A BELIEF NETWORK MODEL FOR EXPERT SEARCH

Craig Macdonald, Iadh Ounis
Department of Computing Science, University of Glasgow, Glasgow, G12 8QQ, UK
craigm@dcs.gla.ac.uk, ounis@dcs.gla.ac.uk

Keywords: Expert Search, Expertise Modelling, People finding, Bayesian Belief Networks

Abstract: Expert search is a task of growing importance in Enterprise settings. In a classical search setting, users normally require relevant documents to fulfil an information need. However, in Enterprise settings, users also have a need to identify the co-workers with relevant expertise to a topic area. An expert search engine assists users with their expertise need, by ranking candidate experts with respect to their predicted expertise about a topic of interest. This work presents a novel model for the expert search, based on a Bayesian belief network. We show how the proposed model can generate several different strategies for ranking candidates by their predicted expertise with respect to a query. The Bayesian belief network model for expert search proposed here is general, as it can be extended in the future to take into account various other types of evidence in the expert search task, such as the social aspect of expert search, where people work within groups and co-author publications.

1 INTRODUCTION

The expert search task is an Information Retrieval (IR) task which is of growing importance in recent years. In this task, the user's 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 user's query.

In our previous work, we showed that expert search can be viewed as a voting process (Macdonald and Ounis, 2006). In such a setting, a profile of documents is built for each candidate, to represent the candidates' expertise to the system. Then, in response to a user query, documents from the collection are ranked. This ranking of documents contains implicit evidence suggesting the candidates with relevant expertise that should be retrieved by the expert search system. For example, candidates associated with many highly ranked documents are more likely to have relevant expertise than candidates with no documents retrieved. These 'votes' from the documents can then be used to rank the candidates. In particular, (Macdonald and Ounis, 2006) proposed twelve voting techniques which defined ways in which a ranking of documents could be transformed into a ranking of candidates. These voting techniques are based on different sources of evidence about how candidates should be ranked with respect to a ranking of documents and the known associations between the documents and the candidates (i.e. the profile of each candidate).

In this work, we show that the expert search task can be modelled through the use of Bayesian belief networks. The novel use of a Bayesian network allows a good understanding of the basis of our Voting Model for expert search (Macdonald and Ounis, 2006), derived from probabilistic considerations. In particular, our network naturally models the complex dependencies between terms, documents and candidates in the Voting Model for expert search. To model these dependencies, the network is based on two sides: The candidate side of the network provides the links between the candidates and their associated profile documents; The query side of the network links the user query to the keywords it contains, and also the keywords to the documents which contain them.

The remainder of this paper is as follows: Section 2 details existing work on modelling the expert search task; Section 3 introduces the concept of a Bayesian network, and highlights previous applications of Bayesian networks in IR; Section 4 details the inference networks model we propose for expert search; Section 5 demonstrates an example expert search query using the Bayesian belief network, and provides some results of applying the equivalent voting techniques on recent expert search test collections; We provide concluding remarks about our Bayesian belief model for expert search and directions for future work in Section 6.

2 EXPERT SEARCH

Modern expert search systems for Enterprise settings work by using documents to form the profile of textual evidence for each candidate expert (Craswell et al., 2006). The candidate's profile represents the expertise of the candidate expert in the expert search system. This documentary evidence can take many forms, such as documents or emails authored by the candidates, or web pages visited by the candidate (see (Macdonald and Ounis, 2006) for an overview). In this work, the profile of a candidate is considered to be a set of documents associated with the candidate. These candidate profiles can then be used to rank candidates automatically in response to a query.

Among the first models for expert search, is that proposed by (Craswell et al., 2001), in which all documents in each candidate's profile are combined into 'virtual documents', which are then directly ranked in response to a user query. However, the contribution of each document in a profile cannot be measured individually, and as a result, this approach is less effective than other subsequent approaches.

The advent of the expert search task in TREC 2005 Enterprise track has stimulated research interest in expert search (Craswell et al., 2006; Soboroff et al., 2007). From this forum, there have been several approaches for expert search: Balog et al. proposed the use of language models in expert search (Balog et al., 2006) based on two formal models. Their first model is based on the virtual document approach (Craswell et al., 2001) described above. Their second model combines the evidence from the distinct documents in the candidates profiles. Their experimental results showed that the second model improves over the simpler first model. Relatedly, the probabilistic approach proposed by Cao et al. in (Cao et al., 2005) and the hierarchical language models proposed by (Petkova and Croft, 2006) use a more fine-grained approach with windowing of documents around candidate name occurrences.

Also successfully applied at TREC is the Voting Model for expert search proposed by Macdonald & Ounis in (Macdonald and Ounis, 2006), which considers the problem of expert search as a voting process. The *ranking of documents*, with respect to the query Q, denoted by $R(Q)$, is assumed to provide inherent evidence about a possible ranking of candidates. The ranking of candidates can then be modelled as a voting process, from the retrieved documents in $R(Q)$ to the profiles of candidates: Every time a document is retrieved and is associated with a candidate, then this is considered to be a vote for that candidate to have relevant expertise to Q. The ranking of the candidate profiles can then be determined by applying a voting technique that appropriately aggregates the votes of the documents. Twelve voting techniques for ranking experts were defined in (Macdonald and Ounis, 2006). Each of these voting techniques employ various sources of evidence derived from the ranking of documents, such as counting the number of documents associated with each candidate that are retrieved (number of votes), or the scores and/or ranks of the associated documents of each candidate (strength of votes).

The aim of this work is to produce a formal grounding for the voting techniques described in (Macdonald and Ounis, 2006), in a probabilistic framework. To this end, we use a Bayesian belief network to naturally combine the connections between a user query, the terms, the documents, and the candidates present in a collection. In particular, we show how several of the voting techniques from the Voting Model can be expressed by the proposed belief network. In the next section, we describe Bayesian networks, while in Section 4 we introduce our modelling of the expert search task.

3 BAYESIAN NETWORKS

Bayesian networks provide a graphical formalism for explicitly representing independencies among the variables of a joint probability distribution. This distribution is represented through a directed graph whose nodes represent the random variables of the distribution. In particular, a Bayesian network is a directed acyclic graph (DAG), where each node represents an event with either a discrete or a continuous set of outcomes, and whose edges encode conditional dependencies between those events. If there is an edge from node X_i to another node X_j, X_i is called a *parent* of X_j, X_j is a *child* of X_i, and moreover X_i is said to *cause* X_j. We denote the set of parents of a node X_i by $parents(X_i)$.

The fundamental principle of a Bayesian network is that known independencies among the random variables of a domain are declared explicitly and that a joint probability distribution is synthesised from the set of independencies. Furthermore, the inference process in a Bayesian network provides mechanisms, such as d-separation, to decide whether a set of nodes is independent of another set of nodes, given a set of evidence. For further details on Bayesian networks, we refer the reader to (Pearl, 1988).

In the network, the joint probability function is the product of the local probability distribution of each node, given its parent nodes:

$$P(X_1,...,X_n) = \prod_{i=1}^{n} P(X_i|parents(X_i)) \qquad (1)$$

Furthermore, if a node has no parents, i.e. it is a *root* node, its local probability distribution is unconditioned, otherwise it is conditional upon its parent nodes. A node X_i is conditionally independent of all nodes that it is not a *descendant* of (i.e. all the nodes from which there is no path to X_i).

The influence of $parents(X_i)$ on X_i (i.e. $P(X_i|parents(X_i))$) can be specified by any set of functions $F_i(X_i, parents(X_i))$ that satisfy

$$\sum_{\forall x_i} F_i(X_i, parents(X_i)) = 1 \qquad (2)$$

$$0 \leq F_i(X_i, parents(X_i)) \leq 1 \qquad (3)$$

This specification is complete and consistent because the product $\prod_{\forall i} F_i(X_i, parents(X_i))$ constitutes a joint probability distribution for the nodes in the network (Pearl, 1988; Ribeiro-Neto and Muntz, 1996).

While there have been many applications of graph-based formalisms applied in IR over the years, the use of Bayesian networks was initiated by Turtle and Croft. In particular, Turtle and Croft (Turtle and Croft, 1990; Turtle, 1991) proposed the inference network model for IR using Bayesian network formalisms. They showed that both the vector space model (Salton and Buckley, 1988) and Fuhr's model for retrieval with probabilistic indexing (RPI) (Fuhr, 1989) could be generated by their inference networks for IR. (Metzler and Croft, 2004) later extended the inference network model to the language modelling framework.

Similarly, (Ribeiro-Neto, 1995) discusses how the Boolean and probabilistic models are subsumed by his belief network model for IR. In his model the root nodes are terms, while, in contrast, the documents were the root nodes in Turtle's inference network model. Ribeiro-Neto further extended his belief network model by using it for combining link and content based Web evidences (Silva et al., 2000), and for integrating evidence from past queries (Ribeiro-Neto et al., 2000).

Other works using Bayesian networks include that of (Tsikrika and Lalmas, 2004) who also combined link and content-based evidence in a Web IR setting, as well as applications of Bayesian networks to other IR-related tasks such as document classification (Denoyer and Gallinari, 2004), question answering (Azari et al., 2004) and video retrieval (Graves and Lalmas, 2002).

The following section introduces our proposed Bayesian network model for expert search. Our model is inspired by the work of (Ribeiro-Neto and Muntz, 1996), but makes additional considerations for candidates in addition to the nodes for the query, terms and documents.

4 A BELIEF NETWORK FOR EXPERT SEARCH

In this paper, a belief network model for expert search is developed. Our work is founded on that of Ribeiro-Neto et al. in building belief networks for classical document IR retrieval (Ribeiro-Neto, 1995; Ribeiro-Neto and Muntz, 1996; Silva et al., 2000). This works extends the belief network model by adding a second stage that considers the ranking of candidates with respect to the query. The remainder of this section is separated into three stages: Firstly, we introduce the definitions that we use in this work; Secondly, we introduce the Bayesian belief network model for expert search, based on these definitions. Finally, we discuss how various expert search ranking strategies can be generated using this model.

4.1 Definitions

Let t be the number of indexed terms in the collection of documents, and k_i be a term. $U = k_1,...,k_t$ is the set of all terms. Moreover, let $u \subset U$ be a concept in U, composed of a set of terms of U. (Ribeiro-Neto and Muntz, 1996) views each index term as an elementary concept. A concept is a subset of U and can represent a document in the collection or a user query.

To each term k_i is associated a binary random variable which is also referred to as k_i. The random variable is set to 1 to indicate that k_i is a member of set u. Let $g_i(u)$ be the value of the variable k_i according to set u. The set u defines a concept in U as the subset formed by the indexes k_i for which $g_i(u) = 1$ (Wong and Yao, 1995; Ribeiro-Neto and Muntz, 1996).

Let N be the number of documents in the collection of documents. A document d in the collection is represented as a set of terms $d = \{k_1, k_2, ..., k_t\}$ where k_1 to k_t are binary random variables which define the terms that are present in the document.

If an index term k_j is used to describe the document d then $g_j(d) = 1$. Likewise, if the same index term also describes a user query q, then $g_j(q) = 1$.

The random variables (i.e. k_i) associated to the index terms are binary because this is the simplest possible representation for set membership. The set u defines a set in U as a subset formed by the terms k_i for which $g_i(u) = 1$. Thus there are 2^t possible subsets of terms in U.

We now extend these definitions of (Ribeiro-Neto and Muntz, 1996) to allow the modelling of candidates in the belief network model:

Let $V = d_1, ..., d_N$ be the set of all documents, which defines the sample space for the candidate side of the model. Let $v \subset V$ be a subset of V. A candidate c in the collection is represented to the system as a set of documents $c = \{d_1, d_2, ..., d_N\}$ where d_1 to d_N are binary random variables which define the documents that are associated to candidate c. The documents associated to each candidate form their expertise profile.

Let $h_i(v)$ be the value of the variable d_i according to set v. The set v defines a set in the space V as a subset formed by the documents d_i for which $h_i(v) = 1$. Moreover, let M be the number of candidates in the collection.

4.2 Network Model

In this section, we propose a Bayesian belief network model for Expert Search based on the definitions introduced above. Furthermore, we show that the voting techniques for ranking candidates according to their expertise to a query q can be reproduced by our belief network.

We model the user query q as a network node to which is associated a binary random variable (as in (Pearl, 1988)) which is also referred to as q. The query node is the child of all term nodes k_i which are contained in the query q.

A document d in the collection is modelled as a network node to which is associated a binary random variable which is also referred to as d. Analogously to the query, the document node d is a child of all term nodes k_i that are contained in the document d.

Each candidate c is modelled as a network node, which is linked to by the nodes of all the documents that are associated to the candidate, to form their expertise profile. Hence, a candidate c in the collection is specified as a subset of the documents in the space V which point to the candidate c, to represent their expertise to the system.

Figure 1 illustrates our belief network model for expert search. The index terms are independent binary random variables (the k_i variables) and hence are the root nodes of the network. Query q is pointed to by the index term nodes which compose the query concept. Documents are treated analogously to user queries, thus a document node d is pointed to by the index term nodes which compose the document. Similarly, a candidate node c is pointed to by the documents which are associated to the candidate.

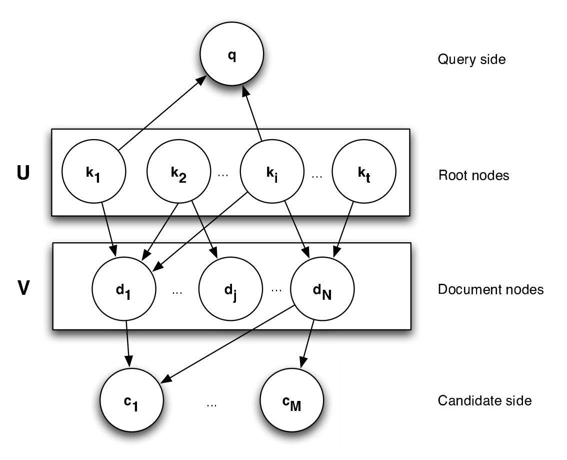

Figure 1: A Bayesian belief network model for expert search.

From Figure 1, it is clear by Equation (1) that the joint probability function of this network is:

$$p(k_1, ..k_t, q, d_1, .., d_N, c_1, .., c_M) = P(u) \cdot P(q|u) \cdot P(v|u) \cdot \prod_{j=1}^{M} P(c_j|v) \quad (4)$$

for some set of terms u and some set of documents v.

We now need to specify how to rank the candidates in the collection relative to their predicted expertise about a query q. We adopt $P(c_j|q)$ as the ranking of the candidate c_j with respect to the query q. Since the system has no prior knowledge of the probability that a concept u occurs in space U, we assume the unconditional probability of the root nodes, the term nodes, is uniform:

$$P(u) = \frac{1}{2^t} \quad (5)$$

To complete our belief network we need to specify the conditional probabilities $P(q|u)$, $P(v|u)$ and $P(c|v)$. Various specifications of these conditional probabilities lead to different ranking strategies for candidates.

4.3 Ranking Strategies for Expert Search

In our network of Figure 1, the similarity (or rank) of a candidate c_j with respect to a user query q

is computed by the conditional probability relationship $P(c_j|q)$. From the conditional probability definition, we can write $P(c_j|q) = \frac{P(c_j,q)}{P(q)}$. Since $P(q)$ is a constant for all candidates, this can be safely disregarded while ranking the candidates, and hence $P(c_j|q) \propto P(c_j,q)$, i.e., the rank assigned to a candidate c_j is directly proportional to $P(c_j,q)$. We can use the joint probability function of the network (Equation (4)) to calculate this, by summing over all nuisance variables (i.e. all variables except c_j and q):

$$\begin{aligned} P(c_j|q) &\propto \sum_{\forall v,k,c} p(k_1,..k_t,q,d_1,..,d_N,c_1,..,c_M) \\ &= \sum_{\forall v,k,c} P(u)\cdot P(q|u)\cdot P(v|u)\cdot P(c_j|v) \\ &\quad \cdot \prod_{c_i,i\neq j} P(c_i|v) \\ &= \sum_{\forall v,k} P(u)\cdot P(q|u)\cdot P(v|u)\cdot P(c_j|v) \quad (6) \end{aligned}$$

Note that in Equation (6) above, the other candidate nodes c_i are separate from c_j, and they are easily marginalised out $(P(c_i|v) + P(\overline{c_i}|v) = 1)$.

In (Macdonald and Ounis, 2006), the authors identified the existence of a relationship between the expertise of a candidate c in relation to a query q, and the extent to which a document d is about a query q, if there is a known relationship between the document and the candidate (for instance, the document was written by the candidate). The types of evidence demonstrating expertise of a candidate in the Voting Model are described in Section 2 above, namely the scores or ranks of associated documents (the strength of votes), and the number of associated documents ranked for the query (number of votes). Various voting techniques were proposed, (for instance Votes, CombMAX, and CombSUM) to combine a ranking of documents into a ranking of candidates.

In the following, we show that several of the voting techniques can be generated by the careful specification of $P(q|u)$, $P(v|u)$ and $P(c_j|v)$ to calculate $P(c_j|q)$. To ensure correctness, the specifications of $P(q|u)$, $P(v|u_q)$ and $P(c_j|v)$ are defined in accordance to Equations (2) & (3).

Firstly, we restrict the concept set of terms u being considered to that of the terms involved in query q, by the following specification of $P(q|u)$:

$$\begin{aligned} P(q|u) &= \begin{cases} 1 & if\, \forall k_i, g_i(q) = g_i(u) \\ 0 & otherwise \end{cases} \quad (7) \\ P(\overline{q}|u) &= 1 - P(q|u) \quad (8) \end{aligned}$$

In this case, $P(q|u)$ is 1 iff $u = q$, and 0 otherwise (i.e. sets q and u contain exactly the same terms activated). We refer to the subset of documents $u = q$ as u_q. Then Equation (6) reduces to $P(c_j|q) \propto \sum_v P(c_j|v)\cdot P(u_q)\cdot P(v|u_q)$.

Next, we restrict the set of documents v being considered for the ranking of candidates to those actually ranked by query q, which we denote v_q. In particular, we adopt $P(d_i|u_q)$ as the relevance score of document d_i with respect to a set of terms u_q, and use this to determine the set of retrieved document v_q. Set v_q is then equivalent to the document ranking $R(Q)$ discussed in the Voting Model for expert search. We restrict v to v_q as follows:

$$\begin{aligned} P(v|u_q) &= \begin{cases} 1 & if\, \forall d_i,\, h_i(v) = \begin{cases} 1 & if\, P(d_i|u_q) > 0 \\ 0 & otherwise \end{cases} \\ 0 & otherwise \end{cases} \quad (9) \\ P(\overline{v}|u_q) &= 1 - P(v|u_q) \quad (10) \end{aligned}$$

Here, we see $P(d_i|u_q)$ as the relevance score of document d_i to the set of query terms q_u, which can be calculated using any probabilistic retrieval model (for instance language modelling (Hiemstra, 2001)). Note that we only consider a constant number of the top-ranked documents (as ranked by $P(d_i|u_q)$) as the set v_u[1]. By this restriction of v to v_q, the last summation from Equation (6) is removed, and it reduces further to $P(c_j|q) \propto P(u_q)\cdot P(c_j|v_q)$.

Since u_q is a set of terms, by Equation (5), the probability $P(u_q)$ is a constant, therefore candidates are ranked by $P(c_j|q) = K\cdot P(c_j|v_q)$ where K is a constant, and v_q is the set of documents ranked for the query q by a given approach to generate $P(d|u_q)$. We now propose several definitions for $P(c|v_q)$, which determine a ranking of candidates with respect to a query, given an input set of documents v_q. These are based on the equivalent voting techniques from (Macdonald and Ounis, 2006).

- **Votes:** In the Votes voting technique (Macdonald and Ounis, 2006), which is based on the number of votes evidence, the predicted expertise of a candidate is equal to the number of documents which were retrieved by the query q. The Votes technique can be represented in the belief network model as:

[1] Some probabilistic retrieval models (for instance Hiemstra's language models using Jelink-Mercer smoothing (Hiemstra, 2001)) do not assign a non-zero probability to a document which does not contain any of the query terms, and instead give a default value. By taking only the top-ranked documents, we prevent documents not matching any query terms from appearing in v_q. Indeed, the number of top-ranked documents can alternatively be considered a constant.

$$P_{Votes}(c_j|v_q) = \frac{\sum_{\forall d_i} h_i(v_q) \cdot h_i(c_j)}{\sum_{\forall c'} \sum_{\forall d_i} h_i(c')} \quad (11)$$

$$P_{Votes}(\overline{c_j}|v_q) = 1 - P_{Votes}(c_j|v_q) \quad (12)$$

In this definition, our belief in the candidate c_j given the set of documents v is dependent on the number of documents in v_q that are associated with c_j. To convert this into a probability, in the range $(0,1)$, we normalise this by the number of total votes present for any candidate in the collection, which has no effect in the ranking $P(c_j|v_q)$. Potentially, $P_{Votes}(c_j|v_q) = 1$ if the candidate was the only candidate in the collection, they were associated to all documents in the collection, and all documents were retrieved in v_q.

- **CombMAX:** In the CombMAX voting technique (Macdonald and Ounis, 2006), candidates are ranked by their strongest vote from the document ranking. The intuition behind this voting technique is that a candidates who has written (for instance) a document that is very close to the required topic area (i.e. the user query), is more likely to be an expert in the topic area than a candidate who has written some documents that are marginally about the topic area. This expertise evidence is the strongest votes for each candidate. We represent the CombMAX voting technique in the belief network model as follows:

$$P_{CombMAX}(c_j|v_q) = \frac{max_{\forall d_i}\{h_i(v_q) \cdot h_i(c_j) \cdot P(d_i|u_q)\}}{\sum_{\forall d_i} h_i(v_q)} \quad (13)$$

$$P_{CombMAX}(\overline{c_j}|v) = 1 - P_{CombMAX}(c_j|v_q) \quad (14)$$

The above definition for $P_{CombMAX}(c_j|v_q)$ is a valid probability distribution, as $P_{CombMAX}(c_j|v) = 1$ iff v_q contained only a single document and this document d had $P(d|u_q) = 1$. Under a probabilistic document retrieval model, this would only happen if d was the only document in the collection, and the query q contained all the terms of d.

- **CombSUM:** In the CombSUM voting technique (Macdonald and Ounis, 2006), candidates are ranked by the sum of the document relevance scores that are associated with the candidate. Again, this technique can be modelled in the Bayesian belief network, as follows:

$$P_{CombSUM}(c_j|v_q) = \frac{1}{M}\sum_{\forall d_i} h_i(v_q) \cdot h_i(c) \cdot P(d_i|u_q) \quad (15)$$

$$P_{CombSUM}(\overline{c_j}|v_q) = 1 - P_{CombSUM}(c_j|u_q) \quad (16)$$

The normalisation by the number of candidates in the collection ($\frac{1}{M}$)) is necessary so that $0 \leq \sum_{\forall c_j} P_{CombSUM}(c_j|v_q) \leq 1$. This is required as in a probabilistic retrieval model, $\sum_{\forall d} P(d|u) \leq 1$, hence the maximal is $P(c_j|v) = \frac{1}{M}$ if a given candidate was associated to every document in the collection, and all documents were ranked in v_q.

It should be noted that this formalisation of $P(c_j|v_q)$ is analogous to Model 2 of (Balog et al., 2006), as well as the models proposed by (Cao et al., 2005) and (Petkova and Croft, 2006), who all use a marginalisation to remove d from $P(c|d,q)$.

The above are three definitions of $P(c_j|v_q)$ show that three voting techniques from the Voting Model can be completely represented using our proposed Bayesian network model for expert search. Using other definitions of $P(c_j|v_q)$ and $P(v_q|u_q)$, other voting techniques defined in (Macdonald and Ounis, 2006) could be modelled. In the following section, we give an illustrative example demonstrating the use of the proposed Bayesian belief network for expert search in the processing of a query.

5 ILLUSTRATIVE EXAMPLE & EXPERIMENTS

This section presents an example belief network and shows how a query is evaluated to produce a ranking of candidates. Moreover, a selection of results are presented using the equivalent voting techniques applied on recent expert search test collections from the TREC Enterprise track.

The example belief network shown in Figure 2 shows three documents (each containing only a few terms each) and two candidates. In particular, document d_1 contains the terms “IR”, “stemming” and “tutorial”; d_2 contains the term “IR” only; and d_3 contains the terms “databases” and “tutorial”. In terms of candidate profiles, candidate c_1 is associated to documents d_1, d_2 and d_3, while candidate c_2 is associated to documents d_2 and d_3. In this case, the query contains only the term “IR”, hence we are looking to rank experts by their predicted expertise about the topic “IR”.

Our experimental setup is as follows: we use the language modelling framework as a probabilistic model with which we rank documents by $P(d|u_q)$. In the language modelling framework, documents are normally ranked by $P(d|q)$. In this case, we replace q by u_q without loss, as both are a set of terms

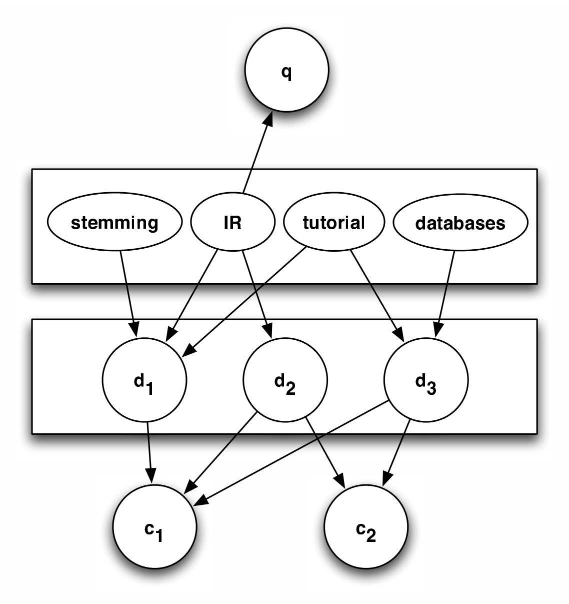

Figure 2: A simple example Bayesian Belief network model in an expert search setting.

representing a query. Then $P(d|u_q)$ is calculated using Bayes rule:

$$P(d|u_q) = \frac{P(u_q|d) \cdot P(d)}{P(u_q)} \tag{17}$$

As $P(u_q)$ does not affect the ranking $P(d|u_q)$, and we assume a uniform document prior $P(d) = \frac{1}{N}$, then

$$\begin{aligned} P(d|u_q) &\propto P(u_q|d) \\ &\propto \prod_i (\lambda \frac{tf_{q_i,d}}{\|d\|} + (1-\lambda)\frac{cf_{q_i}}{\|C\|})^{qf_i} \end{aligned} \tag{18}$$

where $tf_{q_i,d}$ is the frequency of the query term in document d, $\|d\|$ is the number of tokens in document d, cf_{q_i} is the term frequency of the query term in the entire collection, and $\|C\|$ is the number of tokens in the entire collection. qf_i is the frequency of the term in the query. λ is a parameter that controls the smoothing (Zhai and Lafferty, 2001), for which we apply a default value of $\lambda = 0.15$ (Hiemstra, 2001).

Hence, from the network in Figure 2 the following probabilities arise:

$$\begin{aligned} P(u) = (\frac{1}{2^4}) &= 0.0625 \\ p(d_1|u_q) = 0.15 \cdot \frac{1}{3} + 0.85 \cdot \frac{2}{6} &= 0.333 \\ p(d_2|u_q) = 0.15 \cdot \frac{1}{1} + 0.85 \cdot \frac{2}{6} &= 0.433 \\ p(d_3|u_q) = 0.15 \cdot \frac{0}{3} + 0.85 \cdot \frac{2}{6} &= 0.283 \end{aligned}$$

Recall that the set u_q is a set of terms in U for which only the query terms are active. In this example, only the node for the term "IR" is active. Moreover, we only consider the top 2 documents ranked by $P(d_i|u_q)$. This ensures that the set of documents v_q only contains the documents that contain the query terms in u_q (as per the footnote in Section 4.3). Hence, in this example, v_q contains only documents d_1 and d_2 as active.

Using the Votes definition for $P(c_j|v_q)$, the conditional probabilities would be as follows:

$$\begin{aligned} P_{Votes}(c_1|v_q) &= 0.4 \\ P_{Votes}(c_2|v_q) &= 0.2 \end{aligned}$$

This gives a ranking of $c_1 <_{rank} c_2$ (i.e. c_1 ranked first in the ranking), because c_1 achieves two votes, while candidate c_2 achieves only one vote.

Using the CombMAX definition for $P(c_j|v)$, both candidates are ranked equally ($c_1 =_{rank} c_2$), as both candidates are associated to the highest voting document d_2:

$$\begin{aligned} P_{CombMAX}(c_1|v_q) &= 0.433 \\ P_{CombMAX}(c_2|v_q) &= 0.433 \end{aligned}$$

Finally, using the CombSUM definition for $P(c_j|v_q)$, the following probabilities would be calculated:

$$\begin{aligned} P_{CombSUM}(c_1|v_q) &= 0.256 \\ P_{CombSUM}(c_2|v_q) &= 0.144 \end{aligned}$$

which gives a ranking $c_1 <_{rank} c_2$.

This example query illustrates the use of the belief network model for expert search. However, this setting is extremely simple, with documents containing only a few terms, and only two candidates. For comparison purposes, the TREC W3C collection, which is the standard test collection used in the expert search tasks TREC 2005 and TREC 2006 Enterprise tracks, contains 331,037 documents, 1024 candidates, 874,369 unique terms, and 997,241 associations between candidates and documents (this last number is dependent on how the association between candidates and documents is performed). However, the example given in this section is representative as it demonstrates how the belief network can be used to infer a ranking of candidates with respect to a query, and according to a ranking strategy.

It is of note that there is no need to directly evaluate the retrieval accuracy of the belief network model proposed in this work, because the retrieval strategies for ranking candidate experts using the Voting Model have already been thoroughly evaluated in the context of the expert search tasks of the TREC 2005, TREC 2006 and TREC 2007 Enterprise tracks. Moreover,

Voting Technique	TREC 2005		TREC 2006		TREC 2007	
	MAP	P@10	MAP	P@10	MAP	P@10
Year Median	0.1402	-	0.3412	-	0.2468	-
Votes	0.1548	0.2620	0.4883	0.6163	0.2188	0.1000
CombMAX	**0.1983**	0.2620	0.4936	0.5959	**0.3406**	0.1224
CombSUM	0.1672	**0.2720**	**0.5210**	**0.6367**	0.3076	0.1265

Table 1: Results applying three voting techniques on the TREC Enterprise track expert search tasks. Evaluation measures are Mean Average Precision (MAP) and Precision at 10 (P@10). Year Median is the mean of the per-topic median retrieval system. (The median results of P@10 were not provided.) For all years, the voting techniques perform markedly above the median MAP of that year.

we do not have to evaluate the expert search strategies proposed within the belief network model, as they are complete and sound with respect to the voting techniques on which they are based. However, we provide some experimental results here showing the retrieval performance of the equivalent voting techniques. In particular, we index the collections using the Terrier IR platform (Ounis et al., 2006), removing standard stopwords and applying the first two steps of Porter's English stemmer. To identify candidate document associations, we look for documents containing the occurrences of candidates names. Document are ranked (i.e. $P(d_i|u_q)$ is calculated) using Hiemstra's language modelling, as above. Table 1 shows the results for the three TREC tracks. For more evaluation experiments using the voting techniques, the reader is referred to (Lioma et al., 2007; Macdonald and Ounis, 2006; Macdonald and Ounis, 2007; Hannah et al., 2007).

6 CONCLUSIONS

This paper proposed a Bayesian belief network model for the expert search task which allows many different ranking strategies for ranking candidate with respect to a query. Expert search is a more complex task than classical document retrieval, with an additional layer of objects on top of the normal documents. By modelling expert search using a model based on a graphical framework, the dependencies and independencies within the model are clear and easily interpreted, and moreover are derived from probabilistic considerations. While the probabilistic framework proposed can create one ranking strategy that is similar to the models of (Balog et al., 2006), (Cao et al., 2005) and (Petkova and Croft, 2006), this framework is more general, allowing for additional strategies for ranking candidates, such as the Votes and CombMAX voting techniques (Macdonald and Ounis, 2006). Moreover, it is feasible that the probabilistic techniques devised within the framework of the Belief network model can be used to generate previously unknown voting techniques.

It is of note that while this paper does not contain a detailed evaluation of the proposed belief network model using a test collection for expert search, this is not required, as the model produces complete and sound representations of existing voting techniques, which have been evaluated thoroughly and are are well understood. However, in Table 1, we provide some experimental results comparing the voting technique equivalents of the belief network model to current expert search test collections.

Furthermore, the belief network model can be applied to other tasks that involve the ranking of aggregates. Consider a task from the Web IR setting: a user wishes to find blogs that have regularly blogged about a topic, so that the user can subscribe to their feed and read the blog in the future (Macdonald et al., 2007). This task is in fact a large-scale example of expert search on the Blogosphere, and as such, the proposed belief network model and the equivalent voting techniques can be applied in this task (Hannah et al., 2007).

This belief network model for expert search also opens up more facets of research within the expert search task. For instance, using inference, we believe that the proposed model can identify, given a set of input experts, the similar candidates to the input set. Moreover, if a candidate has manually provided some keywords about their interests, then it is possible to integrate these into the model by extending the belief network using links from terms to candidates.

Our future work on our belief network model is focused in two directions: Firstly, we propose to extend the model to take into account non-binary memberships of documents to candidate profiles. This is motivated by our belief that it is likely that one type of document is more likely to be a good indicator of expertise than another type of documents - for instance, a document definitely written by a candidate is probably a better expertise indicator than a document which simply mentions the candidate's name; Secondly, we intend to take into account priors on experts and links between experts. For instance, it may be known that a group of people work in the same team. It is possible that if some members of a team are deemed to have relevant expertise, then the rest of the team may also have relevant expertise. In contrast, a candidate prior might be 'age' of a candidate - if there are many candidates with predicted expertise about a topic area, then the oldest candidate, by virtue of experience, may have more expertise than the younger candidates. Integrating all these sources of evidence into the belief network model will allow it to be applied more generally and perhaps more effectively to the expert search task. We are currently experimenting with these expansions, and will be reporting in other venues.

REFERENCES

Azari, D., Horvitz, E., Dumais, S., and Brill, E. (2004). Actions, answers, and uncertainty: a decision-making perspective on Web-based question answering. *Inf. Process. Manage.*, 40(5):849–868.

Balog, K., Azzopardi, L., and de Rijke, M. (2006). Formal Models for Expert Finding in Enterprise corpora. In *Proceedings of ACM SIGIR 2006*, pages 43–50.

Cao, Y., Li, H., Liu, J., and Bao, S. (2005). Research on Expert search at Enterprise track of TREC 2005. In *Proceedings of TREC-2005*.

Craswell, N., de Vries, A. P., and Soboroff, I. (2006). Overview of the TREC-2005 Enterprise Track. In *Proceedings TREC-2005*, pages 199–204.

Craswell, N., Hawking, D., Vercoustre, A.-M., and Wilkins, P. (2001). Panoptic expert: Searching for Experts not just for Documents. In *Ausweb Poster Proceedings*.

Denoyer, L. and Gallinari, P. (2004). Bayesian Network Model for semi-structured Document Classification. *Inf. Process. Manage.*, 40(5):807–827.

Fuhr, N. (1989). Models for retrieval with probabilistic indexing. *Inf. Process. and Manage.*, 25(1):55–72.

Graves, A. and Lalmas, M. (2002). Video retrieval using an mpeg-7 based inference network. In *Proceedings of ACM SIGIR 2002*, pages 339–346.

Hannah, D., Macdonald C., He B., Peng, J., and Ounis, I. (2007). University of Glasgow at TREC 2007: Experiments in Blog and Enterprise Tracks with Terrier. To appear in *Proceedings of TREC 2007*.

Hiemstra, D. (2001). *Using Language Models for Information Retrieval*. PhD thesis, Univ. of Twente, NL.

Lioma, C., Macdonald, C., Plachouras, V., Peng, J., He, B., and Ounis, I. (2007). University of Glasgow at TREC 2006: Experiments in Terabyte and Enterprise Tracks with Terrier. In *Proceedings of TREC 2006*.

Macdonald, C. and Ounis, I. (2006). Voting for Candidates: Adapting data fusion techniques for an Expert Search task. In *Proceedings of ACM CIKM 2006*.

Macdonald, C. and Ounis, I. (2007). Using Relevance Feedback in Expert Search. In *Proceedings of ECIR 2007*, Lecture Notes in Computer Science. Springer.

Macdonald, C., Ounis, I., and Soboroff I. (2007) Overview of the TREC 2007 Blog Track. To appear in *Proceedings of TREC 2007*.

Metzler, D. and Croft, W. B. (2004). Combining the Language Model and Inference network approaches to retrieval. *Inf. Process. Manage.*, 40(5):735–750.

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

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., 2nd ed.

Petkova, D. and Croft, W. B. (2006). Hierarchical Language models for Expert Finding in Enterprise corpora. In *Proceedings of ICTAI 2006*, pages 599–606.

Ribeiro-Neto, B. (1995). *Approximate Answers in Intelligent Systems*. PhD thesis, Univ. of California, LA.

Ribeiro-Neto, B. and Muntz, R. (1996). A Belief Network Model for IR. In *Proceedings of ACM SIGIR 1996*, pages 253–260.

Ribeiro-Neto, B., lmerio, S., and Muntz, R. (2000). Bayesian Network Models for IR. In Crestani, F. and Pasi, G., editors, *Soft Computing in Information Retrieval: techniques and applications*, pages 259–291. Physica Verlag, Germany.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523.

Silva, I., Ribeiro-Neto, B., Calado, P., Moura, E., and Ziviani, N. (2000). Link-based and Content-based Evidential Information in a Belief Network Model. In *Proceedings of ACM SIGIR 2000*, pages 96–103.

Soboroff, I., de Vries, A. P., and Craswell, N. (2007). Overview of the TREC-2006 Enterprise Track. In *Proceedings of TREC 2006*.

Tsikrika, T. and Lalmas, M. (2004). Combining Evidence for Web Retrieval using the Inference Network Model: an Experimental Study. *Inf. Process. Manage.*, 40(5):751–772.

Turtle, H. and Croft, W. B. (1990). Inference Networks for Document Retrieval. In *Proceedings of ACM SIGIR 1990*, pages 1–24.

Turtle, H. R. (1991). *Inference Networks for Document Retrieval*. PhD thesis, Univ. of Massachusetts, MA.

Wong, S. K. M. and Yao, Y. Y. (1995). On Modeling Information Retrieval with Probabilistic Inference. *ACM Trans. Inf. Syst.*, 13(1):38–68.

Zhai, C. and Lafferty, J. (2001). A study of Smoothing Methods for Language Models applied to ad hoc Information Retrieval. In *Proceedings of ACM SIGIR 2001*, pages 334–342.