Broad Expertise Retrieval in Sparse Data Environments

Krisztian Balog
ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ
Amsterdam, The Netherlands
kbalog@science.uva.nl

Toine Bogers
ILK, Tilburg University
P.O. Box 90153, 5000 LE
Tilburg, The Netherlands
A.M.Bogers@uvt.nl

Leif Azzopardi
Dept. of Computing Science
University of Glasgow,
Glasgow, G12 8QQ
leif@dcs.gla.ac.uk

Maarten de Rijke
ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ
Amsterdam, The Netherlands
mdr@science.uva.nl

Antal van den Bosch
ILK, Tilburg University
P.O. Box 90153, 5000 LE
Tilburg, The Netherlands
Antal.vdnBosch@uvt.nl

ABSTRACT

Expertise retrieval has been largely unexplored on data other than the W3C collection. At the same time, many intranets of universities and other knowledge-intensive organisations offer examples of relatively small but clean multilingual expertise data, covering broad ranges of expertise areas. We first present two main expertise retrieval tasks, along with a set of baseline approaches based on generative language modeling, aimed at finding expertise relations between topics and people. For our experimental evaluation, we introduce (and release) a new test set based on a crawl of a university site. Using this test set, we conduct two series of experiments. The first is aimed at determining the effectiveness of baseline expertise retrieval methods applied to the new test set. The second is aimed at assessing refined models that exploit characteristic features of the new test set, such as the organizational structure of the university, and the hierarchical structure of the topics in the test set. Expertise retrieval models are shown to be robust with respect to environments smaller than the W3C collection, and current techniques appear to be generalizable to other settings.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.4 [**Information Systems Applications**]: H.4.2 Types of Systems; H.4.m Miscellaneous

General Terms

Algorithms, Measurement, Performance, Experimentation

Keywords

Expertise search, expert finding, intranet search, language models

1. INTRODUCTION

An organization's intranet provides a means for exchanging information between employees and for facilitating employee collaborations. To efficiently and effectively achieve this, it is necessary to provide search facilities that enable employees not only to access documents, but also to identify expert colleagues.

At the TREC Enterprise Track [22] the need to study and understand *expertise retrieval* has been recognized through the introduction of Expert Finding tasks. The goal of *expert finding* is to identify a list of people who are knowledgeable about a given topic. This task is usually addressed by uncovering associations between people and topics [10]; commonly, a co-occurrence of the name of a person with topics in the same context is assumed to be evidence of expertise. An alternative task, which using the same idea of people-topic associations, is *expert profiling*, where the task is to return a list of topics that a person is knowledgeable about [3].

The launch of the Expert Finding task at TREC has generated a lot of interest in expertise retrieval, with rapid progress being made in terms of modeling, algorithms, and evaluation aspects. However, nearly all of the expert finding or profiling work performed has been validated experimentally using the W3C collection [24] from the Enterprise Track. While this collection is currently the only publicly available test collection for expertise retrieval tasks, it only represents one type of intranet. With only one test collection it is not possible to generalize conclusions to other realistic settings.

In this paper we focus on expertise retrieval in a realistic setting that differs from the W3C setting—one in which relatively small amounts of clean, multilingual data are available, that cover a broad range of expertise areas, as can be found on the intranets of universities and other knowledge-intensive organizations. Typically, this setting features several additional types of structure: topical structure (e.g., topic hierarchies as employed by the organization), organizational structure (faculty, department, ...), as well as multiple types of documents (research and course descriptions, publications, and academic homepages). This setting is quite different from the W3C setting in ways that might impact upon the performance of expertise retrieval tasks.

We focus on a number of research questions in this paper: Does the relatively small amount of data available on an intranet affect the quality of the topic-person associations that lie at the heart of expertise retrieval algorithms? How do state-of-the-art algorithms developed on the W3C data set perform in the alternative scenario of the type described above? More generally, do the lessons from the Expert Finding task at TREC carry over to this setting? How does the inclusion or exclusion of different documents affect expertise retrieval tasks? In addition to, how can the topical and organizational structure be used for retrieval purposes?

To answer our research questions, we first present a set of baseline approaches, based on generative language modeling, aimed at finding associations between topics and people. This allows us to

SIGIR'07, July 23–27, 2007, Amsterdam, The Netherlands.

formulate the expert finding and expert profiling tasks in a uniform way, and has the added benefit of allowing us to understand the relations between the two tasks. For our experimental evaluation, we introduce a new data set (the UvT Expert Collection) which is representative of the type of intranet that we described above. Our collection is based on publicly available data, crawled from the website of Tilburg University (UvT). This type of data is particularly interesting, since (1) it is clean, heterogeneous, structured, and focused, but comprises a limited number of documents; (2) contains information on the organizational hierarchy; (3) it is bilingual (English and Dutch); and (4) the list of expertise areas of an individual are provided by the employees themselves. Using the UvT Expert collection, we conduct two sets of experiments. The first is aimed at determining the effectiveness of baseline expertise finding and profiling methods in this new setting. A second group of experiments is aimed at extensions of the baseline methods that exploit characteristic features of the UvT Expert Collection; specifically, we propose and evaluate refined expert finding and profiling methods that incorporate topicality and organizational structure.

Apart from the research questions and data set that we contribute, our main contributions are as follows. The baseline models developed for expertise finding perform well on the new data set. While on the W3C setting the expert finding task appears to be more difficult than profiling, for the UvT data the opposite is the case. We find that profiling on the UvT data set is considerably more difficult than on the W3C set, which we believe is due to the large (but realistic) number of topical areas that we used for profiling: about 1,500 for the UvT set, versus 50 in the W3C case. Taking the similarity between topics into account can significantly improve retrieval performance. The best performing similarity measures are content-based, therefore they can be applied on the W3C (and other) settings as well. Finally, we demonstrate that the organizational structure can be exploited in the form of a context model, improving MAP scores for certain models by up to 70%.

The remainder of this paper is organized as follows. In the next section we review related work. Then, in Section 3 we provide detailed descriptions of the expertise retrieval tasks that we address in this paper: expert finding and expert profiling. In Section 4 we present our baseline models, of which the performance is then assessed in Section 6 using the UvT data set that we introduce in Section 5. Advanced models exploiting specific features of our data are presented in Section 7 and evaluated in Section 8. We formulate our conclusions in Section 9.

2. RELATED WORK

Initial approaches to expertise finding often employed databases containing information on the skills and knowledge of each individual in the organization [11]. Most of these tools (usually called yellow pages or people-finding systems) rely on people to self-assess their skills against a predefined set of keywords. For updating profiles in these systems in an automatic fashion there is a need for intelligent technologies [5]. More recent approaches use specific document sets (such as email [6] or software [18]) to find expertise. In contrast with focusing on particular document types, there is also an increased interest in the development of systems that index and mine published intranet documents as sources of evidence for expertise. One such published approach is the P@noptic system [9], which builds a representation of each person by concatenating all documents associated with that person—this is similar to Model 1 of Balog et al. [4], who formalize and compare two methods. Balog et al.'s Model 1 directly models the knowledge of an expert from associated documents, while their Model 2 first locates documents on the topic and then finds the associated experts. In the reported experiments the second method performs significantly better when there are sufficiently many associated documents per candidate. Most systems that took part in the 2005 and 2006 editions of the Expert Finding task at TREC implemented (variations on) one of these two models; see [10, 20]. Macdonald and Ounis [16] propose a different approach for ranking candidate expertise 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. Petkova and Croft [19] propose yet another approach, based on a combination of the above Model 1 and 2, explicitly modeling topics.

Turning to other expert retrieval tasks that can also be addressed using topic–people associations, Balog and de Rijke [3] addressed the task of determining topical expert profiles. While their methods proved to be efficient on the W3C corpus, they require an amount of data that may not be available in the typical knowledge-intensive organization. Balog and de Rijke [2] study the related task of finding experts that are similar to a small set of experts given as input.

As an aside, creating a textual "summary" of a person shows some similarities to biography finding, which has received a considerable amount of attention recently; see e.g., [13].

We use generative language modeling to find associations between topics and people. In our modeling of expert finding and profiling we collect evidence for expertise from multiple sources, in a heterogeneous collection, and integrate it with the co-occurrence of candidates' names and query terms—the language modeling setting allows us to do this in a transparent manner. Our modeling proceeds in two steps. In the first step, we consider three baseline models, two taken from [4] (the Models 1 and 2 mentioned above), and one a refined version of a model introduced in [3] (which we refer to as Model 3 below); this third model is also similar to the model described by Petkova and Croft [19]. The models we consider in our second round of experiments are mixture models similar to contextual language models [1] and to the expanded documents of Tao et al. [21]; however, the features that we use for defininining our expansions—including topical structure and organizational structure—have not been used in this way before.

3. TASKS

In the expertise retrieval scenario that we envisage, users seeking expertise within an organization have access to an interface that combines a search box (where they can search for experts or topics) with navigational structures (of experts and of topics) that allows them to click their way to an expert page (providing the profile of a person) or a topic page (providing a list of experts on the topic).

To "feed" the above interface, we face two expertise retrieval tasks, *expert finding* and *expert profiling*, that we first define and then formalize using generative language models. In order to model either task, the probability of the query topic being associated to a candidate expert plays a key role in the final estimates for searching and profiling. By using language models, both the candidates and the query are characterized by distributions of terms in the vocabulary (used in the documents made available by the organization whose expertise retrieval needs we are addressing).

3.1 Expert finding

Expert finding involves the task of finding the right person with the appropriate skills and knowledge: *Who are the experts on topic X?*. E.g., an employee wants to ascertain who worked on a particular project to find out why particular decisions were made without having to trawl through documentation (if there is any). Or, they may be in need a trained specialist for consultancy on a specific problem.

Within an organization there are usually many possible candidates who could be experts for given topic. We can state this prob-

lem as follows:

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

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

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

where $p(ca)$ is the probability of a candidate and $p(q)$ is the probability of a query. Since $p(q)$ is a constant, it can be ignored for ranking purposes. 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 $p(ca)$ that candidate ca is an expert.

$$p(ca|q) \propto p(q|ca)p(ca) \quad (2)$$

In this paper our main focus is on estimating the probability of a query given the candidate $p(q|ca)$, because this probability captures the extent to which the candidate knows about the query topic. Whereas the candidate priors are generally assumed to be uniform—and thus will not influence the ranking—it has been demonstrated that a sensible choice of priors may improve the performance [20].

3.2 Expert profiling

While the task of expert searching was concerned with finding experts given a particular topic, the task of expert profiling seeks to answer a related question: *What topics does a candidate know about?* Essentially, this turns the questions of expert finding around. The profiling of an individual candidate involves the identification of areas of skills and knowledge that they have expertise about and an evaluation of the level of proficiency in each of these areas. This is the candidate's *topical profile*.

Generally, topical profiles within organizations consist of tabular structures which explicitly catalogue the skills and knowledge of each individual in the organization. However, such practice is limited by the resources available for defining, creating, maintaining, and updating these profiles over time. By focusing on automatic methods which draw upon the available evidence within the document repositories of an organization, our aim is to reduce the human effort associated with the maintenance of topical profiles[1].

A topical profile of a candidate, then, is defined as a vector where each element i of the vector corresponds to the candidate ca's expertise on a given topic k_i, (i.e., $s(ca, k_i)$). Each topic k_i defines a particular knowledge area or skill that the organization uses to define the candidate's topical profile. Thus, it is assumed that a list of topics, $\{k_1, \ldots, k_n\}$, where n is the number of pre-defined topics, is given:

$$profile(ca) = \langle s(ca, k_1), s(ca, k_2), \ldots, s(ca, k_n) \rangle. \quad (3)$$

[1]Context and evidence are needed to help users of expertise finding systems to decide whom to contact when seeking expertise in a particular area. Examples of such context are: *Who does she work with? What are her contact details? Is she well-connected, just in case she is not able to help us herself? What is her role in the organization? Who is her superior?* Collaborators, and affiliations, etc. are all part of the candidate's *social profile*, and can serve as a background against which the system's recommendations should be interpreted. In this paper we only address the problem of determining topical profiles, and leave social profiling to further work.

We state the problem of quantifying the competence of a person on a certain knowledge area as follows:

> What is the probability of a knowledge area (k_i) being part of the candidate's (expertise) profile?

where $s(ca, k_i)$ is defined by $p(k_i|ca)$. Our task, then, is to estimate $p(k_i|ca)$, which is equivalent to the problem of obtaining $p(q|ca)$, where the topic k_i is represented as a query topic q, i.e., a sequence of keywords representing the expertise required.

Both the expert finding and profiling tasks rely on the accurate estimation of $p(q|ca)$. The only difference derives from the prior probability that a person is an expert ($p(ca)$), which can be incorporated into the expert finding task. This prior does not apply to the profiling task since the candidate (individual) is fixed.

4. BASELINE MODELS

In this section we describe our baseline models for estimating $p(q|ca)$, i.e., associations between topics and people. Both expert finding and expert profiling boil down to this estimation. We employ three models for calculating this probability.

4.1 From topics to candidates

Using Candidate Models: Model 1 Model 1 [4] defines the probability of a query given a candidate ($p(q|ca)$) using standard language modeling techniques, based on a multinomial unigram language model. For each candidate ca, a candidate language model θ_{ca} is inferred such that the probability of a term given θ_{ca} is non-zero for all terms, i.e., $p(t|\theta_{ca}) > 0$. From the candidate model the query is generated with the following probability:

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

where each term t in the query q is sampled identically and independently, and $n(t, q)$ is the number of times t occurs in q. The candidate language model is inferred as follows: (1) an empirical model $p(t|ca)$ is computed; (2) it is smoothed with background probabilities. Using the associations between a candidate and a document, the probability $p(t|ca)$ can be approximated by:

$$p(t|ca) = \sum_d p(t|d)p(d|ca),$$

where $p(d|ca)$ is the probability that candidate ca generates a supporting document d, and $p(t|d)$ is the probability of a term t occurring in the document d. We use the maximum-likelihood estimate of a term, that is, the normalised frequency of the term t in document d. The strength of the association between document d and candidate ca expressed by $p(d|ca)$ reflects the degree to which the candidates expertise is described using this document. The estimation of this probability is presented later, in Section 4.2.

The candidate model is then constructed as a linear interpolation of $p(t|ca)$ and the background model $p(t)$ to ensure there are no zero probabilities, which results in the final estimation:

$$p(q|\theta_{ca}) = \prod_{t \in q} \left\{ (1-\lambda) \left(\sum_d p(t|d)p(d|ca) \right) + \lambda p(t) \right\}^{n(t,q)}. \quad (4)$$

Model 1 amasses all the term information from all the documents associated with the candidate, and uses this to represent that candidate. This model is used to predict how likely a candidate would produce a query q. This can can be intuitively interpreted as the probability of this candidate talking about the query topic, where we assume that this is indicative of their expertise.

Using Document Models: Model 2 Model 2 [4] takes a different approach. Here, the process is broken into two parts. Given a candidate ca, (1) a document that is associated with a candidate is selected with probability $p(d|ca)$, and (2) from this document a query q is generated with probability $p(q|d)$. Then the sum over all documents is taken to obtain $p(q|ca)$, such that:

$$p(q|ca) = \sum_{d} p(q|d)p(d|ca). \quad (5)$$

The probability of a query given a document is estimated by inferring a document language model θ_d for each document d in a similar manner as the candidate model was inferred:

$$p(t|\theta_d) = (1-\lambda)p(t|d) + \lambda p(t), \quad (6)$$

where $p(t|d)$ is the probability of the term in the document. The probability of a query given the document model is:

$$p(q|\theta_d) = \prod_{t \in q} p(t|\theta_d)^{n(t,q)}.$$

The final estimate of $p(q|ca)$ is obtained by substituting $p(q|d)$ for $p(q|\theta_d)$ into Eq. 5 (see [4] for full details). Conceptually, Model 2 differs from Model 1 because the candidate is not directly modeled. Instead, the document acts like a "hidden" variable in the process which separates the query from the candidate. This process is akin to how a user may search for candidates with a standard search engine: initially by finding the documents which are relevant, and then seeing who is associated with that document. By examining a number of documents the user can obtain an idea of which candidates are more likely to discuss the topic q.

Using Topic Models: Model 3 We introduce a third model, Model 3. Instead of attempting to model the query generation process via candidate or document models, we represent the query as a topic language model and directly estimate the probability of the candidate $p(ca|q)$. This approach is similar to the model presented in [3, 19]. As with the previous models, a language model is inferred, but this time for the query. We adapt the work of Lavrenko and Croft [14] to estimate a topic model from the query.

The procedure is as follows. Given a collection of documents and a query topic q, it is assumed that there exists an unknown topic model θ_k that assigns probabilities $p(t|\theta_k)$ to the term occurrences in the topic documents. Both the query and the documents are samples from θ_k (as opposed to the previous approaches, where a query is assumed to be sampled from a specific document or candidate model). The main task is to estimate $p(t|\theta_k)$, the probability of a term given the topic model. Since the query q is very sparse, and as there are no examples of documents on the topic, this distribution needs to be approximated. Lavrenko and Croft [14] suggest a reasonable way of obtaining such an approximation, by assuming that $p(t|\theta_k)$ can be approximated by the probability of term t given the query q. We can then estimate $p(t|q)$ using the joint probability of observing the term t together with the query terms, $q_1, \ldots, q_m$, and dividing by the joint probability of the query terms:

$$p(t|\theta_k) \approx p(t|q) = \frac{p(t, q_1, \ldots, q_m)}{p(q_1, \ldots, q_m)} = \frac{p(t, q_1, \ldots, q_m)}{\sum_{t' \in T} p(t', q_1, \ldots, q_m)},$$

where $p(q_1, \ldots, q_m) = \sum_{t' \in T} p(t', q_1, \ldots, q_m)$, and T is the entire vocabulary of terms. In order to estimate the joint probability $p(t, q_1, \ldots, q_m)$, we follow [14, 15] and assume t and $q_1, \ldots, q_m$ are mutually independent, once we pick a source distribution from the set of underlying source distributions U. If we choose U to be a set of document models. then to construct this set, the query q would be issued against the collection, and the top n returned are assumed to be relevant to the topic, and thus treated as samples from the topic model. (Note that candidate models could be used instead.) With the document models forming U, the joint probability of term and query becomes:

$$p(t, q_1, \ldots, q_m) = \sum_{d \in U} p(d)\{p(t|\theta_d) \prod_{i=1}^{m} p(q_i|\theta_d)\}. \quad (7)$$

Here, $p(d)$ denotes the prior distribution over the set U, which reflects the relevance of the document to the topic. We assume that $p(d)$ is uniform across U. In order to rank candidates according to the topic model defined, we use the *Kullback-Leibler* divergence metric (KL, [8]) to measure the difference between the candidate models and the topic model:

$$KL(\theta_k||\theta_{ca}) = \sum_{t} p(t|\theta_k) \log \frac{p(t|\theta_k)}{p(t|\theta_{ca})}. \quad (8)$$

Candidates with a smaller divergence from the topic model are considered to be more likely experts on that topic. The candidate model θ_{ca} is defined in Eq. 4. By using KL divergence instead of the probability of a candidate given the topic model $p(ca|\theta_k)$, we avoid normalization problems.

4.2 Document-candidate associations

For our models we need to be able to estimate the probability $p(d|ca)$, which expresses the extent to which a document d characterizes the candidate ca. In [4], two methods are presented for estimating this probability, based on the number of person names recognized in a document. However, in our (intranet) setting it is reasonable to assume that authors of documents can unambiguously be identified (e.g., as the author of an article, the teacher assigned to a course, the owner of a web page, etc.) Hence, we set $p(d|ca)$ to be 1 if candidate ca is author of document d, otherwise the probability is 0. In Section 6 we describe how authorship can be determined on different types of documents within the collection.

5. THE UVT EXPERT COLLECTION

The UvT Expert collection used in the experiments in this paper fits the scenario outlined in Section 3. The collection is based on the Webwijs ("Webwise") system developed at Tilburg University (UvT) in the Netherlands. Webwijs (`http://www.uvt.nl/webwijs/`) is a publicly accessible database of UvT employees who are involved in research or teaching; currently, Webwijs contains information about 1168 experts, each of whom has a page with contact information and, if made available by the expert, a research description and publications list. In addition, each expert can select expertise areas from a list of 1491 topics and is encouraged to suggest new topics that need to be approved by the Webwijs editor. Each topic has a separate page that shows all experts associated with that topic and, if available, a list of related topics.

Webwijs is available in Dutch and English, and this bilinguality has been preserved in the collection. Every Dutch Webwijs page has an English translation. Not all Dutch topics have an English translation, but the reverse is true: the 981 English topics all have a Dutch equivalent.

About 42% of the experts teach courses at Tilburg University; these courses were also crawled and included in the profile. In addition, about 27% of the experts link to their academic homepage from their Webwijs page. These home pages were crawled and added to the collection. (This means that if experts put the full-text versions of their publications on their academic homepage, these were also available for indexing.) We also obtained 1880 full-text versions of publications from the UvT institutional repository and

converted them to plain text. We ran the `TextCat` [23] language identifier to classify the language of the home pages and the full-text publications. We restricted ourselves to pages where the classifier was confident about the language used on the page.

This resulted in four document types: research descriptions (RD), course descriptions (CD), publications (PUB; full-text and citation-only versions), and academic homepages (HP). Everything was bundled into the UvT Expert collection which is available at `http://ilk.uvt.nl/uvt-expert-collection/`.

The UvT Expert collection was extracted from a different organizational setting than the W3C collection and differs from it in a number of ways. The UvT setting is one with relatively small amounts of multilingual data. Document-author associations are clear and the data is structured and clean. The collection covers a broad range of expertise areas, as one can typically find on intranets of universities and other knowledge-intensive institutes. Additionally, our university setting features several types of structure (topical and organizational), as well as multiple document types. Another important difference between the two data sets is that the expertise areas in the UvT Expert collection are self-selected instead of being based on group membership or assignments by others.

Size is another dimension along which the W3C and UvT Expert collections differ: the latter is the smaller of the two. Also realistic are the large differences in the amount of information available for each expert. Utilizing Webwijs is voluntary; 425 Dutch experts did not select any topics at all. This leaves us with 743 Dutch and 727 English usable expert profiles. Table 2 provides descriptive statistics for the UvT Expert collection.

Universities tend to have a hierarchical structure that goes from the faculty level, to departments, research groups, down to the individual researchers. In the UvT Expert collection we have information about the affiliations of researchers with faculties and institutes, providing us with a two-level organizational hierarchy. Tilburg University has 22 organizational units at the faculty level (including the university office and several research institutes) and 71 departments, which amounts to 3.2 departments per faculty. As to the topical hierarchy used by Webwijs, 131 of the 1491 topics are top nodes in the hierarchy. This hierarchy has an average topic chain length of 2.65 and a maximum length of 7 topics.

6. EVALUATION

We present an experimental evaluation on the UvT Expert collection of Section 4's baseline models for expert finding and profiling. We list our research questions, describe our experimental setup, and then present our results.

	Dutch	English
no. of experts	1168	1168
no. of experts with $\geq$ 1 topic	743	727
no. of topics	1491	981
no. of expert-topic pairs	4318	3251
avg. no. of topics/expert	5.8	5.9
max. no. of topics/expert (no. of experts)	60 (1)	35 (1)
min. no. of topics/expert (no. of experts)	1 (74)	1 (106)
avg. no. of experts/topic	2.9	3.3
max. no. of experts/topic (no. of topics)	30 (1)	30 (1)
min. no. of experts/topic (no. of topics)	1 (615)	1 (346)
no. of experts with HP	318	318
no. of experts with CD	318	318
avg. no. of CDs per teaching expert	3.5	3.5
no. of experts with RD	329	313
no. of experts with PUB	734	734
avg. no. of PUBs per expert	27.0	27.0
avg. no. of PUB citations per expert	25.2	25.2
avg. no. of full-text PUBs per expert	1.8	1.8

Table 2: Descriptive statistics of the Dutch and English versions of the UvT Expert collection.

6.1 Research Questions

We address the following research questions. Both expert finding and profiling rely on the estimations of $p(q|ca)$. The question is how the models compare on the different tasks, and in the setting of the UvT Expert collection. In [4], Model 2 outperformed Model 1 on the W3C collection. How do they compare on our data set? And how does Model 3 compare to Model 1? What about performance differences between the two languages in our test collection?

6.2 Experimental Setup

The output of our models was evaluated against the self-assigned topic labels, which were treated as relevance judgements. Results were evaluated separately for English and Dutch. For English we only used topics for which the Dutch translation was available; for Dutch all topics were considered. The results were averaged for the queries in the intersection of relevance judgements and results; missing queries do not contribute a value of 0 to the scores.

We use standard information retrieval measures, such as Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR). We also report the percentage of topics (%q) and candidates (%ca) covered, for the expert finding and profiling tasks, respectively.

6.3 Results

Table 1 shows the performance of Model 1, 2, and 3 on the expert finding and profiling tasks. The rows of the table correspond to the various document types (RD, CD, PUB, and HP) and to their combinations. RD+CD+PUB+HP is equivalent to the full collection and will be referred as the BASELINE of our experiments.

Looking at Table 1 we see that Model 2 performs the best across the board. However, when the data is clean and very focused (RD), Model 3 outperforms it in a number of cases. Model 1 has the best coverage of candidates (%ca) and topics (%q). The various document types differ in their characteristics and how they improve the finding and profiling tasks. Expert profiling benefits much from the clean data present in the RD and CD document types, while the publications contribute the most to the expert finding task. Adding the homepages does not prove to be particularly useful.

When we compare the results across languages, we find that the coverage of English topics (%q) is higher than of the Dutch ones for expert finding. Apart from that, the scores fall in the same range for both languages. For the profiling task the coverage of the candidates (%ca) is very similar for both languages. However, the performance is substantially better for the English topics.

While it is hard to compare scores across collections, we conclude with a brief comparison of the absolute scores in Table 1 to those reported in [3, 4] on the W3C test set (2005 edition). For expert finding the MAP scores for Model 2 reported here are about 50% higher than the corresponding figures in [4], while our MRR scores are slightly below those in [4]. For expert profiling, the differences are far more dramatic: the MAP scores for Model 2 reported here are around 50% below the scores in [3], while the (best) MRR scores are about the same as those in [3]. The cause for the latter differences seems to reside in the number of knowledge areas considered here—approx. 30 times more than in the W3C setting.

7. ADVANCED MODELS

Now that we have developed and assessed basic language modeling techniques for expertise retrieval, we turn to refined models that exploit special features of our test collection.

7.1 Exploiting knowledge area similarity

Document types	Expert finding			Expert profiling		
	Model 1	Model 2	Model 3	Model 1	Model 2	Model 3
	%q MAP MRR	%q MAP MRR	%q MAP MRR	%ca MAP MRR	%ca MAP MRR	%ca MAP MRR
English						
RD	**97.8** 0.126 0.269	83.5 **0.144 0.311**	83.3 0.129 0.271	**100** 0.089 0.189	39.3 **0.232 0.465**	41.1 0.166 0.337
CD	**97.8** 0.118 0.227	91.7 **0.123 0.248**	91.7 0.118 0.226	**32.8** 0.188 0.381	32.4 0.195 **0.385**	32.7 **0.203** 0.370
PUB	97.8 0.200 0.330	**98.0 0.216 0.372**	**98.0** 0.145 0.257	**78.9** 0.167 0.364	74.5 **0.212 0.442**	**78.9** 0.135 0.299
HP	**97.8 0.081 0.186**	97.4 0.071 0.168	97.2 0.062 0.149	**31.2** 0.150 0.299	28.8 **0.185 0.335**	30.1 0.136 0.287
RD+CD	**97.8** 0.188 0.352	92.9 **0.193 0.360**	92.9 0.150 0.273	**100** 0.145 0.286	61.3 **0.251 0.477**	63.2 0.217 0.416
RD+CD+PUB	97.8 0.235 0.373	**98.1 0.277 0.439**	**98.1** 0.178 0.305	**100** 0.196 0.380	87.2 **0.280 0.533**	89.5 0.170 0.344
RD+CD+PUB+HP	97.8 0.237 0.372	**98.6 0.280 0.441**	98.5 0.166 0.293	**100** 0.199 0.387	88.7 **0.281 0.525**	90.9 0.169 0.329
Dutch						
RD	**61.3** 0.094 0.229	38.4 **0.137 0.336**	38.3 0.127 0.295	**38.0** 0.127 0.386	34.1 **0.138 0.420**	**38.0** 0.105 0.327
CD	**61.3** 0.107 0.212	49.7 0.128 0.256	49.7 **0.136 0.261**	**32.5** 0.151 0.389	31.8 0.158 **0.396**	**32.5 0.170** 0.380
PUB	**61.3** 0.193 0.319	59.5 **0.218 0.368**	59.4 0.173 0.291	**78.8** 0.126 0.364	76.0 **0.150 0.424**	**78.8** 0.103 0.294
HP	**61.3** 0.063 0.169	56.6 **0.064 0.175**	56.4 0.062 0.163	**29.8** 0.108 0.308	27.8 **0.125 0.338**	**29.8** 0.098 0.255
RD+CD	**61.3** 0.159 0.314	51.9 **0.184 0.360**	51.9 0.169 0.324	**60.5** 0.151 0.410	57.2 **0.166 0.431**	60.4 0.159 0.384
RD+CD+PUB	61.3 0.244 0.398	**61.5 0.260 0.424**	61.4 0.210 0.350	**90.3** 0.165 0.445	88.2 **0.189 0.479**	**90.3** 0.126 0.339
RD+CD+PUB+HP	61.3 0.249 0.401	**62.6 0.265 0.436**	**62.6** 0.195 0.344	**91.9** 0.164 0.426	90.1 **0.195 0.488**	**91.9** 0.125 0.328

Table 1: Performance of the models on the expert finding and profiling tasks, using different document types and their combinations. %q is the number of topics covered (applies to the expert finding task), %ca is the number of candidates covered (applies to the expert profiling task). The top and bottom blocks correspond to English and Dutch respectively. The best scores are in boldface.

One way to improve the scoring of a query given a candidate is to consider what other requests the candidate would satisfy and use them as further evidence to support the original query, proportional to how related the other requests are to the original query. This can be modeled by interpolating between the $p(q|ca)$ and the further supporting evidence from all similar requests q', as follows:

$$p'(q|ca) \quad = \quad \lambda p(q|ca) + (1-\lambda) \sum_{q'} p(q|q') p(q'|ca), \quad (9)$$

where $p(q|q')$ represents the similarity between the two topics q and q'. To be able to work with similarity methods that are not necessarily probabilities, we set $p(q|q') = \frac{w(q,q')}{\gamma}$, where γ is a normalizing constant, such that $\gamma = \sum_{q''} w(q'', q')$. We consider four methods for calculating the similarity score between two topics. Three approaches are strictly content-based, and establish similarity by examining co-occurrence patterns of topics within the collection, while the last approach exploits the hierarchical structure of topical areas that may be present within an organization (see [7] for further examples of integrating word relationships into language models).

The Kullback-Leibler (KL) divergence metric defined in Eq. 8 provides a measure of how different or similar two probability distributions are. A topic model is inferred for q and q' using the method presented in Section 4.1 to describe the query across the entire vocabulary. Since a lower KL score means the queries are more similar, we let $w(q, q') = \max(KL(\theta_q||\cdot) - KL(\theta_q||\theta_{q'}))$.

Pointwise Mutual Information (PMI, [17]) is a measure of association used in information theory to determine the extent of independence between variables. The dependence between two queries is reflected by the $SI(q, q')$ score, where scores greater than zero indicate that it is likely that there is a dependence, which we take to mean that the queries are likely to be similar:

$$SI(q, q') = \log \frac{p(q, q')}{p(q)p(q')} \quad (10)$$

We estimate the probability of a topic $p(q)$ using the number of documents relevant to query q within the collection. The joint probability $p(q, q')$ is estimated similarly, by using the concatenation of q and q' as a query. To obtain $p(q|q')$, we then set $w(q, q') = SI(q, q')$ when $SI(q, q') > 0$ otherwise $w(q, q') = 0$, because we are only interested in including queries that are similar.

The *log-likelihood* statistic provides another measure of dependence, which is more reliable than the pointwise mutual information measure [17]. Let k_1 be the number of co-occurrences of q and q', k_2 the number of occurrences of q not co-occurring with q', n_1 the total number of occurrences of q', and n_2 the total number of topic tokens minus the number of occurrences of q'. Then, let $p_1 = k_1/n_1$, $p_2 = k_2/n_2$, and $p = (k_1 + k_2)/(n_1 + n_2)$,

$$\begin{aligned} \ell\ell(q, q') \quad = \quad & 2(\ell(p_1, k_1, n_1) + \ell(p_2, k_2, n_2) \\ & - \ell(p, k_1, n_1) - \ell(p, k_2, n_2)), \end{aligned}$$

where $\ell(p, n, k) = k \log p + (n - k) \log(1 - p)$. The higher $\ell\ell$ score indicate that queries are also likely to be similar, thus we set $w(q, q') = \ell\ell(q, q')$.

Finally, we also estimate the similarity of two topics based on their *distance within the topic hierarchy*. The topic hierarchy is viewed as a directed graph, and for all topic-pairs the shortest path $SP(q, q')$ is calculated. We set the similarity score to be the reciprocal of the shortest path: $w(q, q') = 1/SP(q, q')$.

7.2 Contextual information

Given the hierarchy of an organization, the units to which a person belong are regarded as a context so as to compensate for data sparseness. We model it as follows:

$$\begin{aligned} p'(q|ca) \quad = \quad & \left(1 - \textstyle\sum_{ou \in OU(ca)} \lambda_{ou}\right) \cdot p(q|ca) \\ & + \textstyle\sum_{ou \in OU(ca)} \lambda_{ou} \cdot p(q|ou), \end{aligned}$$

where $OU(ca)$ is the set of organizational units of which candidate ca is a member of, and $p(q|o)$ expresses the strength of the association between query q and the unit ou. The latter probability can be estimated using either of the three basic models, by simply replacing ca with ou in the corresponding equations. An organizational unit is associated with all the documents that its members have authored. That is, $p(d|ou) = \max_{ca \in ou} p(d|ca)$.

7.3 A simple multilingual model

For knowledge institutes in Europe, academic or otherwise, a multilingual (or at least bilingual) setting is typical. The following model builds on a kind of independence assumption: there is no spill-over of expertise/profiles across language boundaries. While a simplification, this is a sensible first approach. That is: $p'(q|ca) = \sum_{l \in L} \lambda_l \cdot p(q_l|ca)$, where L is the set of languages used in the collection, q_l is the translation of the query q to language l, and λ_l is a language specific smoothing parameter, such that $\sum_{l \in L} \lambda_l = 1$.

	Expert finding			Expert profiling		
Language	Model 1	Model 2	Model 3	Model 1	Model 2	Model 3
	%q MAP MRR	%q MAP MRR	%q MAP MRR	%ca MAP MRR	%ca MAP MRR	%ca MAP MRR
English only	97.8 0.237 0.372	98.6 0.280 0.441	98.5 0.166 0.293	**100** 0.199 0.387	88.7 0.281 0.525	90.9 0.169 0.329
Dutch only	61.3 0.249 0.401	62.6 0.265 0.436	62.6 0.195 0.344	91.9 0.164 0.426	90.1 0.195 0.488	91.9 0.125 0.328
Combination	*99.4 0.297 0.444*	**99.7 0.324 0.491**	**99.7** *0.223 0.388*	**100** *0.241 0.445*	*92.1* **0.313 0.564**	*93.2 0.224 0.411*

Table 3: Performance of the combination of languages on the expert finding and profiling tasks (on candidates). Best scores for each model are in italic, absolute best scores for the expert finding and profiling tasks are in boldface.

8. ADVANCED MODELS: EVALUATION

In this section we present an experimental evaluation of our advanced models.

8.1 Research Questions

Our questions follow the refinements presented in the preceding section: Does exploiting the knowledge area similarity improve effectiveness? Which of the various methods for capturing word relationships is most effective? Furthermore, is our way of bringing in contextual information useful? For which tasks? And finally, is our simple way of combining the monolingual scores sufficient for obtaining significant improvements?

8.2 Experimental setup

Given that the self-assessments are also sparse in our collection, in order to be able to measure differences between the various models, we selected a subset of topics, and evaluated (some of the) runs only on this subset. This set is referred as *main topics*, and consists of topics that are located at the top level of the topical hierarchy. (A main topic has subtopics, but is not a subtopic of any other topic.) This main set consists of 132 Dutch and 119 English topics. The relevance judgements were restricted to the main topic set, but were not expanded with subtopics.

8.3 Exploiting knowledge area similarity

Table 4 presents the results. The four methods used for estimating knowledge-area similarity are KL divergence (KLDIV), Pointwise mutual information (PMI), Log-likelihood (LL), and distance within topic hierarchy (HDIST). We managed to improve upon the baseline in all cases, however the improvement is more noticeable for the profiling task. For both tasks, the LL method performed the best. The content-based approaches performed consistently better than HDIST.

Method	Model 1		Model 2		Model 3	
	MAP	MRR	MAP	MRR	MAP	MRR
English						
BASELINE	0.296	0.454	0.339	0.509	0.221	0.333
KLDIV	0.291	0.453	0.327	0.503	0.219	0.330
PMI	0.291	0.453	0.337	0.509	0.219	0.331
LL	**0.319**	**0.490**	**0.360**	**0.524**	**0.233**	**0.368**
HDIST	0.299	0.465	0.346	0.537	0.219	0.332
Dutch						
BASELINE	0.240	0.350	0.271	0.403	0.227	0.389
KLDIV	0.239	0.347	0.253	0.386	0.224	0.385
PMI	0.239	0.350	0.260	0.392	0.227	0.389
LL	**0.255**	**0.372**	**0.281**	**0.425**	0.231	0.389
HDIST	0.253	0.365	0.271	0.407	**0.236**	**0.402**

Method	Model 1		Model 2		Model 3	
	MAP	MRR	MAP	MRR	MAP	MRR
English						
BASELINE	0.485	0.546	0.499	0.548	0.381	0.416
KLDIV	0.510	0.564	0.513	0.558	0.381	0.416
PMI	0.486	0.546	0.495	0.542	0.407	0.451
LL	**0.558**	**0.589**	**0.586**	**0.617**	**0.408**	**0.453**
HDIST	0.507	0.567	0.512	0.563	0.386	0.420
Dutch						
BASELINE	0.263	0.313	0.294	0.358	0.262	0.315
KLDIV	0.284	0.336	0.271	0.321	0.261	0.314
PMI	0.265	0.317	0.265	0.316	0.273	0.330
LL	**0.312**	**0.351**	**0.330**	**0.377**	**0.284**	**0.331**
HDIST	0.280	0.327	0.288	0.341	0.266	0.321

Table 4: Performance on the expert finding (top) and profiling (bottom) tasks, using knowledge area similarities. Runs were evaluated on the main topics set. Best scores are in boldface.

8.4 Contextual information

A two level hierarchy of organizational units (faculties and institutes) is available in the UvT Expert collection. The unit a person belongs to is used as a context for that person. First, we evaluated the models of the organizational units, using all topics (ALL) and only the main topics (MAIN). An organizational unit is considered to be relevant for a given topic (or vice versa) if at least one member of the unit selected the given topic as an expertise area.

Table 5 reports on the results. As far as expert finding goes, given a topic, the corresponding organizational unit can be identified with high precision. However, the expert profiling task shows a different picture: the scores are low, and the task seems hard. The explanation may be that general concepts (i.e., our main topics) may belong to several organizational units.

Second, we performed another evaluation, where we combined the contextual models with the candidate models (to score candidates again). Table 6 reports on the results. We find a positive impact of the context models only for expert finding. Noticably, for expert finding (and Model 1), it improves over 50% (for English) and over 70% (for Dutch) on MAP. The poor performance on expert profiling may be due to the fact that context models alone did not perform very well on the profiling task to begin with.

8.5 Multilingual models

In this subsection we evaluate the method for combining results across multiple languages that we described in Section 7.3. In our setting the set of languages consists of English and Dutch: $L = \{UK, NL\}$. The weights on these languages were set to be identical ($\lambda_{UK} = \lambda_{NL} = 0.5$). We performed experiments with various λ settings, but did not observe significant differences in performance.

Table 3 reports on the multilingual results, where performance is evaluated on the full topic set. All three models significantly improved over all measures for both tasks. The coverage of topics and candidates for the expert finding and profiling tasks, respectively, is close to 100% in all cases. The relative improvement of the precision scores ranges from 10% to 80%. These scores demonstrate that despite its simplicity, our method for combining results over languages achieves substantial improvements over the baseline.

9. CONCLUSIONS

In this paper we focused on expertise retrieval (expert finding

Lang.	Topics	Model 1		Model 2		Model 3	
		MAP	MRR	MAP	MRR	MAP	MRR
Expert finding							
UK	ALL	0.423	0.545	0.654	0.799	0.494	0.629
UK	MAIN	0.500	0.621	0.704	0.834	0.587	0.699
NL	ALL	0.439	0.560	0.672	0.826	0.480	0.630
NL	MAIN	0.440	0.584	0.645	0.816	0.515	0.655
Expert profiling							
UK	ALL	0.240	0.640	0.306	0.778	0.223	0.616
UK	MAIN	0.523	0.677	0.519	0.648	0.461	0.587
NL	ALL	0.203	0.716	0.254	0.770	0.183	0.627
NL	MAIN	0.332	0.576	0.380	0.624	0.332	0.549

Table 5: Evaluating the context models on organizational units.

Lang.	Method	Model 1		Model 2		Model 3	
		MAP	MRR	MAP	MRR	MAP	MRR
Expert finding							
UK	BL	0.296	0.454	0.339	**0.509**	0.221	0.333
UK	CT	**0.330**	**0.491**	**0.342**	0.500	**0.228**	**0.342**
NL	BL	0.240	0.350	**0.271**	0.403	0.227	0.389
NL	CT	**0.251**	**0.382**	0.267	**0.410**	**0.246**	**0.404**
Expert profiling							
UK	BL	0.485	0.546	0.499	0.548	0.381	0.416
UK	CT	**0.562**	**0.620**	0.**508**	**0.558**	**0.440**	**0.486**
NL	BL	0.263	0.313	0.294	0.358	0.262	0.315
NL	CT	**0.330**	**0.384**	**0.317**	**0.387**	**0.294**	**0.345**

Table 6: Performance of the context models (CT) compared to the baseline (BL). Best scores are in boldface.

and profiling) in a new setting of a typical knowledge-intensive organization in which the available data is of high quality, multilingual, and covering a broad range of expertise area. Typically, the amount of available data in such an organization (e.g., a university, a research institute, or a research lab) is limited when compared to the W3C collection that has mostly been used for the experimental evaluation of expertise retrieval so far. To examine expertise retrieval in this setting, we introduced (and released) the UvT Expert collection as a representative case of such knowledge intensive organizations. The new collection reflects the typical properties of knowledge-intensive institutes noted above and also includes several features which may are potentially useful for expertise retrieval, such as topical and organizational structure.

We evaluated how current state-of-the-art models for expert finding and profiling performed in this new setting and then refined these models in order to try and exploit the different characteristics within the data environment (language, topicality, and organizational structure). We found that current models of expertise retrieval generalize well to this new environment; in addition we found that refining the models to account for the differences results in significant improvements, thus making up for problems caused by data sparseness issues.

Future work includes the manual assessments of the automatically generated profiles by the employees themselves.

10. ACKNOWLEDGMENTS

Krisztian Balog was supported by the Netherlands Organisation for Scientific Research (NWO) under project number 220-80-001. Maarten de Rijke was also supported by NWO under project numbers 017.001.190, 220-80-001, 264-70-050, 354-20-005, 600.065.-120, 612-13-001, 612.000.106, 612.066.302, 612.069.006, 640.-001.501, 640.002.501, and by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104. The work of Toine Bogers and Antal van den Bosch was funded by the IOP-MMI-program of SenterNovem / The Dutch Ministry of Economic Affairs, as part of the À Propos project.

11. REFERENCES

[1] L. Azzopardi. *Incorporating Context in the Language Modeling Framework for ad hoc Information Retrieval*. PhD thesis, University of Paisley, 2005.

[2] K. Balog and M. de Rijke. Finding similar experts. In *This volume*, 2007.

[3] K. Balog and M. de Rijke. Determining expert profiles (with an application to expert finding). In *IJCAI '07: Proc. 20th Intern. Joint Conf. on Artificial Intelligence*, pages 2657–2662, 2007.

[4] K. Balog, L. Azzopardi, and M. de Rijke. Formal models for expert finding in enterprise corpora. In *SIGIR '06: Proc. 29th annual intern. ACM SIGIR conf. on Research and development in information retrieval*, pages 43–50, 2006.

[5] I. Becerra-Fernandez. The role of artificial intelligence technologies in the implementation of people-finder knowledge management systems. In *AAAI Workshop on Bringing Knowledge to Business Processes*, March 2000.

[6] C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom. Expertise identification using email communications. In *CIKM '03: Proc. twelfth intern. conf. on Information and knowledge management*, pages 528–531, 2003.

[7] G. Cao, J.-Y. Nie, and J. Bai. Integrating word relationships into language models. In *SIGIR '05: Proc. 28th annual intern. ACM SIGIR conf. on Research and development in information retrieval*, pages 298–305, 2005.

[8] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

[9] N. Craswell, D. Hawking, A. M. Vercoustre, and P. Wilkins. P@noptic expert: Searching for experts not just for documents. In *Ausweb*, 2001.

[10] N. Craswell, A. de Vries, and I. Soboroff. Overview of the TREC-2005 Enterprise Track. In *The Fourteenth Text REtrieval Conf. Proc. (TREC 2005)*, 2006.

[11] T. H. Davenport and L. Prusak. *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, Boston, MA, 1998.

[12] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[13] E. Filatova and J. Prager. Tell me what you do and I'll tell you what you are: Learning occupation-related activities for biographies. In *HLT/EMNLP*, 2005.

[14] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01: Proc. 24th annual intern. ACM SIGIR conf. on Research and development in information retrieval*, pages 120–127, 2001.

[15] V. Lavrenko, M. Choquette, and W. B. Croft. Cross-lingual relevance models. In *SIGIR '02: Proc. 25th annual intern. ACM SIGIR conf. on Research and development in information retrieval*, pages 175–182, 2002.

[16] C. Macdonald and I. Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In *CIKM '06: Proc. 15th ACM intern. conf. on Information and knowledge management*, pages 387–396, 2006.

[17] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.

[18] A. Mockus and J. D. Herbsleb. Expertise browser: a quantitative approach to identifying expertise. In *ICSE '02: Proc. 24th Intern. Conf. on Software Engineering*, pages 503–512, 2002.

[19] D. Petkova and W. B. Croft. Hierarchical language models for expert finding in enterprise corpora. In *Proc. ICTAI 2006*, pages 599–608, 2006.

[20] I. Soboroff, A. de Vries, and N. Craswell. Overview of the TREC 2006 Enterprise Track. In *TREC 2006 Working Notes*, 2006.

[21] T. Tao, X. Wang, Q. Mei, and C. Zhai. Language model information retrieval with document expansion. In *HLT-NAACL 2006*, 2006.

[22] TREC. Enterprise track, 2005. URL: `http://www.ins.cwi.nl/projects/trec-ent/wiki/`.

[23] G. van Noord. TextCat Language Guesser. URL: `http://www.let.rug.nl/~vannoord/TextCat/`.

[24] W3C. The W3C test collection, 2005. URL: `http://research.microsoft.com/users/nickcr/w3c-summary.html`.