Foundations and Trends® in
Information Retrieval
Vol. 6, Nos. 2–3 (2012) 127–256

DOI: 10.1561/1500000024

Expertise Retrieval

By Krisztian Balog, Yi Fang, Maarten de Rijke,
Pavel Serdyukov and Luo Si

Contents

1 Introduction **129**

1.1 The Need for Expertise Retrieval 129
1.2 Challenges in Expertise Retrieval 131
1.3 Organization 132

2 Background **134**

2.1 Expertise Retrieval vs. Expertise Seeking 134
2.2 Early Work 135
2.3 Expertise Retrieval in Information Retrieval 136
2.4 Related Topics and Tasks 139

3 Expertise Retrieval Tasks **142**

3.1 Expertise Retrieval Systems 142
3.2 Two Tasks: Expert Finding and Expert Profiling 145

4 Evaluation **149**

4.1 Measures 149
4.2 Test Collections 150

5 Approaches to Expertise Retrieval **158**

5.1 A Brief Roadmap to Expertise Retrieval Approaches 158
5.2 Generative Probabilistic Models 162
5.3 Discriminative Probabilistic Models 172
5.4 Voting Models 180
5.5 Graph-based Models 185
5.6 Other Models 193
5.7 Comparison of Approaches 196

6 Advanced Components **198**

6.1 Document-candidate Associations 198
6.2 Query Modeling, Relevance Feedback 204
6.3 Document Importance 207
6.4 Document Structure 210
6.5 External Evidence 212
6.6 Candidate Importance 214

7 Discussion **217**

7.1 Practical Considerations 217
7.2 Limitations of Current Approaches 222
7.3 Relation Between Document Retrieval and Expert Finding 224
7.4 Broader Applicability 226
7.5 Emerging Challenges 228

8 Conclusions **232**

Acknowledgments **237**

References **239**

Foundations and Trends® in
Information Retrieval
Vol. 6, Nos. 2–3 (2012) 127–256

DOI: 10.1561/1500000024

Expertise Retrieval

Krisztian Balog[1], Yi Fang[2], Maarten de Rijke[3], Pavel Serdyukov[4] and Luo Si[5]

[1] *Department of Computer and Information Science, NTNU, Sem Sælands vei 7-9, Trondheim, 7491, Norway, krisztian.balog@idi.ntnu.no*
[2] *Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA, yfang.purdue@gmail.com*
[3] *Intelligent Systems Lab Amsterdam, University of Amsterdam, Science Park 904, Amsterdam, 1098 XH, The Netherlands, derijke@uva.nl*
[4] *Yandex, Lva Tolstogo, 16, Moscow, 119021, Russia, pavser@yandex-team.ru*
[5] *Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA, lsi@purdue.edu*

Abstract

People have looked for experts since before the advent of computers. With advances in information retrieval technology and the large-scale availability of digital traces of knowledge-related activities, computer systems that can fully automate the process of locating expertise have become a reality. The past decade has witnessed tremendous interest, and a wealth of results, in expertise retrieval as an emerging subdiscipline in information retrieval. This survey highlights advances in models and algorithms relevant to this field. We draw connections among methods proposed in the literature and summarize them in five groups of basic approaches. These serve as the building blocks for more advanced models that arise when we consider a range of content-based factors that may impact the strength of association between a topic and a person. We also discuss practical aspects of building an expert search system and present applications of the technology in other domains, such as blog distillation and entity retrieval. The limitations of current approaches are also pointed out. We end our survey with a set of conjectures on what the future may hold for expertise retrieval research.

1

Introduction

Believe one who has proved it. Believe an expert.
—Virgil (70 BC–19 BC), *Aeneid*

1.1 The Need for Expertise Retrieval

We call for an expert when we need someone to show us the right path to tackle a problem. There may be large volumes of information available around the problem at hand, but we need an expert to help us find our way. Sometimes the required knowledge is just not freely accessible in digital format. The information that is available might be hard to express in writing or it may be difficult to analyze.

Experts can be in demand not only for being asked for questions, but also for being assigned some role or job in an organizational setting. For instance, conference organizers may search for teams of reviewers, recruiters for talented employees, and consultants may look for other consultants to redirect inquiries and decrease the risk of losing clients.

Research on how to enable people to effectively share expertise can be traced back to at least the 1960s when studies in library and information science explored what sources of information knowledge workers like researchers and engineers use [152]. Subsequent work identified complex information seeking strategies relying on a variety of information sources, including human experts [98, 180]. From results of this type of research grew the realization that the expertise of employees is a major value of an organization and that effective sharing of knowledge can lead to material gains [52, 63, 218].

How do we locate expertise? Relatively early on, the field of knowledge management developed, with the goal of using knowledge within an organization as well as possible. One focus was on developing information systems that could support search for expertise. Initial approaches were mainly focused on how to unify disparate and dissimilar databases of the organization into a single data warehouse that could easily be mined [79, 187]. Resulting tools relied on people to self-assess their skills against a predefined set of keywords, and often employed heuristics generated manually based on current working practice.

Despite the achievements made so far, the question of how to provide effective access to expertise is far from solved, and continues to be addressed from different viewpoints. It has been found that a standard document search engine may be of great help [99], but does not directly address this task: it returns documents, not people. Even in a professional environment, many of us still just "ask around" [101]. According to independent research carried out by Vanson Bourne, who assessed the current information capabilities and needs of 170 organizations in the United Kingdom with more than 1,000 employees, only 55 percent of professional service employees and a mere 27 percent of public sector employees are able to locate expertise using their current enterprise search systems, while 50 percent or more of those surveyed want to be able to locate expertise on a daily basis [167].

But today, as we increasingly live our professional lives online, evidence of expertise can be traced, mined, and organized. Over the past decade, this development, together with the increasingly distributed nature of our working environments, has led to renewed interest in two types of expertise retrieval system: *expert finding* systems (that help answer information needs such as "Find me someone who is an expert on X") and *expert profiling* systems (that help answer information needs such as "Tell me in which topics this person is an expert").

We refer to the general area of linking humans to expertise areas, and vice versa, as *expertise retrieval*.

Expertise retrieval (also known as expertise location or expertise identification) is traditionally regarded as a subject of research in information retrieval. And often, expertise retrieval is taken to mean "expertise retrieval *within a specific organization*." Expertise retrieval is a part of the functionality of a typical enterprise search system, which usually operates within the scope of a single company.

1.2 Challenges in Expertise Retrieval

Finding an expert is a challenging task because *expertise* is a loosely defined concept that is hard to formalize. It is common to refer to expertise as "tacit knowledge" [33], the type of knowledge that people carry in their minds and which is, therefore, difficult to access. It is often contrasted with "explicit knowledge," which is already captured, described, documented, and stored. However, the only way for an expert finding system to assess and access "tacit knowledge" in organizations is through artifacts of "explicit knowledge" (e.g., documents). As a consequence, expertise retrieval inherits many challenges from document retrieval, but there is more to it than just document retrieval.

Expertise retrieval brings new challenges over and above the challenges usually associated with document retrieval. We list the key challenges:

- Candidate experts are usually not represented as retrievable units: they are identified indirectly through the texts associated with them, through authorship, mentions, or citations. We discuss this issue throughout the survey, when talking about test collections (Section 4), about advanced models (Section 6), and about practical considerations (Section 7.1).
- Moreover, expert names are often ambiguous: mentions might be incomplete and a single name may belong to multiple people, even within a single organization; this issue is discussed in Sections 2.4 and 6.1.
- Also, expertise evidence often comes from heterogeneous sources, not all of which are equally important: a brief email

probably carries a different weight than a technical standards document. The heterogeneous nature of expertise evidence is discussed throughout the survey, for instance in Sections 2.3, as part of our discussions of models for expertise retrieval in Section 5, in advanced components that consider document importance and structure (Sections 6.3 and 6.4), and as part of a discussion of practical considerations on expertise retrieval (Section 7.1).
- And finally, determining the strength of the association between a candidate expert and textual evidence of his or her expertise is a complex decision as well. It is the core focus of much of the modeling work presented in Sections 5 and 6.

The challenges listed above make expertise retrieval a multi-faceted research area and building a state-of-the-art expert finding system consists of many steps, each bringing its own scientific challenges.

While there are a number of tasks and problems related to expertise retrieval (see Section 2.4 for a collection of them), to maintain a clear focus throughout this survey, we center our attention around scenarios where an individual wants to contact an expert (as opposed to, for example, building a team of experts). Moreover, we primarily limit ourselves to *topical* aspects of the tasks, thereby largely abstracting away from *cognitive* and *social* considerations. While these are indeed interesting directions, they are research areas on their own and discussing them in detail is beyond the scope of this survey; we highlight, however, some of these works that are of particular relevance within the scope of our survey in Section 7.5.

1.3 Organization

With the multi-faceted nature of expertise retrieval in mind, we structure this survey in the following manner.

To begin, Section 2 focuses on the roots of research on expertise retrieval that served as an inspiration for the approaches developed in the 2000s that form the bulk of the material covered in this survey. Section 3 then introduces the primary tasks on which we focus in this survey: the tasks of expertise retrieval and expertise profiling. Section 4 provides an overview of test collections and evaluation methodology commonly accepted in the research community. Section 5 continues with an overview of approaches and includes probabilistic models (generative and discriminative), voting models, graph-based models, as well as methods that do not fall under any of these headings. Extensions of these models are discussed in Section 6. Section 7 discusses practical considerations including the limitations of current expertise retrieval approaches and recent work aimed at addressing some of them. Finally, Section 8 concludes the survey and identifies some future research directions.

2

Background

2.1 Expertise Retrieval vs. Expertise Seeking

The goal of *expertise retrieval* is to link humans to expertise areas, and vice versa. Research into expertise retrieval has primarily focused on identifying good topical matches between a need for expertise on the one hand and the content of documents associated with candidate experts on the other hand. Much of the research takes a system-centered perspective, which is similar to document search.

In contrast, *expertise seeking* addresses the problem of linking humans to expertise areas from a human-centered perspective. It studies how people search for expertise in the context of a specific task. Expertise seeking has been mainly investigated in the field of knowledge management where the goal is to utilize human knowledge within an organization as well as possible.

In recent years, several studies have tried to combine insights from expertise retrieval with insights from expertise seeking. Amongst others, content-based approaches from expertise retrieval have been combined with factors that may play a role in decisions of what expert to contact or recommend, such as accessibility, reliability, physical proximity, and freshness. We will return to this combination of expertise retrieval and expertise seeking in Sections 7.2 and 7.5; until then, we will focus almost exclusively on content-based approaches to finding expertise.

2.2 Early Work

Research on how to enable people to effectively share expertise can be traced back to at least the 1960s when studies in library and information science explored what sources of information knowledge workers such as researchers and engineers consult [153]. Subsequent work has identified complex information-seeking strategies relying on human experts [98, 99, 180].

As the field of knowledge management was established and developed in the early 1990s, one focus was on developing information systems that could support search for expertise [63]. Early approaches were mainly focused on how to unify disparate databases of the organization into a data warehouse that can be easily mined. The resulting tools were usually called yellow pages, expert locator systems, or expertise management systems [225]. Prominent early systems include Hewlett-Packard's CONNEX, the National Security Agency's KSMS, Microsoft's SpuD, and SAGE People Finder [34, 35]. Please refer to [146, 225] for a comprehensive survey of the early expert locator systems.

These early systems often relied on employees to manually judge their skills against a predefined set of keywords, a task that is both laborious and time-consuming. Moreover, once initial profiles have been created, they soon become outdated and no longer reflect the expertise of an employee accrued through his or her employment. As a consequence, there was an increased demand for intelligent technologies that can automate the process of initializing and updating profiles in expert finding [34]. This demand sparked great interest from the IR community in fully automatically finding experts based on text corpora.

2.3 Expertise Retrieval in Information Retrieval

2.3.1 Work Before the TREC Enterprise Track

Many of the early automatic expertise retrieval systems tended to focus on specific document types. For example, McDonald [147] and McDonald and Ackerman [148] perform a study of locating experts within the technical and support departments of a software company. Mockus and Herbsleb [157] present a tool called Expertise Browser for finding expertise in a collaborative software engineering environment. Others have tried to find expertise residing in email communications, because emails capture candidate experts' activities, interest, and goals in a natural way. Moreover, because people explicitly direct email to one another, social networks are likely to be contained in the patterns of communication. Yimam-Seid and Kobsa [225] provide an overview of early automatic expertise finding systems.

Because of the apparent limitations of the above systems (i.e., focusing on specific document types), both academia and industry had an increased interest in systems that can index and mine heterogeneous sources of evidence accessible within an organization. These systems were meant to enable the search of all kinds of expertise within an organization without being restricted to a single specific domain. The P@noptic system [56] is one of the first published approaches of this kind. The system built representations of each candidate expert by concatenating all the documents within the organization associated with that person. When a query was submitted to the system, it was matched against these representations, as if it were a document retrieval system. Candidates were then ranked according to the similarity of their representation with the query.

2.3.2 The TREC Enterprise Track

The P@noptic system demonstrated the feasibility of expertise retrieval on heterogeneous collections. Based on this insight, an expert finding task was launched as part of the Enterprise Track at the Text REtrieval Conference (TREC) from 2005 to 2008 [10, 29, 54, 198]. The TREC Enterprise Track provided a common platform for researchers to empirically assess methods and techniques devised for expert finding. As a consequence, expert finding received a substantial boost in attention from the IR research community and rapid progress was made, both in modeling, algorithm design, and evaluation.

The TREC Enterprise test collections are based on public facing web pages of large knowledge-intensive organizations, such as the World Wide Web Consortium (W3C) and the Commonwealth Scientific and Industrial Research Organisation (CSIRO); we discuss these in detail in Sections 4.2.1 and 4.2.2, respectively.

At TREC, two principal approaches to expert finding were proposed early on. They were formalized using generative language models in [13] as so-called *candidate* models and *document* models, or *Model 1* and *Model 2*. *Model 1*'s candidate-based approach is also referred to as profile-based method in [83] or query-independent approach in [170]. These approaches build a textual (usually term-based) representation of candidate experts and then rank them based on the query, using traditional *ad-hoc* retrieval models, which is similar to what the aforementioned P@noptic system did [56]. The document models are also referred to as query-dependent approaches in [170]. The idea is to first find documents that are relevant to the topic and then locate the experts associated with these documents. As we will see below, it has been shown that *Model 2* is generally more effective than *Model 1*. Most of the teams participating in the expert finding task at TREC implemented a variant of one of these two approaches. Section 5.2 reviews the generative probabilistic models.

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 associations at the document level, they may be estimated at the paragraph or snippet level. 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 [83], document structure [238], and the use of hierarchical, organizational and topical context and structure [16, 170]. These approaches have been shown to deliver state-of-the-art performance on commonly used test sets. Section 6 reviews these advanced components.

Both candidate and document models are generative language models describing a generative process of how a query or an expert is generated, as does the vast majority of existing work. Recently, discriminative models have been proposed for expert search and they have proved their effectiveness and advantages [88]. *Model 2* and the Arithmetic Mean Discriminative (AMD) discriminative model [88] can be viewed as a classical generative-discriminative pair in the terminology of [163], much like Naive Bayes (NB) and Logistic Regression (LR) for classification, and Hidden Markov Models (HMM) and Conditional Random Fields (CRF) for relational learning [204]. More recently, several other discriminative models have been proposed [140, 158, 201]. These methods have shown competitive empirical performance both on TREC testbeds and in real-world applications. Section 5.3 reviews the discriminative probabilistic models.

Beyond the probabilistic models just sketched, there exist two main alternative families of approaches. Macdonald and Ounis [132] treat the problem of ranking experts as a voting problem based on data fusion techniques. They find that applying field-based weighting models improves the ranking of candidates. Another effective approach is to model the process of expert finding by probabilistic random walks on so-called expertise graphs by relevance propagation [192]. Sections 5.4 and 5.5 review these two approaches in detail.

2.3.3 Work Beyond TREC

Some work has been developed and evaluated based on other testbeds than those provided by TREC. Two prominent examples of testbeds developed outside TREC are the UvT Expert collection [16] and the DBLP collection [72, 73]. Section 4.2 details both of them. In the TREC testbeds, the relationship between documents and experts is ambiguous and therefore a large amount of effort in expert finding research is devoted to modeling candidate-document associations. In contrast, the UvT and DBLP collections have clear candidate-document associations. Moreover, the documents in the UvT collection — as in many realistic scenarios — come from heterogeneous information sources such as publications, course descriptions, homepages, project descriptions, and so on. While the models developed for TREC can be directly applied to this scenario by simply treating the probability of the document-expert association as 1 [16], additional methods have been proposed that exploit the special characteristics of the scenario. For example, in [87], all the documents with the same source can be concatenated without modeling candidate-document associations, and then the effort is dedicated to weighting the importance of various data sources by mixtures of logistic regression models.

2.4 Related Topics and Tasks

Expertise retrieval is closely related to the following tasks or areas:

Enterprise document search. In expert search, a list of people names is returned. In some cases, it would be useful to also produce a list of documents relevant to the query topic. Furthermore, document retrieval is a key ingredient of expertise retrieval and it has a big impact on the end-to-end performance of expertise retrieval systems. In addition, as enterprise users are often willing to express their information needs in a more elaborate form than, say, generic web search engine users, query modeling plays an important role in enterprise document search. Section 6.2 reviews the query modeling aspect of expertise retrieval and Section 7.3 discusses the relations between document search and expert finding.

Finding similar experts. Many web search engines offer a "find similar pages" option next to each page in the result list. It could be quite beneficial to have such a feature for expertise retrieval too [20, 48, 101]. When more than one example is provided, this functionality would be similar to completing a list of names with similar expertise. Moreover, we can infer expertise based (in part) on "expertise-similarity" between people in order to improve expertise retrieval performance.

Expert finding in social networks. Due to the recent rise of online social networks, there is an increased need to find experts in such settings. Email communications are an obvious source for constructing social networks [93, 230]. Further examples concern searching for blogs from the blogosphere that you would like to read [24], as well as the social networks formed by chat logs [81], online discussion forums [231], community-based question answering systems [2, 128], co-authorship information from bibliographic databases [122, 232], or social search [104].

Resource selection. As a task in distributed IR [45], resource selection is the ranking of available document collections in order to select those that are most likely to contain many documents relevant to a given query. Some effective methods for uncooperative environments sample a part of the collection and then sum the relevance of these documents with respect to a query to rank collections [196]. Resource selection bears a great resemblance to document-based expert finding approaches, if we disregard the fact that only a sample of documents is used.

Learning to rank. Learning to rank (L2R) is concerned with automatically constructing a ranking model using training data. It has been successfully applied to many tasks in information retrieval. In essence, expert finding is a ranking problem but so far little work has been done on applying L2R technologies to the task. Most of the existing L2R algorithms have been developed for document search. Section 5.3 reviews recent developments of L2R models for expert search.

Entity retrieval. Expert finding can also be regarded as a specialized entity retrieval task with a restriction for entities to be of the type "people." In general, many more types of entity are usually mentioned in documents and hence can be searched by matching their context to a query. Recently, entity retrieval has attracted increased attention in the IR community. An Entity Ranking track started in 2007 at the INEX workshop [65] and TREC also launched an Entity track in 2009 [25]. Some of the methods developed for expert finding have successfully been adapted and applied to entity retrieval [25].

Web people search. Web people search is a vertical search task: given a query consisting of a person name, find web documents that are relevant to this name [15]. Ambiguity is a serious challenge for people search: names may refer to hundreds and sometimes thousands of people. *Result disambiguation* for web people search is the problem of finding correct referents for all occurrences of the query person name in the search results. The problem has been studied in the Web People Search (WePS) campaigns [5, 6, 7], using search results obtained from a major web search engine; see Section 6.1.1 for further discussions.

Expertise matching. *Expertise matching* is the following problem: given a task or multiple tasks, how do we find a *team* of experts to fulfill the task(s)? Example tasks include conference paper-reviewer assignment, product or proposal reviewer assignment, etc. Interestingly, to address the task of finding a team (as opposed to an individual) one needs to complement content-based rankings of experts and topics with constraints such as diversity, workload, etc. [208]; see Section 5.6 for further discussions.

3

Expertise Retrieval Tasks

In this section we formally introduce the expert finding task (*"Who are the experts on topic X?"*) and the profiling task (*"What topics does person Y know about?"*). We view these tasks as two sides of the same coin; both are cast as a ranking problem, where the set of items to be ranked (people or knowledge areas) are assumed to be given a priori. By way of motivation, we begin by discussing a few real-world examples.

3.1 Expertise Retrieval Systems

To begin with, we briefly introduce a few existing expert search systems. Exploiting the fact that an increasingly large volume of expertise data is available online, several public expert search systems have been recently developed and presented in the literature such as ArnetMiner,[1] INDURE,[2] and Microsoft Academic Search.[3] Most of them are in the academic domain, mainly because the expertise information is often more sensitive in non-academic organizations.

[1] http://www.arnetminer.org/.
[2] http://www.indure.org.
[3] http://academic.research.microsoft.com/.

Fig. 3.1 A ranked list of experts as presented by the ArnetMiner sytem. The panel on the left contains a ranked list of experts returned in response to a topic; the panels on the right-hand side contain lists of conferences and publications related to the topic.

Viewed abstractly, what these systems offer is functionality to go from experts to topics and to go from topics to experts. In addition, they may offer ranking and visualization tools for other types of entity or relations between entities. For instance, ArnetMiner focuses on the computer science domain and provides search results for entities such as experts, conferences, and organizations; see Figure 3.1 for a screen dump of ArnetMiner's expert finding interface. ArnetMiner also allows users to browse those entities in hundreds of specific topics in computer science.

INDURE searches experts in four universities across many disciplines in the State of Indiana in the United States. It provides an expert search function and also allows users to browse expertise information with respect to the ontology of the National Research Council; see Figure 3.2.

Microsoft Academic Search covers many disciplines and also provides rankings of experts and organizations. Like ArnetMiner, Microsoft Academic Search provides graph visualization of relationship information such as co-authors and citations. Figure 3.3 shows the citation graph around an author as visualized by Microsoft Academic Search.

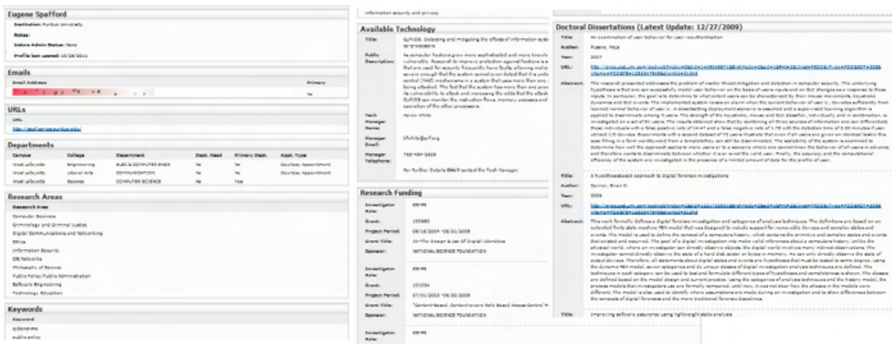

Fig. 3.2 Expert profiling in the INDURE system. The panel on the left shows the contact details and research areas for an expert; the panel in the center lists additional details on technology produced by the expert and research funding obtained; the panel on the right-hand side lists theses supervised by the expert.

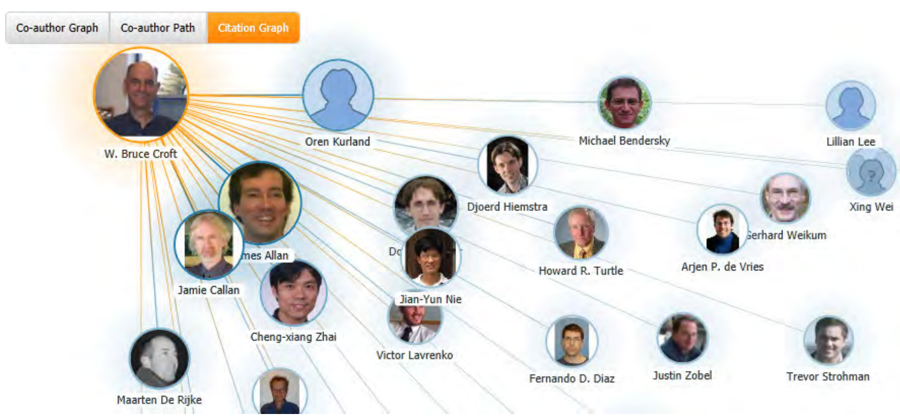

Fig. 3.3 A citation graph in Microsoft Academic Search.

In this survey we mainly focus on modeling and algorithmic aspects of expertise retrieval. Nevertheless, it is important to understand how these retrieval models fit together with other components from which an expertise search engine can be engineered. Similar to other types of search engine, an expertise retrieval system generally consists of the following four components.

Data acquisition. To be able to automatically generate rankings and profiles of experts, expertise evidence needs to be collected from data sources that are available within the organization; see Section 7.1.1 for details.

Preprocessing and indexing. For document retrieval systems, building indexes is a routine task and many open source software tools exist. For expertise retrieval systems there is an additional challenge: occurrences of identifiers of candidate experts (e.g., names or email addresses) need to be recognized in documents and data from heterogeneous sources needs to be integrated around these entities; cf. Section 7.1.1.

Modeling and retrieval. This component is about modeling the association between candidate experts and a user query, and providing a ranking based on the strength of these associations. This is the core focus of later parts of this survey, especially of Sections 5 and 6.

Interaction design. Expertise retrieval systems provide search services and present the results to users in an accessible form. As the examples in this section illustrate, real-world expert search systems usually go beyond a basic keyword-based expert search service. See Section 7.1.2 for further discussions on this point.

3.2 Two Tasks: Expert Finding and Expert Profiling

In this section we detail and formalize the two main expertise retrieval tasks that we consider in this survey: expert finding and expert profiling.

3.2.1 Expert Finding

Expert finding addresses the task of finding the right person 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, the problem is to identify which of these candidates are likely to be an expert. The TREC

2005 Enterprise track (using the W3C collection) defined the expert finding tasks as follows: "Given a topical query, find a list of W3C people who are experts in that topic area" [54]. We assume that each candidate expert (i.e., employee of the organization) is associated with some unique identifier (email address or person ID), which is provided beforehand. Formally, given an input query q, we wish to estimate the level of expertise of each candidate expert e_i, expressed as $score(e_i, q)$, and rank experts in decreasing order of this score. Throughout this survey, we use "expert finding" and "expert search" as interchangeable synonyms.

3.2.2 Expert Profiling

While the task of expert finding is concerned with finding experts given a particular topic, the task of expert profiling turns this around and asks *"What topics does a person know about?"* Balog and de Rijke [19] define the *topical profile* of an individual as "a record of the types and areas of skills and knowledge of that individual, together with an identification of levels of "competency" in each." (The term *topical* is used to emphasize the distinction to other types of profiles, for example, *social* profiles.) Formally, the topical profile of an expert e can be represented as a vector, where the ith element of the vector corresponds to the *knowledge area* ka_i, and $score(e, ka_i)$ reflects the person's knowledge in the given area:

$$profile(e) = \langle score(e, ka_1), score(e, ka_2), \ldots, score(e, ka_n) \rangle.$$

The topical profiling task can naturally be decomposed into two stages: (1) discovering and identifying possible knowledge areas, and (2) measuring the person's competency in each of these areas. Our focus here is on the second step, i.e., on estimating $score(e, ka_i)$, where we assume that the set of possible knowledge areas is given beforehand (for example, defined by the organization [16] or contributed by employees themselves [193]).

3.2.3 Two Sides of the Same Coin

Expert finding and expert profiling are two sides of the same coin. This is easily seen when one visualizes the tasks at hand as computations

Table 3.1. Skills matrix.

	area 1	area 2	area 3	...	area $n-1$	area n
candidate 1	•		•			•
candidate 2		•	•		•	
candidate 3	•	•			•	
⋮						
candidate $m-1$			•		•	
candidate m	•	•				•

using a "skills matrix" in which rows correspond to people and columns to knowledge areas; see Table 3.1. In this simple example, a cell is filled if the candidate in the corresponding row has a certain level of expertise in the area of the corresponding column; in practice, multiple grades are often used to indicate different levels of expertise, ranging, e.g., from "Cannot perform the task" to "Can train others to perform," see, e.g., [124].

A skills matrix is an often-used instrument to capture and represent expertise. Usually, people have to manually input and update their profiles. Becerra-Fernandez [35] identified several drawbacks of this manual approach, having to do both with reliability and scalability. By focusing on automatic methods that 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. This addresses the problems of creating and maintaining the candidate profiles.

Returning to the expert finding and profiling tasks, in terms of a skills matrix, expert finding is the task of filling a column, given a column heading (i.e., given a knowledge area), and expert finding is the task of filling a row, given a row heading (i.e., a candidate expert). While the toy skills matrix in Table 3.1 only has binary values, we view both tasks as ranking tasks. We assume that the labels of the items we want to rank (people for expert finding and knowledge areas for expert profiling) are given. Then, both tasks boil down to automatically estimating cell values, i.e., the level of expertise of a person e on a given topic q, $score(e,q)$ — Sections 5 and 6 are devoted to this. To remain focused, throughout these sections we will primarily concentrate on the expert finding direction; however, as shown in [12, 16], the profiling task generally benefits from improvements made in the context of expert finding.

4

Evaluation

As part of our discussions of expertise retrieval models in Sections 5 and 6, we will occasionally report on the effectiveness of the retrieval methods we consider. To prepare for this, we introduce the test collections and evaluation methodology that we will be using.

We limit ourselves to system-oriented evaluations of expertise retrieval tasks. For the use of alternative evaluation paradigms in expertise retrieval, we refer the reader to [41, 96, 220], which provide good starting points for readers interested in user studies on the subject.

4.1 Measures

According to our usage scenario the user of an expertise retrieval system is interested in a ranked list of answers in response to an information need: experts in case of the expert finding task and knowledge area descriptors in case of the expert profiling task. Our primary interest is in determining the "quality" of these rankings, i.e., measuring the system's ability to rank relevant experts or descriptors above non-relevant ones. In accordance with the tradition established by the TREC community, expert finding methods are evaluated in exactly the same way as document retrieval systems. This is a reasonable choice, since the quality of rankings can be estimated independently of what we rank if quality measures for individual items are alike. Hence, the measures we adopt are standard IR metrics, well-known from document retrieval: (Mean) Average Precision (MAP) and Precision@N (P@N). Under certain usage scenarios the user might be interested only in the top ranked relevant result, in those cases we use (Mean) Reciprocal Rank (MRR). For most test collections the judgment about a person's expertise is binary; in cases when graded assessments are available, Normalized Discounted Cumulated Gain (NDCG) is often reported as well; see [145, Section 8] for general background on these measures.

Not all standard IR metrics are equally natural or appropriate, depending on the (implied) usage scenario. For example, users of expertise retrieval systems have a clear demand for high precision at low ranks, even more so than users of web search engines. The cost of a false recommendation in expert search is much higher than in web search: a conversation with an ignorant person or even the act of reading documents that support an incorrect recommendation from an expertise retrieval system takes much more time than taking a glance at a single non-relevant web page. Therefore, measures that are based on recall alone are quite rare in expertise retrieval research. Besides, for both the expert finding and the expert profiling task it is usually assumed that the set of items to be ranked (experts and knowledge areas, respectively) is given a priori. Since it is not part of the task to discover these, measuring pure recall is often not meaningful.

We limit ourselves to human usage of expert finding or profiling systems and are not discussing measures related to downstream machine processing of the results (misses, false alarms, ROC, etc.).

4.2 Test Collections

When test collections for a retrieval task become available, this is often the trigger for a boost in interest in the task from the IR community. Expert finding is no exception. We describe test collections that have originated from TREC as well as other collections developed by researchers to study aspects missing from TREC collections. Table 4.1

Table 4.1. Overview of test collections. Here, "qrels" denotes relevance judgments, size is given in GB, and "ref" provides a reference.

	Number of						
Content	candidates	docs	queries	qrels	size	public	ref
W3C	1,092	331,037	99	9,860	5.7	Yes	[54]
CERC	≈ 3,500	370,715	127	2,862	4.2	Yes	[11]
UvT	1,168	36,699	1,491	4,318	0.31	Yes	[16]
DBLP+G scholar	574,369	953,774	17	244	20	No	[72]
DBLP in RDF	17,910,795	715,690	—	—	19	No	[103]
ArnetMiner	1,033,050	1,632,440	13	1,781	0.85	Yes	[207]
INDURE	12,535	—	100	6,482	—	No	[87]
Yahoo! Answers	169,819	780,193	67	7,515	—	Yes	[202]
Desktop	—	48,068	6	—	8.1	No	[70]

lists these collections and their key characteristics. For general background on test collection based evaluation of retrieval systems, we refer the reader to [182].

4.2.1 The W3C Collection

The W3C collection,[1] used in the Enterprise Track of the Text REtrieval Conference (TREC) in 2005 and 2006, was the pioneering dataset that initiated research on expert finding within the IR community [54, 198]. It represents the internal documentation of the World Wide Web Consortium (W3C) and was crawled from public W3C (*.w3.org) sites in June 2004. As shown in Table 4.2, the dataset consists of 331,037 documents from several sub-collections: web pages, source code files, mailing lists, etc. Not all parts of the collection are equally useful — for instance, the *dev* part was rarely used despite its size. While there are not so many near-duplicates in the *lists* part, only about 60,000 e-mails are single messages and the rest of them belong to about 21,000 multi-message threads. In contrast, the *www* part contains a lot of "almost near-duplicates," e.g., revisions of the same report document describing W3C standards and guidelines.

The W3C data is supplemented with a list of 1,092 candidate experts represented by their full names and email addresses. Two quite

[1] http://research.microsoft.com/en-us/um/people/nickcr/w3c-summary.html.

Table 4.2. Summary of the W3C collection.

Part	Description	# docs	size (GB)
lists	e-mail discussion forum	198,394	1.855
dev	source code documentation	62,509	2.578
www	web pages	46,975	1.043
esw	wiki	19,605	0.181
other	miscellaneous	3,538	0.047
people	personal homepages	1,016	0.003

different sets of test queries were used by participants in 2005 and 2006. In 2005, 50 queries were created using names of so-called working groups in W3C as topic titles and members of these groups were considered experts on the query topic. An example query topic (EX01) is "Semantic Web Coordination." Judgments were binary, 1 for experts (members) and 0 for non-experts (non-members). Additionally, a set of 10 training topics was made available, also based on W3C working groups; these, however, are seldom used.

In 2006, test queries (49) were contributed by TREC participants and assessments were also created collectively, and manually, based on a set of *supporting* documents provided for each candidate. Here, test queries follow the traditional TREC format and, in addition to the keyword query (i.e., the title field), description and narrative fields are also provided; Figure 4.1 shows an example query topic. A document

```
<num> Number: EX52

<title>ontology engineering</title>

<desc> Description:
Find individuals with expertise regarding ontology engineering.
</desc>

<narr> Narrative:
This topic attempts to find individuals with expertise regarding to
ontology engineering. Ontology engineering concerns the whole
life-cycle of ontologies, such as ontology construction, ontology
learning, ontology mapping, and ontology evolution. We want people
with expertise about ontology engineering rather then other things
related to ontology.
</narr>
```

Fig. 4.1 Example query topic from the TREC 2006 Enterprise track.

is considered *supporting* if it is about the query topic to some extent and mentions the candidate.

The judgment scale was not binary and participants could mark candidates not only as experts and non-experts, but also as "unknown" when they were not sure to which category a candidate belongs. Both sets of test queries and judgments have certain drawbacks. Judgments from 2005 are obviously incomplete, as the members of W3C working groups were probably not the only experts on the topics of these working groups. At the same time, it was not considered whether supporting evidence exists in the document collection for working group members. Judgments from 2006 may suffer from the incompetence of judges — most TREC participants were neither employees of W3C nor experts on the topics underlying the test queries.

4.2.2 The CERC Collection

The CSIRO Enterprise Research Collection (CERC)[2] was the first dataset that used judgments made by employees of the organization at hand, CSIRO (Commonwealth Scientific and Industrial Research Organisation, Australia's national science agency). The collection was used at the Enterprise Track of the Text REtrieval Conference in 2007 and 2008. It is worth pointing out that the definition of an expert changed compared to previous editions of the Enterprise track (2005 and 2006, using the W3C collection) and that it changed again in the second year it was being used [10, 11, 29]. The collection represents a crawl of publicly available pages hosted at the official Web sites (about 100 **.csiro.au* hosts) of CSIRO. The crawl was produced in March 2007. The dataset contains 370,715 documents with a total size of 4.2 GB. There is no official division into sub-collections, but according to [107] around 89% of documents are HTML pages, 4% are pdf, word, or excel documents, and the rest is a mix of multimedia, script, and log files. At least 95% of the pages have one or more outgoing links.

The TREC 2007 and 2008 participants were provided not with a list of candidates but with only a structural template of email addresses used by CSIRO employees: *firstname.lastname@csiro.au*

[2] http://es.csiro.au/cerc/.

(e.g., *John.Doe@csiro.au*). One way in which many participants attempted to generate a complete list of candidate experts was by extracting e-mail addresses from the corpus, which involved getting around spam protection, checking whether similarly looking addresses belong to the same employee, and filtering out non-personal addresses (e.g., *education.act@csiro.au*) [10, 29]. While such an approach makes the expert finding task more complex, it is doubtful whether it becomes more realistic. Usually, all employees are registered with a staff department and hence it should be possible to automatically generate a list of current employees and avoid recommending those who have left the company.

The topic set used in 2007 was created with the help of CSIRO's science communicators. Their responsibilities include interacting with CSIRO industry groups, government agencies, media, and the general public. As part of their job, they act as expert finders on demand: often, the questions they answer are requests to find employees with specific knowledge. The TREC organizers asked several science communicators to develop topics related to the areas of their expertise and provided them with lists of experts. This resulted in 50 test queries, each supplemented with a few "key contacts" — the most authoritative and knowledgeable CSIRO employees on each query topic. The primary requirement was that topics should be broad and important enough to deserve a dedicated overview page at the CSIRO Web site. While it was unknown whether the collection actually contains any evidence of expertise for each of the proclaimed experts, the realism of the experimental setting certainly increased compared to the previous year, when experts were elected by non-experts (participants). However, the number of experts per topic was considerably smaller than in previous years, as only a few most knowledgeable people per topic were regarded as experts (around 3 on average). In 2008, topic descriptions were created again with the help of science communicators, but judgments were made by participants in the same way as in 2006. Test queries (in both years) include a short keyword query and a relatively long narrative, providing some more detailed explanation of the information need behind the query. In 2007, the query topic definition additionally contained some examples of key reference pages (from the

```
<num>CE-013</num>

<query>human clinical trials</query>

<narr>
Overview of the Human Nutrition Clinic facility and the latest
clinical nutrition trials which are recruiting volunteers.
</narr>

<page>CSIRO145-08477954</page>
<page>CSIRO145-11110519</page>
<page>CSIRO142-02910122</page>
```

Fig. 4.2 Example query topic from the TREC 2007 Enterprise track.

organizational intranet) related to the query; an example query topic is shown in Figure 4.2.

4.2.3 The UvT Expert Collection

There exist other collections that were developed outside TREC; the UvT Expert Collection[3] is the largest and most popular amongst them [16]. It was developed using public data about employees of Tilburg University (UvT), the Netherlands. The collection contains information (in English and Dutch) about 1,168 experts. This often includes a page with contact information, research and course descriptions, and publications record (including full-text versions of 1,880 publications). In addition, each expert detailed his/her background by selecting expertise areas from a list of topics (5.8 on average). An example topic is "European Administrative Law." Balog et al. [16] suggested to use 981 of these topics which have both English and Dutch translations for evaluation (in addition, 510 topic names were available only in Dutch). Besides its bilingual character and the fact that the judgments are provided by the employees themselves, the collection has a number of other unique features. For example, all topics are contained in a thesaurus that contains different types of relation between topics (i.e., synonymy, generalization, or just relatedness). Balog et al. [16] also extracted a topic hierarchy with 132 top nodes, an average topic chain length of 2.65, and a maximum length of 7 topics. Besides, all

[3] http://ilk.uvt.nl/uvt-expert-collection/.

candidate experts are placed in an organizational hierarchy whose top level is represented by faculties (and organizational units on the same level such as the University Office), the second level contains all departments (and institutes) within each faculty, and the final level contains the experts in each department.

4.2.4 Other Collections

There are also test collections that have not received as much attention from researchers as the test collections listed above and that focused on different, and often specialized, set-ups. Many of these collections were created for finding experts among academic researchers.

Deng et al. [72] constructed a mash-up of a sample of the DBLP database (953,774 records for articles): they included abstracts of articles downloaded via Google Scholar, plus topical areas of the conferences and journals where they were published, acquired via http://eventseer.net. Assessments were carried out mainly in terms of how many publications an expert candidate has published, how many publications are related to the given query, how many top conference papers she has published, and what distinguished awards she has been awarded. Four grade scores (3, 2, 1, and 0) were assigned, denoting a top expert, expert, marginal expert, and non-expert qualification, respectively. The judgment scores (at levels 3 and 2) were averaged to construct the final ground truth. The data set contained only 7 test queries (extended to 17 in follow-up work [73]).

Similarly, Hogan and Harth [103] describe an expert finding test collection made of the DBLP and CiteSeer databases containing abstracts of computer science publications. The data were integrated and converted into RDF format resulting in the corpus of 19 GB size including 715,690 abstracts.

The ArnetMiner project[4] provides an enriched version of the DBLP database. It comprises a set of 13 test queries from the Computer Science domain with a total of 1,781 relevance judgments,[5] which were

[4] http://www.arnetminer.org/citation.
[5] http://arnetminer.org/lab-datasets/expertfinding.

generated by collecting people from the program committees of important conferences related to the test queries.

Fang et al. [87] conduct experiments on the INDURE (Indiana Database of University Research Expertise) faculty data which come from 4 different sources: (1) the profiles filled out by individual faculty members and/or their department heads, (2) faculty homepages, (3) descriptions of NSF funded projects, and (4) faculty publications and supervised PhD dissertations. The profiles include faculty research areas, which could be keywords from a predefined taxonomy or free keywords that adequately describe the expertise.

The Pilot Challenge of the CriES workshop [202] used a dataset provided by Yahoo! through the Webscope Program[6] that contains questions and answers from Yahoo! Answers. The Challenge identified a subset[7] of this dataset that is suitable for expert search. The questions and answers are mostly written in English, but the dataset also contains German (1%), French (3%), and Spanish (5%) questions. For each language, 15 test queries were chosen. The relevance judgments were created by assessing a pool consisting of the top 10 retrieved experts of all submitted runs to the workshop, for each query topic.

Finally, Demartini and Niederée [70] proposed the task of finding experts using only data from personal desktops. The collection, gathered from desktops of 14 users (researchers), mainly includes e-mails, publications, address books, and calendar appointments. To emulate a standard test collection, all participants provided a set of test queries that reflect typical activities they would perform on their desktop. In addition, each user was asked to contribute activity logs.

[6] http://webscope.sandbox.yahoo.com/index.php.
[7] http://www.multipla-project.org/cries:webscope.

5

Approaches to Expertise Retrieval

The majority of approaches to expertise retrieval cast the task as an exercise in estimating the strength of association between query terms and candidate experts. These associations are established on the basis of (textual) evidence that (identifiers of) people co-occur with.

In this section we start with an overview of approaches to expertise retrieval. The overview identifies five groups: probabilistic models (generative or discriminative), graph-based, based on voting, and others, which we discuss in turn.

5.1 A Brief Roadmap to Expertise Retrieval Approaches

All approaches to expertise retrieval discussed in this section analyze three main ingredients: people, documents, and topics, as shown in Figure 5.1. In the literature, many approaches have been proposed to rank people with respect to their expertise given a query topic. Despite their differences, these approaches all need to address three fundamental questions; in this section we list these, as they allow us to obtain a high-level classification of expertise retrieval methods, based on their answers to these questions.

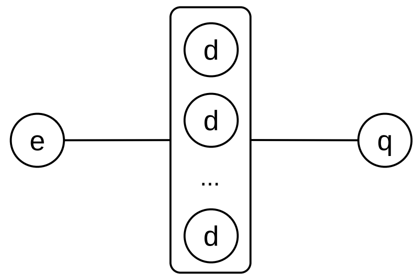

Fig. 5.1 Main ingredients of expertise retrieval: experts (e), topics (q), and documents (d) from which expertise retrieval systems aim to establish a relation between the two.

As pointed out in Section 3, the expert finding and profiling tasks are essentially two sides of the same coin and both boil down to estimating the degree of association between people and topics. For convenience, throughout this section we will be focusing on the expert finding task. We will refer to people as *candidate experts* (or sometimes simply as *candidates*) and to query topics as *queries*.

How to represent candidate experts for the purpose of retrieval? A major challenge in expertise retrieval is that people — unlike documents — are not directly represented as retrievable units. We distinguish between two principal approaches to this issue. *Profile-based methods* "create a textual representation of the individuals' knowledge according to the documents with which they are associated" [13]. These representations (i.e., "pseudo documents" corresponding to people) can then be ranked using standard document retrieval techniques. Since these candidate representations are built off-line, this family of models is also referred to as *query-independent* approaches [170]. *Document-based methods* (also referred to as *query-dependent* approaches [170]) do not directly model the knowledge of a person. They first find documents relevant to the query and then rank candidates mentioned in these documents based on a combination of the document's relevance score and the degree to which the person is associated with that document. A person, therefore, is represented by a weighted set of documents. Because of efficiency considerations, however, most of the time only a query-biased subset of these documents is considered. There are also *hybrid methods* that build candidate profiles in a query-dependent way like the research work that models *documents as mixtures of persons* [191].

What constitutes evidence of expertise? "Commonly, co-occurrence information of the person mentions with the query words in the same context is assumed to be evidence of expertise" [170]. In the simplest case, this context is the document itself, so that "all the evidence within the document is descriptive of the candidate's expertise" [14]. This assumption certainly holds when the person is the author of the document, but can be problematic in many cases (e.g., for long documents with several authors (such as the one you are reading), where each person is responsible for only a specific portion of the document). To better capture the relationship between candidates and terms in documents, one might consider the proximity of terms and candidate mentions 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" [14]. Therefore, terms surrounding candidate mentions form the context of the candidate's expertise and can be defined by a window of a fixed size. "In selecting window sizes, small window sizes often lead to high precision but low recall in finding experts, while large window sizes lead to high recall but low precision" [239]. It is also possible to consider multiple levels of associations in documents, by combining multiple window sizes [14, 239] or by exploiting document structure or metadata [16, 18, 47, 236].

There exist other types of evidence, besides that obtained from the contents of documents. "In an organizational setting, part of a person's knowledge and skills is derived from, and perhaps even characterized by, his or her environment — the knowledge and skills present in colleagues, more broadly, the organization." [12]. Another possibility is to mine information from social relations between people [80, 93, 122, 197, 207, 230, 231, 232]. Most of this survey is focused on textual evidence; we briefly discuss non-textual evidence, such as organization, user, or task dependent factors in Section 7.2.

How to associate query topics to people? The main challenge in any modeling exercise is how to simplify the problem without missing out on important details. In the case of expertise retrieval, the task is commonly modeled in terms of associations between query topics and people (or, viewed more generally, named entities): the stronger the association between a person and a topic, the likelier it is that the person is an expert on that topic. A number of models have been developed to capture these associations between query terms and expert candidates:

- Generative probabilistic models (Section 5.2) estimate associations between query topics and people as the likelihood that the particular topic was generated (i.e., written) by a given candidate (topic generation models) or the other way around, that a probabilistic model based on the query generated the candidate (candidate generation models).
- Discriminative models (Section 5.3) capture associations between query topics and people by directly estimating the binary conditional probability that a given pair of a query topic and a candidate expert is relevant.
- Voting models (Section 5.4) generate associations between query topics and people as a voting process that allows documents ranked with respect to a query to vote for candidate experts by different weight schemes.
- Graph-based models (Section 5.5) determine associations between query models and people by inference on an expertise graph, comprising documents, expert candidates, and different relationships; the graph can be built in a query-dependent or query-independent manner.
- Other models (Section 5.6) use a range of ways of thinking about associations between query topics and people, including modeling people as a distribution of latent variables corresponding to topical themes (author-topic models).

A quick note before we get started: Table 5.1 lists our main notation.

Table 5.1. Variable naming conventions.

Variable	Gloss
q	Query topic
$t \in q$	Query term
$n(t,q)$	Number of times term t occurs in query topic q
e	Person (candidate expert)
d	Document
$n(e,d)$	Number of times person e occurs in document d

5.2 Generative Probabilistic Models

Ever since the first edition of the TREC Enterprise track, generative probabilistic models have formed a popular class of approaches to expertise retrieval, due to their good empirical performance and their potential for incorporating various extensions in a transparent and theoretically sound fashion. Under these models, candidate experts are ranked according to the probability $P(e|q)$, the likelihood of person e being an expert on query q. There are two ways of estimating this probability, and these define the two main sub-classes of generative models. First, it can be computed directly, in which case the candidate e is generated by a probabilistic model based on the query topic q. Following [83], we refer to this family of models as *candidate generation models* and discuss them in Section 5.2.1. Second, by invoking Bayes' Theorem, this probability can be refactored as follows:

$$P(e|q) = \frac{P(q|e)P(e)}{P(q)} \stackrel{\text{rank}}{=} P(q|e)P(e), \tag{5.1}$$

where $P(e)$ 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 e being an expert given query q is proportional to the probability of the query given the candidate ($P(q|e)$), weighted by the *a priori* belief that candidate e is an expert ($P(e)$). This family of models has previously been referred to as *topic generation models* [83]; we present them in Section 5.2.2.

Fang and Zhai [83] derive the above two families of generative models from a general probabilistic framework for expert finding, based on the Probability Ranking Principle [177]. In their derivation relevance is explicitly modeled as a binary random variable. However, estimating the non-relevance models is problematic. "The difficulty comes from the fact that we do not have evidence for a candidate not to be an expert [on the topic q]" [83]. After making the simplifying assumptions thus necessitated, their models correspond to existing expertise retrieval models.

Without exception, the models that we discuss in this section are based on generative language modeling techniques. Some of the models can incorporate other document retrieval models as the underlying document relevance model; this, however, raises challenges when the produced document relevance scores are not true probabilities. Also, as is shown in [236], language models perform slightly better than both TF/IDF and BM25.

The basic idea behind language modeling is to estimate a language model for each document, and then rank documents by the likelihood of the query according to this model, i.e., "what is the probability of observing this query given this document?" This generative notion can then be extended to the problem of ranking experts in various ways, as we shall see in a moment. The approach to which we refer as *standard language modeling* computes the query likelihood $P(q|d)$ as the product of individual term probabilities:

$$P(q|d) = \prod_{t \in q} P(t|d)^{n(t,q)}, \tag{5.2}$$

where $n(t,q)$ denotes the number of times term t is present in query q and $P(t|d)$ is the document language model: the probability that term t is observed in the given document d. In the simplest case, this probability is taken to be a maximum likelihood estimate; that is, the relative frequency of t in d. However, if one ore more query terms do not appear in the document, then the document will be assigned a zero query likelihood, because of the multiplication of the probabilities in Equation (5.2). This is resolved by ensuring that all term probabilities are greater than zero, a technique referred to as *smoothing* [229]. We write $P(t|\theta_d)$ instead of $P(t|d)$ to denote the smoothed document model, whenever it is important to make this distinction. We refer the reader to [227] for a full account on language modeling.

Before continuing, we note that the author-topic model [179], and its various extensions, e.g., [156, 207, 212] are also generative models, but are of a different nature; we will discuss these in Section 5.6.

5.2.1 Candidate Generation Models

Candidate generation models compute the probability of drawing the candidate e from the model estimated by the query q, $P(e|q)$. The sole representative of this family of models is the *two-stage language model*, proposed by Cao et al. [47]. It is an instance of document-centric approaches that combines two components, a relevance model and a co-occurrence model:

$$\begin{aligned} P(e|q) &= \sum_d P(e,d|q) \\ &= \sum_d P(e|d,q)P(d|q), \end{aligned} \tag{5.3}$$

where $P(d|q)$ denotes the (document) relevance model, i.e., the probability that document d is relevant to query q; this is estimated using the standard language modeling approach for document retrieval (note that $P(d|q)$ needs to be rewritten using Bayes' rule, before Equation (5.2) can be applied). The co-occurrence model, denoted by $P(e|d,q)$, represents the extent to which a person is associated with a document given a query. We discuss the estimation of this component in detail in Section 6.1. It is important to note that the actual computation of the co-occurrence model in [47] does not use any information from the query itself, hence (although not stated explicitly) it is assumed that candidate e and query q are conditionally independent, given the document d: $P(e|d,q) = P(e|d)$. Fang and Zhai [83] refer to this model (that makes this independence assumption explicitly) as *candidate generation model*.[1]

[1] Notice that under this simplifying assumption the model is probabilistically equivalent with the *Model 2* approach by Balog et al.[13], discussed in Section 5.2.2; by rewriting both $P(e|d)$ and $P(d|q)$ using Bayes' rule, computing document relevance using the standard query-likelihood method, and moving $P(c)$ and $P(q)$ to the left side of the equation, we arrive at Equation (5.12).

Zhu et al. [239] introduce several extensions to the two-stage model, including incorporating query-independent features, such as PageRank scores, into the document relevance model (see Section 6.3), considering multiple levels of associations via multiple window sizes for the co-occurrence model (see Section 6.1), and utilizing document structure (see Section 6.4).

5.2.2 Topic Generation Models

This family of probabilistic models ranks candidates according to $P(e|q)$, the probability of a candidate e being an expert given the query topic q. Since the query is likely to consist of only a few terms to describe the expertise required, a more accurate estimate can be obtained by invoking Bayes' Theorem as shown in Equation (5.1). Unlike candidate generation models, topic generation models can incorporate candidate importance in a theoretically sound way, in the form of a candidate prior $P(e)$. Candidate priors are generally assumed to be uniform (and so they do not influence the ranking). However, it has been shown that using non-uniform priors can lead to improvements; see, e.g., [83, 171]. We will return to the issue of considering candidate priors in Section 6.6.

Our focus here is on the estimation of $P(q|e)$, the probability that the query topic q has been generated by a probabilistic model based on the candidate (hence the name, *topic generation models*). Balog et al. [13] introduced and formalized two different ways of determining this probability. In their first model, *Model 1* (or *candidate model*), a textual representation of each individual's knowledge, a candidate language model, is built from the documents with which the person is associated. Then, from this representation, the probability of the query topic given the candidate's model is estimated. Their second model, *Model 2* (or *document model*), retrieves the documents that best describe the topic of expertise, and then considers the candidates that are associated with these documents as possible experts. Both *Model 1* and *Model 2* assume a generative process in which candidates generate documents, which, in turn, generate terms. These models make a conditional independence assumption between candidates and terms, meaning that the relationship between words and candidates appearing in the same document is ignored. We also present variations of these models that enable one to capture these dependencies by measuring the proximity between candidate mentions and terms in documents. Finally, we discuss an alternative approach by Serdyukov and Hiemstra [191], under which documents are generators of candidates. Their model combines features from both document-centric and profile-centric methods, but differs from the previous models in that candidates are ranked according to the joint probability $P(e, q)$ (and not to $P(q|e)$). Nevertheless, their approach belongs to this branch of generative models as the generation of topic terms is modeled (although the generators in this case are documents, not candidates). Graphical representations of these topic generation models are shown in Figure 5.2.

5.2.2.1 Using Candidate Models: *Model 1*

According to *Model 1*, a candidate expert e is represented by a multinomial probability distribution over the vocabulary of terms. A candidate model θ_e is inferred for each candidate, such that the probability of a term given the candidate model is $P(t|\theta_e)$. This model is then used to predict how likely a candidate would produce a query q, by taking the product across all the terms in the query (assuming that query terms

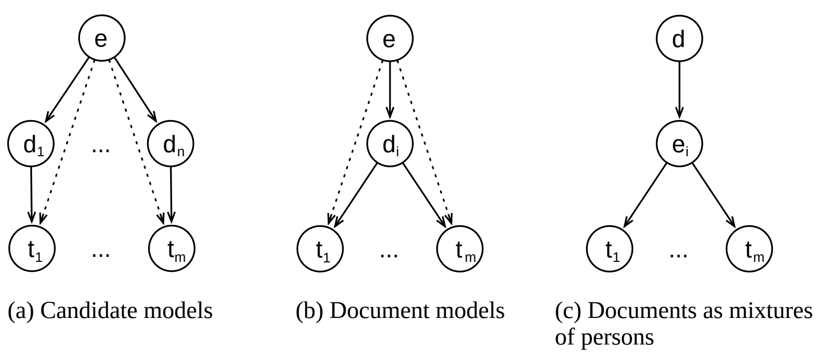

Fig. 5.2 Dependency graphs for topic generation models. The dashed arrows represent the conditional dependence that may exist between terms and entities, given a document. These dependences are not captured in the baseline versions of the models (i.e., in *Model 1* and *Model 2*), only in their proximity-based variations (*Model 1B* and *Model 2B*).

are sampled identically and independently):

$$P(q|\theta_e) = \prod_{t \in q} P(t|\theta_e)^{n(t,q)}. \tag{5.4}$$

To obtain an estimate of $P(t|\theta_e)$, it is necessary to avoid assigning zero probabilities of query terms due to data sparsity. In document language modeling, it is standard to employ smoothing to address this issue:

$$P(t|\theta_e) = (1 - \lambda)P(t|e) + \lambda P(t), \tag{5.5}$$

where $P(t|e)$ is the probability of a term given a candidate, $P(t)$ is the probability of a term in the document repository, and λ is a smoothing parameter. In our context, smoothing adds probability mass to the candidate model according to how likely it is to be generated (i.e., written about) by anyone (on the list of all candidate experts being considered). To approximate $P(t|e)$, documents are used as a bridge to connect the term t and candidate e in the following way:

$$P(t|e) = \sum_{d} P(t|d,e)P(d|e). \tag{5.6}$$

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 ($P(t|d,e)$), weighted by the strength of the association between the document and the candidate ($P(d|e)$). Constructing the candidate model in this way can be viewed as the following generative process: the term t is generated by candidate e by first generating document d from the document collection with probability $P(d|e)$, and then generating the term t from the document d with probability $P(t|d,e)$.[2]

The simplest approach to estimating candidate models assumes that terms and candidates are conditionally independent given a document: $P(t|d,e) = P(t|d)$. $P(t|d)$ is approximated with a standard maximum-likelihood estimate (i.e., the relative frequency of the term in the document). When we put together these choices so far (Equations (5.4),

[2] Notice that $P(d|e)$ can be rewritten using Bayes' rule to $P(e|d)$, which would lead to a different generative process, where documents are generators of persons, similarly to [191]; this reformulation, however, would lead to the removal of candidate priors.

(5.5), and (5.6)), we obtain the following final estimation of the probability of a query given the candidate model:

$$P(e|\theta_q) = \prod_{t \in q} \Big\{ (1-\lambda) \Big(\sum_d P(t|d) P(d|e) \Big) + \lambda P(t) \Big\}^{n(t,q)}. \tag{5.7}$$

Smoothing can have a significant impact on the overall performance of language modeling-based retrieval methods [229]. It is worth pointing out that the original *Model 1* by Balog et al. [13] employs smoothing on the candidate level, and it has been shown that the model is sensitive to the choice of the smoothing parameter [14]. Others [83, 170] treat smoothing differently and use the following simplified version of the candidate model:

$$P(e|\theta_q) = \prod_{t \in q} \Big\{ \sum_d P(t|d) P(d|e) \Big\}^{n(t,q)}. \tag{5.8}$$

Fang and Zhai [83] refer to this model as *topic generation model with profile-based estimation*; it performs smoothing at the document level, that is, $P(t|\theta_d)$ is used for $P(t|d)$. In [170] smoothing is not used at all.

Petkova and Croft [170] propose an extension of the candidate model, referred to as *hierarchical expert model* that combines multiple subcollections (C) to estimate the probability distribution $P(t|\theta_e)$:

$$P(t|\theta_e) = \sum_C \lambda_C P(t|\theta_{e;C}), \tag{5.9}$$

where $P(t|\theta_{e;C})$ is the expert model from subcollection C and $\sum_C \lambda_C = 1$. Other extensions estimate the conditional term probability with respect to both the document and the candidate by considering the proximity of terms and candidate mentions in documents; we review these variations later in this section.

5.2.2.2 Using Document Models: *Model 2*

Instead of creating a term-based representation of a candidate as in the previous subsection, the process of finding an expert can be considered in a slightly different way in which the candidate is not modeled directly. Instead, documents are modeled and queried, then the candidates associated with the documents are considered as possible experts.

Formally, this can be expressed as

$$P(q|e) = \sum_d P(q|d,e)P(d|e), \tag{5.10}$$

where the probability $P(q|d,e)$ is regarded as a measure of how well document d supports the claim that e is an expert on q, and $P(d|e)$ serves as a weight function that indicates how well d describes candidate e. Assuming that query terms are sampled identically and independently, the probability of a query given the candidate and the document is:

$$P(q|d,e) = \prod_{t \in q} P(t|d,e)^{n(t,q)} \tag{5.11}$$

As with candidate models, we can make the simplifying assumption that words and candidates are independent given a document; then, $P(t|d,e) = P(t|\theta_d)$. By substituting Equation (5.11) into Equation (5.10) we obtain the following estimate of the document-based model:

$$P(q|e) = \sum_d \left\{ \prod_{t \in q} P(t|\theta_d)^{n(t,q)} \right\} P(d|e). \tag{5.12}$$

Model 2 is very often chosen as a baseline, as it is intuitive, easy to implement on top of an existing document search engine, and delivers very reasonable performance. As to *Model 1* vs. *Model 2*, Balog [12] concluded that *Model 2* is more robust with respect to parameter settings but is also more insensitive to various extensions, such as document-candidate associations or considering term-candidate proximity. While *Model 1* starts from a lower baseline, it benefits more from these advanced components and with the appropriate parameter settings it can outperform *Model 2* [22].

5.2.2.3 Proximity-based Variations

When using either candidate models or document models, one makes a conditional independence assumption between candidates and terms when computing the probability $P(t|d,e)$ (by setting it to $P(t|d)$). Theoretically, this is a very strong assumption, as it means that the relationship between words and candidates that appear in the same document is ignored. Thus, the person appearing in the document is equally strongly associated with all topics discussed in that document.

To capture the dependence between candidates and terms, Petkova and Croft [171] propose a document representation based on proximity kernels. Specifically, they use the distance between query terms and candidate mentions (measured in term positions) to weight the dependence between q and e. Given a single term t, this candidate-centered document representation is formulated as follows:

$$P(t|d,e) = \frac{1}{Z}\sum_{i=1}^{N} \delta_d(i,t)k(t,e), \tag{5.13}$$

where N is the length of the document, $\delta_d(i,t)$ is an indicator function that returns 1 if the term at position i in d is t, otherwise it returns 0, and Z is a normalizing constant that guarantees that $P(\cdot|d,e)$ is a distribution $\left(Z = \sum_{i=1}^{N} k(t,e)\right)$. Any non-uniform, non-increasing function $k(t,e)$ can be converted into a proximity-based kernel. The simplest kernel is a constant function $(1/N)$ that assigns the same probability to each term in the document, thus corresponds to the bag-of-word representation. Three non-uniform functions are considered in [171]: triangle kernel, Gaussian kernel, and step function. These deliver very similar empirical performance, with no significant differences among them; all non-uniform functions were shown to outperform the baseline constant function.

Balog et al. [13] introduced proximity-based extensions to their models in [14] and termed them *Model 1B* and *Model 2B*. The probability of the term given a document and a candidate expert is estimated within a fixed window of size w, and then these probabilities are combined for multiple window sizes W:

$$P(t|d,e) = \sum_{w \in W} P(t|d,e,w)P(w). \tag{5.14}$$

Equation (5.14) essentially implements the step function based proximity kernel of Petkova and Croft [171]. In [14] only single fixed-size windows are used, with sizes ranging from 20 to 250, based on the suggestions of Cao et al. [47]. Zhu et al. [239] consider the combination of multiple window sizes (although in the context of topic generation models), where smaller windows are given higher weight than larger windows.

5.2.2.4 Modeling Documents as Mixtures of Persons

The topic generation models that we have discussed so far all assume that terms are generated by documents (either on their own or jointly with candidates); see Figures 5.2a and 5.2b. Serdyukov and Hiemstra [191] propose a model in which terms are generated by candidates; the dependencies are depicted in Figure 5.2c. Under this approach, candidate experts are ranked according to their probability of being observed together with query terms in the set of top retrieved documents R, given the query. The joint probability is computed as follows:

$$P(e,q) = \sum_{d \in R} P(q|e)P(e|d)P(d) = P(q|\theta_e) \sum_{d \in R} P(e|d)P(d). \tag{5.15}$$

The summation on the right-hand side of Equation (5.15) can be considered as a person's prior probability, $P(e)$, if set in a query-independent manner. Further, $P(q|\theta_e)$ is the probability of generating the query from the candidate's language model, computed assuming term independence (cf. Equation (5.4)). One way of estimating this probability would be by using candidate models, like *Model 1*. Instead, Serdyukov and Hiemstra [191] model the likelihood of the set of top retrieved documents, R, where documents are represented as mixtures of candidate models and the global language model $P(t)$:

$$P(R) = \prod_{d \in R} \prod_{t \in d} \left\{ (1-\lambda) \left(\sum_{e} P(t|e)P(e|d) \right) + \lambda P(t) \right\}^{n(t,d)}. \tag{5.16}$$

Term generation probabilities from the candidate language model, $P(t|\theta_e)$, are then estimated by maximizing $P(R)$ using the Expectation Maximization (EM) algorithm [71]. Also, the authors propose two ways to estimate association probabilities $P(e|d)$. In one case, they are regarded as observed variables and set using heuristics proposed in [18]. In another case, they are regarded as latent variables and estimated by maximizing the above-mentioned likelihood. The second solution appears to perform better and the proposed mixture model manages to outperform *Model 2*, but only marginally so.

5.3 Discriminative Probabilistic Models

While generative models have been studied for many years in the literature on expert finding, discriminative models, another important class of probabilistic models with a solid statistical foundation, have not been investigated until very recently. There are theoretical results showing that discriminative models tend to have a lower asymptotic error as the training set size increases [163]. Recently, discriminative models have been preferred over generative models in many information retrieval applications such as text classification and information extraction [161] when some training data is available. The success of generative models largely depends on the validity of the model assumptions. These assumptions are sometimes too strong such as the independence assumption of term distributions. Discriminative models typically make fewer model assumptions and they prefer to let the data speak for itself.

Early work on applying discriminative models in information retrieval dates back to the early 1980s in which the maximum entropy approach was investigated to get around the term independence assumption in generative models [53]. More recently, discriminative models have been applied to retrieval problems such as homepage finding [86, 161], e-mail retrieval [222] and question answering [116]. A particularly prominent area in recent years is learning to rank for *ad-hoc* retrieval, which has sparked significant interest in the IR community [126]. Most learning to rank models are discriminative in nature and they have been shown to improve over generative language models in *ad-hoc* retrieval. Benchmark data sets such as LETOR [126] are also available for research on learning to rank.

With respect to expert finding, as shown in previous sections, one key component of generative models is to determine associations between people and documents: associations tend to be ambiguous in the TREC Enterprise settings as well as in many realistic scenarios. In generative models, the number of association signals is very limited but the way of combining them is often heuristic and lacks a clear justification. Another important ingredient in generative models is document evidence, which includes potentially numerous document relevance features. These features include document authority information such as the PageRank, indegree, and URL length [236], internal and external document structures [21], non-local evidence [22], and the evidence that can be acquired outside of an enterprise [190]; see Section 6 for a more detailed discussion. Incorporating more document features (as well as more document-candidate association features) may significantly improve expert finding performance, but it often requires additional and non-trivial modeling assumptions in the generative models.

5.3.1 A Discriminative Learning Framework

In [88], a discriminative learning framework (DLF) is proposed to address the above limitations of generative models in expert finding. DLF is rooted in one of the original IR theories: the probability ranking principle (PRP), which says that one should sort documents by decreasing probability of relevance. Fang and Zhai [83] apply this principle to study the task of expert finding. Formally, let a binary variable $r \in \{1, 0\}$ denote relevance (i.e., 1 is relevance and 0 is non-relevance). Given a candidate e and query q, the odds ratio

$$\log \frac{P(r = 1|e, q)}{P(r = 0|e, q)}$$

is used to rank the candidate. Fang and Zhai [83]'s models utilize Bayes' theorem to reverse the original conditional probability $P(r = 1|e, q)$ to calculate the class conditional $P(e, q|r = 1)$ instead. Fang and Zhai [83] have further shown that their models subsume most existing generative language models proposed for expert finding. Despite the solid theoretical justifications, these models often do not produce good empirical success owing to the difficulty in estimating the class conditionals $P(e, q|r = 1)$ or $P(e, q|r = 0)$. Because the set of relevant documents r is unknown in these generative models, they have to rely on relevance feedback or make possibly inaccurate simplifying assumptions, for instance that $P(e, q|r = 0)$ is uniformly distributed.

In contrast, while retaining the basic framework of probability ranking principle, DLF avoids estimating the class-conditional and instead directly computes $P(r = 1|e, q)$. As stated by Vapnik [214], "one should solve the (classification) problem directly and never solve a more general problem (class-conditional) as an intermediate step." Similar to the Binary Independence Model (BIM, [176]) and [83], DLF re-casts expert finding as a binary classification problem and treats the relevant query-expert pairs as positive data and non-relevant pairs as negative data. Given the relevance judgment r_{mk} for each training expert-query pair (e_k, q_m) which is assumed independently generated, the conditional likelihood L of the training data is as follows

$$L = \prod_{m}^{M} \prod_{k}^{K} P_\theta(r = 1|e_k, q_m)^{r_{mk}} P_\theta(r = 0|e_k, q_m)^{1-r_{mk}}, \qquad (5.17)$$

where M is the number of queries and K is the number of experts. $P(r = 1|e, q)$ is parameterized by θ. $P_\theta(r = 1|e, q)$ can take any proper probability function form with parameter θ. Based on different instantiation of P_θ, the resulting discriminative models may differ; the parameter θ can be estimated by maximizing the likelihood function.

5.3.1.1 Discriminative Models

In [88], two specific probabilistic models are derived from DLF: i.e., the Arithmetic Mean Discriminative (AMD) model and the Geometric Mean Discriminative (GMD) model. AMD directly builds on the same intuition as *Model 2* in which the supporting document d serves as a bridge to connect expert e and query q. More specifically, two factors are considered: (1) whether d is relevant to q; (2) whether e is relevant to d. The final relevance decision for (e, q) is made by averaging over all the documents. Formally, this can be expressed as

$$P_\theta(r = 1|e, q) = \sum_{t=1}^{n} P(r_1 = 1|q, d_t) P(r_2 = 1|e, d_t) P(d_t), \qquad (5.18)$$

where $P(r_1 = 1|q, d_t)$ measures the query-document relevance and $P(r_2 = 1|e, d_t)$ indicates the document-candidate associations. A document d_t with higher values on both probabilities would contribute more to the value of $P_\theta(r = 1|e, q)$. The prior probability of a document, $P(d_t)$, is generally assumed uniform. Both $P(r_1 = 1|q, d_t)$ and $P(r_2 = 1|e, d_t)$ are modeled by a logistic function over a linear combination of features. Formally, they are parameterized as follows:

$$P(r_1 = 1|q, d_t) = \sigma\left(\sum_{i=1}^{N_f} \alpha_i f_i(q, d_t)\right) \tag{5.19}$$

and

$$P(r_2 = 1|e, d_t) = \sigma\left(\sum_{j=1}^{N_g} \beta_j g_j(e, d_t)\right), \tag{5.20}$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the standard logistic function. The term $f_i(q, d_t)$ is a query-document feature such as document retrieval score that indicates the relevance of document with respect to query. The term $g_j(e, d_t)$ is a document-candidate feature such as the Boolean association that describes the strength of document-candidate association. Finally, α_i and β_j are the weight parameters that are learned by maximizing the conditional likelihood of the training data (i.e., Equation (5.17)).

From Equation (5.18), we can see that $P_\theta(r = 1|e, q)$ is essentially the arithmetic mean of $P(r = 1|q, d, e)$ with respect to d. Prior work in machine learning has shown that in certain cases the geometric mean (the product rule) is better than the arithmetic mean (the sum rule) in combining evidence [209]. This observation motivates an alternative discriminative model, which is referred to as the geometric mean discriminative (GMD) model, where $P_\theta(r = 1|e, q)$ is modeled by the geometric mean [88].

5.3.1.2 Generative-discriminative Pair

AMD bears much resemblance to *Model 2* and many other generative models in aggregating document evidence and document-candidate associations through the bridge of documents. They are different in how to estimate these two probabilities. In *Model 2*, the document evidence is computed by language models and the document-candidate associations are estimated by a heuristic combination of document-candidate association features. In AMD, both quantities are modeled by logistic functions with arbitrary features. In essence, *Model 2* and AMD constitute a classical generative-discriminative pair in the terminology of

[163], much like Naive Bayes and logistic regression for classification, and hidden Markov models and conditional random fields for relational learning [204].

As manifested in most of the generative-discriminative pairs, the principal advantage of discriminative models is their ability to use rich, overlapping and complex features. This is especially true and evident for expert finding in which a lot of document features are heterogeneous. Generative language models face difficulties in the meticulous design of different model distributions for these heterogeneous features. In contrast, discriminative models are not plagued by such limitations. As shown in AMD, virtually arbitrary document features and document-candidate features can be effortlessly incorporated into the model. The parameters associated with these features are automatically learned from training data instead of being manually tuned in some generative models. Another advantage of AMD/GMD is that they directly and naturally characterize the notion of relevance. The use of $P(r = 1|e, q)$ instead of $P(e|q)$ makes it explicit that the relevance of an expert is measured with respect to a query; this explicit notion of relevance can help quantify the extent to which a user's information need is satisfied.

5.3.2 Learning to Rank for Expert Search

The task of Learning to Rank (L2R) is to automatically construct a ranking model using training data. In recent years, L2R has been studied extensively for document retrieval. Many L2R algorithms have been proposed in the literature, which can be categorized into three groups: pointwise, pairwise, and listwise.[3] Liu [126] gives a comprehensive survey on this topic.

[3] For the pointwise approach to L2R, the input space contains a feature vector of each single document; the output space contains the relevance score of each single document. For the pairwise approach, the input space contains pairs of documents, both represented by feature vectors; the output space contains the pairwise preference (which takes values from $\{+1, -1\}$) between each pair of documents. The input space of the listwise approach contains a set of documents associated with the query at hand and the output space contains the ranked list of the documents.

In essence, expert search is a ranking problem and thus the existing L2R techniques can be naturally applied to it. In particular, Sorg and Cimiano [201] treat the problem as binary classification and use multilayer perceptrons and logistic regression as classifiers. This falls into the pointwise learning to rank approach. By choosing a set of well designed features, they show the proposed approach outperforms the state-of-the-art generative language models on the dataset extracted from Yahoo! Answers (see Section 4.2). Yang et al. [223] apply Ranking SVM [97] to rank candidate experts for ArnetMiner. Their approach is pairwise learning to rank by predicting the relative order of candidates. Moreira [158] and Moreira et al.[159] conduct a comprehensive evaluation of L2R algorithms for expert search on the DBLP database. Examples from all three groups of approaches (i.e., pointwise, pairwise, and listwise) are tested including Additive Groves, AdaRank, Coordinate Ascent, RankBoost, RankNet, SVMrank, and SVMmap [126]. Experiments show that the Additive Groves pointwise approach and the SVMmap listwise approach outperform all other algorithms. The authors also point out that the pointwise algorithm is very robust. It is worth noticing that the Discriminative Learning Framework (and the AMD and GMD models) in Section 5.3.1 can also be categorized as instances of the pointwise L2R approach. This suggets that pointwise approaches may be preferred in the task of expert search.

5.3.3 Learning Models for Ranking Aggregates

Macdonald and Ounis [140] recently proposed a supervised learning approach to aggregate ranking and apply it to expert and blog search. Their approach essentially falls into a branch of machine learning: ensemble learning [143], which combines many individual learners in an attempt to produce a strong learner. The individual learners in their case are different voting models with three varying factors: document weighting models (e.g., TF.IDF and BM25), ranking cutoffs, and voting techniques (e.g., CombSUM and CombMNZ). This could result in many different instantiations of voting models, which are treated as weak learners. Once these weak learners have been extracted for the sampled candidates, an existing learning to rank technique is then applied to learn the weights associated with these weak learners to generate a stronger ranker.

Ensemble learning has shown good empirical performance on many practical learning tasks and was extensively used in the top performers in the recent Netflix and KDD Cup competitions [36]. The approach is orthogonal to the AMD and GMD learning models. It treats individual models as black boxes to generate features and can be used on top of any expert finding models. In fact, ensemble learning tends to yield better results when there is a significant diversity among the models [118]. Therefore, it is worth exploring to go beyond the class of voting models in [140], and to promote diversity by combining more distinctive classes of models such as *Model 2*, AMD/GMD, and voting models.

5.3.4 Other Discriminative Models

In addition to the work listed above, there exist other discriminative models for expert finding. Cummins et al. [61] use genetic programming to learn a formula for the weights of document associations within the candidate profiles. In particular, they outline two separate approaches: learning query-based aggregation and learning expert-based aggregation. Fang et al. [85, 87] study the issue of differentiating heterogeneous information sources according to specific queries and experts by proposing a series of discriminative models. The work is in a different setting from TREC Enterprise Track: document-expert associations are assumed to be binary, with no ambiguity.

5.3.5 Evaluation

In empirical studies, the proposed discriminative models have demonstrated effectiveness and robustness on the TREC test collections as well as on real-world datasets. Fang et al. [88] conduct a set of experiments to compare AMD and GMD with *Model 2* under different configurations on two TREC Enterprise track corpora (i.e., W3C and CERC). The discriminative models outperformed *Model 2* on various settings. The gaps between them are statistically significant when many features (i.e., 22) are incorporated into AMD/GMD. These results are consistent with those of other IR tasks in the presence of heterogenous features

[161, 204]. Macdonald and Ounis [140]'s ensemble learning approach can generate comparable results to the TREC best runs on CERC without considering the proximity of query and candidate as all the other top runs did. Sorg and Cimiano [201] show that the discriminative models outperform BM25 and the candidate-based *Model 1* on the Yahoo! Answers dataset. Moreira et al. [159] and Moreira [158] also find that all the tested supervised learning to rank algorithms outperform *Model 1* and *Model 2* on the DBLP dataset. Fang et al. [85, 87]'s learning models show significant improvement over *Model 2* and voting models on the INDURE and UvT testbeds. Finally, Cummins et al. [61] also find that high absolute scores can be achieved by their genetic programming learning approach on the TREC test collections.

5.3.6 Discussion

Although discriminative models enjoy a solid theoretical foundation and as well as recent successes in empirical studies, it does not mean they are necessarily superior to generative models. It is worth pointing out the limitations of the discriminative models that may prevent them from widespread use in some expert finding applications. First of all, the number of relevance judgments available may not be sufficient to train discriminative models. Fang et al. [88] chose "out-of-order" training in the experiments as training data is from the 2006 corpus and test data is from 2005 for W3C and training data is from the 2008 corpus and test data is from 2007 for CERC because the setting provides more training data. The "in-order" experiments (i.e., training on 2005 or 2007 and testing on 2006 or 2008, respectively) would allow for fair comparisons with the TREC submitted runs, but they were found not as effective as "out-of-order" training. Thanks to the prevalence of web search, large search engine companies can now utilize abundant implicit feedback from users as training data, but at the time of writing expert finding has not yet reached that stage. A lack of training data may greatly hinder the applicability of discriminative models for expert finding. In addition, the likelihood functions in Equation (5.17) for AMD and GMD are non-convex, which may lead to locally optimal solutions. This is also evident in a lot of learning to rank models [126]. In [88], the BFGS Quasi-Newton method [74] is used for the optimization for AMD and GMD, but it sometimes gets trapped in a local optimum. In practice, good starting points and regularizers are often needed to generate good results. One method that addresses better tuning of the parameters is the annealing procedure [49].

5.3.7 Next Steps

The recent development of discriminative models opens new opportunities. Because there already exists abundant relevant work in other domains, such as learning to rank for *ad-hoc* retrieval, existing techniques can be leveraged for expert finding as shown in Section 5.3.2. We can also study the special characteristics of the expert finding task and design dedicated discriminative models as illustrated with AMD and GMD. Both directions may lead to much improved performance in expert finding because feature engineering can be much more easily done in the discriminative models. Furthermore, generative models may effectively utilize unlabeled or missing data.

It would be interesting to develop a hybrid of discriminative and generative models to obtain the best of both for expert finding. As seen from a 2009 NIPS workshop on the Generative and Discriminative Learning Interface,[4] hybrid models are an active area in machine learning and other applications such as natural language processing, speech recognition, and computer vision. Finally, the recent advent of crowdsourcing is revolutionizing data annotation by dramatically reducing the time, cost, and effort involved. This may expand the applicability of discriminative models in expert finding.

5.4 Voting Models

The Voting Model [132] is inspired by techniques from the field of data fusion. Data fusion techniques, and more generally, ideas of combining evidence from different sources, rankers, or representations, have had a long history in information retrieval, going back at least to the late 1980s [185]. An early retrieval model aimed at combining evidence

[4] http://gen-disc2009.wikidot.com/.

is presented in [213] and early applications of data fusion techniques can be found in meta-search, e.g., [92, 121]. In the context of expert finding, the use of data fusion techniques is based on the intuition that documents ranked with respect to a query can be seen as "voting" for candidate experts that are mentioned in them. Rather than aggregating different systems' scores on a single document as in meta-search, the voting methods for expert search aggregate scores from the same system across members of a document aggregate (i.e., expert candidate). Building on a large number of data fusion techniques, many instantiations of the Voting Model for expert finding have been considered.

The simplest method that Macdonald and Ounis [132] propose assumes that documents provide binary votes given that they appear among the ranked documents returned by a document search engine. Candidate experts are ranked by the number of retrieved documents mentioning them in the text. The authors used this method as a baseline in most follow-up work on the Voting Model. The Reciprocal Rank (RR) [234] data fusion technique has also been adapted for expert search. According to this method, the rank of a document in the combined ranking is determined by the sum of the reciprocal ranks received by the document in each of the individual rankings. Thus, in the case of expert finding, the strength of the votes from all the documents appearing in the document ranking $R(q)$, produced for query q, is computed as follows:

$$Score(e,q) = \sum_{\{d:e \in d, d \in R(q)\}} \frac{1}{rank(d,q)}, \tag{5.21}$$

where $rank(d,q)$ is the rank of document d (containing the mention of candidate expect e) in the document ranking returned by a document retrieval system in response to the query q. Another method, CombSUM [92], considers not the reciprocal ranks, but the actual relevance scores of ranked documents. Finally, the CombMNZ [92] data fusion method can also be put to work as a Voting Model:

$$Score(e,q) = |\{d : e \in d, d \in R(q)\}| \sum_{\{d:e \in d, d \in R(q)\}} s(d,q), \tag{5.22}$$

where $s(d,q)$ represents the document relevance score and $|\{d : e \in d, d \in R(q)\}|$ denotes the number of documents appearing among the ranked documents $R(q)$ returned in response to query q and containing a mention of e. CombMNZ emphasizes both candidates with many ranked documents as well as candidate experts with many associated top ranked documents. Voting Models based on data fusion methods using minimum, maximum, median, and average of relevance scores have also been proposed. Furthermore, Macdonald and Ounis [132] describe variants of the same methods with the score of each document transformed by applying the exponential function (i.e., $\exp(Score)$). Table 5.2 contains short descriptions of instantiations of the Voting Model.

Macdonald and Ounis [138] show that instantiations of the Voting Model can be biased towards candidates with many associated documents, and these candidates are more likely to be mentioned in the retrieved documents (as well as in any document in the collection). This may lead to the situation when a candidate is mentioned in many marginally relevant documents and considered to be more likely an expert than a candidate mentioned in a smaller number of more relevant documents. A proposed solution consists of two normalization techniques. The first approach normalizes the score of the candidate,

Table 5.2. Summary of data fusion techniques and the Voting Models to which they give rise; $M(e)$ is the set of documents that contain a mention of the candidate e and $R(q)$ is the set of documents retrieved in response to the query q [132].

Name	Relevance Score of Candidate is:
Votes	$\|M(e) \cap R(q)\|$
RR	sum of inverse of ranks of documents in $M(e) \cap R(q)$
BordaFuse	sum of K-ranks of documents in $M(e) \cap R(q)$
CombMED	median of scores of documents in $M(e) \cap R(q)$
CombMIN	minimum of scores of documents in $M(e) \cap R(q)$
CombMAX	maximum of scores of documents in $M(e) \cap R(q)$
CombSUM	sum of scores of documents in $M(e) \cap R(q)$
CombANZ	CombSUM$/M(e) \cap R(q)$
CombMNZ	$\|M(e) \cap R(q)\| \cdot$ CombSUM
expCombSUM	sum of exp of scores of documents in $\|M(e) \cap R(q)\|$
expCombANZ	expCombSUM$/\|M(e) \cap R(q)\|$
expCombMNZ	$\|M(e) \cap R(q)\| \cdot$ expCombSUM

as calculated by a voting technique, by the number of potential votes the candidate could receive:

$$Score(e,q)_{Norm_1} = \frac{Score(e,q)}{|\{d : e \in d\}|}, \quad (5.23)$$

where $|\{d : e \in d\}|$ is the number of documents in the collection mentioning e. We refer to this normalization as *Normalization 1*. The second normalization approach (i.e., *Normalization 2*) is more sophisticated:

$$Score(e,q)_{Norm_2} = Score(e,q) \cdot \log_2\left(1 + \alpha * \frac{avg_l}{|\{d : e \in d\}|}\right), \quad (5.24)$$

where avg_l is the average document length in the whole collection and $\alpha > 0$ is a parameter that controls the amount of candidate profile length normalization applied.

5.4.1 Evaluation

Not all data fusion techniques are equally effective for expert search. Macdonald [132], Macdonald and Ounis [129] compare the performance of the 12 data fusion techniques listed in Table 5.2. Among the best performing methods are CombSUM and CombMNZ. Their exponential variants, expCombSUM and expCombMNZ, further improve retrieval performance. The exponential function increases the scores of the top ranked documents much more than the bottom ranked documents, and thus increases the strength of their votes. RR, a rank-based technique, shows good performance on P@10, which suggests that highly ranked documents contribute more to the expertise of a candidate, and should be considered as stronger votes. The importance of the top ranked documents is also illustrated by the good performance of CombMAX, which does not take into account the number of votes for a candidate profile. In contrast, CombANZ, CombMIN, and expCombANZ do not perform well, probably because they focus too much on the lowly scored documents that may not be good indicators of expertise. These results indicate the importance of the underlying document retrieval models; Section 7.3 further investigates this issue in detail.

Macdonald [138], Macdonald and Ounis [129] evaluate the two proposed normalization techniques. They show that *Normalization 2* is generally more effective than *Normalization 1*. The benefit of candidate length normalization varies across the different voting techniques. Specifically, the Votes method is often significantly improved with the application of normalization. This improvement is more often larger for *Normalization 2*. Similar patterns are observed for CombSUM and expCombSUM. In contrast, CombMNZ and expCombMNZ are most often improved with the use of *Normalization 1*. The common feature of these two voting techniques is that they combine the number of votes with the strength of votes. Moreover, CombMAX almost always works best without normalization; it can only receive at most one vote from the document ranking, making the application of candidate length normalization for this technique unnecessary. In addition, the successful application of normalization also depends on the testbed and on how candidates and documents are associated.

5.4.2 Connections with Generative Models

The Voting Model is closely related to probabilistic generative models. In [129], instantiations of the Voting Model are formulated in terms of Bayesian networks [168]. Bayesian networks specify how causes generate effects, and thus they are often called generative models. In the framework of Bayesian networks for expert search, each network is based on two sides to model the dependencies between query terms, documents and candidates. The candidate side of the network specifies the links between the candidates and their associated documents. The query side of the network relates the user query to the keywords it contains, and also links the keywords to the documents that contain them. Based on Bayesian networks, Macdonald [129] derives instantiations of the Voting Model as well as models that are very similar to *Model 2*.

In fact, instantiations of the Voting Model can be viewed to be equivalent to *topic generation models* in Section 5.2.2, under certain conditions. From Equation (5.1), we have

$$P(e|q) = \frac{P(q|e)P(e)}{P(q)} \stackrel{\text{rank}}{=} \sum_d P(q|d)P(d|e)P(e). \tag{5.25}$$

Table 5.3. Equivalence between some representative Voting Models and generative models by varying $P(q|d)$ and $P(e)$ in Equation (5.25).

Name	$P(q|d)$	$P(e)$
Votes	binary	uniform
RR	document reciprocal rank	uniform
CombSUM	document retrieval score	uniform
CombMNZ	document retrieval score	Votes

Specific instantiations of the Voting Model can be derived based on the above equation by varying the measures for $P(q|d)$ and $P(e)$ (the document-candidate association $P(d|e)$ is usually assumed as binary in the Voting Model). For example, the Votes method can be obtained by setting the prior $P(e)$ as uniform and setting $P(q|d)$ as binary (i.e., for all the documents in the candidate profile $M(e)$, $P(q|d) = 1$; otherwise, $P(q|d) = 0$). For the CombMNZ method, $P(e)$ is calculated by the Votes method and $P(q|d)$ is computed based on the document retrieval score (e.g., from language models). It is worth noting that the document relevance scores in the generative models are probabilities, and thus normalized, while scores in the Voting Model are often not normalized. Table 5.3 summarizes how some representative instantiations of the Voting Model are related to the generative models by varying $P(q|d)$ and $P(e)$. Equivalences between other instantiations of the Voting Model and generative models can be obtained in a similar way.

5.5 Graph-based Models

The approaches described earlier in this section build upon an analysis of the textual content of documents to which a candidate expert is *directly related*: he or she can be mentioned in the metadata as an author, or cited somewhere in the body of the document. Indeed, the narrower the textual context surrounding a person, the more valuable the evidence of expertise is that we find in such contexts. Consequently, the methods that analyze the relevance of the text surrounding the mention of a person in the document at the paragraph, sentence, or even term level are usually highly effective [22, 171] (see Section 5.2.2.3). While analyzing documents that directly mention a person naturally proves to be an effective approach to identifying expertise evidence of the person, there may also be such evidence outside the scope of these documents.

In the search for evidence of expertise, some expert finding methods consider various relations among people and documents, both implicit and explicit ones. Such relations are well represented by *expertise graphs*: both documents and candidate experts become vertices and directed edges symbolize containment conditions. An example of such a graph is depicted in Figure 5.3. As we can see, the graph contains many small and disconnected components. However, a significant number of candidate experts are contained in a large connected component.

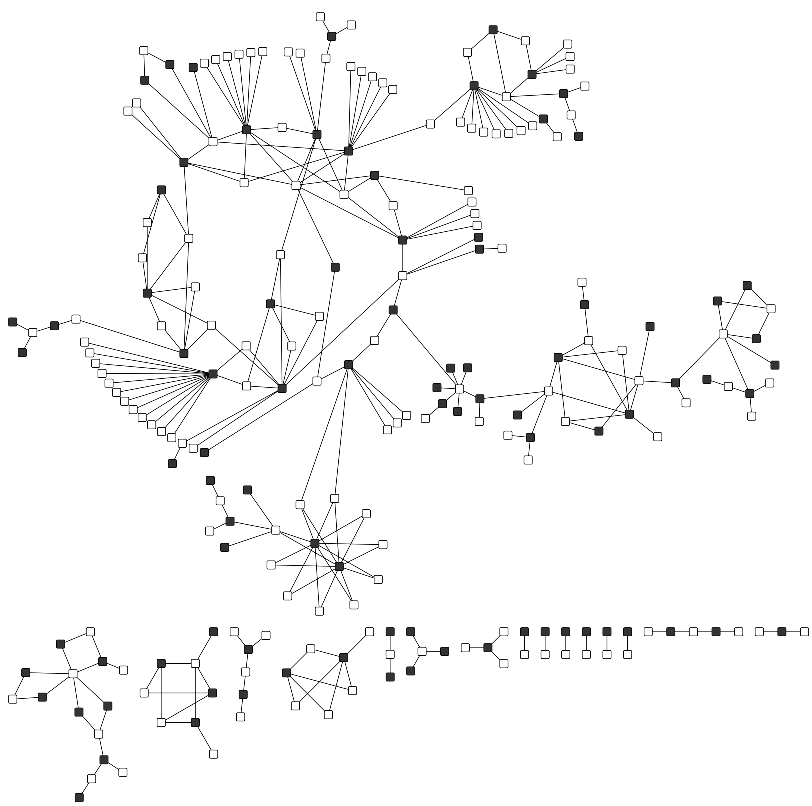

Fig. 5.3 A fragment of the expertise graph with links between documents (white nodes) and candidate experts (black nodes) for the query "sustainable ecosystems" from the CERC dataset (see Section 4.2.2).

This example demonstrates that people with similar expertise tend to be close to each other in an organization, even if the relations among them are built via co-occurrences in documents only without considering their professional relations due to the organizational structure. Often, not only document-person containment relations are known, but also links between documents or organizational connections between candidate experts. Document links are then represented by directed edges. When modeling organizations, person-to-person edges are often symmetric as they represent collaborative relationships, especially for employees working in the same or related sub-divisions, while this may not be the case in general, e.g., for more informal social networks.

Experts and documents do not need to be the only entities in an expertise graph. Even though we are only interested in ranking experts, it might be useful to exploit additional connections via nodes of other types, such as dates, locations, and events. Moreover, organizational units may serve either as mediators in the search for employees, or as the actual objects of search. People outside the company and in other companies may reveal interesting connections as well, if they are somehow related to the entities mentioned above. For instance, it may be useful to incorporate relations extracted from external global professional social networks (e.g., LinkedIn[5]).

The size and density of an expertise graph may depend on the query and the number of retrieved documents. However, in the simplest case, the graph is query-independent and includes the entire document collection (or the entire organization with entities of all types). Sometimes, we define the graph using documents that are relevant to a given query, but their relevance status values are then ignored during the actual step of expertise inference through graph analysis.

5.5.1 Query-dependent Graph-based Expert Finding

One may view the candidate generation model (Section 5.2.1) as the following probabilistic process: a user looking for an expert selects a document among the ones appearing in the initial document ranking returned in response to the query, makes a list of all candidate experts

[5] http://www.linkedin.com.

mentioned in it, and selects one of them as an expert on the query topic. The probability of selecting a document is its relevance score $P(d|q)$ (since the user will most probably search for contacts of knowledgeable people in one of the top ranked documents returned by a search engine). The subsequent selection of a candidate expert depends on $P(e|d,q)$, the probability of the candidate expert e to be related to parts of document d that are relevant to query q.

This process can be understood as a one-step relevance probability propagation process from documents to related candidate experts. It may be beneficial to continue the propagation of document relevance after this initial step. Intuitively, the search for expertise may consist of several, sometimes repeating, user actions, subject to the information encoded in an expertise graph. For example, in an expertise graph with person–document, document–document, and person–person edges, the expertise seeking actions of a user can be viewed as:

(1) At any time: (a) randomly read a document or (b) just pick a random candidate expert;
(2) After reading a document: (a) contact a person mentioned in this document or (b) check for other linked documents and read one of them;
(3) After contacting a candidate expert: (a) read other documents mentioning this candidate expert or (b) contact another candidate expert which is connected to this person (e.g., by recommendation).

While modeling this knowledge acquisition process, one may want to concentrate the random walk around the most relevant documents, building on the assumption that sources of the same knowledge are located close to each other in expertise graphs. Using this intuition, Serdyukov et al. [192] model the multi-step relevance probability dissemination in topic-specific *expertise graphs* consisting of persons and top retrieved documents. Expertise graphs form the background for three expert finding methods: based on a finite, an infinite, or a specialized parameter-free absorbing random walk. We describe the last two methods in this section.

The *infinite random walk model* regards the search for experts as a non-stop process, which may be realistic when users have a constant information need. The user in this model visits document and candidate nodes over and over again, making an infinite number of steps. In order to retain the importance for a candidate to stay in proximity to relevant documents, the model has jump transitions to the document nodes of the expertise graph. The probability $P_J(d)$ of jumping to the specific document d equals its probability of being relevant to the query at hand. This assumption ensures that candidate experts that are situated closer to relevant documents are visited more often in total during consecutive walk steps. The simplest variant of an infinite random walk model on an expertise graph containing no document–document and person–person edges is described by Equations (5.26), (5.27), and (5.28), which are used iteratively until convergence:

$$P_i(d) = \lambda P_J(d) + (1 - \lambda) \sum_{e \to d} P(d|e) P_{i-1}(e), \tag{5.26}$$

$$P_i(e) = \sum_{d \to e} P(e|d) P_{i-1}(d), \tag{5.27}$$

$$P_J(d) = P(d|q), \tag{5.28}$$

where λ is the probability that at any step the user decides stop following outgoing edges of the graph and makes a jump to a document node. $\sum_{e \to d} P(d|e) P_{i-1}(e)$ is the probability of reaching the document by following incoming links. Equation (5.27) means that candidate nodes can be reached by following incoming links only. However, it is possible to introduce jumps to candidate nodes as well. The described Markov process is aperiodic and irreducible (due to introduced jump probabilities), and hence has a stationary distribution P_∞ [9]. Consequently, we can define the expertise of e as being proportional to the stationary probability $P_\infty(e)$.

The *absorbing walk model* computes the probability of reaching the candidate expert in any minimum possible number of steps by starting the walk from one of the relevant documents. The candidate node being evaluated is only self-transient and hence becomes the final destination of the walk. Formally speaking, and as illustrated by Figure 5.4, the method removes all outgoing edges from the candidate being evaluated,

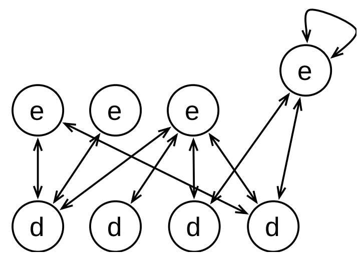

Fig. 5.4 An expertise graph modified by the absorbing random walk model.

adds the self-transition edge to it, and uses the following equations iteratively:

$$P_0(d) = P(d|q), \quad P_0(e) = 0, \tag{5.29}$$

$$P_i(d) = \sum_{e \to d} P(d|e)P_{i-1}(e), \tag{5.30}$$

$$P_i(e) = \sum_{d \to e} P(e|d)P_{i-1}(d) + P_{i-1}(e)P^{self}(e|e). \tag{5.31}$$

Finally, the expertise of e is proportional to the probability $P_\infty(e)$, as in the case with the infinite random walk method. Note that $P^{self}(e|e)$ equals 1.0, since the method removes all edges from any node e under study.

The absorbing random walk based method has some advantages over the infinite random walk model. As the size of the expertise graph is fixed, this method does not need any other parameters to be tuned. It is also a direct generalization of the one-step relevance probability propagation method. In contrast to the one-step approach using one-step probabilities, this multiple-step probability propagation based method calculates the probability $P^{mult}(e|d)$ of finding candidate expert e by making the *minimum sufficient number of steps* starting from document d:

$$P(e|q) = \sum_{d \in Top} P^{mult}(e|d)P(d|q). \tag{5.32}$$

In other words, in contrast to the model described in Section 5.2.1, Equation (5.32) provides the opportunity to propagate relevance to a candidate not only from directly related documents, but from any documents from which there is a path to the candidate expert. Both the infinite and absorbing random walk models outperform the baseline one-step relevance probability propagation method, as reported by Serdyukov et al. [192], with the former performing slightly better than the latter.

Other probabilistic expert finding models can also be extended to take advantage of relations between people and documents in the organization. For example, Karimzadehgan et al. [111] proposed to smooth language models of candidate experts (see Section 5.2.2.1) with language models of their colleagues working in the same organizational subdivision:

$$P(q|\theta'_e) = \alpha P(q|\theta_e) + \frac{1-a}{N}\sum_{i=1}^{N} P(q|\theta_{e'_i}),$$

where α is a weighting parameter, N is the number of colleagues e' of employee e (i.e., a candidate expert). Following similar ideas, Deng et al. [73] represent language models of candidate experts not via language models of directly related documents (in their case, conference papers), but via communities to which these documents belong. Moreover, for each query, they first identify the most relevant communities in whose scope the candidate expert ever published her documents, and then use only their language models to measure the candidate's level of expertise.

5.5.2 Query-independent Graph-based Expert Finding

It is important to mention another line of research that proposed finding experts by measuring their centrality in organizational or public social networks. These approaches often ignore the relevance of content related to candidate experts and utilize documents only as "context," to establish relations between candidates based on co-occurrence. Sometimes they are designed as query independent measures of prior belief that a person is authoritative within some knowledge community and, therefore, able to answer questions on topics popular in the community. For specialized communities this assumption seems plausible, but there is no guarantee that central users from multidisciplinary knowledge networks are "know-it-alls."

Observe, to begin, that some query-dependent graph-based expert finding methods can easily be turned into a query-independent variant. For instance, the infinite random walk method (see Section 5.5.1) becomes query-independent once we assume that $P_J(d)$ is uniform.

Schwartz and Wood [186] located people by observing communication patterns in e-mail logs. A set of heuristic graph algorithms was then used to cluster people by shared interests. Discovering users who are knowledgeable about a particular topic was identified as one of the potential applications. As social network analysis entered into the mainstream of IR, people began to appreciate the potential of this idea and to study it in detail. Campbell et al. [46] analyzed the link structure defined by senders and receivers of e-mails using a modified version of the HITS algorithm [115] to identify authorities; the expertise graph used was created using e-mail headers and from/to fields, and, hence, contained only people as nodes and e-mail messages as edges; they showed that using only the authority scores from HITS for candidate ranking resulted in better precision, but lower recall than for a simple content based method. Dom et al.[76] studied various graph-based ranking measures, including HITS and PageRank [165], for the purpose of ranking e-mail correspondents according to their degree of expertise. The results showed that PageRank performed noticeably better whereas HITS was slightly worse than the other methods. Because the TREC W3C dataset includes e-mail archives, some TREC participants analyzed the e-mail communications [51, 93]. Chen et al. [51] demonstrated that rankings produced by both HITS and PageRank are inferior to the ranking generated by a standard document-based method. Other sources for constructing social networks include chat logs [80], community-based question-answering systems [2], or co-authorship information from bibliographic databases [232]. Zhang et al. [231] analyzed a large, highly specialized help-seeking community (in Java programming) in order to identify users with high levels of expertise. The social graph was built from post/reply user interactions with edges directed from questions to answers, to reward answering activity. Three measures were compared: answers/questions ratio and the HITS and PageRank graph centrality measures. The former measure outperformed the centrality measures, which means that answering questions of users who answer a lot themselves is not an activity indicating a high level of expertise. Noll et al. [164] proposed a method which assumes that an expert should be one who tends to identify useful resources before other users discover them. They also applied a HITS-like algorithm to exploit mutual reinforcement relationship between users and documents and distinguish between followers and discoverers (who are meant to be experts in this case). Finding experts among users of popular social network platforms is also a hot topic nowadays. For instance, Weng et al. [217] proposed TwitterRank, an extension of PageRank algorithm, which is supposed to measure the influence of users in Twitter. TwitterRank differentiates itself from PageRank in that the random surfer performs a topic-specific random walk (via friendship connections), i.e., the transition probability from one twitterer to another is topic-specific. So, it not only discovers the areas of expertise of Twitter users, but also finds experts in these areas.

Finding experts in topic-focused communities boils down to the well-known task of finding authoritative people in large social networks. For instance, the authority (citation index) of scientists in co-authorship networks is traditionally defined by centrality measures: closeness, betweenness, Markov centrality (PageRank), and so on [128]. These measures do a good job for tasks where globally important social actors are to be identified (who are not necessarily active in the scope of a given topic). However, these approaches must experience problems in query-dependent expert finding scenarios, for which it is hard to detect a well-developed and homogeneous social community on the topic of each possible query. This may happen, for example, for novel topics that are just starting to emerge and attract attention.

5.6 Other Models

Since expert search is closely related to document search and other IR retrieval tasks, some classical IR methods have been applied to expert search, such as topic modeling [40], the vector space model [181], and cluster-based retrieval [127].

A number of topic models have been proposed for expertise retrieval. One of the earliest attempts at modeling the expertise of authors is the author-topic-model (ATM) [179], which is an extension of latent Dirichlet allocation (LDA) [40], like most topic models. In ATM, authors and content are simultaneously modeled with coupled hyper-parameters for the expertise of authors and the themes present in text. Mimno and McCallum [156] extended ATM and proposed the author-persona-topic (APT) model. In this model, an extra layer of unobserved variables called "personas" is added. Each author can write under one or more "personas," which are represented as independent distributions over hidden topics. The proposed model was evaluated on a reviewer matching task based on human relevance judgments determining how well the expertise of proposed reviewers matches a submission. As an extension of ATM, author-conference-topic (ACT) models have been proposed [207]. Unlike APT, where unobserved variables are added, ACT incorporates the observed variable "conference." Three variants of the model have been presented by varying the author-conference association. The models have been evaluated and deployed in ArnetMiner. Based on probabilistic latent semantic analysis (PLSA) [102], a mixture model [233] has been proposed for expert search and also evaluated on ArnetMiner, without considering the effect of authors. In addition, Tu et al. [212] have proposed the citation-author-topic (CAT) model, another variant of ATM, for expert search. CAT models the cited author information together with the words and authors. This additional contextual information enhances the author-topic association and thus may produce a more coherent author-topic distribution. All topic models mentioned so far treat topics as latent variables. Deng et al. [72] proposes a topic-based model where the topics are defined and known in advance. It eliminates the estimation of the latent topics, but how to select the topics is a challenging issue. The model has been evaluated on the DBLP dataset discussed in Section 4.2. None of the proposed topic modeling approaches have been evaluated on the TREC testbeds. The reason might be that both the W3C and the CSIRO dataset is a crawl of an intranet and thus heterogeneous in content. The (learned) topics from this data may not be well aligned with expertise areas. Another possible reason is that the semantics of document-candidate associations in DBLP (and in publication data in general) is clear: associations always mean authorship. In intranet data this is not the case; people with different roles may be present, and thus, associated, with documents, e.g., web-masters, secretaries, etc.

In [68] the classical vector space model (VSM) is extended for expert search. Specifically, experts are represented as a linear combination of documents, resulting in expert vectors which are in the same vector space with documents and queries. Existing practices for VSM can then be exploited such as the choice of the vector space basis.

Cluster-based retrieval [127] is a class of retrieval methods that has been proven effective for *ad-hoc* retrieval. In principle, it is very similar to *Model 2* while in a different context (i.e., aggregate ranking for document retrieval). Cao et al. [47] use clustering technique to re-rank experts. Persons are clustered according to their co-occurrences with topics and other persons. Macdonald and Ounis [131], Macdonald et al. [130] try to cluster profiles to use only the most relevant part of them. It is worth noting that several teams participating in TREC [221, 78, 26] resort to homepage detection to identify high quality documents to aid in the expertise assessment. In [206], a topic level expertise framework is proposed to simultaneously model topical aspects of different objects such as papers and conferences within a connected network. The learned topic models have been used for tasks such as expertise ranking, citation tracing analysis, and topical graph search.

One practical application of collective expertise matching is to assign reviewers to conference papers or research proposals according to their expertise. Some valuable work has been conducted for this research. The work in [113] considers multiple aspects of expertise of reviewers to cover all content topics in research papers. The work in [112] and [208] goes beyond content-based matching by considering other constraints such as the desired number of reviewers per paper, the maximum workload of reviewers and the authority requirement of reviewers. Charlin et al. [50] formulate the assignment procedure as an integer programming problem and show how several properties can be encoded as constraints or modications of the objective function. Furthermore, the work in [119] measures the effectiveness of an expert team with required skills by their communication-cost in a social network environment and develops an approximate algorithm for obtaining desired solutions. Adding these constraints is promising to meet the needs in the practical applications of assigning reviewers.

5.7 Comparison of Approaches

We already compared the various approaches to expert finding discussed so far along several dimensions: generative vs. discriminative (in Section 5.3) and the voting and graph-based models vs. their generative counterpart, in Sections 5.4 and 5.5, respectively. In Section 7.1.3 we include a comparison of document-based and profile-based methods from a practical point of view. In this section we briefly mention the scores obtained by testing on the test collections listed in Section 4.

Comparing methods in terms of absolute scores on some retrieval metric would not be meaningful (and fair) here for a number of reasons. First, the core (expertise retrieval) models are embedded within a larger system and are measured on an end-to-end task; thus, their performance is influenced by various components that may vary from system to system: document preprocessing (tokenization, stemming, etc.), query parsing, person name detection and disambiguation, etc. Second, results are often reported for a non-baseline version of the model, but with one or more additional techniques applied on top, such as exploiting document structure or assigning higher prior importance to certain documents or candidates (these are the techniques that we refer to as "additional components" and discuss in the next section). Third, some models require training material while others do not. Also, parameter settings are sometimes fine-tuned (either manually or automatically) to achieve best performance on a specific test collection and/or query set. Given these considerations, instead of quoting numbers from specific papers, we report the ranges of the best scores on the three main test collections.

For the W3C collection (Section 4.2.1), best MAP scores range from 0.2 to 0.3 on the 2005 query set and from 0.45 to 0.65 on the 2006 queries. (This difference between the two years stems from the nature of the queries.) MRR scores are in the ranges of 0.6–0.75 and 0.8–0.95 for 2005 and 2006, respectively. For the CERC collection (Section 4.2.2), best scores are in the 0.45–0.6 range for MAP and in the 0.7–0.9 range for MRR, for both the 2007 and 2008 topic sets. On the UvT collection (Section 4.2.3), the highest scores reported for the expert finding task are in the 0.2–0.4 range for MAP and in the 0.4–0.6 range for MRR. As for the expert profiling task, the numbers are within similar ranges: 0.2–0.3 for MAP and 0.4–0.6 for MRR.

As a general observation, we can conclude that identifying a relevant expert at rank one (similar to the "I'm feeling lucky" feature known from the Google search engine) can be done with very high accuracy; finding all relevant experts and ranking them above non-relevant ones, however, still proves to be challenging on some test sets.

6

Advanced Components

In this section we discuss extensions of the models presented in Sections 5.2–5.6. We discuss document-candidate associations, query modeling and relevance feedback, document importance and document structure, and conclude with a brief survey of the use of external evidence of expertise and of a candidate expert's importance.

6.1 Document-candidate Associations

A feature shared by all expertise retrieval models discussed in Section 5 is their reliance on associations between people and documents. For example, a person strongly associated with an important document on a given topic is more likely to be an expert on the topic than someone who is not associated with any documents on the topic or only with marginally relevant ones. Different retrieval models represent document-candidate association in different ways; for example, generative models capture it as the probability of a candidate generating a document ($P(d|e)$) or the probability of a document generating a candidate ($P(e|d)$), while instantiations of the Voting Model consider the set of documents associated with a person using binary weights for associations. In general, establishing associations may be viewed as a two-step process: (1) for each document in the collection, identify the set of candidates that are associated with that document (e.g., authors assigned explicitly as document metadata or people mentioned in the content), and (2) optionally, for each of these document-candidate pairs found, estimate the strength of the association (e.g., by considering which other people are mentioned in the document or by taking into account what other documents the person is associated with). We discuss these two steps in Sections 6.1.1 and 6.1.2, respectively.

6.1.1 Identifying Associations

There are two main strategies for identifying which candidates are associated with a given document; the document's type and the availability of metadata fields possibly supplied with the document very much dictate which of the two approaches can be applied. In cases where the document has people explicitly attributed to it, associated candidates can be identified unambiguously. Examples of such assignments of people to documents include authors of scholarly publications [16, 207], supervisors of graduates' dissertations [84, 123], teachers of courses [16], participants of research projects [84], and senders and recipients of e-mail messages [18, 88, 170]. The semantic roles of people vary across these examples, but all cases can be viewed as if documents were "tagged" with people; being associated with a document means that the person has knowledge about the topics covered in that document. The UvT Expert Collection is an example of a data set where document-candidate associations are provided explicitly and unambiguously (cf. Section 4.2.3).

In many cases, however, such explicit attribution of documents to people is unavailable. Almost all of the W3C collection (except for the *lists* and *people* parts) and the entire CERC collection are like that (cf. Sections 4.2.1 and 4.2.2). In such cases, candidates have to be recognized in documents through one of their representations, such as name or e-mail address. This is a special and restricted *named entity recognition* (NER) task. A simple and cheap way of performing NER is to use the candidate's identifiers as the query and consider the retrieved documents to be associated with that person [170, 171]. Balog et al. [13] introduce rule-based matching methods to identify candidates by their names (using three matching patterns: exact match, initial plus last name, and last name only) and by their e-mail-addresses. Using exact matches as opposed to less strict patterns (e.g., last name only) results in a lower coverage of candidates, yet it leads to better overall retrieval performance, as the amount of mismatched evidence is reduced [13]. Zhu et al. [240] employ a sophisticated rule-based approach to generate typical variations of a person's name, e.g., given "Deborah L. McGuinness," the automatically generated variants are "Deborah McGuinness," "McGuinness, Deborah L.," "McGuinness, D. L.," and so forth. Additionally, typical correspondence between real names and nicknames (e.g., "Michael" and "Mike," "Deborah," and "Deb") and conventional correspondence between non-English and English characters (e.g., ë→ e, ø→ oe) are made. These variations are then matched against documents using the Aho-Corasick algorithm [3]. A version of the W3C corpus that has been annotated using this method has been made publicly available[1] and has been widely used by other researchers too, e.g., in [14, 21, 171, 239]. Balog [12] made a similar contribution for the CERC collection, which includes the list of candidates and document-candidate associations.[2]

Person names are inherently ambiguous; even when full names and exact matching are used, it is possible that the same pattern matches multiple people within an organization. Matters become worse when people are not referenced by their full name, but only by their last name, for example. Therefore, person names have to be disambiguated and normalized, that is, surface forms need to be mapped to unambiguous references, like e-mail addresses (as has been done, e.g., for the CERC collection) or some internal (organization-wide) person identifiers (as has been done, e.g., for the W3C and UvT Expert collections). There are two principal approaches to treating candidate occurrences identified in documents: (1) resolve possible ambiguity using a set of heuristic rules [32, 240] and map each occurrence to a single person

[1] http://ir.nist.gov/w3c/contrib/.
[2] http://es.csiro.au/cerc/data/balog.

(or ignore it if matching cannot be established with high enough confidence), and (2) instead of attempting to resolve the ambiguity, consider all possible matching candidates, thereby associating each occurrence with one or more persons and (optionally) propagating uncertainty through the retrieval process, e.g., [13, 21, 83, 88, 170, 191, 192, 239]. However, even if all candidate occurrences in a given document were perfectly resolved, "common sense tells us that not all candidates mentioned in the document are equally important" [21]. Therefore, it is reasonable to weigh document-candidate associations, as opposed to taking them to be binary relationships; weighting associations also provides a natural solution for modeling uncertainty for case (2). Before moving on to presenting approaches to estimating the strength of document-candidate associations in Section 6.1.2, we first briefly discuss a number of related research problems in a broader context.

Named entity normalization (NEN) is a special case of the task of *record linkage*: find entries that refer to the same entity in different data sources. This task has been investigated since the 1950s — usually, entries are considered with their attributes (e.g., person with phone, address) [82, 219]. The task proved important because data sources have varying ways of referring to the same real-world entity due to, e.g., different naming conventions, misspellings, or use of abbreviation. The task of *reference normalization* is to analyze and detect these different references [75, 77].

Along with its sister tasks of *name disambiguation* and *resolution*, NEN has been studied very extensively in natural language processing both for named entities, generally (see, e.g., [60, 108]) and for person names, specifically (see, e.g., [144, 200]). As explained in Section 2.4, in the setting of people search engines, person name disambiguation has received renewed attention. Amongst other, the Web People Search (WePS) evaluation campaign was devoted to the problem of identifying different referents of result pages returned by a search engine in response to a person name [5, 6, 7, 8]. Successful approaches use clustering techniques [15]; the growing presence of large volumes of social media profiles calls for dedicated approaches [37].

Recently, NEN has been re-considered in the context of different text mining tasks. One is the task of discovering (suitable anchors of) hypertext links in text (together with a link target); see e.g., [43, 59, 90]. Another variation is assigning unambiguous Wikipedia pages ("concepts") to phrases occurring in queries [149, 150], documents [154, 155] or text streams [151]. Like NEN, these tasks involve identifying and disambiguating references to entities.

6.1.2 Estimating the Strength of Associations

Forming associations between documents and people is a key ingredient in expertise search models, and we have seen various methods for identifying such associations. Under the *boolean* or *set-based* approach these associations are taken to be 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" [21]. A more general approach is to allow document-candidate associations to be real numbers, where the score for each document-candidate pair is proportional to the strength of their association.

The strength of association can be measured by our confidence that the right person has been identified. For example, Balog et al. [13] use different patterns for recognizing candidates in documents and set the association score proportional to the number of matching patterns. In [18], the authors focus on an e-mail sub-collection and give higher weight to candidates that occur in multiple e-mail fields (from, to, cc, or body). The method used by Fang et al. [88] is based on discriminative models, and tries to learn the association between an expert and a document. Given a set of training query topics, relevance judgments of experts for those topics, and a set of features, the mapping function of association is automatically learned for maximizing the associations of those relevant experts to relevant documents with respect to corresponding queries. Features are based on different types of name matches (e.g., last name or full name) and different fields (e.g., to or cc) for e-mail address matches.

Another popular way of estimating the strength of association is to relate it to the number of times the candidate is mentioned in the given document (and in other documents in the collection). Balog and de Rijke [21] refer to it as the *frequency-based* approach and perform an experimental comparison of set-based and frequency-based modeling of associations in the context of topic generation models (*Model 1* and *Model 2*, cf. Section 5.2.2). Specifically, candidate occurrences in the text are replaced with unique identifiers, and are treated as regular terms. The importance of a candidate for a given document is estimated using standard IR term weighting schemes (TF, IDF, TF-IDF, and language models). Because the size of documents can vary greatly, normalization based on document length is desired; the authors propose a lean document representation, in which only candidate identifiers are kept, while all other terms are filtered out. This leads to very substantial improvements for the profile-based method (i.e., *Model 1*). The document-based method (i.e., *Model 2*), on the other hand, is much less sensitive to the weighting of document-candidate associations, and only modest improvements can be achieved over the boolean model of associations. Using language models, the association's strength in [21] is calculated as follows:

$$P(e|d) = (1-\lambda)\frac{n(e,d)}{\sum_{e'} n(e',d)} + \lambda\frac{\sum n(e)}{\sum_{e'} n(e')}, \tag{6.1}$$

where $n(e,d)$ is the number of occurrences of an expert in a document and $n(e)$ is the total number of occurrences of e in all documents (i.e., $n(e) = \sum_d n(e,d)$). A very similar language modeling-based scheme is used in [47, 171], but with Dirichlet smoothing, i.e., by setting $\lambda = \frac{n(e)}{n(e)+\mu}$ and $(1-\lambda) = \frac{\mu}{n(e)+\mu}$, where μ is the average number of candidate mentions in the collection ($\mu = \sum_e n(e)/|e|$). This way, the amount of smoothing applied is dynamically adjusted per candidate, "to prevent favoring candidates who occur in many documents" [171]. It is also possible to use the document frequency of candidates instead of their "term frequency" for $n(e)$ [171]. Macdonald and Ounis [138] propose similar normalization techniques for the Voting Model: one normalizes the overall score of the candidate by the total number of documents mentioning the candidate and another one also uses the average number of documents per candidate in the collection (see Section 5.4).

Associations are usually estimated at the document level, but it is also possible to establish them at a more fine-grained level. For example, associations may be conditioned on the query too ($P(e|q,d)$)

and estimated by considering internal document structure and incremental window sizes, see, e.g., [47, 239, 240], and Sections 5.2.2.3 and 6.4.

6.2 Query Modeling, Relevance Feedback

A keyword query is usually a very sparse representation of an underlying information need, and is not always formulated using the most effective terms. One popular way of enriching the user's query, and thus obtaining a more detailed specification of the underlying information need, is through query expansion, that is, adding new terms to the initial query and/or adjusting term weights in the original query in order to improve retrieval performance.

6.2.1 Document-centric Expansion

In the absence of explicit user feedback, the canonical approach is to treat the top-ranked documents retrieved in response to a query as if they had been marked relevant by the user, and use this pseudo-relevant set to expand the initial query. Pseudo-Relevance Feedback (PRF, also known as blind relevance feedback) has been used for a number of search tasks and has been shown to be useful for *ad-hoc* document retrieval [4], while less so for other tasks, such as topic distillation and known-item finding [55]. A variety of methods have been proposed for selecting terms from the pseudo-relevant set of documents; these can all be applied directly in expert search. This type of query expansion is termed *document-centric* [134], as it considers documents returned in response to the initial query as the basis of expansion. In this manner, the initial query can be reformulated, such that an improved underlying document search component is obtained, with resulting improvements in the accuracy of identifying relevant experts [134].

Using a language modeling framework, Petkova and Croft [170] apply the Relevance Models of Petkova and Croft [120], and Serdyukov et al. [189] use the Model-based Feedback strategy by Zhai and Lafferty [228]. Macdonald and Ounis [134] investigate two statistically different models from the Divergence From Randomness Framework: Bose–Einstein statistics, that is similar to Rocchio [178], and

Kullback–Leibler divergence [4]. Zhu et al. [239] report on very substantial improvements using an adaptation of the Hyperspace Analog to Language (HAL) model [44] for query expansion. First, a word co-occurrence matrix is constructed by accumulating co-occurrence counts in a sliding text window over the set of feedback documents. The combined HAL vector for the query is obtained based on a concept combination algorithm [199]. It has to be noted that Zhu et al. [239] use not only title queries, but additionally assign higher weights to expansion terms that appear in the description or narrative parts of TREC topic definitions (see Section 4 for the description of TREC topic definitions).

Generally speaking, document-centric PRF methods typically consider the top 5–30 documents and expand the original query with up to 30 terms. The results are mixed and range from little or no significant effects [189] to some significant increases [134, 170, 239].

Macdonald and White [142] provide an example of implicit feedback using click-through data from a large organization's intranet. For common information needs (i.e., popular queries) clicked documents alone can be used to identify relevant experts reasonably. Combined with an existing expert search approach (using multiple sources of evidence), click-data can improve retrieval performance by a statistically significant margin.

Finally, Balog and de Rijke [22] consider a scenario where the keyword query is complemented with examples of key reference pages. Arguably, in an enterprise setting, "users are more willing than, say, average web search engine users, to express their information needs in a more elaborate form than by means of a few key words" [30]. Balog and de Rijke [22] use the method proposed in [30] to exploit these so-called example documents for query expansion. Unlike previous work on relevance modeling [120] and blind relevance feedback mechanisms [178], here it is assumed that expansion terms are sampled independently from the original query terms (meaning that they do not necessarily have to co-occur with the original query terms to be selected for inclusion). This helps to address the "aspect recall" problem, by bringing in more "rare" terms that are not identified by standard (query-biased) expansion methods. Indeed, this method improves both precision and recall, substantially.

6.2.2 Candidate-centric Expansion

Instead of sampling expansion terms from documents, relevance feedback can be performed in a *candidate-centric* way: considering the top ranked candidates as the pseudo-relevant set and extracting informative terms from the corresponding candidate profiles [134]. This idea was proposed independently by Macdonald and Ounis [134] and by Serdyukov et al. [189] at the same time. In [189] query expansion is based jointly on the top ranked documents and on the top ranked candidates. Although the differences from the baseline are not statistically significant, the authors find that this method can help queries where the initial ranking is already of high quality; this behavior is already known from *ad-hoc* document search [226]. These results strongly suggest "that a prediction of query performance could be crucial for query modeling" [189].

Macdonald and Ounis [134] perform a thorough experimental comparison of document-centric and candidate-centric query expansion methods. Using default settings, document-centric expansion is found to be more stable and consistently outperforms candidate-centric expansion. In a follow-up study the authors investigate how topic drift within candidate profiles reduces the effectiveness of candidate-centric expansion [133].

6.2.3 Other Types of Query Expansion

Several TREC participants have experimented with exploiting the narrative and description fields of TREC topic definitions (see Section 4.2). For example, Balog et al. [27] added noun phrases from these fields to the original query, but found that it has a negative impact on performance. Duan et al. [78] added selected terms from the narrative using two different selection procedures: either taking terms with the highest IDF, or taking terms that most frequently co-occur with the terms from the title part within the organizational documents. Cao et al. [47] expanded queries using acronym normalization. In acronym normalization, for example, "Extensible Markup Language" is converted into its acronym form "XML."

Instead of expanding the initial query with additional terms, Balog et al. [16] propose to improve the scoring of a query given a candidate by considering what other requests (i.e., queries) that the candidate would satisfy. These similar requests are used as "further evidence to support the original query, proportional to how related the other requests are to the original query" [16]. The method assumes that a list of possible requests is provided, in the form of knowledge areas defined by the organization (the same can also be achieved with query logs, if sufficient volumes are available). Similarity between queries is determined by examining co-occurrence patterns of topics within the collection or by exploiting the hierarchical structure of topical areas that may be present within an organization. Taking topic similarities into account improves for both the expert finding and profiling tasks, but the improvement is more substantial for the latter. Bao et al. [32] transformed the initial query into a set of derived queries, by extracting bi-grams. For example, both "css test" and "test suite" are bi-gram queries constructed from the query "css test suite;" the former should receive a bigger weight as it carries more information. To model this, Bao et al. use the number of returned documents to refine the query weight, and then use the weighted sum of scores produced by different sub-queries to arrive at the document ranking.

6.3 Document Importance

Not all documents (or rather types of document) are equally important for the purpose of finding experts. For example, by comparing the performance of *Model 1* and *Model 2* on six different document types of the W3C collection, Balog [12] shows that the *lists* and *www* parts are more useful than other document types. Petkova and Croft [170] use the proposed expert models as a baseline and illustrate that models built from the *www* subcollection significantly outperform models built from the *lists* subcollection. Similar to models for web retrieval, expertise retrieval can take document importance into account to improve the retrieval performance as demonstrated in prior work. In the relevant literature, two main questions are addressed: (1) how to measure document importance and (2) how to incorporate document importance into the expertise retrieval models.

6.3.1 Measuring Document Importance

One way to measure document importance is to utilize the work in web IR on measuring the quality of a web page. The number of incoming hyperlinks (i.e., indegree) and the URL length of each document have been used to predict document importance for expert finding [130]. Similarly, PageRank, URL length and indegree have been adopted in [236, 239]. In the context of academic networks, document importance is computed as a function of the citation count of the document, which is similar to indegree [72, 73]. The usefulness of clicks on documents as an indicator for document importance is assessed in [142].

The measures listed above are both query- and candidate-independent, which is what we will focus on in this subsection. Of course, there exist document importance measures that are either query-dependent or candidate-dependent or both; those are discussed in later subsections. The document rank (or retrieval score) produced by document retrieval models can be viewed as query dependent (see Section 7.2). Macdonald et al. [130] define three other document quality measures: (1) the probability of being the homepage of the candidate, which is candidate-dependent (see Section 6.1); (2) clustering of candidate profiles to boost the importance of documents representing the candidate's central interests, which is also candidate-dependent; (3) the proximity between candidate and query in the document, which is both query- and candidate-dependent (see Section 5.2.2.3).

Supervised learning has also utilized to learn weights for documents or their types. Balog [12] obtains the importance of a document type based on the MAP score achieved using only documents of that type on the training data. Fang et al. [87] propose discriminative models to learn the importance weights for four document sources on INDURE (as presented in Section 4.2). More sophisticated models have been presented to enable the weights to be candidate-dependent, query-dependent or dependent on both [87]. The motivation is that the best weighting strategy should not be fixed but should adapt to various (latent) categories of candidates and queries.

6.3.2 Incorporating Document Importance

In the framework of probabilistic models, document importance can be encoded as a prior distribution over documents. For both *Model 1* and *Model 2* introduced in Section 5.2, we need to be able to estimate the probability $P(d|e)$ (see Equations (5.6) and (5.10)), which expresses the extent to which document d characterizes candidate expert e. The conditional probability $P(d|e)$ can be rewritten using Bayes rule:

$$P(d|e) = \frac{P(e|d)P(d)}{P(e)}, \tag{6.2}$$

where $P(e|d)$ denotes the probability of the candidate being associated with document, which is discussed in Section 6.1. The term $P(e)$ encodes the candidate importance which will be investigated in more detail in Section 6.6. The document importance can be expressed by $P(d)$. Some probabilistic models explicitly include the term $P(d)$ such as Modeling Documents as Mixtures of Persons (see Equation (5.15)) and the discriminative AMD model (see Equation (5.18)). In the basic models presented in Section 5.2, $P(d)$ is usually assumed to be uniform. In the advanced models, $P(d)$ can be computed based on the quality measures in Section 6.3.1 (subject to normalization). Some work applies certain transformations to the quality scores. For example, feature log odds estimate sigmoid functions [58] are used in [130, 236, 239] and the logarithmic function is applied in [72, 73]. Documents for which $P(d)$ is below a certain threshold sometimes get completely ignored (i.e., $P(d) = 0$) [12, 130]. It is worth noting that $P(d)$ can only encode both query- and candidate-independent document importance.

Balog [12] compares *Model 1* and *Model 2* under different configurations of document priors on the W3C collection. The experimental results show that document priors can improve performance for almost all cases, although the differences are not significant in most cases. Yet, *Model 1* using a lean document representation and machine learned document priors demonstrates significant improvements over the basic model, as it achieves a change of +7% in MAP and +10% in MRR. Working on top of generative language models, Zhu et al. [239] show that PageRank alone does not significantly help improve expertise retrieval performance on the TREC testbeds. Zhu et al. [236] also suggest that URL length is less effective than PageRank and indegree, while PageRank and indegree yield similar performance. Based on the Voting Model, Macdonald et al. [130] proportionally mix the document importance score into the original relevance score and then apply the expCombMNZ voting technique. Their experimental results on the TREC testbeds indicate that proximity and URL length are the best document quality indicators, followed by the candidate profile clustering measure. Fang et al. [87] conduct a set of experiments on the INDURE and UvT Expert collections to evaluate the proposed supervised learning approach. The results demonstrate that even the query- and candidate-independent model can significantly improve retrieval performance (but not on all metrics) by learning the document type importance from training data.

6.4 Document Structure

Documents often contain structural elements such as title, headings, sections, and lists. Taking the document structure into account can lead to better estimates of people/query-document associations, because it allows us to associate candidates or queries with very specific parts of the document instead of with the whole document, which may cover a broad range of topics.

The W3C collection includes technical reports, papers, and e-mails. These documents enjoy a relatively rich document structure. Consequently, most of the prior work about exploiting document structure for expertise retrieval uses the W3C collection as the testbed. Balog et al. [18, 88] consider candidate occurrences in four fields of an e-mail: *from*, *to*, *cc*, and body. Similarly, Petkova and Croft [170] combine evidence from three parts of an e-mail: header (subject, date, *from*, *to*, and *cc* fields), body (the original text of message with reply-to and forwarded text removed), and the thread (the concatenated text of messages making up the thread in which the message occurs). Inspired by work in the TREC Web track [57], Macdonald and Ounis [132] consider three fields of a document: the body, the title, and the anchor text of its incoming hyperlinks. Moreover, Zhu et al. [236] weigh eight structural elements: document body, author, acknowledgements, references, e-mail *from*, e-mail *to*, e-mail *cc*, and *bcc* sections.

To utilize document structure for expertise retrieval, most of the proposed methods weigh different structural elements and then integrate them into the basic models. Balog and de Rijke [18], Petkova and Croft [170] both set the weights manually. Zhu et al. [236] use cross-validation to search for the optimal weights. Macdonald and Ounis [132] apply simulated annealing to optimize the weights. Fang et al. [88] learn the weights automatically from training data using the AMD and GMD models.

There are generally two ways to exploit document structure. The first one aims at improving the estimation of a document's relevance with respect to a query term (i.e., $P(t|d)$), since document retrieval is an important ingredient of expert search. The rich document structure existing in an organizational intranet environment is shown to be effective for document retrieval [57]. Petkova and Croft [170] include a document retrieval component for which a document language model of an e-mail is defined as a linear combination of its three components. Thus, the goal of utilizing document structure is to compute $P(t|d)$. Based on the Voting Model, Macdonald and Ounis [132] also exploit document structure in the phase of document ranking.

Another way of exploiting document structure targets the creation of better document-candidate associations. For example, Balog and de Rijke [18] look at candidate occurrences in different fields of e-mail messages. The final estimate of document-candidate association $P(e|d)$ is defined as a linear combination of its four different components. Zhu et al. [236] focus on the window/proximity based model and consequently the goal is to compute $P(e|d,w,t)$. In the AMD model (Section 5.3.1.1), information about the document structure can be encoded as features for computing $P(r_2 = 1|e,d_t)$ in Equation (5.18), which also measures the document-candidate association. Table 6.1 summarizes the characteristics of the different methods.

Empirical studies have shown that the use of document structure often leads to a marked improvements in retrieval effectiveness. Balog and de Rijke [18] show that their best found combination of weights improves on all measures, and it improves significantly over the baseline

Table 6.1. Overview of methods for utilizing document structure.

Method	# of fields	Weight tuning	Target quantity
Petkova and Croft [170]	3	Manual	$P(t\|d)$
Macdonald and Ounis [132]	3	Automatic	$P(t\|d)$
Balog and de Rijke [18]	4	Manual	$P(e\|d)$
Fang et al. [88]	4	Automatic	$P(r = 1\|e, d)$
Zhu et al. [236]	8	Automatic	$P(e\|d, w, t)$

which treats all the fields equally. They also find that the e-mail *cc* field has a great importance when it is used within a combination. Interestingly, Fang et al. [88] reach the same conclusion by examining the magnitude of the learned weights associated with document structure features. Macdonald and Ounis [132] demonstrate that by utilizing document structure, there are statistically significant improvements in both MAP and P@10 for the vast majority of data fusion schemes. Significant improvements are also being observed in [236] by using language models enhanced by document structure.

6.5 External Evidence

Many of the expert finding techniques discussed so far only use evidence that can be found on the intranet of the organization that harbors the candidate experts. However, the evidence that is located outside the enterprise may be a valuable addition: it may help overcome sparseness issues (and hence lead to better estimations) and may bring in perspectives such as a person's status on a global scale, his popularity and "approachability." The latter are key factors that play a role when people look for expertise [220]. To assess these factors, data available outside the boundaries of an enterprise seems more suitable than data from within the enterprise. The web is a natural source for supporting evidence — and this is what several authors have explored.

Serdyukov and Hiemstra [190] suggest acquiring external expertise evidence from web search engines APIs by querying them with so-called "evidence identification queries" [184] that contain a candidate expert's name, the name of the current employer (for disambiguation purposes), and the expertise query. The number of results returned is used to assess the level of expertise of the candidate expert. The best performance is achieved by aggregating the ranking of candidates built in this way with the ranking built using a state-of-the-art expert finding algorithm extracting evidence from organizational data. Here, the aggregation is realized by the Borda count method and the aggregated ranking is shown to be far better than both component rankings, achieving a 29% improvement in MAP and a 20% in MRR over the ranking built using organizational data only. In later work, Serdyukov et al. [188] attempt to measure the quality and relevance of each (external) item returned by a web search engine, but incorporating these estimations into the expert ranking model does not result in any noticeable improvements.

Jiang et al. [105] follow a different approach: they create a web corpus by downloading, for each candidate expert, web documents that contain the candidate expert's name and then regard this web corpus as the actual document collection to find experts in. The authors generate rankings of candidate experts using a generative modeling based approach similar to *Model 2* (see Section 5.2) for internal, external, and combined evidence (including all found web documents). The rankings built using combined evidence are the most effective. The improvement is not overwhelming. The reason for the limited added value of the added external expertise evidence may lie in the unfocused crawling of information related to a person via a web search engine. The relevant documents related to the person may indeed be on the web, but it is impossible to find them using web search engines, with their limits on the number of returned results, by just using the person's name or e-mail address as a query and not adding the part describing the expertise needed to narrow down the search.

Santos et al. [184] propose a web search engine mimicking approach that involves three main steps: (1) download web documents or their summaries retrieved by "evidence identification queries" (similar to the ones used in [190]); (2) analyze their content as is normally done for the purposes of expert finding using organizational data using their Voting Model; (3) tune the parameters of the retrieval models used to score documents so that they would produce (so to say "mimick") rankings of the downloaded documents/summaries that are most similar to those generated by the web search engine. The last step of tuning improves the performance significantly. Interestingly, using summaries (titles + result snippets) rather than full content of documents is more effective.

Yet another approach is followed by Balog and de Rijke [23]; instead of applying a document-based approach to processing web search engine results (which are either snippets or full documents), they use web evidence in a candidate-based fashion. That is, they create textual profiles from snippets, a step that is performed offline, and then use candidate models based on these profiles (*Model 1* in Section 5.2.2.1). A linear combination of the latter approach with a generative approach built on enterprise data is shown to bring improvements (of 4% in MAP and 10% in P@5) even on top of a combination of document-centric and candidate-centric methods; this combination is the best performing official run at TREC 2008.

6.6 Candidate Importance

Not all documents are made equal (Section 6.3). Likewise, not all candidate experts are equally important either. For example, senior employees are more likely to have relevant expertise. If a particular query is difficult and the system cannot find enough evidence to decide who is an expert, candidate importance can help rank the candidates. Prior work has demonstrated that the performance of expert finding can be boosted by considering candidate importance [22, 83, 191].

Most existing work that utilizes candidate importance is based on probabilistic models. *Topic generation models* can incorporate it in a theoretically sound way, in the form of candidate priors, $P(e)$, as they estimate $P(e|q) \stackrel{\text{rank}}{=} P(q|e)P(e)$ (cf. Equation (5.1)). It is worth noting that not all probabilistic models can employ candidate priors in a straightforward (and probabilistically correct) manner. One example is *candidate generation models*, which estimate $P(e|q)$ directly (cf. Equation (5.3)). Another example is *Model 2* with the *candidate-centric* perspective — this is a variation that rewrites $P(d|e)$ using Bayes' rule, as opposed to estimating it directly (that is, the *document-centric perspective*) [13]. As shown below, the candidate prior $P(e)$ is not present in the *candidate-centric* reformulation:

$$P(e|q) \stackrel{\text{rank}}{=} \sum_d P(q|d)P(d|e)P(e) \;\; \textit{document-centric}$$
$$\stackrel{\text{rank}}{=} \sum_d P(q|d)P(e|d)P(d) \;\; \textit{candidate-centric}.$$

There are two types of approach to estimating $P(e)$. The first one is to directly use prior knowledge or some evidence from the text collection. For example, Balog and de Rijke [22] set the candidate prior to 0 for all science communicators in the CERC collection and otherwise to 1. Essentially, this filters out all the science communicators from the candidate list, because they are generally deemed as non-experts. This leads to statistically significant improvements, in terms of MAP and MRR, over the best performing configurations of *Model 1* and *Model 1B*. Fang and Zhai [83] utilize the e-mail mentions in W3C, the assumption being that a candidate whose e-mail has been mentioned many times should have a high prior probability of being an expert. In particular, $P(e) = \frac{n(e)}{2n(e)+\beta}$, where $n(e)$ is the total number of e-mail mentions for candidate e and β is the parameter to control the skewness of the prior (in the paper $P(R = 1|e)$ is used instead of $P(e)$, but they have the same semantics).

The second class of approaches measures $P(e)$ indirectly through documents:

$$P(e) = \sum_d P(e|d)P(d), \tag{6.3}$$

where $P(e|d)$ indicates the document-candidate association (Section 6.1) and $P(d)$ measures document importance (Section 6.3). Serdyukov and Hiemstra [191] take this approach by making document priors inversely proportional to the document rank (i.e., $P(d) = \frac{1}{rank(d)}$). Experimental results on the W3C collection show that using this candidate prior improves performance over uniform priors for MAP and P@5 measures in almost all the cases. Petkova and Croft [171] also adopt Equation (6.3) to compute $P(e)$ but use a uniform prior distribution for $P(d)$. Their experimental results on the TREC 2006 test set are mixed; using this candidate prior is not as effective as the uniform prior when candidate recognition is improved.

Hofmann et al. [100, 101] propose a number of ways to set candidate priors, using an expert's media experience (the number of mentions in the media), position in the company (e.g., professor vs. PhD student), reliability (the number of publications), up-to-dateness (the number of publications in recent years), and the size of the personal social network. The authors experiment with these priors for the task of finding similar experts and discover that some are helpful (for instance, to find the most authoritative expert on the topic), but not all. The authors also mention other possible indicators to be used as priors, such as availability or familiarity of the candidate with the user, but admit that these are problematic to implement due to the lack of such information in their dataset, the UvT Expert collection.

7

Discussion

In this section we discuss practical considerations as well as limitations of current expertise retrieval approaches, their broader applicability, and recent developments.

7.1 Practical Considerations

We start by considering specific steps in the expertise retrieval pipeline we have (implicitly) been assuming for much of this survey paper. After that we zoom out and discuss more general issues related to making expertise retrieval work in practice.

7.1.1 Preprocessing and Indexing

Many text preprocessing techniques for traditional IR applications (e.g., *ad-hoc* search) are adopted in expert search. For example, many expert search systems remove stopwords from raw text document data, in favor of high precision. Different types of stemming techniques such as Porter or Krovetz stemming [145] have been used in expert search to convert different inflected word forms that present similar semantic meanings into a single stem/root form.

An expert's data is often collected from multiple documents within heterogeneous data sources. This is an important difference with document retrieval that poses technical challenges to both data integration and indexing. First of all, the same names in different sources or documents may refer to different persons and different names may represent the same person. Indeed, named entity recognition (NER) and disambiguation play a key role in most operational expert search systems. Apart from the techniques discussed in Section 6.1.1, in practice, additional domain-specific knowledge can be utilized to develop effective solutions. For example, in the context of mining academic social networks, ArnetMiner considers five types of relationship between publications and utilizes an approach based on Hidden Markov Random Fields to disambiguate names by capturing dependencies between observations (i.e., papers) [207].

How to weigh different types of data source is another challenge. P@noptic [56] deploys a simple strategy that concatenates all the documents of an expert to form one single document, but this may bias the ranking toward the data sources with large chunks of text and blur the ones with little text but with distinguishing features. ArnetMiner utilizes a generative probabilistic model to simultaneously model different types of information by estimating a topic distribution for each type of information [207]. INDURE builds separate indexes for each data source and retrieves the documents from the respective data sources [84]; INDURE obtains its final ranked list of experts by merging and weighing the individually retrieved results, which has the potential to remove the bias of length (e.g., the number of documents) and size (e.g., the average size of documents) of different sources. This process is similar to that of federated search [196].

7.1.2 Interaction Design

Real-world expert search systems usually provide a range of services beyond the basic keyword search. For example, INDURE provides advanced search functions such as search in a department or within a particular data source.[1] In addition to expert search, ArnetMiner

[1] https://www.indure.org/search/advanced.cfm.

provides association search, related expert finding, and various social network mining services. Microsoft Academic Search can perform expert finding within a specific scientific domain. It also offers many additional functionalities such as visualizing the co-author graph, the citation graph, and domain trends. Section 3 shows screenshots for some of these services.

Aiding users in formulating their queries and requests is key to a successful user experience. This aspect, however, has not received much attention in the literature. Some systems provide limited query assistance; for example, Microsoft Academic Search offers "as-you-type" suggestions for query auto-completion; this helps to formulate queries that are within a controlled vocabulary. If one of these recognized queries is selected, the system also displays related topics.

The presentation of expert search results to users is an important issue in practice. Yarosh et al. [224] conducted a lab-based, controlled investigation with 35 enterprise workers and found that presenting additional information about each expert in a search result list led users to make quicker and better-informed selections. However, a simple list of names does not always help the user to judge the relevance of a candidate to the query. In contrast to document search, there is often no single document snippet that can be quickly examined to determine relevance. Therefore, expert search result pages often display not only a ranked list of people, but also of documents, conferences, and journals (as in ArnetMiner and Microsoft Academic Search). This way the user can have an overview of the key documents (and venues) on the given topic that have led to that particular ranking of experts.

Several publications portray the interfaces of their systems [19, 56, 131], giving clues as to the likely useful features. First of all, contact details for each ranked expert appear to be essential to facilitate communication. There is a great deal of work on locating personal homepages; many of the proposed methods utilize supervised learning techniques [62, 86, 207]. Balog and de Rijke [18] mine contact information from e-mail signatures. Tang et al. [207] further extract profile properties such as affiliations and addresses from the identified homepages for ArnetMiner. Secondly, photos of experts are important, because users may need to ascertain the likely seniority or familiarity of an expert before contacting him/her. For example, they may look for someone of comparable age or experience to themselves. Thirdly, related information such as affiliations and related documents, including publications and project descriptions, appear to help the user ascertain that the expert is likely to have relevant expertise. Serdyukov et al. [193] stress the need to summarize the output of an expert finding system by presenting a concise description of expertise for each returned employee. Since evidence about an employee's knowledge can be spread over many disparate sources in the enterprise and since it is difficult to find pieces of text that summarize even parts of personal expertise, they suggest to show tags describing the expertise of candidate experts along with their contact information in the returned result. The task of automatic assignment of such tags to people has been mentioned previously in this survey as the task of expertise profiling (Section 3.2.2).

The user's judgment of relevance may depend on the outcome of a dialog with the candidate expert. The judgment of relevance may be accomplished through follow-up e-mails, phone conversations, or face-to-face conversations with the suggested expert. Richardson and White [175] demonstrate that logged conversations (e.g., instant messaging) between an expert and the user are good sources of evidence about whether the asker was satisfied by the answers received. Predictions can be made at many points of the question lifecycle (e.g., when the question is entered or halfway through the asker-answerer dialog). Horowitz and Kamvar [104] experiment with different existing communication channels that people use to ask and answer questions: IM, e-mail, SMS, iPhone, Twitter, and web-based messaging. They find that a chat-like interface is the most efficient, since a private 1-to-1 online conversation creates an intimacy which encourages both honesty and freedom within the constraints of real-world social norms. Besides, in a real-time conversation, an answerer may request clarifying information from the asker about her question, or for the asker to follow-up with further reactions or inquiries to the answerer. System support of the follow-up communication between the user and the expert is important both for better learning to recommend experts and for improving the user experience of interaction with the expert finding system.

7.1.3 Scalability and Efficiency

Most existing research on expert search focuses on effectiveness, i.e., the "quality" of the ranking. As more and more large-scale expert search systems emerge, efficiency is becoming an increasingly important research topic. Efficiency issues need to be considered for various components in an expert search system. For example, how to provide incremental indexing functionality as new expertise evidence (e.g., new publications) becomes available? And, how to keep response times low, even when there are many expert candidates and documents to search through? Prior research on efficiency and scalability in traditional information retrieval (e.g., *ad-hoc* search) can be very valuable to address these issues.

Specific to expertise retrieval, the efficiency of *document-based* methods can be greatly improved by focusing only on the top documents relevant to the query (instead of considering all documents that contain any of the query terms); as shown in [12, 88, 215], applying such rank-based cut-offs has a positive impact on effectiveness too. For *profile-based* methods, the standard practice is to build a separate index for people from the contents of documents they are associated with.

A general strategy for improving efficiency (of any information retrieval system) is to move the computationally intensive processing offline, wherever possible; this can be done in a distributed fashion, if the size of the datasets requires so. For example, LinkedIn relies on MapReduce [66] to tackle the challenge of scalability in expert finding and many other services [31]. Tang et al. [205] propose MapReduce-based Topical Affinity Propagation to model the topic-level social influence and apply it to expert finding.

7.1.4 Getting Started

Given the plethora of expertise retrieval models available, what is a reasonable method to start with? Document-centric models (e.g., *Model 2* in Section 5.2.2.2) are a good choice to get going. They can be implemented with very limited effort on top of any existing document search engine and have decent overall performance over various testbeds (cf. Section 5.7). In the simplest case, a document-centric expert finding model can be implemented as follows:

(1) Perform a standard document retrieval run;
(2) For each relevant document d: for each candidate expert e associated with d, increase the candidate's score ($score(e,q)$) with the document's relevance score (this amounts to taking document-candidate associations to be boolean).

Later, this approach can easily be extended to include more advanced techniques, like the ones discussed in Section 6.

7.2 Limitations of Current Approaches

The models discussed in this survey do not directly model the concept of "expertise." Instead, they capture the degree of association between a topic and a person. Thus, they essentially answer a weaker, but related question: how strongly is the person related to the topic? While it has been shown that this simplification works well in practice, there exist some noticeable limitations.

First of all, a person who merely co-occurs a lot with the topic may not be the real expert. Frequent mentions of a candidate expert (i.e., a high document frequency for the person's name) does not entail the candidate's expertise. For instance, in the CERC collection (Section 4.2.2), many public-facing web pages mention contact people. Some of them are web masters or science communicators; their occurrence does not make them experts on all the topics discussed in the given document. One can correct for this by assigning a lower prior to people with particular job titles (in the CERC collection, science communicators are typically called "communication officer/manager/advisor" or "manager public affairs communication"), or by identifying "proper" experts based on their document frequency or on the coherence of the set of documents in which they occur [22]. Likewise, frequent mentions of an expertise topic in a document do not necessarily indicate a strong association between the topic and the document (and a similar rationale holds for the person-document association). A possible remedy is to examine where and how the person co-occurs with the topic. For example, if the topic is frequently mentioned in the person's homepage or the curriculum vitae (especially in the segments of expertise areas), it is very likely that the person possesses that expertise. Moreover, we can identify the expertise of the person with high certainty by detecting certain surface text patterns in the document, e.g., "... Dr. X's research interests include Y1, Y2 and Y3 ..." In other words, to overcome the above limitations, one should exploit advances in information extraction and text mining techniques that can pinpoint the expertise at the syntactic/semantic-level instead of at the word-level.

The models discussed so far only focus on the topical match but neglect other factors that may play a role in real-world expert finding. Section 2.1 discusses the human-centered perspective with expertise seeking. This line of work identifies contextual factors that influence how people locate and select experts, which go beyond the textual content. Some recent work attempts to integrate contextual factors with topical retrieval models. Hofmann et al. [101] address the task of finding similar experts and show that models combining content-based and contextual factors can significantly outperform existing content-based models. They also illustrate that while content-based features are the most important, users do take contextual factors into account, such as media experience and organizational structure. In addition, Smirnova and Balog [197] investigate social factors to estimate the time to contact an expert. They consider social graphs based on organizational hierarchy, geographical location, and collaboration, as well as the combination of these. They propose a user-oriented approach by taking the above factors into account and demonstrate substantial improvements over a baseline method that ranks people solely based on their knowledge. In fact, their approach can be viewed as personalization for expertise retrieval which has been rarely studied in the literature. Similarly, Kukla et al. [117] make use of the organization's social network together with the information about people's areas of expertise by propagating the query to friends and colleagues of the expert through the acquaintance chain.

Another limitation of current approaches is that they often assume a static collection of reasonably clean and properly edited content. With the rise of user-generated content, many ingredients become increasingly dynamic such as the documents, the topics, the people, and their expertise areas. The fact that someone may be a rising (or declining) expert on a topic may make him a more interesting expert than someone with a stable expertise profile [220]. In addition, the quality of the textual evidence needed for establishing people-topic associations may be highly variable. As a consequence, new challenges arise. For example, how to model changing people-topic associations? How to distinguish credible from non-credible evidence [216]? How to adapt the current expertise retrieval models to those circumstances? How does named entity normalization in user-generated content affect the retrieval models [108, 114]? These are open questions that need further investigation in the future work.

Finally, while current expertise retrieval models are able to establish associations between people and topics, they do not really provide a human-interpretable *explanation* for the association. For instance, in an academic setting, someone may be considered to be an expert on topic X because he has written the standard text book on X, or because his publications on X receive many citations. Again, a closer integration of retrieval models with fine-grained information extraction technology is called for to be able to supplement person-topic associations with explanations.

7.3 Relation Between Document Retrieval and Expert Finding

Document ranking is one of the key components of expert search systems. This is because enterprise documents are important sources of evidence for personal expertise and because in many cases the level of expertise of a person is assumed to correlate with the chance of the person being mentioned in relevant documents on the topic. The quality of document ranking with respect to the query topic can seriously affect the expert finding performance, yet, improved document retrieval does not necessarily imply improved expert finding performance.

Macdonald and Ounis [137] investigate the impact of the retrieval effectiveness of the underlying document search component. They experiment with fictitious perfect document rankings, to attempt to identify an upper-bound for expert search performance. However, they discover that non-relevant documents can bring useful expertise evidence, and that removing these does not lead to an upper-bound for retrieval performance. For instance, these documents may not exactly be on-topic (so would have been judged irrelevant during document judging), but they are about the same general topic area, and are associated to relevant candidate(s). In retrieving these documents, a document search engine may bring more evidence of expertise than a perfect document search engine that only retrieves relevant documents. Similar findings are reported by Zhu [235], who find in that although the improvement in terms of document retrieval may be dramatic due to tuning of some parameters, the expert finding performance need not improve and may even degraded. Furthermore, the work in [139] explores a document ranking that puts documents with mentions of relevant experts at the top. This type of document ranking increases the correlation with accurate expert ranking. On the other hand, the perfect expert ranking is still not achieved by this document ranking method, amongst others because some documents mention both relevant and irrelevant experts and some relevant experts are not mentioned in any document.

As confirmed by Macdonald and Ounis [135], the accuracy of expertise retrieval techniques depends on the accuracy of the underlying ranking of documents, though some of them suffer from imperfect document rankings considerably more than others. However, if one wants to improve the performance of an expert finding method that is based on a document retrieval component, as Macdonald and Ounis [135] suggest, optimizing Mean Average Precision of document ranking may be the most effective strategy.

Balog [12] and Weerkamp et al. [215] investigate the effect of the number of documents retrieved on the expert finding performance, specifically on *Model 2*. Instead of using the full collection for calculating the scoring of a candidate, the authors use only a subset of documents, defined by the top relevant documents returned (by a standard document retrieval run) in response to a query. They find that using this topically restricted subset of documents not only improves responsiveness, but in fact improves performance, both in terms of Mean

Average Precision and Mean Reciprocal Rank. A similar pattern was observed in [88] with the discriminative AMD and GMD models.

7.4 Broader Applicability

Viewed abstractly, the expert finding methods discussed in this survey compute associations between a certain type of metadata and textual material that surrounds it. The type on which we focused is ⟨`person`⟩...⟨`/person`⟩, but this is not essential for these approaches to work. In this section we first put expert finding methods to work in a different scenario: blog distillation. Then, we generalize the expert search task to finding entities of arbitrary types.

7.4.1 Blog Distillation

To provide an example of how expert finding methods can be used in a domain very different from organizational intranets, we look at the task of *blog distillation*: identifying key blogs with a recurring interest in a given topic, that provide credible information about the topic [141]. From a modeling point of view, blog distillation bears a strong resemblance to expert finding; blog posts correspond to documents and blogs correspond to people. Balog et al. [24] adopt *Model 1* and *Model 2* to the blog distillation task and refer to them as Blogger Model and Posting Model, respectively. The authors find that, unlike their expert finding counterparts, the Blogger Model significantly outperforms the Posting Model, and conclude that "there is a qualitative difference between the expert finding and feed distillation tasks, as a result of which an effective strategy for identifying key bloggers is to explicitly model them and the main themes that occupy them" [24]. In follow-up work, Weerkamp et al. [215] focus on using blog specific associations, combining the models, and improving efficiency. Macdonald and Ounis [136] also tackle blog distillation as an expert search problem, by adapting their Voting Model. Additionally, they introduce techniques to enhance the underlying ranking of blog posts, and to favor blogs with a recurring interest, by estimating "how well the posts in a single blog are spread throughout the time frame covered by the test collection" [141].

7.4.2 Entity Retrieval

Recent years have witnessed a growing interest in generalizing the kind of typed search introduced with expert finding to the retrieval of entities of other types. In 2007, INEX launched an Entity Ranking track (INEX-XER) [65], which also ran in 2008 [67] and in 2009 [69]. Here, entities are assumed to have a corresponding Wikipedia page, and queries asking for an entity are typed (i.e., asking for entities belonging to certain Wikipedia categories) and may come with examples. Specifically, two tasks are considered: (1) *entity ranking*, where query and target categories are given, and (2) *list completion*, where a textual query, example entities, and (optionally) target categories are given. Despite the fact that entities have direct representations (i.e., the Wikipedia page corresponding to the entity), there is a wide range of approaches looking for evidence in other documents. Zhu et al. [237] and Jiang et al. [106] adopt expert finding methods to build a co-occurrence model, which takes into account the co-occurrence of the entity and query terms (or example entities) in other documents. Tsikrika et al. [211] use random walks to model multi-step relevance propagation between linked entities. Others utilize the link structure of Wikipedia, e.g., as link priors [109] or by exploiting link co-occurrences [91, 169] to improve the effectiveness of entity ranking. A very distinctive feature, that sets apart the INEX-XER task from plain document retrieval, is the availability of category information, both as metadata to Wikipedia articles and as an additional piece of input provided along with the keyword query. A variety of approaches have been proposed to make use of category information; we refer to [17] for an overview.

The TREC Entity track started in 2009 with the aim to build test collections to evaluate entity-oriented search on web data, and introduced the *related entity finding* (REF) task: return a ranked list of entities (of a specified type) that engage in a given relationship with a given source entity [25]. Typically, REF systems identify a set of candidate entities that co-occur with the input entity, examine the contexts of the co-occurrence, and apply type filtering to arrive at a final ranking of entities. According to the track's setup, entities are represented by their homepages, therefore homepage detection completes the pipeline [25, 28]. A great deal of methods have been proposed, including generative probabilistic models [42, 110], an adaptation of the Voting Model [183], and learning-to-rank approaches [125].

An increased amount of structured data is being published on the web as Linked Data (such as DBpedia, Freebase, and others) and as metadata embedded inside web pages (RDF, RDFa, Microformats, and others). Inherently, much of this data is organized around entities. The Semantic Search Challenge series [39, 95] introduced a platform for evaluating the task that has been termed *ad-hoc entity retrieval*: "answering arbitrary information needs related to particular aspects of objects [entities], expressed in unconstrained natural language and resolved using a collection of structured data" [173]. Commonly, a document-based representation is built from RDF triples associated with a given entity; these pseudo-documents can then be ranked using (fielded extensions of) standard document retrieval methods [39, 95, 162].

7.5 Emerging Challenges

There are some lines of recent research on expertise retrieval that try to look at the problem of expertise mining from alternative angles and propose to tackle previously unaddressed challenges. Some of these have to do with evaluation methodology, others with multi-linguality or with contextual factors.

To begin with the latter, as pointed out in Section 2.1, previous research in expertise seeking has found that other factors besides content-based ones may play a role as well. In a study of trust-related factors in expertise recommendation, Heath et al. [96] find that *experience* and *impartiality* of the expert may play a role, and may additionally depend on a task's *criticality* and *subjectivity*. Borgatti and Cross [41] show that knowing about an expert's knowledge, valuing that knowledge, and being able to gain access to an expert's knowledge influenced which experts would be contacted for help. Differences between job roles regarding the amount and motivation of expert search, as well as the type of tools used indicate a possible influence of work tasks [81]. The use of social network information is expected to benefit expert search based on domain analysis [210] and users are more likely to select expert search results that include social network information [194]. Woudstra and Van den Hooff [220] focus on factors related to quality and accessibility in source selection, i.e., the task of choosing which expert candidate to contact in a specific situation. Quality-related factors include reliability and up-to-dateness of the expert, and accessibility includes physical proximity and cognitive effort expected when communicating with the expert. Further evidence of the usefulness for information seeking of individual contextual factors, such as social network information, is provided by systems that apply expertise retrieval. *Answer Garden 2* is a distributed help system that includes an expert finding component [1]. Besides topical matches the system implements heuristics found to be used in human expertise seeking, such as "staying local," i.e., first asking members of the same group or collaborators. *K-net*, a social matching system, aimed at improving sharing of tacit knowledge by increasing awareness of others' knowledge [195]. The system uses information on the social network, existing skills, and needed skills of a person, which are provided explicitly by the users. *SmallBlue* mines an organizations' electronic communication to provide expert profiling and expertise retrieval [80]. Both textual content of messages and social network information (patterns of communication) are used.

The expertise retrieval research community has started to integrate contextual factors with the content-based approaches mostly discussed in this survey. Section 7.2 lists several examples. Hofmann et al. [100, 101] integrate contextual factors of the type listed above with content-based methods for the task of similar expert finding and Smirnova and Balog [197] realize a similar integration, but for the more traditional task of expert finding. One feature that these publications share is that their task setting is, by necessity, more specific than the abstract scenario adopted at TREC. This allows them to identify, model, and study the contribution of specific contextual factors but it also brings the risk of findings that are hard to generalize. Future research in the area should target on developing generic expertise retrieval models that allow for transparent and effective integration of contextual factors.

De Rijke et al. [64] suggest that expertise profiling systems can be assessed not only in terms of precision and recall values of the descriptive terms produced (see Section 4.1), but that novelty and diversity should also play a role: (near) duplicate entries in the result list should be avoided and the result list should cover as many aspects of the expert being profiled as possible. The authors motivate their proposal by the fact that the presence of closely related descriptors in a result set (at the cost of omitting descriptors that highlight different aspects of an entity) is viewed unfavorably by users. For example, in their user study, one person complained about being automatically profiled with the descriptors of *international public law*, *international law*, and *international private law* that were all correct but also near-synonyms. Serdyukov et al. [193] make similar observations when analyzing the performance of their automatic people tagging system (which is essentially an expert profiling system). In many cases, the system assigned tags that were very similar in meaning to the same person (e.g., *networking* and *networks*). To address the issue, De Rijke et al. [64] propose two quality measures that take the novelty of each additionally added descriptor into account by considering the probability that two descriptors describe the same concept. According to Plachouras [172], taking diversity seriously may also be beneficial for expertise retrieval. Interestingly, the authors discuss several types of diversity, including topical diversity of the candidate result set (the presence of candidates with expertise on different sub-topics within the more general topic of the query), geographical diversity (the presence of candidates affiliated with different institutions or organizational departments), and supporting document diversity (the presence of candidates whose expertise evidence have been mined from sources of different types). Su et al. [203] investigate the task of diversifying expert finding in the context of academic social network and leverage supervised learning to learn a diversity retrieval function.

Finally, as fewer and fewer enterprises are limited to the borders of a single linguistic community, multilinguality is an extremely natural aspect to take into account for expertise retrieval. The UvT Expert collection (Section 4.2.3) first used by Balog et al. [16] is bilingual and the authors consider a simple multilingual model (a linear combination of the monolingual models), both for expert finding and expert profiling. The resulting coverage of topics and candidates for the expert finding and profiling tasks, i.e., the fraction of requests for which results are returned, is close to 100% in all cases. The relative improvement of the precision scores ranges from 10% to 80%. These scores demonstrate that it pays off to use multilingual information, if available. Another development in this area is the CriES workshop held at CLEF 2010 [202], which has drawn attention to the problem of cross-lingual expertise retrieval. It provides an expertise retrieval dataset (a crawl of Yahoo! Answers) including 60 topics in 4 languages: English, German, French, and Spanish. Experts on the same topic in the same community are found to often speak different languages. This corroborates the fact that in a multi-national organization expertise retrieval systems should rely on a cross-lingual notion of expertise.

8

Conclusions

> We do good work. Everybody has an expertise and we utilize them.
>
> —Brian Miller

The first decade of the 21st century has witnessed tremendous interest, and a wealth of results, in expert retrieval as an emerging subdiscipline in IR. We have presented a comprehensive survey highlighting advances on models, algorithms, and evaluation relevant to this field. We traced the roots of modern work on expertise retrieval to early work in library science and knowledge management, and contrasted it with current research in IR which aims at a fully automatic search process based on text corpora.

Throughout this survey, we have summarized the key modeling issue in expertise retrieval as how to associate query topics to people. Based on this common theme, five groups of methods were identified as the basic approaches. Generative probabilistic models (Section 5.2) estimate associations between query topics and people based on the likelihood of a topic being generated by a given candidate (i.e., topic generation models), or the other way around, based on how the candidate is generated by the query (i.e., candidate generation models).

Discriminative models (Section 5.3) determine the associations by directly estimating the conditional probability that a given pair of a query topic and a candidate expert is relevant or not. Voting models (Section 5.4) capture associations between query topics and people by a voting process that allows documents ranked with respect to a query to vote for candidate experts by different weighting schemes. Graph-based models (Section 5.5) estimate associations by expertise inference via graph analysis in an expert graph with documents, expert candidates, and different relationships, which can be built in a query-dependent or query-independent manner. Other models (Section 5.6) use a range of ways for capturing associations between query topics and people, including modeling people as a distribution of latent variables corresponding to topical themes.

Based on the basic approaches mentioned above, advanced models can be developed by considering a range of content-based factors that may impact the strength of association between a topic and a person: query modeling, document–candidate associations, document and candidate importance, document structure, and external evidence. We have seen that adding these advanced components on top of the basic models often leads to improved performance in practice.

We have also discussed practical aspects of building an expert search system and presented applications of the technology in other domains such as blog distillation and entity retrieval. Some limitations of the current approaches have also been discussed.

After this brief summary, let us look forward and conjecture what the future may hold for expertise retrieval research, specifically in relation to themes such as personalization, interaction, structured data, and social media.

Most existing work on expert finding has focused on a global notion of "expertise." To further improve the user experience we should consider personalized views on expertise, views that which can be tailored to user-specified information needs. As discussed in Section 7.2, initial work on user modeling for expert search [197] is a first step toward the direction but a number of advances must be made before expertise retrieval fulfills the promise of personalization. For instance, personalization requires the capability of modeling the user's preferences and interests. In web search, this is usually done by aggregating a user's interaction with the system including his previous queries, click-through, and even eye-tracking. It has been shown that query logs of expert search have quite different characteristics from those of web search and search by children [89]. Consequently, adapting existing web search personalization techniques to expert search is a non-trivial research challenge.

For expertise retrieval, user feedback is rarely collected and available. One of the reasons may be that meetings between the user and experts often take place offline. Today's expert finding systems rarely know what happens beyond the step of presenting the top candidate experts to the user. Future expert finding systems that provide online communication tools would have the chance to benefit from analyzing conversations between the user and the recommended expert and facilitate the interaction between them so as to improve the knowledge acquisition process. Recent studies by Richardson and White [175] show that it is possible to predict if the user was satisfied by the answer from the expert by using a broad range of features derived from the scripted dialog between them. The ability of expert finding systems to leverage asynchronous (e.g., email exchange) or synchronous (e.g., instant messaging) communication channels between users and experts may not only improve the user experience and lead to faster answers, but also increase the availability of experts that might be hard to access face-to-face (due to their busy agendas or distant locations) and provides interesting opportunities for online learning to rank experts algorithms.

Expertise retrieval has traditionally been studied in the context of enterprise intranets. But it is already common for people to search for expertise outside enterprise contexts, e.g., locating a PhD supervisor, looking for industry-research collaborations, etc. Guan et al. [94] investigated the general task of searching experts on the Web, where documents (web pages) could be of varying quality and inevitably carry noise. Additionally, a large number of people indicate their expertise or qualifications through social media, even for leisure-related usage. Although some prior research has addressed the task of expert finding on question-answering sites [2, 128, 174, 231], for most popular social platforms, such as Twitter or Facebook, the potential to assist in finding experts has not been studied yet. However, people actively use those services to search for expertise and share their own. Morris et al. [160] conduct a study where participants post a question to their Facebook social network and simultaneously searched the web. The authors find that over half of their participants received responses from their social network before completing their web search. Still, identifying experts in these contexts is very challenging, given the wide variety of potential expertise areas and interests and the large number of users. Furthermore, expert finding based on social media is often sensitive to location and time. As pointed out in Section 7.2, modeling highly dynamic people-topic associations in a location and time dependent manner is still an open problem. As an example, Pal et al. [166] study the evolution of experts in community question answering: expert users differ from ordinary users in terms of their contributions; as the probability of providing a best answer increases for experts it decreases for ordinary users over time. Social media provides an ideal testbed for the personalization of expertise retrieval due to the explicit user interactions through Web 2.0 tools. User feedback, such as bookmarking, rating, "likes," commenting, and blogging, provides an explicit indication of the user's interests and can be safely used without violating the user's privacy, both to develop feature-based ranking models and for evaluation purposes [38].

Finally, current expertise retrieval approaches have very limited inference mechanisms due to their heavy use of unstructured data. What they often lack is a formal understanding of relevant data and of relations amongst items in the data. On today's web, there is an increasing volume of open data sets that are structured and accessible in compliance with Linked Data principles.[1] The pervasiveness of (semi-)structured expertise data is also evident in large recruitment websites, such as LinkedIn[2] and Monster.[3] These sites can capture

[1] http://www.w3.org/DesignIssues/LinkedData.html.
[2] http://www.linkedin.com/.
[3] http://www.monster.com/.

structured candidate expert data at scale by incentivizing candidates to not only meticulously build their professional profiles through responses to specific questions, but also to keep their profiles up to date. By being structured and heavily interlinked, this growing volume of data is likely to bring new research opportunities to the field of expertise retrieval.

Acknowledgments

We are grateful to the editors-in-chief, Doug Oard and, especially, Fabrizio Sebastini, for advice, encouragement, and patience along the way. We are also extremely grateful to our three reviewers: their valuable comments and suggestions helped us improve the paper.

Krisztian Balog was supported by the COMIDOR project, funded by the Norwegian Research Council.

The research of Yi Fang and Luo Si was partially supported by the NSF research grants IIS-0746830, CNS-1012208, IIS-1017837 and research grants from Purdue Cyber Center as well as Google. This work is also partially supported by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

Maarten de Rijke was partially supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430 (Galateas), the European Community's Seventh Framework Programme (FP7/2007–2013) under grant agreements nr 258191 (PROMISE Network of Excellence) and 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.814, 612.061.815, 640.004.802, 380-70-011, 727.011.005, 612.001.116, the Center for Creation, Content and Technology (CCCT), the Hyperlocal Service Platform project funded by the Service Innovation & ICT program, the WAHSP and BILAND projects funded by the CLARIN-nl program, the Dutch national program COMMIT, and the ESF Research Network Program ELIAS.

References

[1] M. S. Ackerman and D. W. Mcdonald, "Collaborative support for informal information in collective memory systems," *Information Systems Frontiers*, vol. 2, no. 3–4, pp. 333–347, 2000. ISSN 1387-3326.

[2] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and Yahoo! Answers: Everyone knows something," in *Proceedings of the International World Wide Web Conference, (WWW '08)*, New York, NY, USA: ACM, pp. 665–674, 2008. ISBN 978-1-60558-085-2.

[3] A. V. Aho and M. J. Corasick, "Efficient string matching: An aid to bibliographic search," *Communications ACM*, vol. 18, pp. 333–340, June 1975.

[4] G. Amati, "Probabilistic models for information retrieval based on divergence from randomness," PhD thesis, Department of Computing Science, University of Glasgow, 2003.

[5] J. Artiles, A. Borthwick, J. Gonzalo, S. Sekine, and E. Amigó, "Weps-3 evaluation campaign: overview of the web people search clustering and attribute extraction tasks," in *Cross-Language Evaluation Forum 2010 Working Notes*, 2010.

[6] J. Artiles, J. Gonzalo, and S. Sekine, "The SemEval-2007 WePS evaluation: Establishing a benchmark for the web people search task," in *Proceedings of the International Workshop on Semantic Evaluations, (SemEval '07)*, Stroudsburg, PA, USA: ACL, pp. 64–69, 2007.

[7] J. Artiles, J. Gonzalo, and S. Sekine, "WePS 2 evaluation campaign: Overview of the web people search clustering task," in *Web People Search Evaluation Workshop (WePS 2009), WWW Conference*, 2009.

[8] J. Artiles, J. Gonzalo, and F. Verdejo, "A testbed for people searching strategies in the WWW," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '05)*, New York, NY, USA, pp. 569–570, 2005. ISBN 1-59593-034-5.

[9] S. Asmussen, *Applied Probability and Queues*. Wiley, 1987. ISBN 0387002111.

[10] P. Bailey, N. Craswell, A. P. de Vries, and I. Soboroff, "Overview of the TREC 2007 enterprise track," in *Proceedings of the Text REtrieval Conference, (TREC '07)*, Gaithersburg, MD, 2008.

[11] P. Bailey, N. Craswell, I. Soboroff, and A. de Vries, "The CSIRO enterprise search test collection," *ACM SIGIR Forum*, vol. 41, pp. 42–45, 2007.

[12] K. Balog, "People search in the enterprise," PhD thesis, University of Amsterdam, June 2008.

[13] K. Balog, L. Azzopardi, and M. de Rijke, "Formal models for expert finding in enterprise corpora," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '06)*, New York, NY, USA, pp. 43–50, 2006. ISBN 1-59593-369-7.

[14] K. Balog, L. Azzopardi, and M. de Rijke, "A language modeling framework for expert finding," *Information Processing and Management*, vol. 45, pp. 1–19, January 2009. ISSN 0306-4573.

[15] K. Balog, L. Azzopardi, and M. de Rijke, "Resolving person names in web people search," in *Weaving Services, Locations, and People on the World Wide Web*, Springer, pp. 301–323, 2009. ISBN 978-3-642-00569-5.

[16] K. Balog, T. Bogers, L. Azzopardi, M. de Rijke, and A. van den Bosch, "Broad expertise retrieval in sparse data environments," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '07)*, New York, NY, USA, pp. 551–558, 2007. ISBN 978-1-59593-597-7.

[17] K. Balog, M. Bron, and M. de Rijke, "Query modeling for entity search based on terms, categories, and examples," *ACM Transactions on Information Systems*, vol. 29, November 2011. ISSN 1046-8188.

[18] K. Balog and M. de Rijke, "Finding experts and their details in e-mail corpora," in *Proceedings of the International World Wide Web Conference, (WWW '06)*, New York, NY, USA, pp. 1035–1036, 2006. ISBN 1-59593-323-9.

[19] K. Balog and M. de Rijke, "Determining expert profiles (with an application to expert finding)," in *Proceedings of the International Joint Conference on Artifical Intelligence, (IJCAI '07)*, San Francisco, USA, pp. 2657–2662, 2007.

[20] K. Balog and M. de Rijke, "Finding similar experts," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '07)*, New York, NY, USA, pp. 821–822, 2007. ISBN 978-1-59593-597-7.

[21] K. Balog and M. de Rijke, "Associating people and documents," in *Proceedings of the European Conference on IR Research*, (C. M. et al., ed.), *(ECIR '08)*, Berlin, Heidelberg, pp. 296–308, 2008. ISBN 978-3-540-78645-0.

[22] K. Balog and M. de Rijke, "Non-local evidence for expert finding," in *Proceeding of the ACM International Conference on Information and Knowledge*

Management, *(CIKM '08)*, New York, NY, USA, pp. 489–498, October 2008. ISBN 978-1-59593-991-3.

[23] K. Balog and M. de Rijke, "Combining candidate and document models for expert search," in *Proceedings of the Text REtrieval Conference*, *(TREC '08)*, Gaithersburg, MD, 2009.

[24] K. Balog, M. de Rijke, and W. Weerkamp, "Bloggers as experts: Feed distillation using expert retrieval models," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '08)*, New York, NY, USA, pp. 753–754, 2008. ISBN 978-1-60558-164-4.

[25] K. Balog, A. P. de Vries, P. Serdyukov, P. Thomas, and T. Westerveld, "Overview of the TREC 2009 Entity Track," in *Proceedings of the Text REtrieval Conference*, *(TREC '09)*, Gaithersburg, MD, February 2010.

[26] K. Balog, K. Hofmann, W. Weerkamp, and M. de Rijke, "The University of Amsterdam at the TREC 2007 enterprise track," in *Proceedings of the Text REtrieval Conference*, *(TREC '06)*, Gaithersburg, MD, 2007.

[27] K. Balog, K. Hofmann, W. Weerkamp, and M. de Rijke, "Query and document models for enterprise search," in *Proceedings of the Text REtrieval Conference*, *(TREC '07)*, Gaithersburg, MD, 2008.

[28] K. Balog, P. Serdyukov, and A. P. de Vries, "Overview of the TREC 2010 entity track," in *Proceedings of the Text REtrieval Conference*, *(TREC '10)*, Gaithersburg, MD, February 2011.

[29] K. Balog, I. Soboroff, P. Thomas, N. Craswell, A. P. de Vries, and P. Bailey, "Overview of the TREC 2008 enterprise track," in *Proceedings of the Text REtrieval Conference*, *(TREC '08)*, Gaithersburg, MD, 2009.

[30] K. Balog, W. Weerkamp, and M. de Rijke, "A few examples go a long way: Constructing query models from elaborate query formulations," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '08)*, New York, NY, USA, pp. 371–378, 2008. ISBN 978-1-60558-164-4.

[31] B. Bansal, "Large scale computing @ linkedin," *Talk*, 2011. URL http://cacs.usc.edu/education/cs596/usc_bbansal.pdf.

[32] S. Bao, H. Duan, Q. Zhou, M. Xiong, Y. Cao, and Y. Yu, "Research on expert search at enterprise track of TREC 2006," in *Proceedings of the Text REtrieval Conference*, *(TREC '06)*, Gaithersburg, MD, 2007.

[33] P. Baumard, *Tacit knowledge in organizations.* Thousand Oaks, CA, USA: Sage Publications, 2001.

[34] I. Becerra-Fernandez, "The role of artificial intelligence technologies in the implementation of people-finder knowledge management systems," *Knowledge-Based Systems*, vol. 13, no. 5, pp. 315–320, 2000. ISSN 0950-7051.

[35] I. Becerra-Fernandez, "Searching for experts on the web: A review of contemporary expertise locator systems," *ACM Transactions on Internet Technology*, vol. 6, no. 4, pp. 333–355, 2006. ISSN 1533-5399.

[36] J. Bennett and S. Lanning, "The Netflix prize," in *Proceedings of KDD Cup and Workshop*, vol. 2007 of *(KDD '07)*, New York, NY, USA: ACM, p. 35, 2007. ISBN 978-1-59593-609-7.

[37] R. Berendsen, B. Kovachev, E. Nastou, M. de Rijke, and W. Weerkamp, “Result disambiguation in web people search,” in *Proceedings of the European Conference on Information Retrieval*, *(ECIR ’12)*, Berlin, Heidelberg, 2012.

[38] R. Berendsen, E. Tsagkias, M. de Rijke, and E. Meij, “Generating pseudo test collections for learning to rank scientific articles,” in *CLEF 2012: Conference and Labs of the Evaluation Forum*, Rome, Italy, 2012.

[39] R. Blanco, H. Halpin, D. M. Herzig, P. Mika, J. Pound, H. S. Thompson, and T. T. Duc, “Entity search evaluation over structured web data,” in *Proceedings of the International Workshop on Entity-Oriented Search*, *(EOS ’11)*, pp. 65–71, 2011.

[40] D. Blei, A. Ng, and M. Jordan, “Latent Dirichlet allocation,” *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003. ISSN 1532-4435.

[41] S. P. Borgatti and R. Cross, “A relational view of information seeking and learning in social networks,” *Management Science*, vol. 49, no. 4, pp. 432–445, 2003. ISSN 0025-1909.

[42] M. Bron, K. Balog, and M. de Rijke, “Ranking related entities: Components and analyses,” in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM ’10)*, New York, NY, USA, pp. 1079–1088, 2010. ISBN 978-1-4503-0099-5.

[43] M. Bron, B. Huurnink, and M. de Rijke, “Linking archives using document enrichment and term selection,” in *Proceedings of the International Conference on Theory and Practice of Digital Libraries*, *(TPDL ’11)*, Heidelberg, Germany, 2011. ISBN 978-3-642-24468-1.

[44] C. Burgess, K. Livesay, and K. Lund, “Explorations in context space: Words, sentences, discourse,” *Discourse Processes*, vol. 25, pp. 211–257, 1998.

[45] J. Callan, “Distributed information retrieval,” *Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval*, vol. 7, pp. 127–150, 2002.

[46] C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom, “Expertise identification using email communications,” in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM ’03)*, New York, NY, USA, pp. 528–531, 2003. ISBN 1-58113-723-0.

[47] Y. Cao, J. Liu, S. Bao, and H. Li, “Research on expert search at enterprise track of TREC 2005,” in *Proceedings of the Text REtrieval Conference*, *(TREC ’05)*, Gaithersburg, MD, 2006.

[48] S. Cetintas, M. Rogati, L. Si, and Y. Fang, “Identifying similar people in professional social networks with discriminative probabilistic models,” in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR ’11)*, New York, NY, USA, pp. 1209–1210, 2011. ISBN 978-1-4503-0757-4.

[49] O. Chapelle and M. Wu, “Gradient descent optimization of smoothed information retrieval metrics,” *Information retrieval*, vol. 13, no. 3, pp. 216–235, 2010. ISSN 1386-4564.

[50] L. Charlin, R. Zemel, and C. Boutilier, “A framework for optimizing paper matching,” in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, *(UAI’11)*, pp. 86–95, 2011. ISBN 978-0-9749039-7-2.

[51] H. Chen, H. Shen, J. Xiong, S. Tan, and X. Cheng, "Social network structure behind the mailing lists: Ict-iiis at TREC 2006 expert finding track," in *Proceedings of the Text REtrieval Conference*, *(TREC '05)*, Gaithersburg, MD, 2006.

[52] D. Constant, L. Sproull, and S. Kiesler, "The kindness of strangers: The usefulness of electronic weak ties for technical advice," *Organization Science*, vol. 7, no. 2, pp. 119–135, 1996.

[53] W. Cooper, "Exploiting the maximum entropy principle to increase retrieval effectiveness," *Journal of the American Society for Information Science and Technology (JASIST)*, vol. 34, no. 1, pp. 31–39, 1983. ISSN 0002-8231.

[54] N. Craswell, A. de Vries, and I. Soboroff, "Overview of the TREC-2005 enterprise track," in *Proceedings of the Text REtrieval Conference*, *(TREC '05)*, Gaithersburg, MD, 2006.

[55] N. Craswell and D. Hawking, "Overview of the TREC-2002 web track," in *Proceedings of the Text REtrieval Conference Proceedings*, *(TREC '01)*, Gaithersburg, MD, 2002.

[56] N. Craswell, D. Hawking, A. M. Vercoustre, and P. Wilkins, "P@noptic expert: Searching for experts not just for documents," in *Ausweb Poster Proceedings*, 2001.

[57] N. Craswell, D. Hawking, R. Wilkinson, and M. Wu, "Overview of the TREC 2003 web track," in *Proceedings of the Text REtrieval Conference Proceedings*, vol. 2003 *of TREC '02*, Gaithersburg, MD, pp. 78–92, 2003.

[58] N. Craswell, S. Robertson, H. Zaragoza, and M. Taylor, "Relevance weighting for query independent evidence," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '05)*, New York, NY, USA, pp. 416–423, 2005. ISBN 1-59593-034-5.

[59] A. Csomai and R. Mihalcea, "Linking documents to encyclopedic knowledge," *IEEE Intelligent Systems*, vol. 23, pp. 34–41, September 2008. ISSN 1541-1672.

[60] S. Cucerzan, "Large-scale named entity disambiguation based on wikipedia data," in *Proceedings of Empirical Methods in Natural Language Processing*, *(EMNLP '07)*, Stroudsburg, PA, USA: ACL, pp. 708–716, 2007.

[61] R. Cummins, M. Lalmas, and C. O'Riordan, "Learning aggregation functions for expert search," in *Proceedings of the European Conference on Artificial Intelligence*, *(ECAI '10)*, pp. 535–540, 2010. ISBN 978-1-60750-605-8.

[62] S. Das, P. Mitra, and C. Giles, "Learning to rank homepages for researcher-name queries," *The International Workshop on Entity-Oriented Search*, pp. 53–58, 2011.

[63] T. Davenport and L. Prusak, *Working Knowledge: How Organizations Manage What They Know.* Harvard Business Press, 2000. ISBN 0875846556.

[64] M. de Rijke, K. Balog, T. Bogers, and A. van den Bosch, "On the evaluation of entity profiles," in *Proceedings of the 2010 Conference on Multilingual and Multimodal Information Access Evaluation*, *(CLEF '10)*, pp. 94–99, September 2010. ISBN 978-3-642-15997-8.

[65] A. de Vries, A.-M. Vercoustre, J. A. Thom, N. Craswell, and M. Lalmas, "Overview of the INEX 2007 entity ranking track," in *Proceedings of the Initiative on the Evaluation of XML Retrieval, (INEX '07)*, Berlin, Heidelberg, pp. 245–251, 2008. ISBN 978-3-540-85901-7.

[66] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008. ISSN 0001-0782.

[67] G. Demartini, A. de Vries, T. Iofciu, and J. Zhu, "Overview of the INEX 2008 entity ranking track," in *Proceedings of the Initiative on the Evaluation of XML Retrieval, (INEX '08)*, Berlin, Heidelberg, pp. 243–252, 2009. ISBN 978-3-642-03760-3.

[68] G. Demartini, J. Gaugaz, and W. Nejdl, "A vector space model for ranking entities and its application to expert search," *Advances in Information Retrieval*, pp. 189–201, 2009.

[69] G. Demartini, T. Iofciu, and A. de Vries, "Overview of the INEX 2009 entity ranking track," in *Proceedings of the Initiative on the Evaluation of XML Retrieval*, vol. 6203 of *(INEX '09)*, (S. Geva, J. Kamps, and A. Trotman, eds.), Berlin, Heidelberg, pp. 254–264, 2010. ISBN 978-3-642-14555-1.

[70] G. Demartini and C. Niederée, "Finding experts on the semantic desktop," in *Proceedings of the International Semantic Web Conference, (ISWC '08)*, 2008. ISBN 978-3-540-88563-4.

[71] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977. ISSN 1467-9868.

[72] H. Deng, I. King, and M. R. Lyu, "Formal models for expert finding on DBLP bibliography data," in *Proceedings of the IEEE International Conference on Data Mining, (ICDM '08)*, Washington, DC, USA, pp. 163–172, 2008.

[73] H. Deng, I. King, and M. R. Lyu, "Enhancing expertise retrieval using community-aware strategies," in *Proceeding of the ACM International Conference on Information and Knowledge Management, (CIKM '09)*, New York, NY, USA, pp. 1733–1736, 2009. ISBN 978-1-60558-512-3.

[74] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial Mathematics, 1996. ISBN 0898713641.

[75] A. Doan and A. Y. Halevy, "Semantic-integration research in the database community," *AI Magazine*, vol. 26, no. 1, pp. 83–94, 2005. ISSN 0738-4602.

[76] B. Dom, I. Eiron, A. Cozzi, and Y. Zhang, "Graph-based ranking algorithms for e-mail expertise analysis," in *Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, (DMKD '03)*, New York, NY, USA, pp. 42–48, 2003.

[77] X. Dong, A. Halevy, and J. Madhavan, "Reference reconciliation in complex information spaces," in *Proceedings of the ACM SIGMOD International Conference on Managementof Data, (SIGMOD '05)*, New York, NY, USA, pp. 85–96, 2005. ISBN 1-59593-060-4.

[78] H. Duan, Q. Zhou, Z. Lu, O. Jin, S. Bao, and Y. C. Y. Yu, "Research on enterprise track of TREC 2007 at SJTU APEX lab," in *Proceedings of the Text REtrieval Conference, (TREC '06)*, Gaithersburg, MD, 2007.

[79] ECSCW'99 Workshop, *Beyond Knowledge Management: Managing Expertise*. 1999. URL http://www.informatik.uni-bonn.de/~prosec/ECSCW-XMWS/.

[80] K. Ehrlich, C. Lin, and V. Griffiths-Fisher, "Searching for experts in the enterprise: Combining text and social network analysis," in *Proceedings of the 2007 International ACM SIGGROUP Conference on Supporting Group Work, (GROUP '07)*, New York, NY, USA, pp. 117–126, 2007. ISBN 978-1-59593-845-9.

[81] K. Ehrlich and N. S. Shami, "Searching for expertise," in *Proceeding of the Annual SIGCHI Conference on Human Factors in Computing Systems, (CHI '08)*, New York, NY, USA, pp. 1093–1096, 2008. ISBN 978-1-60558-011-1.

[82] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate record detection: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, pp. 1–16, January 2007. ISSN 1041-4347.

[83] H. Fang and C. Zhai, "Probabilistic models for expert finding," in *Proceedings of the European Conference on IR Research, (ECIR'07)*, Berlin, Heidelberg, pp. 418–430, 2007. ISBN 978-3-540-71494-1.

[84] Y. Fang, L. Si, and A. Mathur, "Facfinder: Search for expertise in academic institutions," Technical Report: SERC-TR-294, Department of Computer Science, Purdue University, 2008.

[85] Y. Fang, L. Si, and A. Mathur, "Ranking experts with discriminative probabilistic models," in *SIGIR Workshop on Learning to Rank for Information Retrieval, (LR4IR '09)*, 2009.

[86] Y. Fang, L. Si, and A. Mathur, "Discriminative graphical models for faculty homepage discovery," *Information Retrieval*, vol. 13, no. 6, pp. 618–635, 2010. ISSN 1386-4564.

[87] Y. Fang, L. Si, and A. Mathur, "Discriminative probabilistic models for expert search in heterogeneous information sources," *Information Retrieval*, vol. 14, no. 2, pp. 158–177, 2011. ISSN 1386-4564.

[88] Y. Fang, L. Si, and A. P. Mathur, "Discriminative models of integrating document evidence and document-candidate associations for expert search," in *Proceeding of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '10)*, New York, NY, USA, pp. 683–690, 2010. ISBN 978-1-4503-0153-4.

[89] Y. Fang, N. Somasundaram, L. Si, J. Ko, and A. Mathur, "Analysis of an expert search query log," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '11)*, New York, NY, USA, pp. 1189–1190, 2011. ISBN 978-1-4503-0757-4.

[90] S. Fissaha Adafre and M. de Rijke, "Discovering missing links in wikipedia," in *Proceedings of the International Workshop on Link Discovery, (LinkKDD '05)*, New York, NY, USA, pp. 90–97, 2005. ISBN 1-59593-135-X.

[91] S. Fissaha Adafre, M. de Rijke, and E. T. K. Sang, "Entity retrieval," in *Recent Advances in Natural Language Processing*, *(RANLP '07)*, McGraw-Hill, September 2007. ISBN 978-90-272-4825-1.

[92] E. A. Fox and J. A. Shaw, "Combination of multiple searches," in *Text REtrieval Conference (TREC-2)*, NIST, pp. 243–252, 1994.

[93] Y. Fu, R. Xiang, Y. Liu, M. Zhang, and S. Ma, "Finding experts using social network analysis," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, *(WI '07)*, Washington, DC, USA, pp. 77–80, 2007. ISBN 0-7695-3026-5.

[94] Z. Guan, G. Miao, R. McLoughlin, X. Yan, and D. Cai, "Co-occurrence based diffusion for expert search on the web," *IEEE Transactions on Knowledge and Data Engineering*, 2012. ISSN 1041-4347.

[95] H. Halpin, D. M. Herzig, P. Mika, R. Blanco, J. Pound, H. S. Thompson, and T. T. Duc, "Evaluating ad-hoc object retrieval," in *Proceedings of the International Workshop on Evaluation of Semantic Technologies*, *(IWEST '10)*, 2010.

[96] T. Heath, E. Motta, and M. Petre, "Person to person trust factors in word of mouth recommendation," in *Proceedings of the CHI2006 Workshop on Reinventing Trust, Collaboration, and Compliance in Social Systems*, *(Reinvent '06)*, 2006.

[97] R. Herbrich, T. Graepel, and K. Obermayer, "Large margin rank boundaries for ordinal regression," *Advances in Large Margin Classifiers*, vol. 88, no. 2, pp. 115–132, 2000.

[98] M. Hertzum, "People as carriers of experience and sources of commitment: Information seeking in a software design project," *New Review of Information Behaviour Research*, vol. 1, pp. 135–149, 2000. ISSN 1471-6313.

[99] M. Hertzum and A. M. Pejtersen, "The information-seeking practices of engineers: searching for documents as well as for people," *Information Processing and Management*, vol. 36, no. 5, pp. 761–778, 2000. ISSN 0306-4573.

[100] K. Hofmann, K. Balog, T. Bogers, and M. de Rijke, "Integrating contextual factors into topic-centric retrieval models for finding similar experts," in *Proceedings of the SIGIR Workshop on Future Challenges in Expertise Retrieval*, *(fCHER '08)*, 2008.

[101] K. Hofmann, K. Balog, T. Bogers, and M. de Rijke, "Contextual factors for finding similar experts," *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 994–1014, May 2010. ISSN 0002-8231.

[102] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '99)*, New York, NY, USA, pp. 50–57, 1999. ISBN 1-58113-096-1.

[103] A. Hogan and A. Harth, "The expertfinder corpus 2007 for the benchmarking and development of expert-finding systems," in *Proceedings of the International ExpertFinder Workshop*, *(EFW '07)*, 2007.

[104] D. Horowitz and S. Kamvar, "The anatomy of a large-scale social search engine," in *Proceedings of the International World Wide Web Conference*, *(WWW '10)*, New York, NY, USA, pp. 431–440, 2010. ISBN 978-1-60558-799-8.

[105] J. Jiang, S. Han, and W. Lu, "Expertise retrieval using search engine results," in *Proceedings of the SIGIR Workshop on Future Challenges in Expertise Retrieval*, *(fCHER '08)*, ACM, 2008.

[106] J. Jiang, W. Liu, X. Rong, and Y. Gao, "Adapting language modeling methods for expert search to rank Wikipedia entities," in *Proceedings of the Initiative on the Evaluation of XML Retrieval*, *(INEX '08)*, Berlin, Heidelberg, pp. 264–272, 2009. ISBN 978-3-642-03760-3.

[107] J. Jiang, W. Lu, and D. Liu, "CSIR at TREC 2007 expert search task," in *Proceedings of the Text REtrieval Conference*, *(TREC '06)*, Gaithersburg, MD, 2007.

[108] V. Jijkoun, M. A. Khalid, M. Marx, and M. de Rijke, "Named entity normalization in user generated content," in *Proceedings of the workshop on Analytics for Noisy Unstructured Text Data*, *(AND '08)*, New York, NY, USA: ACM, pp. 23–30, 2008.

[109] R. Kaptein and J. Kamps, "Finding entities in Wikipedia using links and categories," in *Proceedings of the Initiative on the Evaluation of XML Retrieval*, *(INEX '08)*, Berlin, Heidelberg: Springer Verlag, pp. 273–279, 2009. ISBN 978-3-642-03760-3.

[110] R. Kaptein, P. Serdyukov, A. de Vries, and J. Kamps, "Entity ranking using Wikipedia as a pivot," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '10)*, New York, NY, USA, pp. 69–78, 2010. ISBN 978-1-4503-0099-5.

[111] M. Karimzadehgan, R. W. White, and M. Richardson, "Enhancing expert finding using organizational hierarchies," in *Proceedings of the European Conference on IR Research*, *(ECIR'09)*, Berlin, Heidelberg, pp. 177–188, 2009. ISBN 978-3-642-00957-0.

[112] M. Karimzadehgan and C. Zhai, "Constrained multi-aspect expertise matching for committee review assignment," in *Conference on Information and Knowledge Management (CIKM'09)*, pp. 1697–1700, 2009. ISBN 978-1-59593-991-3.

[113] M. Karimzadehgan, C. Zhai, and G. Belford, "Multi-aspect expertise matching for review assignment," in *Proceeding of the ACM International Conference on Information and Knowledge Management*, *(CIKM '08)*, New York, NY, USA, pp. 1113–1122, 2008. ISBN 978-1-59593-991-3.

[114] M. Khalid, V. Jijkoun, and M. de Rijke, "The impact of named entity normalization on information retrieval for question answering," in *Proceedings of the European Conference on IR Research*, *(ECIR '08)*, Berlin, Heidelberg, pp. 705–710, 2008. ISBN 978-3-540-78645-0.

[115] J. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999. ISSN 0004-5411.

[116] J. Ko, E. Nyberg, and L. Si, "A probabilistic graphical model for joint answer ranking in question answering," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '07)*, New York, NY, USA, pp. 343–350, 2007. ISBN 978-1-59593-597-7.

[117] G. Kukla, P. Kazienko, P. Bródka, and T. Filipowski, "Soclake: social latent knowledge explorator," *The Computer Journal*, vol. 55, no. 3, pp. 258–276, 2012. ISSN 0010-4620.

[118] L. Kuncheva and C. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine learning*, vol. 51, no. 2, pp. 181–207, 2003.

[119] T. Lappas, K. Liu, and E. Terzi, "Finding a team of experts in social networks," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *(KDD '09)*, New York, NY, USA, pp. 467–476, 2009. ISBN 978-1-60558-495-9.

[120] V. Lavrenko and W. B. Croft, "Relevance based language models," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '01)*, New York, NY, USA, pp. 120–127, 2001. ISBN 1-58113-331-6.

[121] J. H. Lee, "Combining multiple evidence from different properties of weighting schemes," in *Special Interest Group on Informations Retrieval SIGIR'95*, pp. 180–188, 1995.

[122] J.-Z. Li, J. Tang, J. Zhang, Q. Luo, Y. Liu, and M. Hong, "Eos: expertise oriented search using social networks," in *Proceedings of the International World Wide Web Conference*, *(WWW '07)*, New York, NY, USA, pp. 1271–1272, 2007. ISBN 978-1-59593-654-7.

[123] R. Liebregts and T. Bogers, "Design and implementation of a university-wide expert search engine," in *Proceedings of the European Conference on IR Research*, *(ECIR'09)*, Berlin, Heidelberg, pp. 587–594, April 2009. ISBN 978-3-642-00957-0.

[124] J. Liker and D. Meier, *Toyota Talent: Developing Your People the Toyota Way*. McGraw-Hill, 2007. ISBN 0071477454.

[125] B. Lin, K. D. Rosa, R. Shah, and N. Agarwal, "Lads: Rapid development of a learning-to-rank based related entity finding system using open advancement," in *Proceedings of the International Workshop on Entity-Oriented Search*, *(EOS '11)*, pp. 14–19, 2011.

[126] T. Liu, "Learning to rank for information retrieval," *Foundations and Trends in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009. ISSN 1554-0669.

[127] X. Liu and W. B. Croft, "Cluster-based retrieval using language models," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '04)*, New York, NY, USA, pp. 186–193, 2004. ISBN 1-58113-881-4.

[128] X. Liu, W. B. Croft, and M. Koll, "Finding experts in community-based question-answering services," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '05)*, New York, NY, USA, pp. 315–316, 2005. ISBN 1-59593-140-6.

[129] C. Macdonald, "The voting model for people search," PhD thesis, University of Glasgow, Feburary 2009.

[130] C. Macdonald, D. Hannah, and I. Ounis, "High quality expertise evidence for expert search," in *Proceedings of the European Conference on IR Research*, *(ECIR '08)*, Berlin, Heidelberg, pp. 283–295, 2008. ISBN 978-3-540-78645-0.

[131] C. Macdonald and I. Ounis, "Searching for expertise using the Terrier platform," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '06)*, New York, NY, USA, pp. 732–732, 2006. ISBN 1-59593-369-7.

[132] C. Macdonald and I. Ounis, "Voting for candidates: Adapting data fusion techniques for an expert search task," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '06)*, New York, NY, USA, pp. 387–396, 2006. ISBN 1-59593-433-2.

[133] C. Macdonald and I. Ounis, "Expertise drift and query expansion in expert search," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '07)*, New York, NY, USA, pp. 341–350, 2007. ISBN 978-1-59593-803-9.

[134] C. Macdonald and I. Ounis, "Using relevance feedback in expert search," in *Proceedings of the European Conference on IR Research*, *(ECIR'07)*, Berlin, Heidelberg, pp. 431–443, 2007. ISBN 978-3-540-71494-1.

[135] C. Macdonald and I. Ounis, "Expert search evaluation by supporting documents," in *Proceedings of the European Conference on IR Research*, *(ECIR '08)*, Berlin, Heidelberg, pp. 555–563, 2008. ISBN 978-3-540-78645-0.

[136] C. Macdonald and I. Ounis, "Key blog distillation: Ranking aggregates," in *Proceeding of the ACM International Conference on Information and Knowledge Management*, *(CIKM '08)*, New York, NY, USA, pp. 1043–1052, 2008. ISBN 978-1-59593-991-3.

[137] C. Macdonald and I. Ounis, "The influence of the document ranking in expert search," in *Proceeding of the ACM International Conference on Information and Knowledge Management*, *(CIKM '09)*, New York, NY, USA, pp. 1983–1986, 2009. ISBN 978-1-60558-512-3.

[138] C. Macdonald and I. Ounis, "Searching for expertise: experiments with the voting model," *Computer Journal: Special Focus on Profiling Expertise and Behaviour*, vol. 52, no. 7, pp. 729–748, 2009. ISSN 0010-4620.

[139] C. Macdonald and I. Ounis, "The influence of the document ranking in expert search," *Information Processing and Management*, vol. 74, no. 3, pp. 376–390, 2010. ISSN 0306-4573.

[140] C. Macdonald and I. Ounis, "Learning models for ranking aggregates," in *Proceedings of the European Conference on IR Research*, *(ECIR '11)*, Berlin, Heidelberg, pp. 517–529, 2011. ISBN 978-3-642-20160-8.

[141] C. Macdonald, R. L. Santos, I. Ounis, and I. Soboroff, "Blog track research at TREC," *SIGIR Forum*, vol. 44, pp. 58–75, August 2010. ISSN 0163-5840.

[142] C. Macdonald and R. W. White, "Usefulness of click-through data in expert search," in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '09)*, New York, NY, USA, pp. 816–817, 2009. ISBN 978-1-60558-483-6.

[143] R. Maclin and D. Opitz, "Popular ensemble methods: An empirical study," *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999.

[144] G. S. Mann and D. Yarowsky, "Unsupervised personal name disambiguation," in *Proceedings of the Conference on Natural Language Learning at HLT-NAACL*, Stroudsburg, PA, USA, pp. 33–40, 2003.

[145] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008. ISBN 0521865719.

[146] M. Maybury, "Expert finding systems," Technical Report: MTR-06B000040, 2006.

[147] D. W. McDonald, "Evaluating expertise recommendations," in *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, *(GROUP '01)*, New York, NY, USA, pp. 214–223, 2001.

[148] D. W. McDonald and M. S. Ackerman, "Expertise recommender: a flexible recommendation system and architecture," in *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, *(CSCW '00)*, New York, NY, USA, pp. 231–240, 2000.

[149] E. Meij, M. Bron, L. Hollink, B. Huurnink, and M. de Rijke, "Mapping queries to the linking open data cloud: A case study using dbpedia," *Journal of Web Semantics*, vol. 9, no. 4, pp. 418–433, 2011. ISSN 1570-8268.

[150] E. Meij, M. Bron, B. Huurnink, L. Hollink, and M. de Rijke, "Learning semantic query suggestions," in *Proceedings of the International Semantic Web Conference*, *(ISWC '09)*, pp. 415–430, 2009. ISBN 978-3-642-04929-3.

[151] E. Meij, W. Weerkamp, and M. de Rijke, "Adding semantics to microblog posts," in *Proceedings of the ACM International Conference on Web Search and Data Mining*, *(WSDM '12)*, New York, NY, USA, 2012.

[152] H. Menzel, *Review of Studies in the Flow of Information Among Scientists*. New York: Columbia University, Bureau of Applied Social Research, 1960.

[153] H. Menzel, "Information needs and uses in science and technology," *Annual Review of Information Science and Technology*, vol. 1, pp. 41–69, 1966.

[154] R. Mihalcea and A. Csomai, "Wikify!: Linking documents to encyclopedic knowledge," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '07)*, New York, NY, USA, pp. 233–241, 2007. ISBN 978-1-59593-803-9.

[155] D. Milne and I. H. Witten, "Learning to link with Wikipedia," in *Proceeding of the ACM International Conference on Information and Knowledge Management*, *(CIKM '08)*, New York, NY, USA, pp. 509–518, 2008. ISBN 978-1-59593-991-3.

[156] D. Mimno and A. McCallum, "Expertise modeling for matching papers with reviewers," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *(KDD '07)*, New York, NY, USA, pp. 500–509, 2007. ISBN 978-1-59593-609-7.

[157] A. Mockus and J. D. Herbsleb, "Expertise browser: A quantitative approach to identifying expertise," in *Proceedings of the International Conference on Software Engineering*, *(ICSE '02)*, New York, NY, USA, pp. 503–512, 2002.

[158] C. Moreira, "Learning to rank academic experts," Master Thesis, Technical University of Lisbon, 2011.

[159] C. Moreira, P. Calado, and B. Martins, "Learning to rank for expert search in digital libraries of academic publications," *Progress in Artificial Intelligence*, pp. 431–445, 2011.

[160] M. R. Morris, J. Teevan, and K. Panovich, "A comparison of information seeking using search engines and social networks," in *Proceedings of the International Conference on Weblogs and Social Media, (ICWSM '10)*, pp. 291–294, 2010.

[161] R. Nallapati, "Discriminative models for information retrieval," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '04)*, New York, NY, USA, pp. 64–71, 2004. ISBN 1-58113-881-4.

[162] R. Neumayer, K. Balog, and K. Nørvåg, "On the modeling of entities for adhoc entity search in the web of data," in *European Conference on Information Retrieval (ECIR 2012)*, pp. 133–145, 2012.

[163] A. Ng and M. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Proceedings of the Advances in Neural Information Processing Systems, (NIPS '01)*, MIT Press, pp. 841–848, 2001. ISBN 0-262-02550-7.

[164] M. G. Noll, C. m. Au Yeung, N. Gibbins, C. Meinel, and N. Shadbolt, "Telling experts from spammers: expertise ranking in folksonomies," in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR '09)*, New York, NY, USA, pp. 612–619, 2009. ISBN 978-1-60558-483-6.

[165] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," Technical Report 1999–66, 1999.

[166] A. Pal, S. Chang, and J. Konstan, "Evolution of experts in question answering communities," in *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pp. 274–281, 2012.

[167] M. Peacock, "The search for expert knowledge continues," *CMSWire*, 2009. http://www.cmswire.com/cms/enterprise-cms/the-search-for-expert-knowledge-continues-004594.php.

[168] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988. ISBN 1558604790.

[169] J. Pehcevski, A.-M. Vercoustre, and J. A. Thom, "Exploiting locality of wikipedia links in entity ranking," in *Proceedings of the European Conference on IR Research, (ECIR '08)*, Glasgow, UK, pp. 258–269, 2008.

[170] D. Petkova and W. B. Croft, "Hierarchical language models for expert finding in enterprise corpora," in *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence, (ICTAI '06)*, Washington, USA, pp. 599–608, 2006.

[171] D. Petkova and W. B. Croft, "Proximity-based document representation for named entity retrieval," in *Proceedings of the ACM International Conference on Information and Knowledge Management, (CIKM '07)*, New York, NY, USA, pp. 731–740, 2007. ISBN 978-1-59593-803-9.

[172] V. Plachouras, "Diversity in expert search," in *Proceedings of the Workshop on Diversity in Document Retrieval*, (C. Macdonald, J. Wang, and C. Clarke, eds.), pp. 63–67, 2011.

[173] J. Pound, P. Mika, and H. Zaragoza, "Ad-hoc object retrieval in the web of data," in *Proceedings of the International World Wide Web Conference, (WWW '10)*, New York, NY, USA, pp. 771–780, 2010. ISBN 978-1-60558-799-8.

[174] F. Riahi, Z. Zolaktaf, M. Shafiei, and E. Milios, "Finding expert users in community question answering," in *Proceedings of the International Conference on World Wide Web — Workshop on Community Question Answering on the Web*, *(CQA'12)*, pp. 791–798, 2012. ISBN 978-1-4503-1229-5.

[175] M. Richardson and R. W. White, "Supporting synchronous social q&a throughout the question lifecycle," in *Proceedings of the International World Wide Web Conference*, *(WWW '11)*, New York, NY, USA, pp. 755–764, 2011. ISBN 978-1-4503-0632-4.

[176] S. Robertson and K. Jones, "Relevance weighting of search terms," *Journal of the American Society for Information Science and Technology*, vol. 27, no. 3, pp. 129–146, 1976. ISSN 0002-8231.

[177] S. E. Robertson, "The probability ranking principle in IR," *Journal of Documentation*, vol. 33, no. 4, pp. 294–304, 1977. ISSN 0022-0418.

[178] J. Rocchio, *Relevance Feedback in Information Retrieval.* Englewood Cliffs, N.J.: Prentice Hall, 1971.

[179] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, *(UAI '04)*, pp. 487–494, 2004. ISBN 0-9749039-0-6.

[180] V. Rosenberg, "Factors affecting the preferences of industrial personnel for information gathering methods," *Information Storage and Retrieval*, vol. 3, no. 3, pp. 119–127, 1967. ISSN 0020-0271.

[181] G. Salton, A. Wong, and C. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975. ISSN 0001-0782.

[182] M. Sanderson, "Test collection based evaluation of information retrieval systems," *Foundations and Trends in Information Retrieval*, vol. 4, no. 4, pp. 247–375, 2010. ISSN 1554-0669.

[183] R. L. T. Santos, C. Macdonald, and I. Ounis, "Voting for related entities," in *Adaptivity, Personalization and Fusion of Heterogeneous Information, (RIAO '10)*, Paris, France, pp. 1–8, 2010.

[184] R. L. T. Santos, C. Macdonald, and I. Ounis, "Mimicking web search engines for expert search," *Information Processing and Management*, vol. 47, no. 4, pp. 467–481, 2011. ISSN 0306-4573.

[185] T. Saracevic and P. B. Kantor, "A study of information seeking and retrieving. III. searchers, searches, overlap," *Journal of the American Society for Information Science and Technology*, vol. 39, pp. 197–216, 1988.

[186] M. F. Schwartz and D. C. M. Wood, "Discovering shared interests using graph analysis," *Communications of the ACM*, vol. 36, pp. 78–89, August 1993. ISSN 0001-0782.

[187] D. Seid and A. Kobsa, *Demoir: A Hybrid Architecture for Expertise Modeling and Recommender Systems.* 2000.

[188] P. Serdyukov, R. Aly, and H. Djoerd, "University of Twente at the TREC 2008 enterprise track: Using the global web as an expertise evidence source," in *Proceedings of the Text REtrieval Conference, (TREC '08)*, Gaithersburg, MD, 2009.

[189] P. Serdyukov, S. Chernov, and W. Nejdl, "Enhancing expert search through query modeling," in *Proceedings of the European Conference on IR Research, (ECIR'07)*, Berlin, Heidelberg, pp. 737–740, 2007. ISBN 978-3-540-71494-1.

[190] P. Serdyukov and D. Hiemstra, "Being omnipresent to be almighty: The importance of the global web evidence for organizational expert finding," in *Proceedings of the SIGIR Workshop on Future Challenges in Expertise Retrieval, (fCHER '08)*, pp. 17–24, 2008.

[191] P. Serdyukov and D. Hiemstra, "Modeling documents as mixtures of persons for expert finding," in *Proceedings of the European Conference on IR Research, (ECIR '08)*, Berlin, Heidelberg, pp. 309–320, 2008. ISBN 978-3-540-78645-0.

[192] P. Serdyukov, H. Rode, and D. Hiemstra, "Modeling multi-step relevance propagation for expert finding," in *Proceeding of the ACM International Conference on Information and Knowledge Management, (CIKM '08)*, New York, NY, USA, pp. 1133–1142, 2008. ISBN 978-1-59593-991-3.

[193] P. Serdyukov, M. Taylor, V. Vinay, M. Richardson, and R. W. White, "Automatic people tagging for expertise profiling in the enterprise," in *Proceedings of the European Conference on IR Research, (ECIR '11)*, Berlin, Heidelberg, pp. 399–410, 2011. ISBN 978-3-642-20160-8.

[194] S. N. Shami, K. Ehrlich, and D. R. Millen, "Pick me!: Link selection in expertise search results," in *Proceeding of the Annual SIGCHI Conference on Human Factors in Computing Systems, (CHI '08)*, New York, NY, USA, pp. 1089–1092, 2008. ISBN 978-1-60558-011-1.

[195] S. N. Shami, C. Y. Yuan, D. Cosley, L. Xia, and G. Gay, "That's what friends are for: facilitating 'who knows what' across group boundaries," in *Proceedings of the 2007 International ACM Conference on Supporting Group Work, (GROUP '07)*, New York, NY, USA, pp. 379–382, 2007. ISBN 978-1-59593-845-9.

[196] M. Shokouhi and L. Si, "Federated search," *Foundations and Trends in Information Retrieval*, vol. 5, no. 1, 2011. ISSN 1554-0669.

[197] E. Smirnova and K. Balog, "A user-oriented model for expert finding," in *Proceedings of the European Conference on IR Research, (ECIR '11)*, Berlin, Heidelberg, pp. 580–592, 2011. ISBN 978-3-642-20160-8.

[198] I. Soboroff, A. de Vries, and N. Crawell, "Overview of the TREC-2006 enterprise track," in *Proceedings of the Text REtrieval Conference, (TREC '06)*, Gaithersburg, MD, 2007.

[199] D. Song and P. D. Bruza, "Towards context sensitive information inference," *Journal of the American Society for Information Science and Technology*, vol. 54, pp. 321–334, February 2003.

[200] Y. Song, J. Huang, I. C. Councill, J. Li, and C. L. Giles, "Efficient topic-based unsupervised name disambiguation," in *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries, (JCDL '07)*, New York, NY, USA, pp. 342–351, 2007. ISBN 978-1-59593-644-8.

[201] P. Sorg and P. Cimiano, "Finding the right expert: Discriminative models for expert retrieval," in *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, (KDIR '11)*, 2011.

[202] P. Sorg, P. Cimiano, A. Schultz, and S. Sizov, "Overview of the cross-lingual expert search (cries) pilot challenge," in *Workshop of Cross-Language Evaluation Forum, (CLEF '10)*, September 2010. ISBN 978-88-904810-0-0.

[203] H. Su, J. Tang, and W. Hong, "Learning to diversify expert finding with subtopics," in *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, (PAKDD'12)*, 2012. ISBN 978-3-642-30216-9.

[204] C. Sutton and A. McCallum, "An introduction to conditional random fields for relational learning," *Introduction to Statistical Relational Learning*, p. 93, 2007.

[205] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (KDD '09)*, New York, NY, USA, pp. 807–816, 2009. ISBN 978-1-60558-495-9.

[206] J. Tang, J. Zhang, R. Jin, Z. Yang, K. Cai, L. Zhang, and Z. Su, "Topic level expertise search over heterogeneous networks," *Machine Learning Journal*, vol. 82, no. 2, pp. 211–237, 2011.

[207] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: Extraction and mining of academic social networks," in *Proceeding of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (KDD '08)*, New York, NY, USA, pp. 990–998, 2008. ISBN 978-1-60558-193-4.

[208] W. Tang, J. Tang, T. Lei, C. Tan, B. Gao, and T. Li, "On optimization of expertise matching with various constraints," *Neurocomputing*, vol. 76, no. 1, pp. 71–83, 2012.

[209] D. Tax, M. V. Breukelen, R. Duin, and J. Kittler, "Combining multiple classifiers by averaging or by multiplying?," *Pattern Recognition*, vol. 33, no. 9, pp. 1475–1485, 2000. ISSN 0031-3203.

[210] L. Terveen and D. W. Mcdonald, "Social matching: A framework and research agenda," *ACM Transactions on Computer-Human Interaction*, vol. 12, no. 3, pp. 401–434, 2005. ISSN 1073-0516.

[211] T. Tsikrika, P. Serdyukov, H. Rode, T. Westerveld, R. Aly, D. Hiemstra, and A. de Vries, "Structured document retrieval, multimedia retrieval, and entity ranking using PF/Tijah," in *Proceedings of the Initiative on the Evaluation of XML Retrieval*, vol. 4862 *of (INEX '07)*, Berlin, Heidelberg: Springer Verlag, pp. 306–320, March 2008. ISBN 978-3-540-85901-7.

[212] Y. Tu, N. Johri, D. Roth, and J. Hockenmaier, "Citation author topic model in expert search," in *Proceedings of the International Conference on Computational Linguistics: Posters, (COLING '10)*, pp. 1265–1273, 2010.

[213] H. Turtle and W. B. Croft, "Evaluation of an inference network-based retrieval model," *ACM Transactions on Information Systems*, vol. 9, pp. 187–222, 1991.

[214] V. Vapnik, *The Nature of Statistical Learning Theory.* Springer Verlag, 2000. ISBN 0387987800.

[215] W. Weerkamp, K. Balog, and M. de Rijke, "Blog feed search with a post index," *Information Retrieval*, vol. 14, pp. 515–545, 2011. ISSN 1386-4564.

[216] W. Weerkamp and M. de Rijke, "Credibility improves topical blog post retrieval," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, *(ACL'08: HLT)*, Stroudsburg, PA, USA, pp. 923–931, June 2008. ISBN 978-1-932432-04-6.

[217] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proceedings of the ACM International Conference on Web Search and Data Mining*, *(WSDM '10)*, New York, NY, USA, pp. 261–270, 2010. ISBN 978-1-60558-889-6.

[218] K. M. Wiig, "Knowledge management: Where did it come from and where will it go?," *Expert Systems with Applications*, vol. 13, pp. 1–14, July 1997.

[219] W. E. Winkler, "The state of record linkage and current research problems," Technical Report, Statistical Research Division, U.S. Census Bureau, 1999.

[220] L. S. E. Woudstra and B. J. Van den Hooff, "Inside the source selection process: Selection criteria for human information sources," *Information Processing and Management*, vol. 44, pp. 1267–1278, 2008. ISSN 0306-4573.

[221] Y. Xue, "Thuir at TREC2008: enterprise track," in *Proceedings of the Text REtrieval Conference*, *(TREC '07)*, Gaithersburg, MD, 2008.

[222] S. Yahyaei and C. Monz, "Applying maximum entropy to known-item email retrieval," in *Proceedings of the European Conference on IR Research*, *(ECIR '08)*, Berlin, Heidelberg, pp. 406–413, 2008. ISBN 978-3-540-78645-0.

[223] Z. Yang, J. Tang, B. Wang, J. Guo, J. Li, and S. Chen, "Expert2bole: From expert finding to bole search," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *(KDD '09)*, New York, NY, USA, pp. 1–4, 2009. ISBN 978-1-60558-495-9.

[224] S. Yarosh, T. Matthews, and M. Zhou, "Asking the right person: supporting expertise selection in the enterprise," in *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems*, *(CHI'12)*, pp. 2247–2256, 2012.

[225] D. Yimam-Seid and A. Kobsa, "Expert finding systems for organizations: Problem and domain analysis and the demoir approach," *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 1, pp. 1–24, 2003. ISSN 1091-9392.

[226] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow, "Learning to estimate query difficulty: Including applications to missing content detection and distributed information retrieval," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *(SIGIR '05)*, New York, NY, USA, pp. 512–519, 2005. ISBN 1-59593-034-5.

[227] C. Zhai, "Statistical language models for information retrieval a critical review," *Foundations and Trends in Information Retrieval*, vol. 2, pp. 137–213, March 2008. ISSN 1554-0669.

[228] C. Zhai and J. Lafferty, "Model-based feedback in the language modeling approach to information retrieval," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, *(CIKM '01)*, New York, NY, USA, pp. 403–410, 2001. ISBN 1-58113-436-3.

[229] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to information retrieval," *ACM Transactions on Information Systems*, vol. 22, pp. 179–214, April 2004.

[230] J. Zhang and M. S. Ackerman, "Searching for expertise in social networks: a simulation of potential strategies," in *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*, *(GROUP '05)*, New York, NY, USA, pp. 71–80, 2005. ISBN 1-59593-223-2.

[231] J. Zhang, M. S. Ackerman, and L. Adamic, "Expertise networks in online communities: Structure and algorithms," in *Proceedings of the International World Wide Web Conference*, *(WWW '07)*, New York, NY, USA, pp. 221–230, 2007. ISBN 978-1-59593-654-7.

[232] J. Zhang, J. Tang, and J. Li, "Expert finding in a social network," *Advances in Databases: Concepts, Systems and Applications*, pp. 1066–1069, 2007.

[233] J. Zhang, J. Tang, L. Liu, and J. Li, "A mixture model for expert finding," in *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, *(PAKDD '08)*, pp. 466–478, 2008. ISBN 978-3-540-68124-3.

[234] M. Zhang, R. Song, C. Lin, S. Ma, Z. Jiang, Y. Jin, Y. Liu, and L. Zhao, "Expansion-based technologies in finding relevant and new information: THU TREC2002 novelty track experiments," in *Proceedings of the Text Retrieval Conference (TREC)*, pp. 586–590, 2002.

[235] J. Zhu, "A study of the relationship between *ad-hoc* retrieval and expert finding in enterprise environment," in *Proceeding of the ACM Workshop on Web Information and Data Management*, *(WIDM '08)*, New York, NY, USA, pp. 25–30, 2008.

[236] J. Zhu, X. Huang, D. Song, and S. Ruger, "Integrating multiple document features in language models for expert finding," *Knowledge and Information Systems*, vol. 23, no. 1, pp. 29–54, 2010. ISSN 0219-1377.

[237] J. Zhu, D. Song, and S. Rüger, "Integrating document features for entity ranking," in *Proceedings of the Initiative on the Evaluation of XML Retrieval*, *(INEX '07)*, Berlin, Heidelberg, pp. 336–347, 2008. ISBN 978-3-540-85901-7.

[238] J. Zhu, D. Song, S. Rüger, and X. Huang, "Modeling document features for expert finding," in *Proceeding of the ACM International Conference on Information and Knowledge Management*, *(CIKM '08)*, New York, NY, USA, pp. 1421–1422, 2008. ISBN 978-1-59593-991-3.

[239] J. Zhu, D. Song, and S. M. Rüger, "Integrating multiple windows and document features for expert finding," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 4, pp. 694–715, 2009. ISSN 0002-8231.

[240] J. Zhu, D. Song, S. M. Rüger, M. Eisenstadt, and E. Motta, "The open university at trec 2006 enterprise track expert search task," in *Proceedings of the Text REtrieval Conference*, *(TREC '06)*, Gaithersburg, MD, 2007.