A Two-Stage Model for Expert Search

Yunbo Cao [1,2,*], Jingjing Liu [3], Shenghua Bao[1], Hang Li[2], and Nick Craswell[4]

1 Shanghai Jiao Tong University, Shanghai China
2 Microsoft Research Asia, Beijing China
3 Massachusetts Institute of Technology, Massachusetts U.S.A.
4 Microsoft Research Cambridge, Cambridge England
* Corresponding Author
TEL: 86-10-58963128. FAX: 86-10-88097305. E-mail Address: yunbo.cao@microsoft.com

ABSTRACT

This paper is concerned with expert search, a search task where the user types a query representing a topic and the search system returns a ranked list of people who are considered experts on the topic. The system does this using the evidence existing in a document collection. We proposed a model for performing the task in our TREC 2005 submission, referred to as two-stage model. Since that, a number of groups adopted the model in their systems and achieved good performances in their TREC submissions. This paper aims to give a comprehensive description on the two-stage model and provide more experimental results on the use of the model. Two-stage model is capable of employing many types of association relationships among query terms, documents and people (experts). The model consists of two parts: relevance model and co-occurrence model. The relevance model characterizes the relevance of documents to queries. The co-occurrence model characterizes the co-occurrence between people and terms (i.e., queries) in various types. In this paper, we report our new experimental results on two-stage model using the data in both TREC 2005 and TREC 2006 expert search tasks. We also show that our approach is applicable beyond TREC W3C corpus, by introducing experimental results on expert finding at Microsoft Research[1]. Our experimental results, once again, demonstrate the effectiveness of the two-stage model.

Keywords: Expert Search, Probabilistic Model, Information Retrieval

1. INTRODUCTION

In an enterprise, knowing ‘who knows about what’, in other words, finding experts on certain topics, is a critical issue (Hawking, 2004). One way to solve the problem is to manually construct and maintain a database storing necessary expert-expertise information. However, (a) it is costly to create and frequently update such a database; furthermore, (b) it is difficult to make the information complete and specific. An alternative is to automatically discover and search expert-expertise information from the document data available at the company (e.g., data on its intranet). Although this approach is also not perfect, it at least does not suffer from the problem (a) explained above. In expert search, the user submits a query representing a topic, the search system finds the people (experts) strongly associated with the topic using the document collection, ranks the people according to the strength of association with the topic, and returns the ranked list of people.

Various association relationships among terms, documents, and people in different contexts in the document collection might be useful for expert search. Relevance between terms and documents, co-occurrence between terms and people in bodies of documents, co-

[1] http://research.microsoft.com/

occurrence between terms and people within titles and authors of documents, and co-occurrence between people and people in documents are examples of potentially useful association information.

Two questions arise here. (a) What kinds of association information are really useful for expert search? (b) Is there a general framework that can effectively use all the association information?

In this paper, we propose a unified model for expert search, referred to as, two-stage expert search model. The model can incorporate many types of association information described above in a unified and theoretically-sound way. The two-stage model consists of two parts: relevance model and co-occurrence model. The relevance model characterizes the relevance of documents to queries. The co-occurrence model characterizes the association between people and terms (i.e., queries). The co-occurrence model may be further decomposed into sub-models, each representing one type of co-occurrence. There are five co-occurrence sub-models: window-based sub-model, title-author sub-model, block-based sub-model, neighbor-based sub-model, and cluster-based sub-model.

There was not much previous work on expert search, until TREC started to offer a task on expert search in 2005. P@NOPTIC (Craswell et al., 2001) and Expert/Expert- Locating (EEL) (Steer and Lochbaum, 1988) were examples of earlier expert search systems before TREC. Both of them only used one type of co-occurrence information, namely co-occurrences of people and terms within bodies of documents. Many systems of TREC 2005 and TREC 2006 also managed to use similar types of information in expert search. As far as we know, it is us who first proposed the use of two-stage model for conducting expert search, at our TREC 2005 submission (Cao et al., 2005). Our system was among the two best performers of TREC 2005, and it worked the best depending on evaluation measures. In TREC 2006, a number of systems adopted the same two-stage model, including the two top-running systems (Zhu and Song, 2006; Bao et al., 2006). Similar models were proposed independently by other researchers at TREC 2005 and TREC 2006. For example, Balog et al. (2006) considered an expert search model in which document becomes a hidden variable connecting query and expert in a similar way as in our two-stage model. Balog et al.'s model, however, does not utilize various sorts of co-occurrences as in our two-stage model.

We used three data sets to re-verify the effectiveness of two-stage model. Two of the data sets are the expert search data sets at TREC 2005 and TREC 2006, and another is created at Microsoft Research (MSR). Our experimental results show that two-stage model can perform better than the traditional profile-based method in expert search, and can achieve the best results reported in TREC 2005 and TREC 2006. This paper is also the first to verify the effectiveness of such approaches on a separate test collection (MSR), indicating that the approach and several of our sub-models can be successfully applied beyond the TREC W3C test collection.

The rest of the paper is organized as follows. In Section 2, we introduce related work, and in Section 3, we give sources of evidence that we use for expert search. In Section 4 we describe our two-stage model of expert search. Section 5 gives our experimental results. We make concluding remarks in Section 6.

2. RELATED WORK

2.1 Expert Search System

Discovering right experts on certain topics within or outside a company has become more and more important for conducting business at enterprise. Several methods of automatically finding experts from documents were proposed (Craswell et al., 2001; Mattox et al., 1999; Steer and Lochbaum, 1988).

P@NOPTIC employs what is referred to as the 'profile-based' approach in searching for experts (Craswell et al., 2001). In the approach, texts from documents mentioning a specific person are combined. The combined document is used as the profile of the person. Given a query, it uses as the answers the persons whose profiles are most relevant to the query. Expert/Expert-Locating (EEL) system (Steer and Lochbaum, 1988) uses the same approach in searching for expert groups. The profile of each group is represented by a cluster of documents. Latent Semantic Indexing (LSI) is used when matching profiles against queries. DEMOIR (Yimam-seid and Kobsa, 2002) enhances the profile-based approach by separating co-occurrences into different types: documents written by experts, documents profiling experts, and document mentioning experts. DEMOIR further uses a scheme called “matrix fusion” for combining term-document frequencies and expert-document frequencies,

Another setting for expert search is to assume that data from other resources is available. For instance, Expertise Recommender (McDonald and Ackerman, 2000), Expertise Browser (Mockus and Herbsleb, 2002) and the system in (McDonald and Ackerman, 1998) make use of log data in software development systems to find experts. Yet another approach is to mine expert and expertise from email data (Campbell et al., 2003; Dom et al., 2003; Sihn and Heeren, 2001).

The method proposed in this paper makes use of not only profile-based information, but also other types of information such as co-occurrences between people and people. More importantly it is based on a unified probabilistic framework, while the previous methods are not.

2.2 Expert Search at TREC

At TREC, a task on expert search was organized within the enterprise track[2]. The task is equivalent to answering the question of “who in the organization is an expert on X?” A participating system should return a list of people (not documents) who are experts on a given query. This is exactly the problem addressed in this paper. Nine groups participated in the task at TREC 2005, twenty five groups in the task at TREC 2006, and more than thirty-five groups in the task at TREC 2007.

In the task of both TREC 2005 and TREC 2006, the data at the web site of W3C[3] was used as the document collection. The ground truth of TREC 2005 was obtained from an existing database of W3C working groups. The names of working groups were used as queries, and the members of a working group were the experts for that query. The ground truth of TREC 2006 was built manually by the participants of the task. At TREC 2007, a new corpus of CSIRO[4] was introduced and the ground truth was built manually by the experts from CSIRO.

Many systems of TREC 2005 also used co-occurrence information in expert search. For example, Balog et al. (2006) employed a model similar to the two-stage model proposed in this paper. However, their work is different from our work in several aspects. For example, they do not employ a smoothing technique for estimating the probabilities and do not utilize various sorts of co-occurrence information, as we do in this paper.

More systems were developed for expert search in TREC 2006. Some of them adopted the basic idea in the two-stage model. For example, Fang et al. extended the two-stage model by introducing a prior distribution of experts and relevance feedback (Fang et al., 2006). Petkova and Croft further extended the profile-based method by using a hierarchical language model (Petkova and Croft, 2006). The two top-running systems of TREC 2006 (Bao et al., 2006; Zhu and Song, 2006) also made use of the two-stage model. Bao et al. (2006) expanded the notion of two-stage modeling by further considering expert matching quality, query matching quality, and document quality. Zhu and Song (2006) employed the two-stage model as well as other types of information such as document quality, document structure, and size of co-occurrence window.

Other directions for expert search were also studied within the TREC setting (Macdonald and Ounis, 2006; Macdonald and Ounis, 2007; Fu et al., 2007). For example, Macdonald and Ounis (2006) investigated the effectiveness of the voting approach and other data fusion techniques for expert search. For other work related to expert search or expertise search, see (Balog and Rijke, 2006; Balog et al., 2007; Serdyukov et al., 2007).

[2] TREC Enterprise Search guideline. http://www.ins.cwi.nl/projects/trec-ent/wiki/index.php/Main_page

[3] World Web Consortium. http://w3.org

[4] http://www.csiro.au/

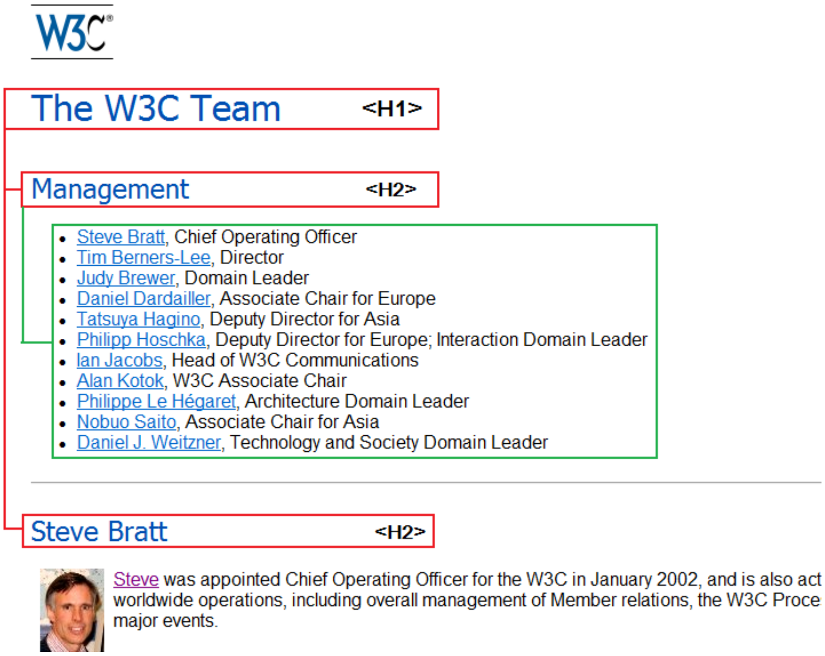
3. SOURCES OF EVIDENCE

Association relationships among terms, documents, and people existing in the document collection can be used as evidence in discovering and ranking experts.

It has been made clear in previous profile-based work that co-occurrences between people and terms in documents are useful for expert search. There are other types of association which can also be helpful.

(1) As noted, if a person appears together with a term describing a topic, that is evidence of association between the two. One consideration is proximity, perhaps only considering co-occurrences within a pre-determined window size. Obviously, if the two are located far away, then they might not be related.

(2) Another source of co-occurrence information is document structure. For example, in Figure 1 people who are related to “W3C management” will be easily found, based on document hierarchy. The “W3C” occurs in a heading, “management” occurs in a subheading, and person names occur within the scope of that subheading.



W3C

The W3C Team

Management

- Steve Bratt, Chief Operating Officer
- Tim Berners-Lee, Director
- Judy Brewer, Domain Leader
- Daniel Dardailler, Associate Chair for Europe
- Tatsuya Hagino, Deputy Director for Asia
- Philipp Hoschka, Deputy Director for Europe; Interaction Domain Leader
- Ian Jacobs, Head of W3C Communications
- Alan Kotok, W3C Associate Chair
- Philippe Le Hégaret, Architecture Domain Leader
- Nobuo Saito, Associate Chair for Asia
- Daniel J. Weitzner, Technology and Society Domain Leader

Steve Bratt

Steve was appointed Chief Operating Officer for the W3C in January 2002, and is also act
worldwide operations, including overall management of Member relations, the W3C Proce
major events.

Figure 1. Co-occurrence within document structure

(3) If a person appears immediately before or after a term, then it is likely that the person is strongly related to the term. The information should also be used.

(4) Co-occurrences in different positions of documents can have different degrees of importance. We identified the title and author fields as potentially important. For example, if a query “timed text” appears in the <title> of a document and a person “Glenn Adams” in the <author> of the document, it is very likely that Glenn Adams is an expert on timed text, because he has authored a document on the topic.

(5) Co-occurrences between people and people may also be useful in identifying experts. It is likely that people often appearing in the same documents (e.g., co-workers in the same group) share the same expertise areas. The information should also be used in expert search.

(6) We hypothesize that relevance of documents to queries can influence the results of expert search. For example, given a query ‘timed text’, the co-occurrences included in a document more relevant to the query should be more indicative of expertise of person names appearing in the document.

In summary, all the association information described above should be useful for expert search. How to incorporate all of them into a model becomes a key question for expert search.

4. TWO-STAGE MODEL FOR EXPERT SEARCH

As input in expert search, we receive a query. The query is usually a noun phrase representing a topic. We automatically retrieve a list of people associated with the topic and rank them according to the likelihood of their being the experts on the topic.

We assume, in this paper, that a list of candidate personal names is made available to the system, and the expert search system selects only from that list. In a real expert search system this could come, for example, from the organization's staff list.

We formalize the problem of expert search - finding and ranking people *e* who are experts for a given topic *q* - as that of estimating a conditional probability *p(e|q)*. Thus, the problem is transformed as how to estimate the probability accurately.

To determine the conditional probability *p(e|q)*, we propose employing a two-stage model for expert search. The two-stage model is defined as

$$p(e \mid q) = \sum_{d} p(e,d \mid q) = \sum_{d} p(d \mid q) p(e \mid d,q) \qquad (1)$$

where $p(d \mid q)$ denotes the relevance model and $p(e \mid d,q)$ denotes the co-occurrence model. The relevance model $p(d \mid q)$ can represent document relevance information described at (6) in Section 3. The co-occurrence model $p(e \mid d,q)$ can represent various types of co-occurrence information including those of (1) ~ (5) in Section 3.

We further decompose the co-occurrence model into several sub-models and let each of the sub-models represent one type of co-occurrence information. In principle, it is also possible to decompose the document relevance model into several sub-models, perhaps as in (Ponte and Croft, 1998). However, in this paper, we do not decompose the document relevant model because the focus of this paper is to study various co-occurrence sub-models.

The co-occurrence model is defined as a linear combination of the co-occurrence sub-models:

$$p(e \mid d,q) = \sum_{m} \lambda_m p_m(e \mid d,q) \qquad (2)$$

where $p_m(e \mid d,q)$ denotes the m^{th} sub-model. $\lambda_m (0 < \lambda_m < 1)$ denotes the weight of the m^{th} sub-model.

In our work, we make use of five co-occurrence sub-models. They are window-based sub-model, title-author sub-model, block-based sub-model, neighbor-based sub-model, and cluster-based sub-model. We will explain each of them in detail in this section.

We employ the language modeling technique used in IR for constructing both the relevance model $p(d \mid q)$ and the co-occurrence model. Specifically, we estimate the relevance model as in (Ponte and Croft, 1998). Moreover, we base most of the co-occurrence sub-models (except cluster-based model) on the equation as follows:

$$p_m(e \mid d,q) = \mu \frac{pf_m(e,d,q)}{DL_m(d,q)} + (1-\mu) \sum_{d':e \in d'} \frac{pf_m(e,d',q)}{DL_m(d',q)} \Big/ df_e \qquad (3)$$

where $pf_m(e, d, q)$ is frequency of person *e* of special relation *m* with query *q* in document *d*, $DL_m(d, q)$ is total frequency of persons of special relation *m* with the query *q* in *d*, and df_e is document frequency of person *e*. We use Dirichlet prior (Ogilvie and Callan, 2003) for smoothing:

$$\mu = \frac{DL_m(d,q)}{DL_m(d,q) + \kappa} \qquad (4)$$

where κ is average length of term frequency of persons in the collection as (Ogilvie and Callan, 2003).

Actually, the maximum likelihood estimation (MLE) of the probability *e* given *d* and *q* under the people distribution for document *d* is the first term of equation (3), namely $\frac{pf_m(e,d,q)}{DL_m(d,q)}$. However, there are several problems with this MLE estimator. The most obvious problem is that we do not wish to assign a probability of zero to a person *e* that is missed in the document *d*. In most cases, there are only a few person names in a document. We cannot safely infer that a person associated to a topic *q* in a document *d* has much more expertise on the topic *q* than a person missed in the document *d* (non-zero probability vs. zero probability). Thus, we use as a smoothing the average of probabilities over the document collection $\sum_{d':e\in d'} \frac{pf_m(e,d',q)}{DL_m(d',q)} \Big/ df_e$. This gives us equation (3). Another issue is that we assume that people *e* occur independently given a query *q* and a document *d*. Obviously, this is not true, as noted in (5) of Section 3. In fact, we seek to take advantage of this dependence, in the cluster-based co-occurrence sub-model (Section 4.5).

The difference between the co-occurrence sub-models, except cluster-based model, lies in relation *m* (in equation (3)). Thus, in the following subsections, we focus on elaborating various ways of building the relation.

4.1 Window-based Sub-model

In the window-based sub-model, co-occurrence is said to happen when the query and person appear within a window of a pre-determined size.

In our work, we also limit the range of each window by considering the natural boundaries of document structure. More precisely, we use 'block' for restricting the range of co-occurrences between terms and people. Here, a block is the text within one of the HTML tags listed in Table 1. When the query matches terms in a document, we look for persons within the smallest surrounding block, but also within a window of a fixed size (in words). If the left or the right boundary of the block exceeds the left or right boundary of the window, the left or right boundary of the window will be used.

Table 1. HTML tags for identifying blocks

<Table>, <ol>, <p>, <ul>, <pre>, <li>, <dl>, <dt>, <tr>, <hr>

When equation (3) is evaluated in the window-based sub-model, it is with respect to whether person *e* and the query *q* satisfy this window constraint.

We note that our window-based sub-model is almost equivalent to conventional profile-based approach. The difference is that we add an HTML block constraint, while profile-based approaches might use the window alone.

The window-based co-occurrence has been used in other applications (c.f., Maarek and Smadja, 1989) as well.

4.2 Title-Author Sub-model

We extract metadata such as title and author from documents. The title-author sub-model represents the association between a query that appears in the title and the person names that appear as authors of the document. Thus, in the sub-model, $pf_m(e, d, q)$ equals 1 when the query *q* and person *e* occur in the title and author of the document *d*, respectively. $DL_m(d, q)$ equals the number of the persons appears as the author.

For example, for web pages, we can construct the <*Title*> from either the 'title' metadata or the content of web pages [15]. As for Word documents, we can extract the <*Title*> from the contents [15]. As for e-mail, we can use the 'subject' field as the <*Title*>.

As for extraction of title and author from web pages, we employ a supervised learning based approach (Hu et al., 2005; Hu et al., 2006). As for email documents, we use the 'subject' field as title and the 'sender' field as author.

4.3 Block-based Sub-model

In this sub-model, we use the format information of HTML document to create blocks and take into consideration of co-occurrence within the tree of created blocks, referred to as block tree.

A block tree denotes the relationships between blocks in an HTML document. If block A contains block B, then the corresponding node of block A becomes parent node of the corresponding node of block B. Here, the HTML tags <H1>, <H2>, <H3>, <H4>, <H5>, and <H6> determine the boundaries of blocks. Note that there is a preference order over the tags: <H1>≻<H2>≻<H3> ≻<H4>≻<H5>≻<H6>. Figure 2 shows an example of block tree.

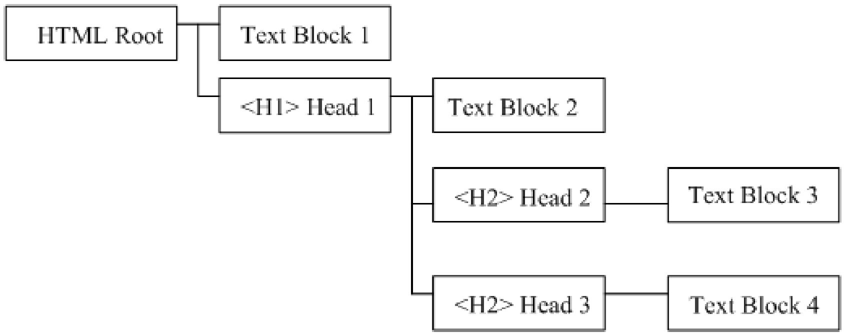


Figure 2. Example of block tree

The block-based sub-model represents the association of queries and people within the block tree. We calculate co-occurrences of terms and people within a block tree with the algorithm in Figure 3. At line 3, function *ancestor_path(T_i)* concatenates all the texts in path from block T_i to root. At line 4, function *exist_in (q, ancestor_path (Ti))* is true, when query *q* appears in texts from *ancestor_path(T_i)*; it is false, otherwise. Thus, Figure 3 gives another relation between the query *q* and the person *e* that can be used in the evaluation of equation (3).

```
1)  For a given query $q$
2)   Foreach block $T_i${
3)     Get ancestor path ancestor_path ($T_i$);
4)     if (exist_in ($q$, ancestor_path ($T_i$)){
5)       Foreach person e appearing in block $T_i${
6)       count frequency of $e$ as co-occurrence with $q$;}
7)   }
8) }
```

Figure 3. Calculating co-occurrence in block tree

4.4 Neighbor-based Sub-model

In the neighbor-based sub-model, the co-occurrence is about whether the query and a person appear immediately next to each other.

The sub-model is equivalent to a window-based sub-model using a small window size. It is used to support the intuition that, the closer to a query *q* appears a person *e*, the more important the person *e* is. As we will discuss in Section 5.2, we separate it from the window-based sub-model so that we can emphasize it separately.

4.5 Cluster-based Sub-model

Cluster-based sub-model utilizes person-to-person co-occurrence information among people. The rationale is that people co-occurring in documents should have similar expertise and thus the use of the information can improve estimation on *co-occurrences between terms and people*.

First, for each person we count the frequencies of terms (including personal names) appearing within the windows surrounding the person in the documents. We create a vector of term frequencies for each person, which represents a profile of the person. Then, we cluster the people according to the similarity in their profile vectors. In this paper, we use K-Means as the clustering algorithm. Finally, the clustering results are used in estimation of the sub-model representing co-occurrences between terms and people:

$$p_{new}(e \mid d,q) = (1-\lambda)\, p(e \mid d,q) + \lambda \sum_{e' \in E(e)} \frac{p(e' \mid d,q)}{\mid E(e) \mid} \tag{5}$$

where *E*(*e*) denotes the cluster which e belongs to and |*E*(*e*)| denotes the size of *E*(*e*). Equation (5) can be seen as another form of smoothing for the model in equation (2).

5. EXPERIMENTAL RESULTS

In the experiments, we evaluated the effectiveness of the two-stage model for expert search with three data sets. The data sets are TREC2005, TREC 2006, and that of MSR.

5.1 Evaluation Measures

We made use of three measures in evaluation. They are MAP, R-precision, and Top N precision (P@N).

MAP is the mean of average precisions over a set of queries. Given a query q_i, average precision is defined as the average of precision after each correct (expert) answer is retrieved. Given a query q_i, its average precision ($AvgP_i$) is calculated as

$$AvgP_i = \sum_{j=1}^{M} \frac{P(j) \times pos(j)}{number\ of\ correct\ answers\ in\ ground\ truth} \tag{6}$$

where *j* is the rank, *M* is the number of candidates returned, *pos(j)* is a binary function to indicates whether the candidate in the rank *j* is in the ground truth, and *P(j)* is the precision at the given cut-off rank *j*:

$$P(j) = \frac{number\ of\ correct\ (expert)\ answers\ in\ top\ j\ positions}{j} \tag{7}$$

R-precision is defined as

$$R - precision = \frac{\sum_{i=1}^{K} P(R_i)}{K} \tag{8}$$

where R_i is number of experts for the query *i* in the ground truth. *K* is the total number of queries (topics) in the evaluation set.

Top *N* precision (*P@N*) is defined as

$$P @ N = \frac{\sum_{i=1}^{K} P(N)}{K} \tag{9}$$

Here, *N* = 1, 2, ..,10.

5.2 Personal Name Identification

We assume that a list of experts is provided to the system, containing all possible candidate experts in the organization. In our experiments, each candidate is represented as a triple of <candidate_id, full_name, email>. Note that it is possible that two persons in different candidate_id's can have same full_names. We used a rule-based system for identifying personal names in the document set. Specifically, for each person in the list, we matched it against the documents and located all the positions of it using the heuristic rules as Table 2. We could then index the information with regard to each of the persons.

Table 2. Heuristic rules for identifying personal names

1) Every occurrence of a candidate's email address is normalized to the appropriate candidate_id.

2) Every occurrence of a candidate's full_name is normalized to the appropriate candidate_id if there is not ambiguity; otherwise, the occurrence is normalized to the candidate_id of the most frequent candidate with that full_name.

3) Every occurrence of abbreviated name is normalized to the appropriate candidate_id if there is not ambiguity; otherwise, the occurrence may be normalized to the candidate_id of a candidate whose full name has also appeared in the document.

4) All the personal occurrences other than those covered by Heuristic 1) ~ 3) are ignored.

We also constructed an annotated data set for evaluating our heuristic rules. We sampled 500 documents from the W3C document collection and then asked an annotator to annotate all the personal name occurrences with their corresponding candidate_id. Besides the documents, the annotator was also supplied with a list of persons used in the evaluation of TREC expert search task. The result shows that the rule-based system can really work well on the data set. The precision and recall are 100% and 90%, respectively. Most of the missed names are those irregular abbreviations (e.g. 'danc' for 'Dan Connolly').

For the abbreviation name of a full_name, we used the last name, first name, and various variations. For example, Table 3 illustrates all the abbreviations of the name of "Linda Jane Smith".

Table 3. Abbreviations of "Linda Jane Smith"

Last Name	Smith	
First Name	Linda	Linda J.
	Linda Jane	
Variations	Linda J Smith	Smith, Linda J
	Linda Smith	Smith, Linda
	L J Smith	Smith, L J
	L Smith	Smith, L

5.3 Expert Search at W3C (TREC 2005)

In this experiment, we used the data set in the expert search task of enterprise search track at TREC 2005. The document collection is a crawl of the public W3C sites in June 2004. The crawl comprises in total 331,307 web pages. It includes the main W3C site, email archives, a wiki site and log of source control.

The ground truth on expert search is obtained from an existing database of W3C working groups. It consists of two parts, namely training set and test set. The training set and test set include 10 and 50 topics. For each topic, a list of people is associated as experts on it. The people are from a list of 1,092 candidates. In this case, the topics are the names of W3C working groups and the experts are the members of the groups.

In the following experiments, we used the training set to tune parameters and the test set to evaluate results. *Thus, all the parameters in the sub-models and the linear combination of the sub-models were tuned with the training data.*

The baseline method is our window-based co-occurrence sub-model, whose performance is assumed to be equivalent to a profile-based approach. Evaluation on the training set gave us a window size of 50, as indicated in Figure 4. We can see that the model performs badly when the size of window is small (for example, 20). Analysis indicates that the main problem with the model having a small window size is small coverage, missing useful associations. Despite this, a small window size usually means a high accuracy. For that reason we introduce a neighbor-based co-occurrence sub-model.

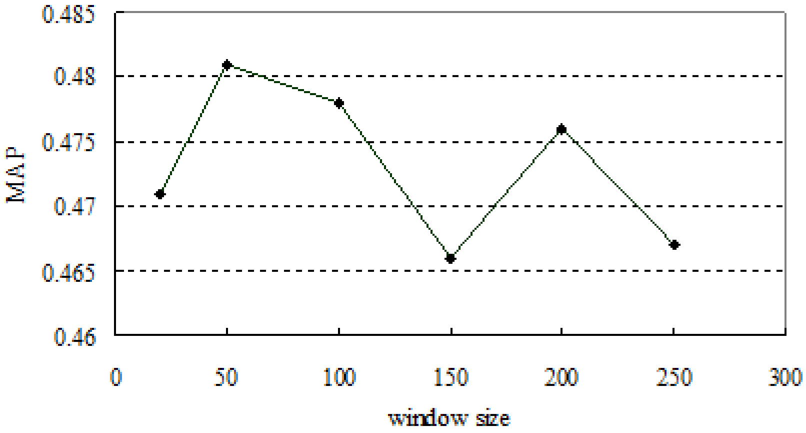


Figure 4. Window-based model in varied window sizes

We incrementally added to the baseline method the proposed components, namely relevance model, title-author co-occurrence sub-model, block-based co-occurrence sub-model, neighbor-based co-occurrence sub-model, and cluster-based sub-model.

For the relevance model, the Dirichlet prior for smoothing was utilized (Ogilvie & Callan, 2003). In the title-author co-occurrence sub-model, we used heuristic methods for title and author extraction. Given a web page, we used the title field as its title. Furthermore, we regarded the names appearing in the top 10 lines in the page as authors. As for email data, we used the contents after “subject” and “from” fields as title and author respectively. In the cluster-based sub-model, we tried various values for the number of clusters. It turned out that when the number of clusters is 20 the performance achieves the best. Furthermore, the parameter λ in equation (5) was set to 0.5, according to evaluation results on the training data (c.f. Figure 5). The other co-occurrence sub-models do not have parameters to be tuned.

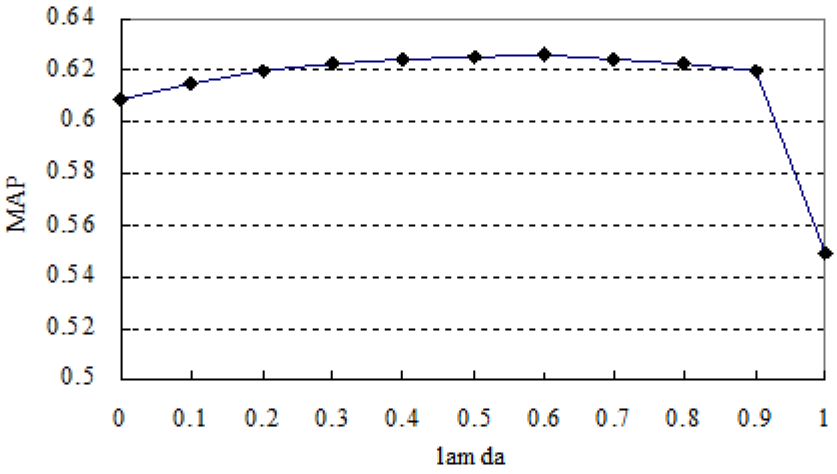


Figure 5. The effect of different λ for cluster-based co-occurrence sub-model

The weights in the linear combination (2) for the title-author, block-based, and neighbor-based co-occurrence sub-model were set 45, 1000, and 40 respectively, based on evaluations on the training set. At the first glance, the block-based co-occurrence sub-model would dominate. However, this is not always true, because in some cases the block-based co-occurrence sub-model may not be applicable, and the other two sub-models would influence the results.

Table 4. Our approach vs. baseline (TREC 2005)

	MAP	**R-precision**	**P@10**
Baseline	0.1632	0.2009	0.238
+ Relevance	0.2236	0.2478	0.320
+ Title-author	0.2255	0.2537	0.318
+ Block	0.2644	0.3001	0.372
+ Neighbor	0.2675	0.2943	0.374
+ Cluster (Two-Stage model)	0.2812	0.3199	0.372

Table 5. Our approach vs. top runs of TREC 2005

	MAP	**R-precision**	**P@10**
Two-Stage model	0.2812	0.3199	0.3720
THUENT0505	0.2749	0.3330	0.4520
MSRA054	0.2688	0.3192	0.3700

Table 4 shows the results obtained from the test data set of TREC 2005. We see that each of the proposed components can boost performance incrementally (not always significantly). We also tried to compare our results with the top runs at the enterprise search track of TREC 2005. The runs that we selected, namely THUENT0505 and MSRA054, were ranked as top 2. From Table 5, we can see that our approach can achieve comparable results to the top runs of TREC.

The results presented in Table 4 and 5 are not as good as those in Figure 4 and 5. This is because the former is from the test data set and the latter is from the training data set. We note that the training data set and the test data set in TREC 2005 differ largely. For example, the test set contains long queries like "Mobile Web Initiative Workshop Program Committee", while the training set does not. (In general long queries are difficult to handle in expert search.) Furthermore, the ground truth of several test queries is hard to be found from the document collection. For example, in TREC 2005, no participating system could find answers for the queries "WCAG reviewers", "AC Meeting attendees", and "MWI Device Descriptions".

5.4 Expert Search at W3C (TREC 2006)

In this experiment, we used the data set at TREC 2006. The document collection of W3C is same as that used in the evaluation of TREC 2005. The ground truth of TREC 2006 was developed by the participants of the expert search task. The evaluation set consists of 49 topics and their corresponding experts. In the experiment, we used all the 49 topics as the test set and used all the parameters selected in Section 5.3.

We adopted the window-based co-occurrence sub-model as the baseline method. Then we incrementally added to the baseline method the proposed components, namely relevance model, title-author co-occurrence sub-model, block-based co-occurrence sub-model, neighbor-based co-occurrence sub-model, and cluster-based sub-model.

Table 6. Our approach vs. baseline (TREC 2006)

	MAP	**R-precision**	**P@10**
Baseline	0.5102	0.5142	0.6265
+ Relevance	0.5335	0.5284	0.6184
+ Title-author	0.5346	0.5336	0.6265
+ Block	0.5652	0.5659	0.6531
+ Neighbor	0.5672	0.5706	0.6551
+ Cluster (Two-Stage model)	0.5684	0.5728	0.6673

Table 7. Our approach vs. top runs of TREC 2006

	MAP	**R-precision**	**P@10**
Two-Stage model	0.5684	0.5728	0.6673
kmiZhu1	0.6431	0.6242	0.7347
SJTU04	0.5947	0.5783	0.7041

Table 6 shows the results. Again, each proposed component can incrementally boost the performance. We also compared our results with those of the top two runs at TREC 2006: kmiZhu1 and SJTU04. From Table 7, we can see that our approach can achieve comparable results to the best performing systems. Note that both kmiZhu1 and SJTU04 employed the two-stage model. kmiZhu1 made use of the two stage model as well as other types of information such as document quality, document structure, and various sizes of co-occurrence windows (Zhu and Song, 2006). SJTU04 expanded the two-stage model by further considering expert matching quality, query matching quality, and document quality (Bao et al., 2006).

5.5 Expert Search at MSR

As we developed our method, we performed an additional experiment in the context of an industrial research lab, Microsoft Research (MSR). In this experiment, the expert candidates are the research staff of the lab. The document set is a crawl of the external web site of the lab. The crawl comprises 31,754 web pages, including group homepages, personal homepages, technical reports, and research papers.

We selected 32 research topics from the query log of Live Search[5] and then built a ground truth against the topics with the following steps. First, for each topic, we manually made a set of seed experts by looking at an existing expertise database of MSR and searching with Google Scholar[6], CiteSeer[7] and ACM DL[8]. We did so, because we intended to make the set of seed experts independent from the document collection (that is, to make the task harder). Then, we invited every researcher in the seed expert set to annotate the topics that he/she involves in. The invited researchers could vote on all the seed expert candidates, introduce more candidates that were not in the seed sets, or remove themselves from the list of candidates. Every time a researcher was newly introduced as an expert on a topic, we would send an invitation to include the researcher in the annotation. Via this viral process from person to person more researchers were introduced. At last, we obtained as the ground truth the final expert sets when the procedure converged. In the ground truth, each topic has 10 experts on average. There were in total 810 researchers involved in this viral process, which were viewed as expert candidates in the experiment. The persons who were nominated and not removed from the list of candidates on a certain topic were considered as experts.

In the experiment, we used all the 32 topics as the test set and used all the parameters obtained in Section 5.3.

Our main purpose in this experiment was to further verify the use of relevance model, since it is one of the major differences of our model from previous work. We also considered an alternate cluster-based co-occurrence sub-model. In this case, the cluster information was not obtained with a clustering procedure but from an existing database of organization chart of the lab. Table 8 illustrates the effectives of relevance model and cluster-based co-occurrence sub-model, for the researcher search.

Table 8. Our approach vs. baseline (MSR)

	MAP	**R-precision**	**P@10**
Baseline	0.5016	0.4594	0.4433
+ Relevance	0.5499	0.5402	0.4733
+ Cluster (Two-Stage model)	0.5573	0.5432	0.4733

The uses of the title-author sub-model, block-based sub-model, and neighbor-based sub-model to this dataset did not show gain. This indicates that useful information for expert search can vary according to data sets. Specifically, in the MSR data the title of a personal home page of a researcher is usually the name of the researcher, and thus the co-occurrence relationship between title and author can hardly be

[5] Live Search. http://www.live.com

[6] Google Scholar. http://scholar.google.com

[7] CiteSeer. http://citeseer.ist.psu.edu

[8] ACM Digital Library. http://portal.acm.org/dl.cfm

used in expert-expertise relation finding. Neighboring co-occurrence between topic and person is also not strong in the MSR data. Furthermore, the block tree structure in the TREC data (e.g., the structure in Figure 2, in which the *head* nodes contain topics and the *text block* nodes contain personal names) seldom appears in the MSR data. Despite this non-positive result on some sub-models, our core model with the incorporation of document relevance information is helpful on the MSR dataset.

6. CONCLUSION

In this paper, we have investigated the problem of expert search, mainly with the data of TREC 2005, TREC 2006, and MSR. We think that the key issue for the study on expert search is to find a unified and theoretically sound framework for introducing diverse evidence of expertise. This is the main focus of this work.

We have proposed a unified framework for expert search. The model, referred to as a two-stage model, can incorporate many types of association information for expert search. It includes relevancies between query terms and documents, co-occurrences between people and terms in bodies of documents, co-occurrences between people and terms in titles and authors, and co-occurrences between people and people in documents. All of them can be represented within sub-models of the two-stage model. The two-stage model has two advantages: it is theoretically sound and it is easy to incorporate new types of sub-models.

Our major findings include:

First, the two-stage model outperforms existing methods such as the profile-based method. It can achieve the best results reported in TREC.

Second, relevance of documents to queries and occurrences between people and terms in a number of ways are useful for expert search. .

Third, our approach generalizes to other enterprise datasets, yielding good results on a MSR dataset. This indicates that the success of two-stage models will not be restricted to the W3C test collection.

Fourth, useful information for expert search can change according to data set. For example, co-occurrence sub-models making use of document structure were more useful on TREC data than the MSR data.

The proposed approach is useful in answering the question “who in the organization is an expert on X?”, and has been proved successful at TREC. In future, the expert search problem might be thought of as part of a wider class of problem: entity search. These problems are of the form “which entities are associated with X?” The entities might be people, or they might be for example products, publications, groups of people or countries. In that case, the two stage model could be applied, as well as analogous and extended techniques relating to document structure.

REFERENCES

Balog, K, Azzopardi, L., and Rijke, M. Forma (2006). Formal models for expert finding in enterprise corpora. In Proceedings of the 29th Annual International ACM SIGIR Conference. pp. 43-50.

Balog, K. and Rijke, M. (2006). Finding Experts and their details in e-mail corpora. In Proceedings of the 15th International World Wide Web Conference.

Balog, K., Bogers, T., Azzopardi, L., Rijke, M., and Bosch, A. (2007). Broad Expertise Retrieval in Sparse Data Environments. In Proceedings of the 30th Annual International ACM SIGIR Conference.

Bao, S. Duan, H., Zhou, Q., Xiong, M., Cao, Y. and Yu, Y. (2006). Research on expert search at enterprise track of TREC 2006. In Proceedings of the Text REtrieval Conference 2006.

Campbell, C.S., Maglio, P., Cozzi, A. and Dom, B. (2003). Expertise identification using email communications. In Proceedings of ACM Twelfth Conference on Information and Knowledge Management.

Cao, Y., Liu, J., and Bao, S., and Li, H. (2005). Research on expert search at enterprise track of TREC 2005. In Proceedings of the Text REtrieval Conference 2005.

Craswell, N., Hawking, D., Vercoustre, A. M. and Wilkins, P. (2001). P@NOPTIC Expert: Searching for experts not just for documents. In Ausweb, 2001.

Dom, B., Eiron, I., Cozzi A. and Yi, Z. (2003). Graph-based ranking algorithms for e-mail expertise analysis. In Proceedings of the Eighth ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery.

ExpertNet: http://www.cvcp.ac.uk/expertnet.htm.

Fang, H., Zhou, L., Zhai, C.(2006). Language models for expert finding-UIUC TREC 2006 enterprise track experiments, In Proceedings of Text Retrieval Conference 2006.

Fu, Y., Xiang, R., Liu, Y., Zhang, M., Ma, S. (2007). A CDD-based formal model for expert finding. In Proceedings of ACM Sixteenth Conference on Information and Knowledge Management,

Hawking, D. (2004). Challenges in enterprise search. In Proceedings of the 15th Australasian Database Conference.

Hertzum, M. and Pejtersen, A. M. (2000). The information-seeking practices of engineers: Searching for documents as well as for people. Information Processing Management, 36(5):761–778.

Hu, Y., Li, H., Cao, Y., Meyerzon, D., Teng, L., and Zheng, Q. (2006), Automatic extraction of titles from general documents using machine learning. Information Processing and Management, 2006.

Hu, Y., Xin G., Song, R., Hu, G., Shi, S., Cao, Y., and Li, H. (2005). Title extraction from bodies of HTML documents and its application to web page retrieval. In Proceedings of the 28th Annual International ACM SIGIR Conference.

Kautz, H., Selman, B., and Milewski, A. (1996). Agent amplified communication. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, pp. 3-9.

Maconald, C. and Ounis, I.(2006). Voting for candidates: adapting data fusion techniques for an expert search task. In Proceedings of ACM Fifteenth Conference on Information and Knowledge Management, pp.387-396.

Macdonald, C. and Ounis, I. (2007). Expertise drift and query expansion in expert search. In Proceedings of ACM Sixteenth Conference on Information and Knowledge Management,

Maarek, Y. S. and Smadja F. Z. (1989). Full text indexing based on lexical chains – An application: software libraries. In Proceedings of the Twelfth Annual International ACM SIGIR Conference.

Mattox, D., Maybury, M., and Morey, D. (1999). Enterprise expert and knowledge discovery. In Proceedings of the HCI International '99 on Human-Computer Interaction: Communication, Cooperation, and Application Design pp. 303-307.

McDonald, D. W. and Ackerman, M. S. (1998). Just talk to me: A field study of expertise location. In Proceedings of the Twelfth ACM conference on Computer Supported Cooperative Work, pp. 315-324.

McDonald, D. W. and Ackerman, M. S. (2000). Expertise recommender: A flexible recommendation system and architecture. In Proceedings of the 14th ACM Conference on Computer Supported Cooperative Work, pp. 231-240.

Mockus, A. and Herbsleb, J.D. (2002). Expertise browser: A quantitative approach to identifying expertise. In Proceedings of the 24th International Conference on Software Engineering.

Ogilvie, P. and Callan, J. (2003). Combining document representations for known-item search. In Proceedings of the 26th Annual International ACM SIGIR Conference.

Petkova, D., and Croft, W. B. (2006). Hierarchical language models for expert finding in enterprise corpora. In Proceedings. of 18th IEEE International Conference on Tools with Artificial Intelligence, pp.599-608.

Ponte, J. and Croft, W. (1998). A language modeling approach to information retrieval. In Proceedings of the 21st Annual International ACM SIGIR Conference.

ProfNet: http://www.profnet.com/.

Serdyukov, P., Hiemstra, D., Fokkinga, M., and Apers, P.M.G. (2007). Generative Modeling of Persons and Documents for Expert Search. In Proceedings of the 30th Annual International ACM SIGIR Conference, pp. 827-828.

Sihn, W. and Heeren F. (2001). Xpertfinder - Expert finding within specified subject areas through analysis of e-mail communication. In Proceedings of the Sixth Annual Scientific conference on Web Technology,

Skillview: htttp://www.skillview.com

Streeter, L.A. and Lochbaum, K.E. (1988). An expert/expert locating system based on automatic representation of semantic structure. In Proceedings of the Fourth IEEE Conference on Artificial Intelligence Applications.

Yimam-seid, D. and Kobsa, A. (2002). Expert finding systems for organizations: Problem and domain analysis and the DEMOIR approach. Journal of Organizational Computing and Electronic Commerce, 2002.

Zhu, J. and Song, D. (2006). The Open University at TREC 2006 enterprise track expert search task. In Proceedings of the Text REtrieval Conference 2006.