Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

A language modeling framework for expert finding

Krisztian Balog [a], Leif Azzopardi [b], Maarten de Rijke [a,*]

[a] ISLA, University of Amsterdam, ISLA, Kruislaan 403, 1098 Amsterdam, Netherlands
[b] Department of Computing Science, University of Glasgow, 18 Lilybank Gardens, Glasgow, G12 8QQ, United Kingdom

ARTICLE INFO

Article history:
Received 30 December 2007
Received in revised form 11 June 2008
Accepted 13 June 2008
Available online xxxx

Keywords:
Expert finding
Language modeling
Expertise search
Intranet search

ABSTRACT

Statistical language models have been successfully applied to many information retrieval tasks, including expert finding: the process of identifying experts given a particular topic. In this paper, we introduce and detail language modeling approaches that integrate the representation, association and search of experts using various textual data sources into a generative probabilistic framework. This provides a simple, intuitive, and extensible theoretical framework to underpin research into expertise search. To demonstrate the flexibility of the framework, two search strategies to find experts are modeled that incorporate different types of evidence extracted from the data, before being extended to also incorporate co-occurrence information. The models proposed are evaluated in the context of enterprise search systems within an intranet environment, where it is reasonable to assume that the list of experts is known, and that data to be mined is publicly accessible. Our experiments show that excellent performance can be achieved by using these models in such environments, and that this theoretical and empirical work paves the way for future principled extensions.

1. Introduction

Human resources are a valuable asset to an organization because they possess a range of knowledge and expertise which can benefit the organization. However, ensuring that this expertise and knowledge is accessed and utilized is a major challenge (Maybury, 2006). Expertise is rare and difficult to quantify, experts vary in experience, and their expertise is continually changing. Within an organization experts may be dispersed geographically, functionally, or structurally, depending on the size and nature of the organization. For instance, an employee in a different division, while only down the hall, may never be utilized by a project team in need of his or her expertise. Knowledge provided by experts can help a project team support collaboration between individuals no matter where the experts are situated and prevent investment in unnecessary effort, or costly decisions. To help address the challenges involved current research is now being directed at the problem through the TREC initiative, where the focus has been on one of the main tasks in locating expertise: *expert finding*, that is, finding experts given a particular topic (Yimam-Seid & Kobsa, 2003).

The need to find experts may arise for many reasons; an employee may want to obtain some background on a project and find out why particular decisions were made without having to trawl through documentation (if there is any), or they may need particular skills for their current project. Alternatively, they may require a highly trained specialist to consult about a specific problem; see (Davenport & Prusak, 1998) for further examples. Expert finding is also particularly useful when individuals are new to an organization and need advice and assistance, or when individuals are within a very large and/or distributed organization. By identifying the relevant experts costs may be reduced and better solutions could be obtained. In short, facilitating collaborations through expert finding applications is a fundamental part of ensuring that the expertise within an organization is effectively utilized.

[*] Corresponding author.
E-mail addresses: kbalog@science.uva.nl (K. Balog), leif@dcs.gla.ac.uk (L. Azzopardi), mdr@science.uva.nl (M. de Rijke).

0306-4573/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2008.06.003

Computer systems that augment the process of finding the right expert for a given problem within an organization are becoming more feasible, largely due to the widespread adoption of technology in organizations coupled with the massive amounts of online data available within the organization. Indeed, an organization's internal and external websites, e-mail, database records, agendas, memos, logs, blogs, and address books are all electronic sources of information which connect employees and topics within the organization. These sources provide valuable information about the employee which can be utilized for the purpose of expert search. In order to perform expertise retrieval tasks such as expert finding, a list of candidate experts (employees, for instance) needs to be identified or obtained. This could be performed through using a named entity recognition system, or extracted from current records of employees. Then, the data is mined to extract associations between documents and candidates. These associations can be used to build representations of the candidates' expertise areas to support expert finding, and other tasks such as expert profiling (Balog & de Rijke, 2007a). These tasks can be seen as two sides of the same coin, where expert finding is the task of finding experts given a topic describing the expertise is required, and expert profiling is the task of identifying the topics for which a candidate is an expert.

In this paper we describe the application of probabilistic generative models, specifically statistical language models, to address the expert finding task. In recent years, language modeling approaches to information retrieval have attracted a lot of attention (Hiemstra, 2001; Ponte & Croft, 1998; Zhai & Lafferty, 2001). These models are very attractive because of their foundations in statistical theory, the great deal of complementary work on language modeling in speech recognition and natural language processing, and the fact that very simple language modeling applied to retrieval performs very well empirically. The basic idea underlying these approaches is to estimate a language model for each document, and then rank documents by the likelihood of the query according to the estimated language model, i.e., "what is the probability of seeing this query given this document?" To model the process of expert search, we adapt this process in two ways; the first uses the associations between people and documents to build a candidate model and match the topic against this model, and the second matches the topic against the documents and then uses the associations to amass evidence for a candidate's expertise. These two approaches represent the main search strategies employed for this task.

The main contribution of this paper is the introduction of a general probabilistic framework for modeling the expert finding task in a principled manner. Using the probabilistic generative framework, we demonstrate how these models can be further extended in a transparent fashion to incorporate the strength of association between candidates and topic terms, along with other forms of evidence. The models are then empirically validated on TREC test collections for expert finding. We demonstrate that these two approaches deliver state-of-the art performance on the expert finding task, and address the following research questions concerning them:

- How do the baseline models (without co-occurrence information) for expert finding compare in terms of effectiveness?
- How can these models be extended to incorporate co-occurrence information between a topic term and a candidate? We extend the baseline models with term-candidate associations, which are not based on entire documents like in the baseline models, but on the proximity of the term given the candidate in the document.
- How do these extended models compare in terms of effectiveness, and how do the different window sizes (used for determining proximity) affect effectiveness?
- How does the strength of the associations between a document and a candidate affect performance? These associations play a very important part in the expert finding models as they determine how representative the text is of a candidate's expertise.
- Finally, how sensitive are the models to the parameter settings in terms of effectiveness? Since there are a number of parameters that need to be estimated, we check how robust the performance is to their change.

While the framework can be extended in many ways, our aim is not to explore all the possibilities, but rather to show how it can be extended and empirically explore this in detail to convincingly demonstrate the extension. Further, we also wish to maintain the generality of the approaches. While the task we address is in the context of expert search, our models do not embody any specific knowledge about what it means to be an "expert." Generally, a co-occurrence of a (reference to a) person with the topic terms in the same context is assumed to be evidence to suggest "expertise." Thus, our approach is very general and can also be applied to search for other named entities such as places, events, and organizations, where the task at hand can be modeled in terms of associations between topics and entities.

The remainder of the paper is organized as follows. In Section 2 we discuss related work on expertise search. Section 3 is devoted to a detailed description of the task that we consider in this paper—expert finding—, and in Section 4 we detail our models for addressing this task. A key component of the models, document-candidate associations, is discussed in Section 5. After detailing our experimental setup in Section 6, we present an experimental evaluation of our models in Section 7. We discuss and analyze our findings in Section 8 and conclude in Section 9.

2. Related work

The need for managing the expertise of employees has been identified by the knowledge management field (Davenport & Prusak, 1998), where early approaches were mainly focused on how to unify disparate and dissimilar databases of the organization into a data warehouse that can easily be mined. Most of this early work was performed by the Knowledge Management and Computer Supported Cooperative Work community, usually called yellow pages, people-finding systems or expertise-management (ECSCW'99 Workshop, 1999; Yimam, 1996). These tools relied on employees to self-assess their skill against predefined set of keywords. However, the task of creating and updating profiles is laborious and consequently rarely performed. Even with initial profiles created, they soon become antiquated and no longer reflect the expertise of an employee accrued through his or her employment. Consequently, the need for intelligent technologies that can enhance the process of initializing and updating profiles of expert finding was recognized repeatedly; see e.g., (Becerra-Fernandez, 2000).

Yimam-Seid and Kobsa (2003) provide an overview of early automatic expertise finding systems. Many early systems tended to focus on specific document genres only, such as email (Campbell, Maglio, Cozzi, & Dom, 2003) or software and software documentation (Mockus & Herbsleb, 2002) to build profiles and find experts. However, the limitations of such applications are apparent, and so there has been increased interest (in both academia and industry) in systems that (1) index and mine published intranet documents along with other heterogeneous sources of evidence which are accessible within the organization, and (2) enable the search of all kinds of expertise within the organization (and are not restricted to one domain). One of the first published approaches to overcome these limitations was the P@noptic system (Craswell, Hawking, Vercoustre, & Wilkins, 2001). This system built representations of each candidate by concatenating all the documents within the organization associated with that candidate. When a query is submitted to the system, it is matched against these representations, as if it were a document retrieval system. Candidates are then ranked according their similarity with the query.

Given the feasibility of expertise search on heterogenous collections, the task of expert finding has received a significant amount of attention. This is due in part to the launch of an expert finding task as part of the annual enterprize track at the Text REtrieval Conference (TREC) in 2005 (TREC, 2005). TREC has provided a common platform for researchers to empirically assess methods and techniques devised for expert finding. In the 2005 and 2006 tracks the following scenario is presented: Given a crawl of the World Wide Web Consortium's web site consisting of emails, lists, personal homepages, etc, a list of candidate experts and a set of topics, the task is to find experts for each of these topics (Craswell, Vries, & Soboroff, 2006; Soboroff, de Vries, & Craswell, 2007).

At TREC, it emerged that there are two principal approaches to expert finding—or rather, to capturing the association between a candidate expert and an area of expertise—, which have been first formalized and extensively compared by Balog, Azzopardi, and de Rijke (2006), and are called *candidate* and *document* models; in this paper, these models are referred to as *Model 1* and *Model 2*, respectively—see Section 4. Model 1's candidate-based approach is also referred to as profile-based method in Fang and Zhai (2007) or query-independent approach in Petkova and Croft (2006). These approaches build a textual (usually term-based) representation of candidate experts, and rank them based on query/topic, using traditional ad-hoc retrieval models. These approaches are similar to the P@noptic system (Craswell et al., 2001). The other type of approach, document models, are also referred to as query-dependent approaches in Petkova and Croft (2006). Here, the idea is to first find documents which are relevant to the topic, and then locate the associated experts. Thus, Model 2 attempts to mimic the process one might undertake to find experts using a document retrieval system. Nearly all systems that took part in the 2005 and 2006 editions of the Expert Finding task at TREC implemented (variations on) one of these two approaches. In this paper, we formalize the two approaches using generative probabilistic models. We focus exclusively on these models because they provide a solid theoretical basis upon which to extend and develop theses approaches.

Building on either candidate or document models, further refinements to estimating the association of a candidate with the topic of expertise are possible. For example, instead of capturing the associations at the document level, they may be estimated at the paragraph or snippet level. In this paper, we model both approaches, with document level associations, and then extend each model to handle snippet level associations. The generative probabilistic framework naturally lends itself to such extensions, and to also include other forms of evidence, such as document and candidate evidence through the use of priors (Fang & Zhai, 2007), the document structure (Zhu, Song, Ruger, Eisenstadt, & Motta, 2007), and the use of hierarchical, organizational and topical context and structure (Petkova & Croft, 2006; Balog, Bogers, Azzopardi, van den Bosch, & de Rijke, 2007). For example, Petkova and Croft (2006) propose another extension to the framework, where they explicitly model the topic, in a manner similar to relevance models for document retrieval (Lavrenko & Croft, 2001). The topic model is created using pseudo-relevance feedback, and is matched against document and candidate models. Serdyukov and Hiemstra (2008) propose a person-centric method that combines the features of both document- and profile-centric expert finding approaches. Fang and Zhai (2007) demonstrate how query/topic expansion techniques can be used within the framework; the authors also show how the two families of models (i.e., Model 1 and 2) can be derived from a more general probabilistic framework. Petkova and Croft (2007) introduce effective formal methods for explicitly modeling the dependency between the named entities and terms which appear in the document. They propose candidate-centered document representations using positional information, and estimate $p(t|d,ca)$ using proximity kernels. Their approach is similar to the window-based models that we use below, in particular, their step function kernel corresponds to our estimate of $p(t|d,ca)$ in Eq. (8) below. Balog and de Rijke (2008) introduce and compare a number of methods for building document-candidate associations. Empirically, the results produced by such models have been shown to deliver state of the art performance (see Balog et al., 2006; Petkova & Croft, 2006; Petkova & Croft, 2007; Fang & Zhai, 2007; Balog et al., 2007; Balog & de Rijke, 2008).

Finally, we highlight two alternative approaches that do not fall into the categories above (i.e., candidate or document models). Macdonald and Ounis (2007b) propose to rank experts with respect to a topic based on data fusion techniques, without using collection-specific heuristics; they find that applying field-based weighting models improves the ranking of candidates. Macdonald, Hannah, and Ounis (2008) integrate additional evidence by identifying home pages of candidate experts and clustering relevant documents. Rode, Serdyukov, Hiemstra, and Zaragoza (2007) represent documents, candidates, and associations between them as an entity containment graph, and propose relevance propagation models on this graph for ranking experts. For other models and techniques, we refer the reader to numerous variations proposed during the TREC track (see Craswell et al., 2006; Soboroff et al., 2007).

While in this paper we concentrate on the task of expert finding, it is worth noting that other expertise retrieval tasks have also been developed based on these models. For example, Balog and de Rijke (2007a) address the task of expert profiling, and Balog and de Rijke (2007b) address the task of finding *similar* experts.

3. The expert finding task

In this section we define the expert finding task in some detail and we formalize the process using generative probabilistic models. Central to the proposed models is the estimation of the probability of the query topic being generated by the candidate expert. Put differently, how likely would this query topic be talked/written about by this candidate? The different approaches to expert search lead to different language models, i.e., candidate or document language models, being used to estimate this probability.

To undertake the modeling of the task of searching for experts within an organization, we assume that there is a sufficiently large repository (or set of repositories) of textual content available in electronic form. These repositories would comprise of a mixture of document types which are indexable and potentially useful in describing the expertise of individuals within an organization. Example document types include home pages, reports, articles, minutes, emails, and so forth. Further, we also assume that there is a list of candidate experts to whom the repository contains references.

Expert finding addresses the task of finding the right person(s) with the appropriate skills and knowledge: *"Who are the experts on topic X?"* Within an organization, there may be many possible candidates who could be experts for a given topic. For a given query, then, the task is to identify which of the candidates are likely to be an expert, or, put differently:

> what is the probability of a candidate ca being an expert given the query topic q?

That is, we wish to determine $p(ca|q)$, and rank candidates ca according to this probability. The candidates with the highest probability given the query are deemed to be the most likely experts for that topic. The challenge, of course, is how to accurately estimate this probability. Since the query is likely to consist of very few terms to describe the expertise required, we should be able to obtain a more accurate estimate by invoking Bayes' Theorem:

$$p(ca|q) = \frac{p(q|ca) \cdot p(ca)}{p(q)}, \tag{1}$$

where $p(ca)$ is the probability of a candidate and $p(q)$ is the probability of a query. Since $p(q)$ is a constant (for a given query), it can be ignored for the purpose of ranking. Thus, the probability of a candidate ca being an expert given the query q is proportional to the probability of a query given the candidate $p(q|ca)$, weighted by the *a priori* belief that candidate ca is an expert $(p(ca))$:

$$p(ca|q) \propto p(q|ca) \cdot p(ca). \tag{2}$$

A considerable part of this paper is devoted to estimating the probability of a query given the candidate, $p(q|ca)$ (see Section 4), because this probability captures the extent to which the candidate knows about the query topic. The candidate priors, $p(ca)$, are generally assumed to be uniform, and so they will not influence the ranking. It has however been shown that using candidate priors can lead to improvements; see, e.g., Fang and Zhai, 2007; Petkova and Croft, 2007. In this paper, we assume that the priors $p(ca)$ are uniform, and so make no assumption about the prior knowledge we have about the candidates.

4. Modeling the expert finding task

In order to determine the probability of a query given a candidate $(p(q|ca))$, we adapt generative probabilistic language models used in Information Retrieval in two different ways. In our first model we build a textual representation of an individual's knowledge according to the documents with which he or she is associated. Previously, this model has been referred to as a candidate model because a language model for the candidate is inferred; we will refer to it as *Model 1*. From this representation we then estimate the probability of the query topic given the candidate's model. In our second model we retrieve the documents that best describe the topic of expertise, and then consider the candidates that are associated with these documents as possible experts. Because language models for documents are being inferred, this model has previously been referred to as a document model; we will refer to it as *Model 2*.

4.1. Using candidate models: models 1 and 1B

Our first formal model for estimating the probability of a query given a candidate, $p(q|ca)$, builds on well-known intuitions from standard language modeling techniques applied to document retrieval (Ponte & Croft, 1998; Hiemstra, 2001). A candidate expert ca is represented by a multinomial probability distribution over the vocabulary of terms. Therefore, a candidate model θ_{ca} is inferred for each candidate ca, such that the probability of a term given the candidate model is $p(t|\theta_{ca})$.

The model is then used to predict how likely a candidate would produce a query q. Each query term is assumed to be sampled identically and independently. Thus, the query likelihood is obtained by taking the product across all the terms in the query, such that:

$$p(q|\theta_{ca}) = \prod_{t\in q} p(t|\theta_{ca})^{n(t,q)}, \tag{3}$$

where $n(t,q)$ denotes the number of times term t is present in query q. Intuitively, the candidate model $p(t|\theta_{ca})$ expresses the likelihood of what kind of things a candidate expert would write about. The presumption is that the more likely a candidate is to talk about something, the more likely he or she is to be an expert about it. The generation of the query given this candidate model is like asking whether this candidate is likely to write about this query topic.

However, to obtain an estimate of $p(t|\theta_{ca})$, we must first obtain an estimate of the probability of a term given a candidate, $p(t|ca)$, which is then smoothed to ensure that there are no non-zero probabilities due to data sparsity. In document language modeling, it is standard to smooth with the background collection probabilities:

$$p(t|\theta_{ca}) = (1-\lambda_{ca})\cdot p(t|ca) + \lambda_{ca}\cdot p(t), \tag{4}$$

where $p(t)$ is the probability of a term in the document repository. In this context, smoothing adds probability mass to the candidate model according to how likely it is to be generated (i.e., written about) by anyone in the organization. To approximate $p(t|ca)$, we use the documents as a bridge to connect the term t and candidate ca in the following way:

$$p(t|ca) = \sum_{d\in D_{ca}} p(t|d,ca)\cdot p(d|ca). \tag{5}$$

That is, the probability of selecting a term given a candidate is based on the strength of the co-occurrence between a term and a candidate in a particular document $(p(t|d,ca))$, weighted by the strength of the association between the document and the candidate $(p(d|ca))$. Constructing the candidate model this way can be viewed as the following generative process: the term t is generated by candidate ca by first generating document d from the set of supporting documents D_{ca} with probability $p(d|ca)$, and then generating the term t from the document d with probability $p(t|d,ca)$. The set of supporting documents is made up of documents associated with ca: $D_{ca} = \{d : p(d|ca) > 0\}$. Alternatively, D_{ca} can be set differently, by using a topically focused subset of documents or taking the top n documents most strongly associated with ca. In Section 5, we describe various way in which $p(d|ca)$ can be estimated. Next, however, we discuss the estimation of $p(t|d,ca)$.

4.1.1. Model 1

Our first approach to estimating candidate models assumes that the document and the candidate are conditionally independent. That is: $p(t|d,ca) \approx p(t|d)$, where $p(t|d)$ is the probability of the term t in document d. We approximate it with the standard maximum-likelihood estimate of the term, i.e., the relative frequency of the term in the document. Now, if we put together our choices so far (Eqs. (3)–(5)), we obtain the following final estimation of the probability of a query given the candidate model:

$$p(q|\theta_{ca}) = \prod_{t\in q}\left\{(1-\lambda_{ca})\cdot\left(\sum_{d\in D_{ca}} p(t|d)\cdot p(d|ca)\right) + \lambda_{ca}\cdot p(t)\right\}^{n(t,q)}, \tag{6}$$

where λ_{ca} is a general smoothing parameter. Here we set λ_{ca} equal to $\frac{\beta}{\beta+n(ca)}$ where $n(ca)$ is the total number of term occurrences in the documents associated with the candidate. Essentially, the amount of smoothing is proportional to the amount of information available about the candidate (and is like Bayes smoothing with a Dirichlet prior). So if there are very few documents about a candidate then the model of the candidate is more uncertain, leading to a greater reliance on the background probabilities. This, then, is our Model 1, which amasses all the term information from all the documents associated with the candidate and uses this to represent that candidate. The probability of the query is directly generated from the candidate's model.

4.1.2. Model 1B

Model 1 assumes conditional independence between the document and the candidate. However, this assumption is quite strong as it suggests that all the evidence within the document is descriptive of the candidate's expertise. This may be the case if the candidate is the author of the document, but here we consider an alternative. We can consider the probability of a term given the document and the candidate, $p(t|d,ca)$, based on the strength of the co-occurrence between a term and a candidate in a particular document. In this case, both the document and the candidate determine the probability of the term.

One natural way in which to estimate the probability of co-occurrence between a term and a candidate, is by considering the proximity of the term given the candidate in the document, the idea being that the closer a candidate is to a term the more likely that term is associated with their expertise. We draw upon previous research on estimating probabilities of term co-occurrences within in a window (Azzopardi, Girolami, & Crowe, 2005) and adapt it for the present case. Here, we assume that the candidate's name, email, etc. have been replaced within the document representation with a candidate identifier, which can be treated much like a term, referred to as ca. The terms surrounding either side of ca form the context of the candidate's expertise and can be defined by a window of size w within the document. For any particular distance (window size) w between a term t and candidate ca, we can define the probability of a term given the document, candidate, and distance:

$$p(t|d,ca,w) = \frac{n(t,d,ca,w)}{\Sigma n(t',d,ca,w)} \tag{7}$$

where $n(t,ca,w,d)$ is the number of times term t co-occurs with ca at a distance of at most w in document d. Now, the probability of a term given the candidate and document is estimated by taking the sum over all possible window sizes W:

$$p(t|d,ca) = \sum_{w \in W} p(t|d,ca,w) \cdot p(w), \tag{8}$$

where $p(w)$ is the prior probability that defines the strength of association between the term and the candidate at distance w, such that $\sum_{w \in W} p(w) = 1$.

The final estimate of a query given the candidate model using this window-based approach is shown in Eq. (9):

$$p(q|\theta_{ca}) = \prod_{t \in q} \left\{ (1-\lambda_{ca}) \cdot \left(\sum_{d \in D_{ca}} \left(\sum_{w \in W} p(t|d,ca,w) \cdot p(w) \right) \cdot p(d|ca) \right) + \lambda_{ca} \cdot p(t) \right\}^{n(t,q)}. \tag{9}$$

This is Model 1B, which amasses all the term information within a given window around the candidate in all the documents that are associated with the candidate and uses this to represent that candidate. Then, as in Model 1, the probability of the query is directly generated from the candidate's model. Clearly, other ways in which to estimate $p(t|d,ca)$ are possible which would lead to variations of candidate-based models. For instance, if the type of reference to the candidate was known i.e., author, citation, etc., then the appropriate extraction could be performed. However, we leave this for further work.

4.2. Using document models: models 2 and 2B

Instead of creating a term-based representation of a candidate as in Models 1 and 1B, the process of finding an expert can be considered in a slightly different way in which the candidate is not directly modeled. Instead, documents are modeled and queried, then the candidates associated with the documents are considered as possible experts. The document acts like a "hidden" variable in the process which separates the querying process from the candidate finding. Under this model, we can think of the process of finding an expert as follows. Given a collection of documents ranked according to the query, we examine each document and if relevant to our problem, we then see who is associated with that document and consider this as evidence of their knowledge about the topic.

Thus, the probability of a query given a candidate can be viewed as the following generative process:

- Let a candidate ca be given.
- Select a document d associated with ca (i.e., generate a supporting document d from ca).
- From this document and candidate, generate the query q, with probability $p(q|d,ca)$.

By taking the sum over all documents $d \in D_{ca}$, we obtain $p(q|ca)$. Formally, this can be expressed as

$$p(q|ca) = \sum_{d \in D_{ca}} p(q|d,ca) \cdot p(d|ca). \tag{10}$$

Assuming that query terms are sampled identically and independently, the probability of a query given the candidate and the document is:

$$p(q|d,ca) = \prod_{t \in q} p(t|d,ca)^{n(t,q)}. \tag{11}$$

By substituting Eq. (11) into Eq. (10) we obtain the following estimate of the document-based model:

$$p(q|ca) = \sum_{d \in D_{ca}} \prod_{t \in q} p(t|d,ca)^{n(t,q)} \cdot p(d|ca). \tag{12}$$

Similarly to Models 1 and 1B, there are two ways of estimating $p(t|d,ca)$, which are discussed next.

4.2.1. Model 2

We can compute the probability $p(q|ca)$ by assuming conditional independence between the query and the candidate. Here, $p(t|d,ca) \approx p(t|\theta_d)$, hence, for each document d a document model θ_d is inferred, so that the probability of a term t given the document model θ_d is:

$$p(t|\theta_d) = (1-\lambda_d) \cdot p(t|d) + \lambda_d \cdot p(t). \tag{13}$$

By substituting $p(t|\theta_d)$ for $p(t|ca,d)$ into Eq. (12), the final estimation of Model 2 is:

$$p(q|ca) = \sum_{d \in D_{ca}} \prod_{t \in q} \{(1-\lambda_d) \cdot p(t|d) + \lambda_d \cdot p(t)\}^{n(t,q)} \cdot p(d|ca), \tag{14}$$

where λ_d is set proportional to the length of the document $n(d)$, such that $\lambda_d = \frac{\beta}{\beta+n(d)}$. Unlike Model 1, which builds a direct representation of the candidate's knowledge, Model 2 mimics the process of searching for experts via a document collection. Here, documents are found that are relevant to the expertise required, and they are used as evidence to suggest that the associated candidate is an expert. After amassing all such evidence, possible candidates are identified.

4.2.2. Model 2B

Similarly to Model 1B, we can consider the probability of a term given the document and the candidate $p(t|d,ca)$ without resorting to the conditional independence assumption. To estimate the probability of co-occurrence between a term and a candidate $(p(t|d,ca))$, we take the sum of the distance-based co-occurrence probabilities $(p(t|d,ca,w))$ over all possible window sizes, as defined in Eq. (8). This creates a localized representation of the document given the candidate (or candidate biased document model) which is used in the querying process. The final estimate of a query given the candidate using this approach is shown in Eq. (15):

$$p(q|ca) = \sum_{d \in D_{ca}} \prod_{t \in q} \left\{(1-\lambda_d) \cdot \left(\sum_{w \in W} p(t|d,ca,w) \cdot p(w)\right) + \lambda_d \cdot p(t)\right\}^{n(t,q)} \cdot p(d|ca). \tag{15}$$

Before turning to an experimental evaluation of the models introduced in this section, we need to explain how we estimate document-candidate associations, $p(d|ca)$.

5. Establishing document-candidate associations

For each of the models introduced in Section 4, we need to be able to estimate the probability $p(d|ca)$, which expresses the extent to which document d characterizes candidate ca. It is important to note that the reading of $p(d|ca)$ is different for the two families of models. In case of Models 1 and 1B (Section 4.1), it reflects the degree to which the candidate's expertise is described using this document d. For Models 2 and 2B (Section 4.2), it provides a ranking of candidates associated with a given document d, based on their contribution made to d.

If we consider the probability $p(d|ca)$ from a different point of view by invoking Bayes' Theorem, we obtain:

$$p(d|ca) = \frac{p(ca|d) \cdot p(d)}{p(ca)}. \tag{16}$$

This decomposition explicitly shows how prior knowledge about the importance of the documents can be encoded within the modeling process, via $p(d)$. For instance, a journal article may be more indicative of expertise than an email. Thus, certain types of documents can be favored over others. Also, prior knowledge with respect to a candidate being an expert can be encoded via $p(ca)$. For instance, if the candidate is known to be an authority within the organization, or a senior member of the organization, this could increase the probability of them being an expert. Here, we assume that $p(d)$ and $p(ca)$ follow uniform distributions, but see (Fang & Zhai, 2007) for examples using priors. Consequently, the task boils down to the estimation of $p(ca|d)$.

We assume that all candidates' occurrences (name, email address, etc.) have been recognized in documents, and $n(ca,d)$ denotes the number of times candidate ca is present (mentioned) in document d. Below, we distinguish between two ways of converting these raw frequencies into probabilities.

5.1. The boolean model of associations

Under the boolean model to establishing document-candidate associations, associations are binary decisions; they exist if the candidate occurs in the document, irrespective of the number of times the person or other candidates are mentioned in that document. Thus, we simply set

$$p(ca|d) = \begin{cases} 1, & \text{if } n(ca,d) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

Clearly, this boolean model of associations makes potentially unrealistic assumptions. In fact, $p(ca|d)$ constructed this way is not a probability distribution. Nevertheless, at this point, our aim is to establish a baseline and to take the simplest choice using this boolean model.

5.2. A frequency-based approach

Our goal with this second estimate for establishing document-candidate associations is to formulate $p(ca|d)$ in such a way that it indicates the strength, and not only the presence, of the association between candidate ca and document d. We approach it by adopting the popular TF.IDF weighting scheme commonly used within the vector space retrieval model. The rationale behind using the TF.IDF formula is that it expresses the candidate's importance within a particular document, while also incorporating the candidate's "general importance" (i.e., candidates who occur only in a handful of documents will be compensated with higher IDF values). To avoid document length normalization problems, we use a "lean" document representation for this task. That is, for the estimation given below, documents consist of only candidate identifiers, and all other terms are filtered out. Formally, this can be expressed as:

$$p(ca|d) \propto \frac{n(ca,d)}{\sum_{ca'} n(ca'd)} \cdot \log \frac{|D|}{|\{d' : n(ca,d') > 0\}|}. \quad (18)$$

Note that Eq. (18) is computed only for candidates that occur in document d ($n(d,ca) > 0$). We refer the reader to Balog and de Rijke (2008) for an extensive study on document-candidate associations.

6. Experimental setup

Now that we have detailed our models, we present an experimental evaluation of our models. We specify our research questions, describe our data set, the way in which we identify candidate experts, our evaluation metrics, and our estimation of smoothing parameters before, finally, addressing our research questions in Section 7.

6.1. Research questions

We address the following research questions:

- How do our expert finding models perform compared to each other? That is, how do Model 1 and Model 2 compare?
- What are optimal settings for the window size(s) to be used in Models 1B and 2B? Do different window sizes lead to different results, in terms of retrieval effectiveness?
- What is the effect of lifting the conditional independence assumption between the query and the candidate (Model 1 vs. Model 1B, Model 2 vs. Model 2B)?
- Which of the two ways of capturing document-candidate associations is most effective: the boolean approach or the frequency-based approach?

6.2. Test collection

We use the test sets of the 2005 and 2006 edition of the TREC Enterprise track (Craswell et al., 2006; Soboroff et al., 2007). The document collection used in both years is the W3C corpus (W3C, 2005), a heterogenous document repository containing a mixture of document types crawled from the W3C web site. The six different types of web pages were lists (email forum; 198,394 documents), dev (code; 62,509 documents), www (web; 45,975 documents), esw (wiki; 19,605 documents), other (miscellaneous; 3,538 documents), and people (personal homepages; 1,016 documents). The W3C corpus contains 331,037 documents, adding up to 5.7GB.

We used the entire corpus, and simply handled all documents as HTML documents. That is, we did not resort to any special treatment of document types, nor did we exploit the internal document structure that may be present; instead, we represented all documents as plain text. We removed a standard list of stopwords, but did not apply stemming.[1] The TREC Enterprise 2005 topics (50) are names of working groups of the W3C organization. Members of the corresponding working group were regarded as experts of the topic. The 2006 topics (49) were contributed by TREC participants and were assessed manually. We used only the titles of the topic descriptions.

6.3. Personal name identification

In order to form document-candidate associations $n(d,ca)$, we need to be able to recognize candidate experts' occurrences within documents. In the TREC setting, a list of possible candidates is given, where each person is described with a unique *person_id*, one or more *names*, and one or more *e-mail* addresses. While this is a specific way of identifying a person, and different choices are also possible (e.g., involving social security number, or employee number instead of, or in addition to, the representations just listed), nothing in our modeling depends on *this* particular choice.

The recognition of candidate occurrences in documents (through one of these representations) is a restricted (and specialized) information extraction task, that is often approached using various heuristics. In Bao et al. (2007), six different match types (MT) of person occurrences are identified, based on full name, email address, and various name abbreviations. Balog et al. (2006) take a similar approach and introduce four types of matching; three attempt to identify candidates by their name, and one uses the candidate's email address. To facilitate comparison, we decided to use annotations of candidate

[1] We also experimented with a stemmed version of the collection, using the Porter stemmer, but did not observe significant differences.

Table 1
Value of β (rounded to integers) for each representation size

Model	β										
	15	25	50	75	100	125	150	200	250	300	All
1/1B	22,507	33,980	57,392	75,270	89,529	101,822	112,714	131,322	146,833	160,181	170,550
2/2B	67	101	171	224	267	303	336	391	438	477	500

"All" corresponds to the full document representation (Model 1 and 2).

occurrences provided by Zhu (2006) to participants in the TREC Enterprise track. In this preprocessed version of the W3C data set candidates are recognized by various representations using the Aho-Corasick matching algorithm.

6.4. Evaluation metrics

The evaluation measures on which we report for the task of finding experts are mean average precision (MAP) and mean reciprocal rank (MRR) (TREC, 2005). Evaluation scores were computed using the `trec_eval` program.[2] For significance testing we use a two-tailed, matched pairs Student's t-test, and look for improvements at significance levels $^{(1)}$ 0.95, $^{(2)}$ 0.99, and $^{(3)}$ 0.999.

6.5. Smoothing parameters

It is well-known that smoothing can have a significant impact on the overall performance of language modeling-based retrieval methods (Zhai & Lafferty, 2001). Here, we use Bayes smoothing with a *Dirichlet prior* (Mackay & Peto, 1994) to improve the estimated document language model. Specifically, as detailed in Sections 4.1.1 and 4.2.1, we need to estimate a smoothing parameter λ that is defined as $\lambda = \frac{\beta}{\beta + n(x)}$, where $n(x)$ is the sum of the lengths of all documents associated with a given candidate (Model 1), or the document length (Model 2). We set the value of β based on the average (candidate/document) representation length, thus dynamically adjusting the amount of smoothing:

- For Model 1 we estimate $\beta = \beta_{ca}$ as follows:

$$\beta_{ca} = \frac{\sum_{ca} n(ca)}{|ca|}, \tag{19}$$

where $|ca|$ is the total number of candidates and $n(ca)$ is the total number of term occurrences associated with the candidate, approximated with the number of documents associated with the candidate, times the average document length: $n(ca) = | \{d : n(ca, d) > 0\} | \cdot | d |$, As before, $n(ca,d)$ denotes the number of times candidate ca is present in document d, while $|d|$ is the average document length.
- Our estimation of $\beta = \beta_{ca,w}$ for Model 1B is given by

$$\beta_{ca,w} = \frac{\sum_{ca}\sum_{d} n(ca, d, w)}{|ca|}, \tag{20}$$

where $n(ca, d, w)$ denotes the number of terms co-occurring with candidate ca in document d at a distance of at most w.
- For Model 2 we take $\beta = |d|$, i.e., the average document length in the collection.
- And, finally, for Model 2B $\beta = \beta_{d,w}$ is defined by:

$$\beta_{d,w} = \frac{\sum_{ca} \sum_{d} n(ca, d, w)}{\sum_{ca} |\{d : p(d, ca) > 0\}|}. \tag{21}$$

The actual numbers obtained for β by using the choices specified above are reported in Table 1.

7. Experimental results

We now present the outcomes of our experiments. One by one, we address the research questions listed in Section 6.1.

7.1. Model 1 vs. Model 2

Which of Model 1 and Model 2 is most effective for finding experts? We compare the two models on the 2005 and 2006 editions of the TREC Enterprise test sets, using the measures listed in Section 6.4 and using the boolean document-candidate association method. The results are presented in Table 2.

[2] For registered participants, `trec_eval` is available from the TREC web site <http://trec.nist.gov>.

Table 2
Model 1 vs. Model 2 on the expert finding task, using the TREC 2005 and 2006 test collections

Model	TREC 2005		TREC 2006	
	MAP	MRR	MAP	MRR
1	.1883	.4692	.3206	.7264
2	**.2053**	**.6088**$^{(2)}$	**.4660**$^{(3)}$	**.9354**$^{(3)}$

Best scores for each year are in boldface.

Several things are worth noting. First, in absolute terms, the scores achieved on the 2006 collection are high, for all measures. Second, the scores on the 2006 topic set are substantially higher than the 2005 scores; this is most likely due to the differences in assessment procedure used. Third, on the 2006 collection Model 2 clearly outperforms Model 1, on all measures.

Moreover, all differences on the 2006 collection are statistically significant. On the 2005 collection, the picture is more subtle: Model 2 outperforms Model 1 in terms of MAP and MRR; however, the difference in MAP scores is not significant.

In conclusion, then, Model 2 outperforms Model 1, significantly so on both the 2005 and 2006 test collection in terms of a precision oriented measure such as MRR. The difference as measured in terms of MAP is significant only on the 2006 test set with its more lenient (human generated) ground truth.

7.2. Window-based models: models 1B and 2B

Next we look for performance differences between models based on different window sizes, i.e., for Models 1B and 2B. Recall that for Models 1B and 2B the candidate-term co-occurrence is calculated for a given window size w, after which a weighted sum over various window sizes is taken (see Eq. (8)). Here, we consider only the simplest case: a single window with size w, thus $W = \{w\}$ and $p(w) = 1$.

To be able to compare the models, first the optimal window sizes (for MAP and MRR) are empirically selected for each model and topic set. The range considered is $w = 15, 25, 50, 75, 100, 125, 150, 200, 250, 300$.[3] In all cases we use the boolean document-candidate association method. The MAP and MRR scores corresponding to each window size w are displayed in Fig. 1.

According to the plots on the left-hand side of Fig. 1, in terms of MAP the ideal window size is between 100 and 250, and MAP scores show small variance within this range. Model 1B on the TREC 2005 topic set seems to break this pattern of behavior, and delivers best performance in terms of MAP at window size 25. In terms of MRR, however, smaller window sizes tend to perform better on the 2005 collection; this is not suprising, as smaller windows are more likely to generate high-precision co-occurrences.

It is worth pointing out that for both measures (MAP and MRR), for both years (2005 and 2006), and both models (1B and 2B), the difference between the best-performing and worst-performing window size is statistically significant.

7.3. Baseline vs window-based models

What is the effect of lifting the conditional independence assumption between the query and the candidate? That is, what if any, are the performance differences between the baseline models (Model 1 and Model 2) and the window-based models (Model 1B and Model 2B, respectively)? For the window-based models, we use the best performing configuration, i.e., window size, according to the results of the previous subsection. We present two sets of results, one based on window sizes optimized for MAP, and one based on window sizes optimized for MRR. In all cases we use the boolean document-candidate association method.

Looking at the results in Table 3 we find that, for the MAP-optimized setting, Model 1B improves upon Model 1 in nearly all cases. On the TREC 2005 topic set, the improvement is most noticeable in early precision: MRR +26% vs. MAP +7%; the difference in MRR is significant. On the TREC 2006 topics the advance of Model 1B over Model 1 is even more substantial, achieving as much as 32% improvement in MAP; the differences in MAP and MRR are highly significant. In contrast, the benefits of Model 2B over Model 2 are not as obvious. On the TREC 2005 collection, Model 2B delivers slight, but non-significant, improvements on both measures. On TREC 2006, however, the window-based model (Model 2B) is outperformed by the baseline (Model 2), although the differences are not significant. Finally, Model 2B performs better than Model 1B, but the gap between them is smaller than between Model 2 and 1. None of the differences between Model 1B and 2B are significant (i.e., neither for MAP, MRR, 2005, nor 2006).

Next we turn to a comparison between the baseline and window-based models based on MRR-optimized settings; see Table 4. Model 1B improves over Model 1, and in all cases except 2005 (MAP) the improvement is significant. Comparing Models 2 and 2B we observe a slight improvement in MRR but losses in MAP; none of the differences are significant. And finally, as to the differences between Model 1B and Model 2B, only the difference in MAP on the TREC 2006 topic set is significant (at the 0.99 level).

[3] Here we followed (Cao, Liu, Bao, & Li, 2006), who considered window sizes 20,. . ., 250; note that the average document length is approximately 500 words.

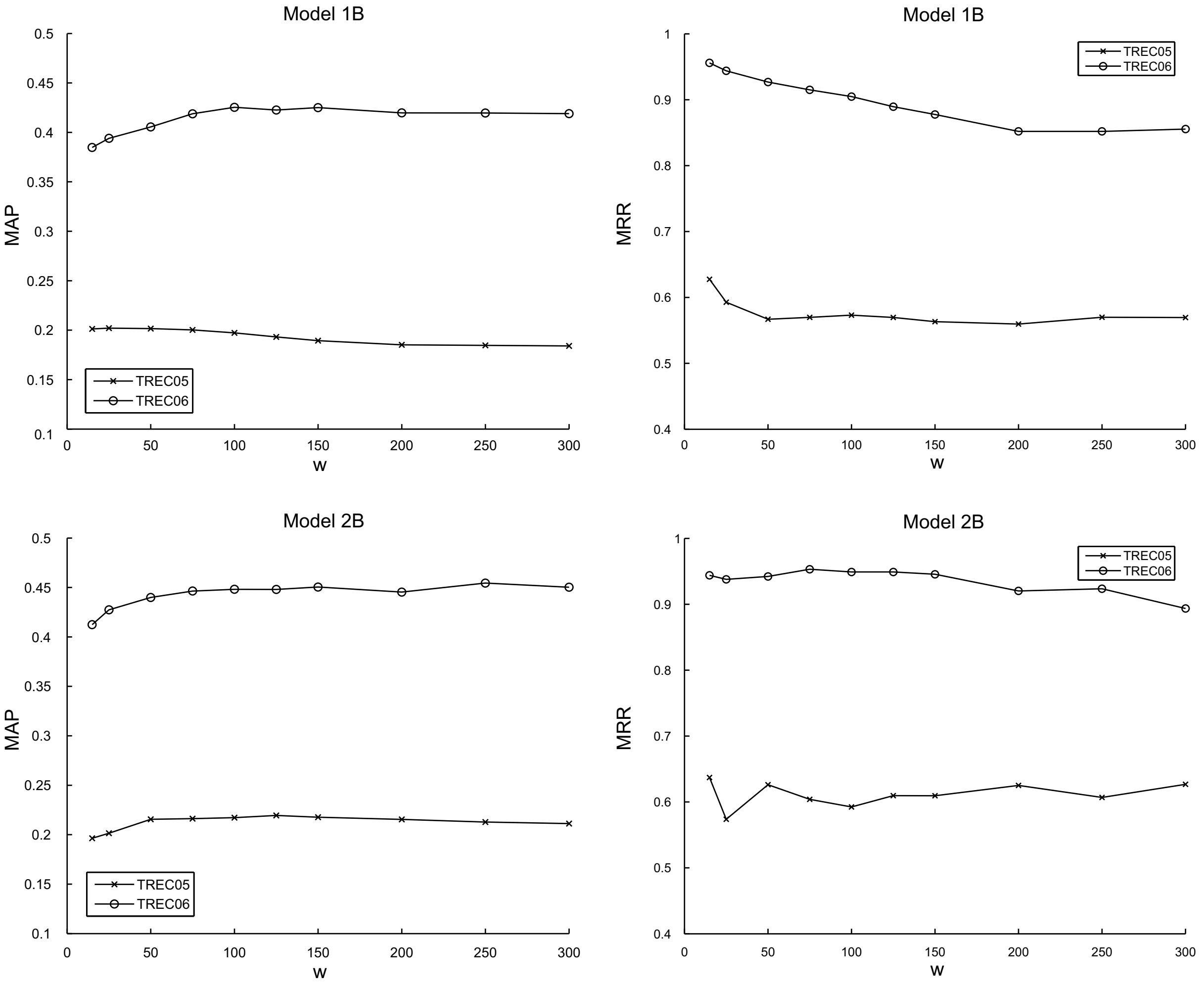

Fig. 1. Effect on MAP (Left) and MRR (Right) of varying the window size *w* for Model 1B (Top) and Model 2B (Bottom).

Table 3
Overall results on the expert finding task; window sizes optimized for MAP

Model	TREC 2005			TREC 2006		
	w	MAP	MRR	*w*	MAP	MRR
1	–	.1883	.4692	–	.3206	.7264
1B	25	.2020	.5928$^{(1)}$	100	.4254$^{(3)}$	.9048$^{(3)}$
2	–	.2053	.6088	–	**.4660**	**.9354**
2B	125	**.2194**	**.6096**	250	.4544	.9235

Best scores (per measure) for each year are in boldface.

7.4. Association methods

Finally, we turn to a comparison of the two document-candidate association methods that we consider in this paper: the boolean approach and the frequency-based approach. The results are listed in Table 5.

The frequency-based association method is beneficial for Model 1: on the 2005 test set the improvements are significant, both in terms of MAP and in terms of MRR; while still beneficial on the 2006 test set, only the MAP scores improve significantly. As to Model 2, the usage of frequency-based document-candidate associations leads to increases in MAP scores and losses in MRR scores, suggesting that the frequency-based association method has a recall-enhancing effect at the expense of some loss in early precision, although none of the differences are significant.

The usage of frequency-based document-candidate associations has a mixed impact on the performance of the windows-based models (Model 1B and Model 2B). The impact on Model 2B is mostly positive, but not significant; in contrast, the

Table 4
Overall results on the expert finding task; window sizes optimized for MRR

Model	TREC 2005			TREC 2006		
	w	MAP	MRR	w	MAP	MRR
1	–	.1883	.4692	–	.3206	.7264
1B	15	.2012	.6275$^{(2)}$	15	.3848$^{(1)}$	**.9558**$^{(3)}$
2	–	**.2053**	.6088	–	**.4660**	.9354
2B	15	.1964	**.6371**	75	.4463	.9531

Best scores (per measure) for each year are in boldface.

Table 5
Comparison of association methods

Model	assoc. method	TREC 2005		TREC 2006	
		MAP	MRR	MAP	MRR
1	boolean	.1883	.4692	.3206	.7264
	freq.	**.2321**$^{(3)}$	*.5857*$^{(2)}$	*.3501*$^{(2)}$	*.7789*
1B (opt. for MAP)	boolean	*.2020*	.5928	*.4254*	*.9048*
	freq.	.1745$^{(2)}$	*.6103*	.3849$^{(3)}$	.8646$^{(1)}$
1B (opt. for MRR)	boolean	*.2012*	*.6275*	*.3848*	*.9558*
	freq.	.1746$^{(3)}$	.6179	.3485$^{(3)}$	.9081$^{(1)}$
2	boolean	.2053	*.6088*	.4660	*.9354*
	freq.	*.2093*	.6083	**.4803**	.9150
2B (opt. for MAP)	boolean	*.2194*	.6096	.4544	.9235
	freq.	.2098	*.6130*	*.4614*	*.9303*
2B (opt. for MRR)	boolean	*.1964*	**.6371**	.4463	.9531
	freq.	.1766$^{(2)}$	.5883	*.4540*	**.9659**

Results on the expert finding task on the TREC 2005 and 2006 test collections. Best scores for each model are in italic. Best scores (per measure) for each year are in boldface. Significance results are reported for differences between boolean and frequency-based versions of the same model.

impact on Model 1B is mostly negative, and in many cases significantly so. This is because association strength is estimated at the document level (based on the number of other candidates associated with that document); however, in the window-based models only part of the document (terms within distance w) is used to describe that candidate. If s and s' are snippets from documents d and d', respectively, terms in s will be taken into account with more weight than s' if fewer candidates occur in s. As the results for Model 1B show, this may favor snippets whose accuracy for describing candidates is limited.

8. Discussion

In this section we discuss and qualify the main findings obtained in Section 7. We start by providing a topic-level analysis of the experimental results from Section 7, follow with an analysis of the sensitivity of our models to the smoothing parameter, comment on the generalizability of our approach, compare our performance to other approaches, and conclude by discussing the preferred model.

8.1. Topic-level analysis

We turn to a topic-level analysis of the comparisons detailed in Sections 6 and 7. Rather than detailing every comparison of approaches from Section 7, we illustrate that section's main findings at the topic level.

To start, we consider the comparison between Model 1 and 2. In Fig. 2 we plot the differences in performance (per topic) between Model 1 and Model 2; topics have been sorted by performance gain. The plots reflect the findings reported in Table 2: In most cases the differences between Model 1 and 2 favor Model 2 (shown as positive). The plots show that Model 1 is preferable only for a handful of topics and that Model 2 is substantially better at retrieving experts at higher ranks for most topics.

Now we turn our attention to a topic-level comparison between Model 1 and 1B and between Model 2 and 2B; see Fig. 3. Again, we see the significance (or lack thereof) of the differences between the approaches clearly reflected in the plots—compare Tables 3 and 4. Interestingly, on the 2006 topic set in terms of reciprocal rank no topic is affected negatively by changing from Model 1 to Model 1B; in terms of average precision, though, some topics do suffer although the overall difference (.3206 vs. .4254) in MAP is positive (significantly so). As to Model 2, it is clear that moving to Model 2 has very little overall impact, both in terms of MAP and, even more clearly, in terms of MRR.

Finally, we turn to boolean vs frequency-based document-candidate associations, comparing their impact on top of our baseline models (Model 1 and 2); see Fig. 4. The aggregated findings from Table 5 are clearly reflected in the plots in Fig. 4: no

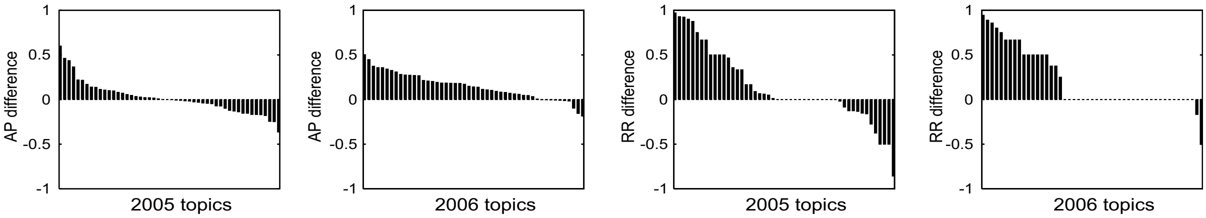

Fig. 2. Topic-level differences in scores, Model 1 (baseline) vs Model 2. From left to right: AP 2005, AP 2006, RR 2005, RR 2006.

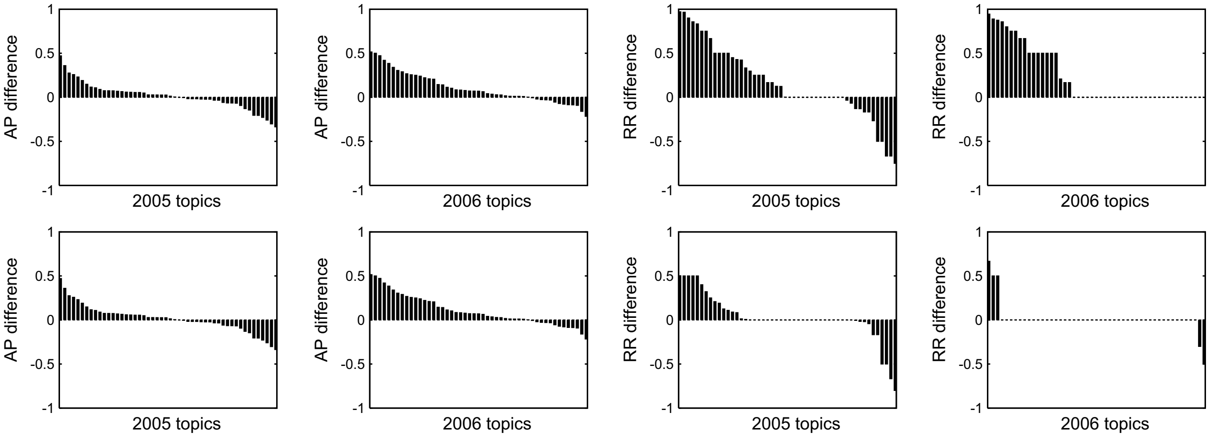

Fig. 3. Topic-level differences in scores. (Top): Model 1 (baseline) vs Model 1B (optimized for MAP or MRR). (Bottom): Model 2 (baseline) vs Model 2B (optimized for MAP or MRR). From left to right: AP 2005, AP 2006, RR 2005, RR 2006.

significant differences for Model 2 (bottom row), while the differences for Model 1 (top row) are significant (accept for MRR on the 2006 topic set). The 2005 topic that takes the biggest hit (in terms of reciprocal rank) by changing from boolean to frequency-based associations on top of Model 2 is no. 18: *compound document formats* (RR -0.6667); when using Model 1 this very topic's reciprocal rank goes up by .4286 if we replace boolean associations by frequency-based ones, suggesting that different topics may perform best with different model/association settings.

8.2. Parameter sensitivity analysis

Our models involve a smoothing parameter, denoted λ_{ca} in case of Model 1 and 1B and λ_d in case of Model 2 and 2B. The value of λ is set to be proportional to the length of the (candidate/document) representation, thus essentially is Bayes

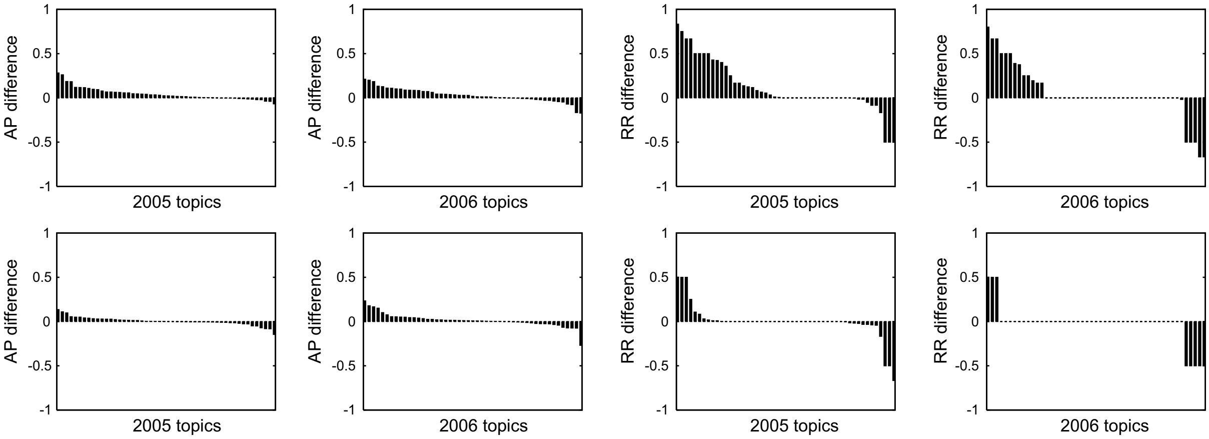

Fig. 4. Topic-level differences in scores. (Top): Model 1 with boolean associations (baseline) vs frequency-based associations. (Bottom): Model 2 with boolean associations (baseline) vs. frequency-based associations. From left to right: AP 2005, AP 2006, RR 2005, RR 2006.

smoothing with a Dirichlet prior β. We set β according to the average representation length, as described in Section 6.5. In this section we examine the parameter sensitivity for our models. That is, we plot MAP and MRR scores as a function of β. Our aim with the following analysis is to determine

1. to which extent we are able to approximate the optimal value of β;
2. how smoothing behaves in the two TREC topic sets; and
3. whether MAP and MRR scores display the same behavior (especially, if they achieve their maximum value with the same β).

Throughout this subsection we use boolean document-candidate associations.

The results for Model 1 and Model 2 are displayed in Fig. 5. The y-axis shows the value of β on a log-scale; notice that the ranges used in the top and bottom plots are different, as are the ranges used for the MAP scores and for the MRR scores. The vertical line indicates our choice of β, according to Table 1.

Our findings are as follows. First, our estimate of β is close to the optimal for Model 2 (in terms of both MAP and MRR), but is underestimated in case of Model 1. Second, with one exception (Model 1, MAP) the curves for the TREC 2005 and 2006 topic sets follow the same general trends, and maximize both MAP and MRR around the same point ($\beta = 10^7$ for Model 1, $\beta = 400$ for Model 2). Third, results show small variance, especially in terms of MAP scores, in the range $\beta = 10^6 - 10^8$ for Model 1, and $\beta = 1 - 400$ for Model 2.

Next, we perform a similar analysis for Model 1B and Model 2B. These models have an extra parameter, the window size, w, which is set to 125. The plots are presented in Fig. 6. The two topic sets follow the same trends in case of Model 2B, but for

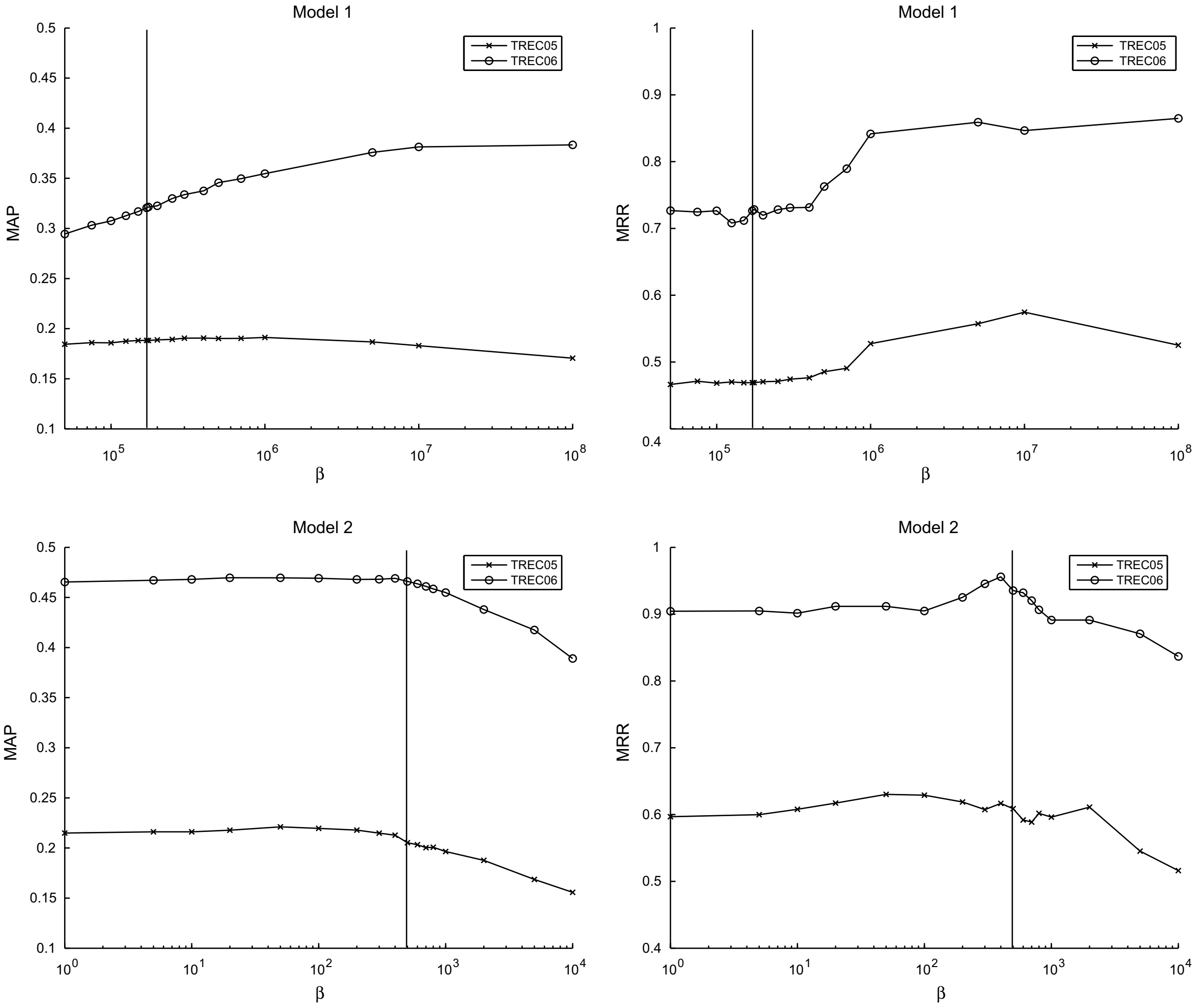

Fig. 5. The effect of varying β on Model 1 (Top) and Model 2 (Bottom). (Left): The effect on MAP. (Right): The effect on MRR.

Model 1B, the difference between the two topic sets is apparent. On the TREC 2005 topic set performance deteriorates for $\beta > 10^2$, while on the TREC 2006 set it is relatively stable throughout a wide range ($\beta \geqslant 10^4$). Our estimation of β delivers close to the best performance for all models/topic sets, with the exception of Model 1B on the TREC 2005 topics. This may be caused by the fact that the TREC 2005 and 2006 topics were created and assessed in a different manner (see Section 6.2). In particular, the TREC 2005 topics are names of working groups, and the assessments (“membership of the working group”) are independent of the document collection.

To conclude this subsection, we include a comparison of the estimated and optimal values of β in terms of MAP and MRR scores in Table 6. Overall, we can see that our estimation performs very well on the 2006 topic set for all models except Model 1, where our method tends to underestimate β and runs created with optimal settings for β significantly outperform runs created estimated settings for β (for MRR on both topic sets, for MAP only on the 2006 set). On the 2005 topic set the results are mixed, but on the whole Model 2 and Model 2B are much less sensitive to smoothing than Model 1 and Model 1B.

8.3. Generalizability of the models

While most methods and approaches to expert search introduced since the launch of the TREC Enterprise track in 2005 have been validated experimentally using the W3C collection (including the work presented in this paper), it is important to note that the W3C collection represents only one type of intranet. With only one collection it is not possible to verify whether results and conclusions generalize to other enterprise settings.

To the best of our knowledge at the time of writing there are two more collections publicly available for expertise retrieval. The CSIRO Enterprise Research Collection (CERC) has been introduced and first used at the 2007 edition of the TREC Enterprise track (Bailey, Craswell, de Vries, & Soboroff, 2007). The UvT Expert Collection was introduced by Balog et al.

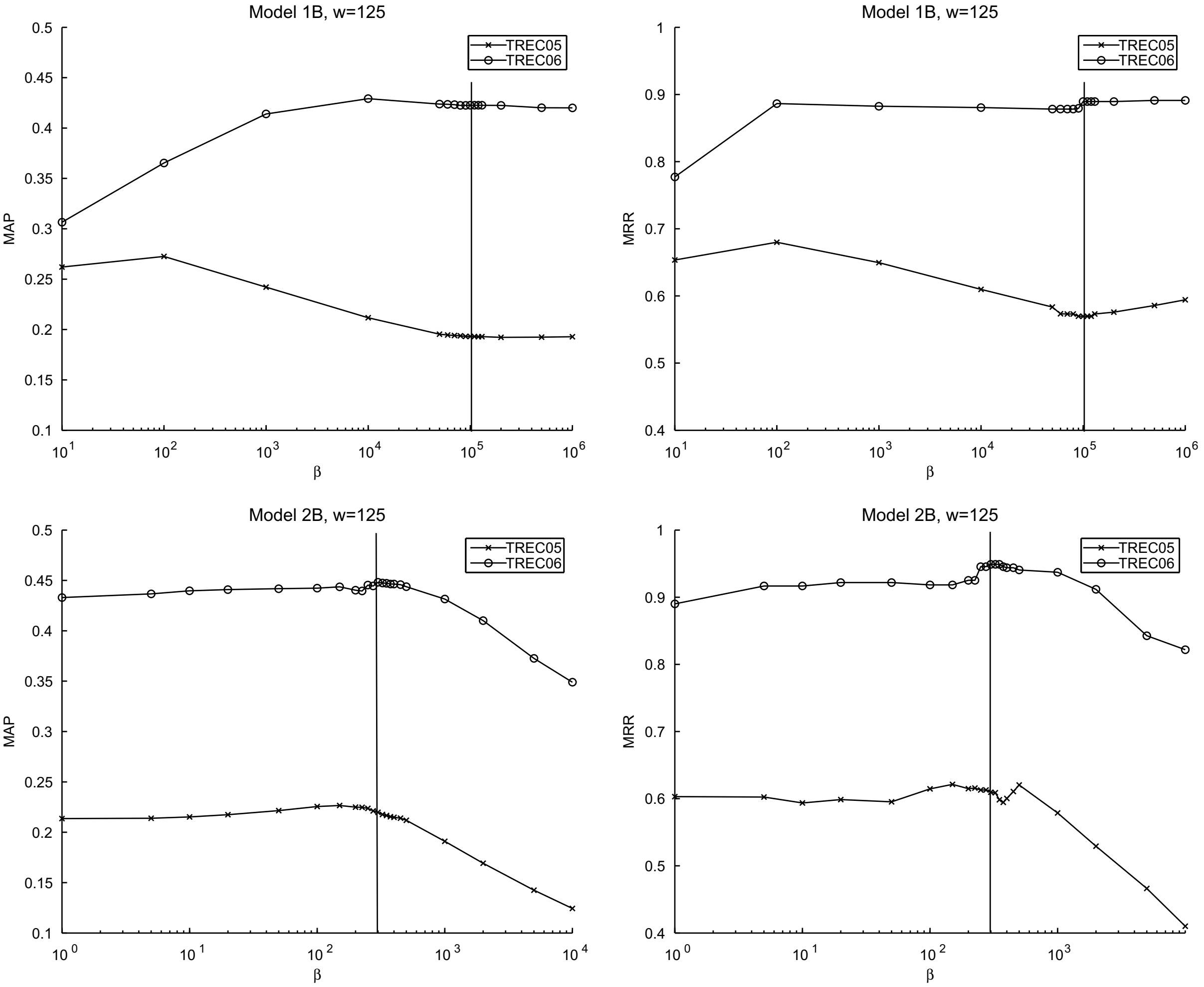

Fig. 6. The effect of varying β on Model 1B (Top), and Model 2B (Bottom), for a fixed window size $w = 125$. (Left): The effect on MAP. (Right): The effect on MRR.

Table 6
Parameter sensitivity: summary

Model	TREC	MAP				MRR			
		Estimated		Optimal		Estimated		Optimal	
		β	MAP	β	MAP	β	MRR	β	MRR
1	2005	$\sim 1.7 \cdot 10^5$	.1883	10^6	.1912	$\sim 1.7 \cdot 10^5$	.4692	10^7	.5747$^{(1)}$
	2006		.3206	10^8	.3834$^{(2)}$		.7264	10^8	.8647$^{(2)}$
1B	2005	$\sim 10^5$	.1931	10^2	.2725$^{(3)}$	$\sim 10^5$	.5696	10^2	.6800$^{(1)}$
	2006		.4226	10^4	.4291		.8895	$5 \cdot 10^5$	.8912
2	2005	500	.2053	50	.2211	500	.6088	50	.6302
	2006		.4660	20	.4697		.9354	400	.9558
2B	2005	~ 300	.2194	150	.2266$^{(1)}$	~ 300	.6096	150	.6213
	2006		.4481	300	.4481		.9490	300	.9490

The significance tests concern comparisons between runs based on estimated settings for β and runs based on optimal settings for β, i.e., column 4 vs column 6 and column 8 vs column 10. (For the window-based models, a fixed size $w = 125$ was used; in all cases boolean document-candidate associations were used).

(2007). Experimental results reported by Balog et al. (2007) and Balog (2008) confirm that our models generalize well, and that findings carry over to different collections. For the sake of comparison, we have included the results of our models on the CSIRO Enterprise Research Collection in the overview table (Table 7) below.

8.4. Comparison with other approaches

In this subsection we compare our methods to other published approaches. Table 7 highlights the highest scoring results for each. We start our discussion by looking at the top three performing teams from the 2005–2007 editions of the TREC Enterprise track.

The top two approaches from 2005 are conceptually similar to our Models 1B and 2B. Fu et al. (2006) use a candidate-centric method that collects and combines information to organize a document which describes an expert candidate (therefore they call this method "document reorganization"). Cao et al. (2006) propose a two-stage language model approach that is similar to our Model 2B, however, the probability of a candidate given the query is estimated directly (i.e., unlike applying Bayes' rule as we do in Eq. (1)). This leads to a different factorization of this probability, $p(ca|q) = \sum_d p(ca|d,q) \cdot p(d|q)$, where $p(d|q)$ is referred as the relevance model and $p(ca|d,q)$ is called the co-occurrence model. The co-occurrence model is computed based on metadata extraction (for example, recognizing whether the candidate is the author of the document and the query matches the document's title) and window-based co-occurrence. Yao, Peng, He, and Yang (2006) use a document-based method, where the query is constructed from the concatenation of the topic phrase and a person name phrase.

The top three approaches at TREC 2006 all employ—a variation of—the two-stage LM approach. Zhu et al. (2007) take the documents' internal structure into account in the co-occurrence model, moreover, they consider a weighted combination of

Table 7
Numbers reported in the literature

Approach	TREC 2005		TREC 2006		TREC 2007	
	MAP	MRR	MAP	MRR	MAP	MRR
TREC Enterprise 2005–2007 top 3 official runs						
2005 1st Fu et al. (2006)	.2749	.7268				
2005 2nd Cao et al. (2006)	.2688	.6244				
2005 3rd Yao et al. (2006)	.2174	.6068				
2006 1st Zhu et al. (2007)			.6431	.9609		
2006 2nd Bao et al. (2007)			.5947	.9358		
2006 3rd You et al. (2007)			.5639	.9043		
2007 1st Fu et al. (2008)					.4632	.6333
2007 2nd Duan et al. (2008)					.4427	.6131
2007 3rd Zhu et al. (2008)					.4337	.5802
Other approaches						
Fang and Zhai (2007)*	.204		.465			
Macdonald and Ounis (2007a)	.1983		.5210		.3406	
Macdonald and Ounis (2007b)*	.2917		.5712			
Petkova and Croft (2006)	.2850	.6496				
Petkova and Croft (2007)			.6193	.9541		
This paper						
Model 1B	.2725	.6800	.4291	.8912	.4633	.6236
Model 2	.2211	.6302	.4697	.9558	.4142	.5671

Approaches marked with * use additional techniques, e.g., document structure, relevance feedback, priors, etc.

multiple window sizes. Bao et al. (2007) improve personal name identification (based on email aliases) and block-based cooccurance extraction. You, Lu, Li, and Yin (2007) experiment with various weighting methods including query phrase weighting and document field weighting.

It is important to note that the top performing systems at TREC tend to use various kinds of document- or collection-specific heuristics, and involve manual effort which we have avoided here. For example, (Fu et al., 2006 & Yao et al., 2006) exploited the fact that the 2005 queries were names of working groups by giving special treatment to group and personal pages and directly aiming at finding entry pages of working groups and linking people to working groups. Zhu et al., 2007 employed query expansion that "helped the performance of the baseline increase greatly," however there are no details disclosed how this expansion was done. You et al. (2007) tuned parameters manually, using 8 topics from the test set.

At TREC 2007 the emphasis was mainly on extracting candidate names (as the list of possible experts was not given in advance). Two out of the top three teams used the same models as they used in earlier years; Fu et al. (2008) used the candidate-based model proposed in Fu et al. (2006) and Zhu et al. (2008) used the multiple window based co-occurrence model as described in Zhu et al. (2007). Duan et al. (2008) computed an ExpertRank analogous to PageRank, based on the co-occurrence of two experts. Further, they computed a VisualPageRank to degrade pages that are unhelpful or too noisy to establish good evidence of expertise.

The second group of entries in Table 7 (below the header "*Other approaches*") were discussed in the related work section (Section 2).

The last two rows of the table correspond to our best performing candidate-based model (Model 1B) and document-based model (Model 2). Note that we report the numbers for optimal smoothing settings, but use boolean document-candidate associations. The numbers reported on the TREC 2007 data set (corresponding to the same configuration that is used for 2005 and 2006) have been taken from (Balog, 2008). Compared with the official results of the TREC Enterprise track, our best baseline results, using automatic smoothing parameter estimation, would be in the top 3 (based on MAP) for 2005 (.2321, using Model 1 and frequency-based document-candidate associations) and in the top 10 for 2006 (.4803, using Model 2 and frequency-based document-candidate associations).

8.5. *Preferred model*

In the case of Model 2 there is little overhead over document search, which makes it easily deployable in an online application. To see this, observe that Model 2 does not require a separate index to be created, like Model 1, but, given the set of associations, can be applied immediately on top an existing document indexed. In practical terms this means that Model 2 can be implemented using a standard search engine with limited effort and does not require additional indexing, but only a lookup/list of document-candidate associations.

Another reason to prefer the document-based Models 2 and 2B over Models 1 and 1B is that they are less sensitive to the smoothing settings and that they perform close-to-optimal with unsupervised smoothing estimations.

As to Model 2 vs Model 2B, the extension of incorporating co-occurrence information marginally increases the performance of both MAP or MRR. These results suggest that, without a better estimate of $p(t|d, ca)$ using windows w, that Model 2 is preferable. Practically, this means less additional implementation effort, less estimation effort in terms of parameters, and more efficient ranking at almost negligible cost to the effectiveness.

Our experiments showed that Model 2 outperforms Model 1 in many conditions, but, given the right smoothing setting, Model 1B outperforms Model 2 (on the 2005 and 2007 test collections). Moreover, additional features (such as candidate priors, collection structure and relevance feedback) tend to benefit Model 1B more than they help Model 2 (see Balog, 2008).

The upshot, then, is that if a lean and effective approach to expert finding is wanted, to run on top of an existing document search engine, Model 2 is the preferred choice. However, if a highly effective approach is wanted, one in which additional ranking features may be successfully integrated, perhaps at the expense of efficiency, Model 1B is the model of choice.

9. Conclusions

In this paper we introduced a general framework for people related search tasks. We defined two baseline models, both based on language modeling techniques, that implement various expertise search strategies. According to one model (Model 1) we identify expertise by collecting, for every candidate, all documents associated with that candidate and then the determine the prominent topics in these documents. According to the second model (Model 2) we first identify important documents for a given a topic and determine who is most closely associated with these documents. We found that Model 2 was to be preferred over Model 1, both because of effectiveness reasons—in terms of average precision and reciprocal rank—and because Model 2 is easier to implement, only requiring a regular document index. We found that window-based extensions of our baseline models could lead to improved effectiveness, especially on top of Model 1, leading to a model (Model 1B) that outperforms Model 2 in a number of cases. Frequency-based document-candidate associations were especially helpful for Model 1, but also helped improve the effectiveness of Model 2.

The models we have developed in this paper have been shown to be simple, flexible and effective for the expert finding task. These models provide the basic framework which can be extended to incorporate other variables and sources of evidence for better estimates and better performance. However, the models we empirically tested here, did not have any specific knowledge about what it means to be an expert, nor did we use any other a priori knowledge. Yet, they deliver excellent and performance comparable to other state-of-the-art approaches. Since the approach is very general, it can also be applied to search for people in other settings or to locate other named entities such as places, events, organizations. For instance, finding bloggers that talk about a topic (Weerkamp, Balog, & de Rijke, 2008), or describing locations and events by the context in which they are described.

For other future work, we see a number possibilities. First, in our modeling we made a few simplifying assumptions, e.g., by assuming uniform priors on candidates, documents, and document types, and including document fields and structure (see Balog, 2008). Reliably estimating such priors and integrating them in the modeling is an obvious next step. Second, an analysis of our estimation of the smoothing parameter shows that our estimation performs very well on one topic set (the TREC 2006 expert finding topics), for all models, but on the 2005 topic set the results were mixed, suggesting that our optimization method for that year was suboptimal and needs further research. Third, as our topic-level analysis revealed, the choice of optimal model and optimal document-candidate association method depends on the query; how can we reliably perform topic-dependent model and association method selection?

Acknowledgement

Balog and de Rijke were supported by the Netherlands Organisation for Scientific Ripm1203 (NWO) under project number 220-80-001. De Rijke was also supported by NWO under Nos. 017.001.190, 640.001.501, 640.002.501, STE-07-012, 612.061.814, 612.061.815, and by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH Contract IST-033104.

References

Azzopardi, L., Girolami, M., & Crowe, M. (2005). Probabilistic hyperspace analogue to language. In *SIGIR'05: Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 575–576). ACM Press.

Bailey, P., Craswell, N., de Vries, A.P., & Soboroff, I. (2007). Overview of the TREC 2007 enterprise track. In *TREC 2007 Working Notes*.

Balog, K. (2008). *People search in the enterprise*. PhD thesis, University of Amsterdam.

Balog, K., Azzopardi, L., & de Rijke, M. (2006). Formal models for expert finding in enterprise corpora. In *SIGIR'06: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 43–50). New York, NY, USA: ACM Press.

Balog, K., Bogers, T., Azzopardi, L., van den Bosch, A., & de Rijke, M. (2007). Broad expertise retrieval in sparse data environments. In *SIGIR'07: Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 551–558). New York, NY, USA: ACM Press.

Balog, K., & de Rijke, M. (2007a). Determining expert profiles (with an application to expert finding). In *IJCAI'07: Proceedings of the 20th international joint conference on artificial intelligence* (pp. 2657–2662).

Balog, K., & de Rijke, M. (2007b). Finding similar experts. In *30th Annual international ACM SIGIR conference on research and development on information retrieval* (pp. 821–822). ACM Press.

Balog, K., & de Rijke, M. (2008). Associating people and documents. In *30th European conference on information retrieval (ECIR 2008)* (pp. 296–308).

Bao, S., Duan, H., Zhou, Q., Xiong, M., Cao, Y., & Yu, Y. (2007). Research on expert search at enterprise track of TREC 2006. In *The fifteenth text retrieval conference proceedings (TREC 2006)*.

Becerra-Fernandez, I. (2000). The role of artificial intelligence technologies in the implementation of people-finder knowledge management systems. In *Proceedings AAAI 2000 workshop on bringing knowledge to business processes*.

Campbell, C. S., Maglio, P. P., Cozzi, A., & Dom, B. (2003). Expertise identification using email communications. In *CIKM'03: Proceedings of the twelfth international conference on Information and knowledge management* (pp. 528–531). ACM Press.

Cao, Y., Liu, J., Bao, S., & Li, H. (2006). Research on expert search at enterprise track of TREC 2005. In *The fourteenth text retrieval conference proceedings (TREC 2005)*.

Craswell, N., Hawking, D., Vercoustre, A. M., & Wilkins, P. (2001). P@noptic expert: Searching for experts not just for documents. In *Ausweb*. URL: <http://es.csiro.au/pubs/craswell_ausweb01.pdf>.

Craswell, N., Vries, A. D., & Soboroff, I. (2006). Overview of the TREC-2005 enterprise track. In *The fourteenth text retrieval conference proceedings (TREC 2005)*.

Davenport, T. H., & Prusak, L. (1998). *Working knowledge: How organizations manage what they know*. Boston, MA: Harvard Business School Press.

Duan, H., Zhou, Q., Lu, Z., Jin, O., Bao, S., Cao, Y., et al. (2008). Research on enterprise track of TREC 2007 at SJTU APEX Lab. In *The sixteenth text retrieval conference (TREC 2007)*.

ECSCW'99 Workshop (1999). Beyond knowledge management: Managing expertise. URL: <http://www.informatik.uni-bonn.de/~prosec/ECSCW-XMWS/>.

Fang, H., & Zhai, C. (2007). Probabilistic models for expert finding. In *Proceedings of the 29th european conference on information retrieval (ECIR 2007)* (pp. 418–430).

Fu, Y., Xue, Y., Zhu, T., Liu, Y., Zhang, M., & Ma, S. (2008). THUIR at TREC 2007: Enterprise track. In *The sixteenth text retrieval conference (TREC 2007)*.

Fu, Y., Yu, W., Li, Y., Liu, Y., Zhang, M., & Ma, S. (2006). THUIR at TREC 2005: Enterprise track. In *The fourteenth text retrieval conference proceedings (TREC 2005)*.

Hiemstra, D. (2001). *Using language models for information retrieval*. PhD thesis, University of Twente.

Lavrenko, V., & Croft, W. B. (2001). Relevance based language models. In *In SIGIR'01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 120–127). New York, NY, USA: ACM Press.

Macdonald, C., Hannah, D., & Ounis, I. (2008). High quality expertise evidence for expert search. In *Proceedings of 30th European conference on information retrieval (ECIR 2008)* (pp. 283–295).

Macdonald, C., & Ounis, I. (2007a). A belief network model for expert search. In *Proceedings of 1st conference on theory of information retrieval (ICTIR)*.

Macdonald, C., & Ounis, I. (2007b). Voting techniques for expert search. *Knowledge and information systems*.

Mackay, D. J. C., & Peto, L. (1994). A hierarchical dirichlet language model. *Natural Language Engineering, 1*(3), 1–19.

Maybury, M. T. (2006). Expert finding systems. technical report MTR06B000040, The MITRE corporation, Center for integrated intelligence systems, Massachuetts.

Mockus, A., & Herbsleb, J. D. (2002). Expertise browser: A quantitative approach to identifying expertise. In *ICSE'02: Proceedings of the 24th International Conference on software engineering* (pp. 503–512). ACM Press.

Petkova, D., & Croft, W.B. (2006). Hierarchical language models for expert finding in enterprise corpora. *The Proceedings of the 18Th IEEE International Conference on Tools With Artifical Intelligence (ICTAI'06)*, 0:599–608.

Petkova, D., & Croft, W. B. (2007). Proximity-based document representation for named entity retrieval. In *CIKM'07: Proceedings of the sixteenth ACM conference on conference on information and knowledge management* (pp. 731–740). New York, NY, USA: ACM.

Ponte, J. M., & Croft, W. B. (1998). A language modeling approach to information retrieval. In *In SIGIR'98: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval* (pp. 275–281). New York, NY, USA: ACM Press.

Rode, H., Serdyukov, P., Hiemstra, D., & Zaragoza, H. (2007). Entity ranking on graphs: Studies on expert finding. Technical report TR-CTIT-07-81, Centre for telematics and information technology, University of Twente, Enschede.

Serdyukov, P., & Hiemstra, D. (2008). Modeling documents as mixtures of persons for expert finding. In *30th European conference on information retrieval (ECIR 2008)* (pp. 309–320).

Soboroff, I., de Vries, A., & Craswell, N. (2007). Overview of the TREC 2006 enterprise track. In *The fifteenth text retrieval conference proceedings (TREC 2006)*.

TREC (2005). Enterprise track. URL: <http://www.ins.cwi.nl/projects/trec-ent/wiki/>.

W3C (2005). The W3C test collection. URL: <http://research.microsoft.com/users/nickcr/w3c-summary.html>.

Weerkamp, W., Balog, K., & de Rijke, M. (2008). Finding key bloggers, one post at a time. In *18th European conference on artificial intelligence (ECAI 2008)*.

Yao, C., Peng, B., He, J., & Yang, Z. (2006). CNDS expert finding system for TREC2005. In *The fourteenth text retrieval conference proceedings (TREC 2005)*.

Yimam, D. (1996). Expert finding systems for organizations: Domain analysis and the demoir approach. In *ECSCW 999 workshop: Beyond knowledge management: Managing expertise* (pp. 276–283). New York, NY, USA: ACM Press.

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

You, G., Lu, Y., Li, G., & Yin, Y. (2007). Ricoh research at TREC 2006: Enterprise track. In *The fifteenth text retrieval conference proceedings (TREC 2006)*.

Zhai, C., & Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 334–342). ACM Press.

Zhu, J. (2006). W3C Corpus Annotated with W3C People Identity. URL: <http://ir.nist.gov/w3c/contrib/W3Ctagged.html>.

Zhu, J., Song, D., & Rüger, S. (2008). The Open University at TREC 2007 Enterprise Track. In *The sixteenth text retrieval conference (TREC 2007)*.

Zhu, J., Song, D., Ruger, S., Eisenstadt, M., & Motta, E. (2007). The open university at TREC 2006 enterprise track expert search task. In *The fourteenth text retrieval conference proceedings (TREC 2006)*.