Lin, Y. Liu, and L. Zhao. Expansion-based technologies in finding relevant and new information: THU TREC2002: Novelty Track experiments. In *Proceedings of TREC-2002*, Gaithersburg, MD, 2002.