Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Bachelor's thesis

Expert Retrieval using weighted query-based Expertise Graphs on Citation Networks

in the Language Technology (LT) group

by

Tim Fischer

born on February 29, 1996 in Lübeck

Matriculation number 6818629

Field of study Computer Science

submitted November 13, 2018

Supervisor: Steffen Remus

First Reviewer: Chris Biemann

Second Reviewer: Steffen Remus

Abstract

This work is primarily concerned with the expert finding task. This task can be easily described as the identification of relevant experts given a particular topic. An expert finding system, therefore, allows a user to type a simple text query and to retrieve a ranked list of names and preferable further information about the individuals that posses the expertise described in the user's query. Two baseline approaches to expert finding are presented and explained in detail. The first baseline approach is based on generative language modeling aimed at finding expertise relations between topics and people whereas the other baseline approach models candidate experts (people), documents and various relations among them with expertise graphs. Most approaches to expert finding are quite effective in simple domains like organizations and universities, however, it has to be examined how these approaches deal with more complex domains such as citation networks, hierarchical organizations or expert social networks. Especially in citation networks, there exists a huge number of documents and candidate experts as well as a very large number of possible topics. These facts make it difficult to identify the true experts among many candidate experts. In order to address these problems, this work proposes an enhancement of the second baseline. By assigning meaningful weights to the edges of the expertise graph, documents and candidate experts can be ranked and distinguished better. The weighting is done by including additional information like number of collaborations, h-index and topicality. The *ACL Anthology Network* is used as the dataset for the experimental evaluation of the three different expert finding methods as it is a clean and easy to use citation network. This evaluation has two purposes. On the one hand, the applicability of the two baseline methods on citation networks has to be explored. On the other hand, the two different approaches to expert finding as well as this work's proposal have to be compared. To conduct the evaluation, an expert finding system was developed, which incorporates the mentioned baselines, this work's proposed method as well as several other methods for expert retrieval. Experimental results using the developed expert finding system show that our proposed method can improve the performance and results of expert finding compared to the two baseline approaches. Moreover, the evaluation reveals that the standard expert retrieval methods are shown to be robust to other domains like citation networks and they appear to be generalizable to other settings.

Contents

1 Introduction

1.1 Motivation

1.1.1 Expert Finding

Human expertise is one of the noteworthy resources in the world. However, expertise is rare and difficult to quantify, experts vary in experience and their expertise is continually changing. It is a challenging but very rewarding task to keep track of people's expertise in such a way that experts for specific topics can be identified.

Expert finding is a problem that has many real-world applications. Employees may want to obtain some background knowledge on a project. A project leader may need persons with particular skills for a successful project. Companies may require a highly trained specialist who is consultable about a specific problem. Also, expert finding is very useful when individuals are new to an organization and need advice and assistance, especially when they are within a very large or distributed organization. Other applications of expert finding could be that an organizer of a conference may need to assign submissions to the members based on their expertise or that the customer service of a company may have to decide which staff should be assigned to solve a given problem. Recruiters may search for talented employees while consultants may search for other consultants to redirect requests and not lose clients (Fang et al. 2007; Balog et al. 2009; Serdyukov et al. 2008).

There are many reasons why people search for actual persons rather than for relevant documents. The needed information may not be accessible because it was not considered important enough to be released, because it is not available in electronic format or because it is just hardly expressible in written language. In these cases asking people becomes the only way to find an answer (Craswell et al. 2001). Experts are often able to explain the search topic in detail and even guide the searcher further into related areas. Furthermore, they can be in demand not only for asking them questions, but also for assigning them to some task (Serdyukov et al. 2008).

Particularly for organizations it is fundamental that the expertise is effectively utilized. Some most valuable knowledge in an enterprise resides in the minds of its employees. It is a significant challenge within any organization to manage the expertise of employees, so that experts for specific knowledge areas can be found. Moreover, because identifying experts may reduce costs and facilitate a better solution than could be achieved otherwise, expert finding is often critical to the success of projects being undertaken (Balog et al. 2006). Especially if experts within an organization are dispersed geographically, functionally or structurally facilitating collaborations through expert identification will ensure that the expertise is used effectively. Since resources and documents hold a range of knowledge and expertise they are a valuable asset to organizations. An organization's intranet provides evidence for employees' expertise and is, as a consequence, a very important foundation for expert finding (Balog et al. 2009).

In the past databases housing the skills and knowledge of each individual were set-up to approach expert finding. These databases were constructed and consulted manually. Users had to identify the experts manually, which was obviously labor-intensive and time-consuming while administrators had to put in considerable effort for setup and maintenance. Thus, it was very interesting for many researches to study how to automatically identify experts for specified expertise areas.

With the launching of the *Enterprise Track* of the *Text REtrieval Conference (TREC)* expert finding gained much attention. The *TREC* provided a common platform for researchers to empirically assess methods and techniques developed for expert finding. Researchers were presented with the following scenario: Given a crawl of the World Wide Web Consortium's website, a list of candidate experts and a set of topics, the task was to provide a ranked list of candidates for each of these topics. The expertise retrieval task gained popularity in the research community during the *TREC Enterprise Track* (2005-2008) (Balog et al. 2008; Bailey et al. 2007; Soboroff et al. 2006; Craswell et al. 2005) and has remained relevant ever since.

1.1.2 Expert Profiling

"People not only are interested in searching for different types of information (such as authors, conferences and papers) but also are keen on finding semantic-based information (such as structured researcher profiles)" (Tang et al. 2008). People searching for expertise are often looking for experts. However, the desired output should be more informative than a ranked list of person names (Hawking 2004). In order to improve expert finding system it is therefore important to include context and evidence to help users of these tools decide whom to contact when seeking expertise in a certain knowledge area. A candidate expert's topical profile should consist of a list of knowledge areas and the level of competence in each whereas an expert's social profile should contain information about her collaborative network. Especially the social profile including the colleagues and collaborators contributes greatly to the value of an expert; an isolated expert might be able to answer specific questions whereas a well-connected expert might put us on track to explore new or additional areas (Balog et al. 2007b).

1.2 Formalization

1.2.1 Expert Finding

Expert finding is one of the challenging types of search, which concerns itself with ranking people who are knowledgeable in a given topic: *"Who are the experts on topic X?"*. For a given query, the task is to identify which of the candidates are likely to be an expert. Following the notation of Balog et al. 2006 this can be also stated as: *"What is the probability of a candidate* $ca \in CA$ *being an expert given the query topic* q*?"*. The goal of expert finding is to identify a ranked list of people who are knowledgeable about a given topic. Therefore, it is necessary to determine $p(ca|q)$ and rank candidates CA according to this probability. Following the probabilistic ranking principle the candidates with the highest probability given the query are supposed to be experts for that topic. The challenge here is to accurately estimate this probability. Normally, $p(ca|q)$ would be calculated this way:

$$p(ca|q) = p(ca,q)/p(q)$$

However, there is no information available to estimate $p(ca,q)$. As a consequence, Bayes' Theorem is invoked to model the probability $p(ca|q)$:

$$p(ca|q) = \frac{p(q|ca) \times p(ca)}{p(q)}$$

where $p(ca)$ is the a priori belief that a candidate ca is an expert and $p(q)$ is the prior probability of a query. Because $p(q)$ is a constant it can be ignored for the purpose of ranking. For that reason, the probability of a candidate ca being an expert given the query q is proportional to the probability of a query given the candidate $p(q|ca)$, multiplied with the prior probability of a candidate expert $p(ca)$[1]:

$$p(ca|q) \propto p(q|ca) \times p(ca)$$

A considerable amount of work and research has been done to estimate the probability of a query given the candidate, $p(q|ca)$. Many methods and models were developed that link up to that formalization. The two most famous and very successful approaches are adaptations to generative probabilistic language models, that are widely used in Information Retrieval, which were first proposed by Balog et al. (2006) and were revised from another perspective by Fang et al. (2007) later.

Other approaches to expert finding interpret the task in another way. Serdyukov et al. (2008), for example, view expert finding as a process of walking through a topic-specific expertise graph of candidate experts and their associated documents. The language model adaption as well as this graph-based method are explained in detail in Chapter 3 since they will serve as a baseline for comparison in this work.

1.2.2 Expert Profiling

While the goal of expert finding is to identify a ranked list of people who are experts on a given topic, the task of expert profiling seeks an answer to a related question: *"What topics does a candidate know about?"*. Hence, the task of expert profiling is to return a list of topics that a person is knowledgeable about. Essentially, this turns the question of expert finding around. The profiling involves identification of areas of skills and an estimation of the level of proficiency in each of theses areas. This is often called topical profile.

A topical profile of a candidate expert can be defined as a vector where each element i of the vector corresponds to the candidate expert's ca expertise on a given topic k. The *expertise score*, which determines the before mentioned level of proficiency, can be denoted as $s(ca,k_i)$. Each topic k_i defines a particular knowledge area that for example an organization uses to define the candidate's topical profile. In most expert profiling tasks it is assumed that a list of topics $k_1,...,k_n$ is given where n is the number of pre-specified topics. Thus, a topical profile can be formally defined as:

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

1. Formula and definition is adopted from Balog et al. 2009

The problem of quantifying a person's expertise on a certain knowledge area can be stated as follows:

> *"What is the probability of a knowledge area k_i being part of the candidate's ca topical profile?"*

This means basically that $s(ca, k_i)$ is defined by $p(k_i|ca)$. The *expertise score* for a given topic k_i is therefore defined as the probability that a candidate ca concerns herself with the topic k_i. Then, the challenging task is to estimate $p(k_i|ca)$, which is equivalent to the problem of estimating $p(q|ca)$, since the topic k_i can also be represented as a query topic q.[2]

Both, expert finding and expert profiling tasks rely on the accurate estimation of the probability that a candidate deals with the query topic q. The probability of the query topic being associated to a candidate expert plays a key role in the process of expert finding and expert profiling. Therefore, it can be stated that the main challenge in both expert finding and expert profiling is to infer the association between a person and an expertise area from resources that contain evidence of expertise (Balog et al. 2007a; Balog et al. 2007b).

Similar Experts

While most work concentrates on the task of expert finding and some work has been done regarding expert profiling it is also worth noting that other expert retrieval tasks have been identified. Balog et al. (2007c) address the task of finding similar experts in their work.

1.3 Problem Statement

As elaborated in Section 1.1.1, experts are highly useful for a number of reasons, and identifying experts is critical to many applications. Therefore, this work will concentrate on the task of expert finding, which has been defined in Section 1.2.

Identification of knowledgeable persons on a specific academic topic can be of great value to many applications, for example recognizing qualified experts to supervise new researchers, assigning a paper to reviewers (Neshati et al. 2012), forming a team of experts (Lappas et al. 2009) and recommending panels of reviewers for state research grant applications (Wang et al. 2015). Moreover, finding people that have expertise on a specific academic topic can benefit students and researchers. Presenting them an overview of relevant authors for a given topic can support them with their studies and provide them a different insight into a topic as well as lead them to related research areas or related information.

Most previous work regarding expert finding has concentrated on simple domains such as organizations (Balog et al. 2012) and universities (Balog et al. 2007a). However, little work has been done on methods of expert finding in any specific academic field, even though this is an important practical problem. While initial approaches are effective in the mentioned simple domains like organizations and universities, they are not suitable for complex domains such as bibliographic network (Deng et al. 2008), hierarchical organizations (Karimzadehgan et al. 2009) and expert's social networks (Neshati et al. 2012). It turns out that expert finding has been

2. Formula and definition is adopted from Balog et al. 2007a

studied thoroughly in the business area while the academic field and bibliographic networks are largely unexplored regarding expert finding.

In typical expert finding methods, the associated documents of each candidate are considered as the main evidence of her expertise. However, a good expert finding solution should take into account more valuable evidence, which is available in extended domains. These evidences can be social interactions between individuals, social connections, temporal behaviours of them as well as document quality indicators, like number of citations or type of the document, which are usually independent of the document's content (Hashemi et al. 2013). The following example is intended to illustrate the problem with standard expert finding methods: suppose a query q has the same relevance to two documents d_1 and d_2, which are associated with authors ca_1 and ca_2 respectively. In addition to that, d_1 is cited by 200 documents, whereas d_2 is cited by 10 documents. Using standard expert finding methods and the above information, the author ca_1 has the same expertise as the author $a2$ given the query q. When considering the citation count however, it is intuitively more reasonable that ca_1 has a higher expertise than ca_2.

In order to address this problem, this work will focus on three goals. First, we want to analyze the performance of standard expert retrieval methods on extended domains. Second, we want to explore useful additional information available in extended domains that enhances the performance of expert finding methods as well as to measure the impact of this information on the expert ranking. Last, we want to introduce an expert finding method based on weighted query-based expertise graphs exploiting additional information available in extended domains since to the best of our knowledge graph based methods for expert finding have not been used on extended domains before. In order to achieve these goals, our proposed method as well as two standard baseline methods for expert finding will be evaluated and compared using the *ACL*[3] *Anthology Network* as the extended domain.

An important first step to reach the goals is the acquisition of a data set containing documents, candidate experts as well as further information from which expertise can be accessed. The *ACL Anthology Network* is a perfectly suited corpus for this task, since it contains more than 23,000 papers written by more than 18,000 authors from conferences and journals in the field of computational linguistics and natural language processing. In scientific research, the publication of a researcher could be assumed to be representative of her expertise (Chakraborty et al. 2018). Extended search domains like bibliographic networks contain various types of documents (e.g. conference proceedings, journal articles), experts (e.g. students and supervisors) and relationships between them (e.g. coauthor and citation relationships) (Deng et al. 2008). This is also true for the *ACL Anthology Network* as it is a semi-automatic curated database of citations, collaborations and publications. Therefore, the *ACL Anthology Network* provides enough documents, authors and other information in order to form the foundation for this work.

Expert finding in a bibliographic network is a challenging task, mainly because of the following reasons stated by Hashemi et al. (2013):

1. In contrast with simple domains, there exist a huge number of documents and candidate experts in a bibliographic network, making it difficult to find the distinguished experts among numerous candidate experts.

2. Besides the content of associated documents of a person in these networks, it is necessary to consider some other important factors such as the quality of documents, social interactions and temporal behavior of authors for expert ranking. As a result, identification of

3. Association of Computational Linguistics

these effective features is an important step toward building a high-quality search engine for bibliographic networks.

3. While scientific publications are usually a product of cooperation among members of a research group, it is not obvious how to estimate the contribution / importance of each author in a multi-author publication.

The rest of this work is organized as follows: In Chapter 2, related work on expertise retrieval is summarized. Chapter 3 provides detailed descriptions of the standard expert finding methods, which are used as baselines. Next, the proposed model based on weighted, query-based expertise graphs is formally described in Chapter 4. In Chapter 5, the baselines and the proposed method are evaluated and compared. Finally, the work is concluded and some future work is discussed in Chapter 6.

2 Related Work

Text mining can be described as the automatic discovery of new, previously unknown information from unstructured textual data. It deals with the machine supported analysis of text and is often seen as the combination of three major tasks: information retrieval, information extraction as well as natural language processing. Defining text mining is difficult, as its definition is motivated by the specific perspective of the research areas. Dependent on the research area, text mining is interpreted as the task of information extraction or as the application of algorithms from the fields of machine learning to find useful patterns (Hotho et al. 2005). Typical text-mining tasks include, besides others, text categorization, concept or entity extraction, sentiment analysis, and document summarization. These technique were successfully applied, for example, in automatic indexing and classifying consumer complaints. Text mining was also used in genome analysis, media analysis, and indexing of documents in large databases for retrieval purposes (Himmel et al. 2009). Most importantly however, text mining, if seen as information extraction or rather information retrieval, was utilized to search for expertise in large document repositories.

Expert finding gained its popularity when Microsoft, Hewlett-Packard and NASA made their experience in building expert finding systems public (Davenport 1997a; Davenport 1997b; Becerra-Fernandez 2000). These systems were not fully automated and represented data repositories of manually created skill descriptions of their employees with simple search functionality. Early approaches like that relied on employees to self-assess their skill against a prespecified set of keywords. It is obvious that the explication of this information for each individual in the organization was laborious and costly and consequently rarely performed. Moreover, the static nature of databases often rendered them incomplete and antiquated. These databases soon became outdated and did not reflect the current expertise of the employees. As a result, the need for automated technologies rose. To address these disadvantages, several systems were proposed, aimed at automatically discovering expertise information.

Early automatic expert finding systems usually focused on specific domains like email (Campbell et al. 2003) or software engineering and software documentation (Mockus et al. 2002) to build profiles and to find experts. Nevertheless, approaches like this also had apparent disadvantages and limitations. Consequently, instead of focusing on solely a specific document type there was increased interest in systems that mine documents of an organization's intranet along with other resources of evidence and enable the search of all kinds of expertise within organizations.

First pioneering approaches to expert finding, which automatically extract expertise from any kind of document, can be categorized as profile centric. One of these published approaches was the P@noptic system by Craswell et al. (2001), which was one of the first to overcome these limitations.

The follow up document centric approaches were developed in the course of the *TREC*. Due to the inclusion of expert finding task in the *TREC Enterprise Track* the task of expert finding received a significant amount of attention from 2005 - 2008. This led to the proposal of many expert finding methods. *TREC* provided with the enterprise track a common platform for researchers to evaluate and assess methods and techniques. The following scenario was presented to the researchers: Given a crawl of the *World Wide Web Consortium (W3C)* the task was to find experts for each of a pre-defined set of topics (*TREC Enterprise Track* 2005-2008 – Balog et al. 2008; Bailey et al. 2007; Soboroff et al. 2006; Craswell et al. 2005).

At *TREC* it emerged that there are two main approaches to expert finding: candidate and document models, or with other words, profile centric and document centric models. While profile centric methods model the knowledge of a specific expert from her associated documents directly, document centric methods first locate documents that are related to the topic and then find associated experts. Most systems that participated in the expert finding task implemented one of these two models. One of the most famous formalizations of these two methods is from Balog et al. (2006).

Even though a few variations of profile centric methods performed better than document centric methods, the document centric model was generally more effective and easier to implement than the profile centric method (Balog et al. 2006). Therefore, most of the following approaches, developed during and after *TREC* to solve the expert finding task, were basically extensions or further refinements of the document centric method.

Since the expert finding task gained a huge amount of interest, especially because of the *Enterprise Track* of *TREC*[1], many contributions and ideas exist that focus on improving the two standard models. In most existing work, document and candidate priors were assumed to be uniform. Fang et al. (2007), however, used prior knowledge for example to encode the importance of a document or candidate within the modeling process. They showed that the usage of candidate or document priors can greatly improve the performance of expert finding methods. Shortly thereafter, Deng et al. (2008) proposed the weighted language model, which used the citation count as a document prior. Fundamentally, this was just like the suggestion by Hashemi et al. (2013). They proposed to consider the quality of a document by taking the number of incoming citations into account (for research papers) as well as to consider social interactions and temporal behaviour. Their main contribution for enhancing the existing models, though, was to improve document-author associations. Especially with research papers with multiple co-authors, it is often impossible to figure out who contributed the most to a paper. In order to solve this problem the authors suggested a method that detects the leading author. Balog et al. (2007a) claimed that in real world applications, the number of topics is around 30 times that of the standard *TREC W3C* corpus. For that reason, they proposed a method to find knowledge area similarities so that an expert finding method can utilize other related topics as further evidence to support the original query. Fang et al. (2007) had a very similar idea to that called topic expansion or query expansion. Other suggestions utilized the document structure or exploited the hierarchical, organizational and topical context and structure (Zhu et al. 2006; Petkova et al. 2006; Balog et al. 2007a).

Generative probabilistic models were in the focus and the favored method among researchers for the tasks of expert retrieval. Such models fell in the categories of candidate generation models, topic generation models and document generation models. The most famous formalization of the document and candidate generation models by (Balog et al. 2006) was revised by Fang et al. (2007). In their work, the authors derived the document and candidate generation models from a more general probabilistic framework. Other researchers, for example Petkova et al. (2006), tried a combination of the above mentioned candidate and document model while explicitly modeling topics to further improve the performance of expert finding methods.

1. Text REtrieval Conference

Despite of the domination of generative probabilistic models, alternative approaches exist that do not fall into the same category. Beyond unsupervised generative approaches, there were graph-based approaches based on random walks proposed by Serdyukov et al. (2008) and voting based approaches with respect to a topic based on data fusion proposed by Macdonald et al. (2006). Furthermore, there were other strategies calculating the centrality of experts in their organizational social network to rank and to find experts (Campbell et al. 2003; Jurczyk et al. 2007; Zhang et al. 2007).

However, more recent work went in another direction again. The unsupervised log-linear model developed by Van Gysel et al. (2016) combined the advantages of the standard document and candidate models. Their proposed model had the ranking performance of the best document model with an inference time complexity in the number of candidate experts as standard candidate models. Moreover, Demartini et al. (2009) proposed a vector space-based method for entity ranking. Their developed framework extended vector spaces operating on documents to entities. Even though vector space models yield very good results using TF-IDF[2] and LSI[3], this approach was not able to cope with the previously mentioned log-linear model.

Turning to other expert retrieval tasks that can also be addressed using topic-people associations, Balog et al. (2007b) addressed the task of determining topical expert profiles in their paper. Also, in another paper, Balog et al. (2007c) studied the related task of finding experts that are similar to small set of experts given as input.

2. Term frequency–inverse document frequency
3. Latent semantic indexing

3 Baseline Models

3.1 Model2

This section presents the popular document model Model2 created by Balog et al. (2006), which is used as a baseline for comparison in this work.

As discussed in Section 1.2.1, the expert finding task can be modeled as

$$p(ca|q) \propto p(q|ca) \times p(ca) \tag{3.1}$$

where ca is a candidate, q is the query and $p(ca)$ is the prior probability of a candidate. Therefore, the main problem is to estimate the conditional probability of a query given the candidate $p(q|ca)$.

For the Model2 proposed by Balog et al. (2006), it is necessary to estimate the probability that a document d is associated with a candidate ca. To define this probability an association $a(ca,d)$ has to be calculated for each document and each candidate. A simple function for $a(ca,d)$ could be defined as:

$$a(ca,d) = \begin{cases} 1, & \text{if } ca \text{ is author of } d \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

Using this association it is possible to estimate the strength of the association between d and ca in terms of the probability $p(d|ca)$ or $p(ca|d)$. These probabilities are defined as:

$$p(d|ca) = \frac{a(ca,d)}{\sum_{d' \in D} a(ca,d')} \tag{3.3}$$

$$p(ca|d) = \frac{a(ca,d)}{\sum_{ca' \in CA} a(ca,d')} \tag{3.4}$$

where D is the set of documents and CA is the set of candidate experts. A simple example follows to demonstrate the utility of these equations: If all documents for a given candidate are ranked using the probability $p(d|ca)$, the top documents are those with which the candidate is most strongly associated.

The most important step in expert finding is the estimation of $p(q|ca)$. Balog et al. (2006) claim that the probability of a query given a candidate can be estimated by the following generative process:

1. Let a candidate ca be given.
2. Select a document d associated with ca.
3. From this document, generate the query q, with probability $p(q|d,ca)$.

Then, by applying this generative process to all documents $d \in D$, $p(q|ca)$ can be obtained, which can be formally expressed as:

$$p(q|ca) = \sum_{d \in D} p(q|d,ca) \times p(d|ca)\,. \tag{3.5}$$

The process of expert finding with Model2 can also be described as follows: Given a collection of documents ranked according to the query, each document is examined and if it is relevant, the candidate associated with the document is noticed, since it is assumed that the document is evidence for the candidate's expertise.

In order to rank documents based on their relevance to the query or to generate a query from a document, the probability of a query given a document and a candidate has to be determined. Assuming that query terms are sampled identically and independently, the probability of a query given the document and the candidate is:

$$p(q|d,ca) = \prod_{t \in q} p(t|d,ca)^{n(t,q)}\,. \tag{3.6}$$

Lastly, the probability of a document generating a certain term, $p(t|d,ca)$, has to be estimated. One way to do this, is to assume conditional independence between the query and the candidate. Balog et al. (2012) presented an alternative way without the independence assumption in their paper. As a result, $p(t|d,ca)$ becomes proportional to $p(t|\theta_d)$. Then, $p(t|d,ca)$ is estimated by inferring a document model θ_d for each document d, so that the probability of a term $t \in$ query q given the document model θ_d is:

$$p(t|\theta_d) = (1-\lambda) \times p(t|d) + \lambda \times p(t)\,. \tag{3.7}$$

The parameter λ is used for smoothing. Balog et al. (2006) for instance used the Jelinek-Mercer smoothing method with $\lambda = 0.5$ in their paper. They stated that even though this is not the best possible setting for all models, it is a reasonable setting that provides acceptable performance across the parameter space. By substituting Equation 3.6 into Equation 3.5 the following estimate of the document-based model is obtained:

$$p(q|ca) = \sum_{d \in D} \prod_{t \in q} p(t|d,ca)^{n(t,q)} \times p(d|ca)\,. \tag{3.8}$$

Substituting $p(t|\theta_d)$ from Equation 3.7 into $p(t|d,ca)$ from Equation 3.8, which is possible due to the conditional independence assumption, leads to the final estimation of Model2:

$$p(q|ca) = \sum_{d \in D} \prod_{t \in q} \left((1-\lambda) \times p(t|d) + \lambda \times p(t)\right)^{n(t,q)} \times p(d|ca)\,. \tag{3.9}$$

The Model2 by Balog et al. (2006) is very popular. This is mainly because it yields reasonable and good results as well as it is easy to implement on top of a standard document index. Moreover, this method embodies a general strategy to find experts and operates on documents of any kind. Thus, it can be said that Model2 is domain-independent. For said reasons as well as the fact that many researchers compare their expert finding methods with Model2, this method is also used for comparison in this work.

3.2 Relevance Propagation

The relevance propagation approach to the expert finding task proposed by Serdyukov et al. (2008) is more advanced and builds upon Model2 by Balog et al. (2006), which was explained in detail in Section 3.1. This approach utilizes graphs for identifying and ranking experts and differs therefore from the common methods.

3.2.1 Expertise Graphs

Serdyukov et al. (2008) were the first to introduce so-called expertise graphs with the goal to model appropriate graphs that represent the association between candidate experts and documents in a certain knowledge area.

A simple expertise graph is constructed as follows: First, a set of documents associated with scores determining their relevance to a given query is obtained from initial standard document retrieval. Next, a set of candidate experts is extracted from these ranked documents by scanning each document for authorship. Finally, this authorship relation between candidates and documents is represented in an expertise graph. Documents and candidate experts are represented as nodes, whereas directed edges symbolize the authorship relations. It is important to note that methods utilizing expertise graphs are query dependent solely due to the restriction to the top ranked documents. The number of top ranked documents is also a very influential parameter on the graph's size and density. Figure 3.1 demonstrates how a simple expertise graph could look like.

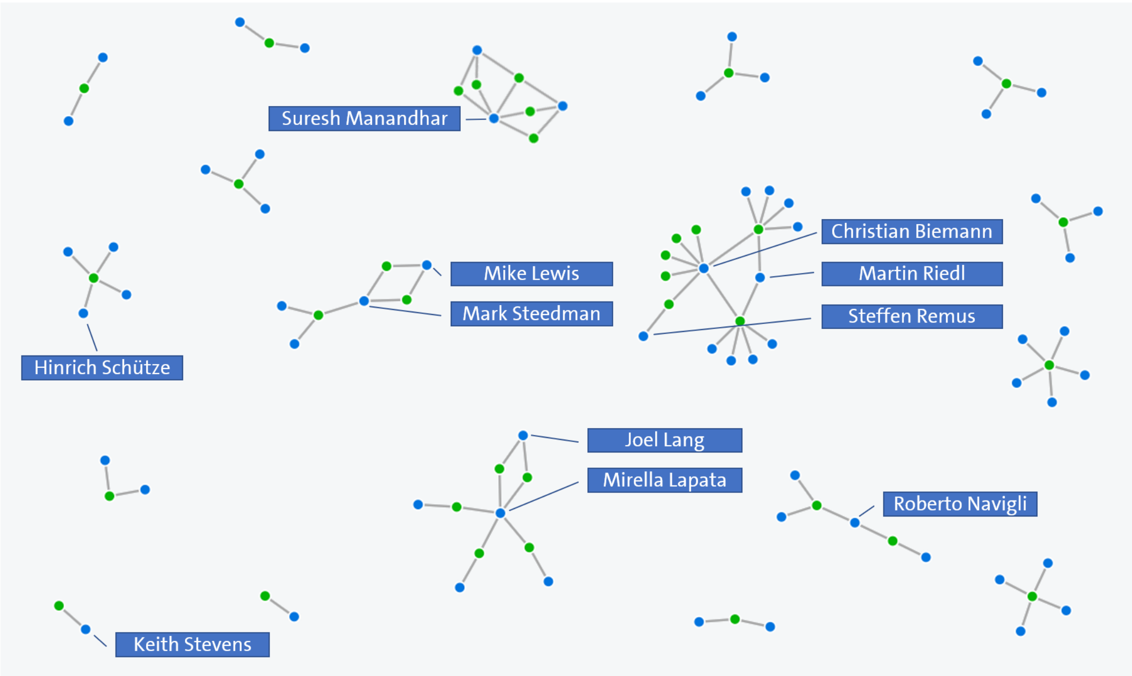

Figure 3.1: A simple expertise graph. Blue nodes represent candidate experts, green nodes represent documents. Edges represent the authorship relations. Some author names were annotated manually.

Since the graph will be used later to propagate relevance, and a well-connected graph is beneficial for that, it is very important to exploit all available relations between the entities of the graph.

Depending on the domain not only authorship relations, but also links between documents or connections between candidate experts are known. Including such additional edges will increase the graph's density and will ensure more intense relevance propagation. Inter-document links are likely to be represented by directed edges while connections from expert to expert are usually bidirectional. Figure 3.2 shows the same simple expertise graph as Figure 3.1, however, this graph was created using citations as document-document relations, collaborations as author-author relations and authorship as author-document relations.

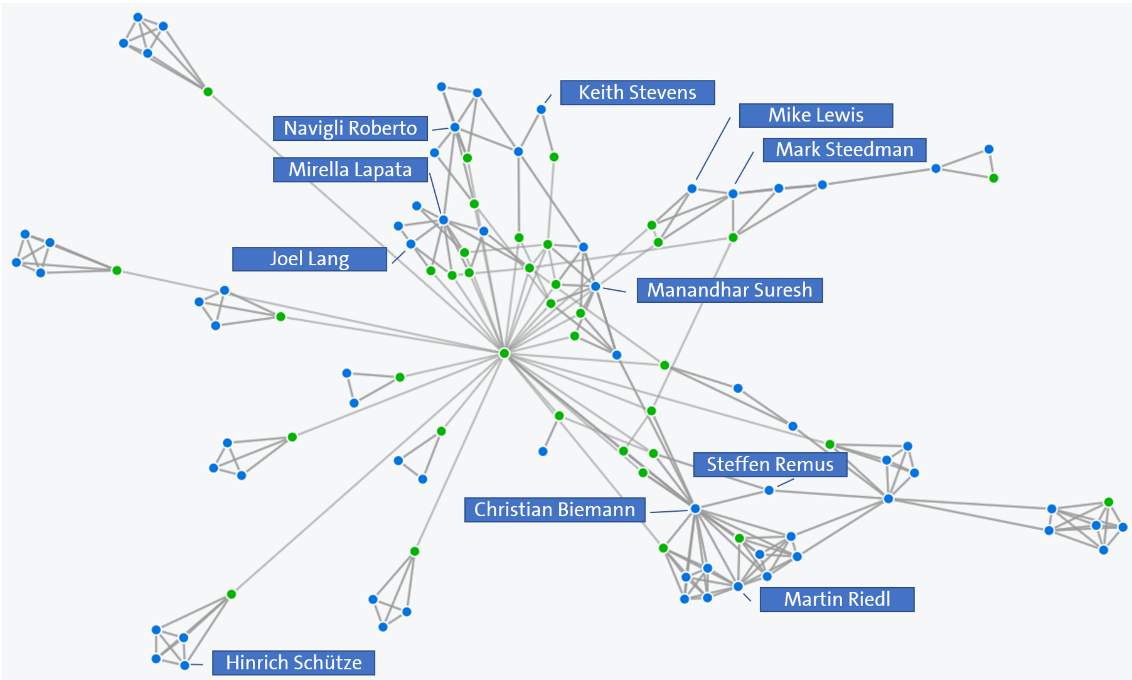

Figure 3.2: An expertise graph utilizing various relations between the entities. Blue nodes represent candidate experts, green nodes represent documents. Edges represent the different relations: authorship, citation, collaboration. Some author names were annotated manually.

Serdyukov et al. (2008) also suggest the inclusion of further entity types, so that the density of the graph gets even higher. Even though the expert finding task is mainly interested in the ranking of experts, it still might be helpful for the relevance propagation to increase the number of nodes and relations by introducing new entities. Depending on the domain, these entity types could be for example dates, locations or events.

3.2.2 One-step Relevance Propagation

Serdyukov et al. (2008) interpret the process of expert finding with Model2 by Balog et al. (2006) in a different way, which is described in the following.

In their point of view, Model2 defines a probabilistic process with three steps. First, a user selects a document among the ones appearing in the initial document retrieval on the user's query. The probability of selecting a certain document is its probabilistic relevance score as the user will most likely search for useful information and contacts of knowledgeable people in one of the top documents. Next, the user will read or skim through the document and enlist all mentioned candidate experts. Finally, the user will refer to one of the enlisted candidate experts with her current information need. The selection of the candidate expert depends on her contribution to the document. For example, it is more likely that the main author will be selected.

Serdyukov et al. (2008) interpret this process as a One-step Relevance Propagation from documents to related candidate experts.

3.2.3 Multi-step Relevance Propagation

The One-step Relevance Propagation might not be the best solution to point the user to the most knowledgeable person on a certain topic as it is not likely that reading only one document and consulting only one person is enough to entirely satisfy ones information need.

Therefore, Serdyukov et al. (2008) suggest an alternative way. According to them, the expert finding process should consist of the following repeating stages of gradual knowledge acquisition:

1. At any time: (a) randomly reading a document, or (b) just picking a random candidate
2. After reading a document: (a) consulting with a person mentioned in this document, or (b) checking for other linked documents and reading one of them
3. After consulting with a person: (a) reading other documents mentioning this person, or (b) consulting with another candidate expert, which is recommended by this person.

Serdyukov et al. (2008) try to overcome the limitations of the One-step Relevance Propagation with this process. The authors model the task of expert finding as a random walk through the expertise graph in seek of the most knowledgeable person on a given topic. In their paper, they propose three different random walk techniques, one of which, the infinite random walk, is presented in the following.

The infinite random walk approach is based on the assumption that the walk to find experts is a non-stop process. The initial probability of a document is equal to its relevance whereby the relevance of a document is defined as the probability that a document d generates the query q, same as in Model2. The probability of a candidate expert ca being an expert at the time t = 0 is zero. These initial probabilities are shown for candidate experts and documents in the Equations 3.10 and 3.11 respectively.

$$p_{t=0}(ca) = 0 \tag{3.10}$$

$$p_{t=0}(d) = p(q|d,ca) \tag{3.11}$$

The following equations characterize the infinite random walk and utilize the whole expertise graph by considering author-document (or document-author), document-document as well as author-author relations and by performing random jumps. These equations are calculated and applied each iteration until convergence:

$$\begin{aligned} p_{t+1}(ca) = \lambda p_j(ca) + (1-\lambda)\Big[& \\ & \big[(1-\mu_{ca}) \sum_{d \to ca} p(ca|d) p_t(d)\big] \\ & + \big[\mu_{ca} \sum_{ca' \to ca} \frac{1}{|N_{ca'}|} p_t(ca')\big]\Big] \end{aligned} \tag{3.12}$$

$$\begin{aligned} p_{t+1}(d) = \lambda p_j(d) + (1-\lambda)\Big[& \\ & \big[(1-\mu_d) \sum_{ca \to d} p(d|ca) p_t(ca)\big] \\ & + \big[\mu_d \sum_{d' \to d} \frac{1}{|N_{d'}|} p_t(d')\big]\Big] \end{aligned} \tag{3.13}$$

To disentangle the complexity of the two Equations 3.12 and 3.13, a detailed descriptions follows.

The first line in Equations 3.12 and 3.13 describes the probability that a random walker jumps to an entity. The parameter λ represents the probability that at any step the random walker decides to jump and not to follow links between entities anymore, while $p_j(ca)$ and $p_j(d)$ define the probability to jump to a certain candidate or a certain document, respectively. These probabilities can be defined as:

$$p_j(ca) = \frac{cf(ca, Top_d)}{|Top_d|} \tag{3.14}$$

$$p_j(d) = p(q|d, ca) \tag{3.15}$$

It is assumed that the more often a candidate is involved in top documents, the more likely the candidate expert is selected as a target for a random jump. Top_d are the top documents, $cf(ca, Top_d)$ is the number of top documents the candidate ca is associated with and $|Top_d|$ is the number of top documents. The probability to jump to a certain document, however, is equal to its relevance to the query same as in Equation 3.11.

The second line in Equations 3.12 and 3.13 encodes the probability that the random walker follows document-author links or author-document links respectively, whereby $p(ca|d)$ and $p(d|ca)$ indicate the probabilities of selecting a candidate given a document and of selecting a document given a candidate. These probabilities were already defined by Balog et al. 2006 and are estimated as defined in Equations 3.4 and 3.3.

The parameters μ_{ca} and μ_d in the last line of Equations 3.12 and 3.13 express the probability that the random walker follows author-author links or document-document links. Moreover, $N_{ca'}$ represents the number of outgoing author-author links from the candidate ca', while $N_{d'}$ represents the number of outgoing document-document links from the document d'.

Even though the Multi-step Relevance Propagation method is not as popular as the Model2 by Balog et al. (2006), it is still a good solution to the expert finding task. It utilizes way more information than Model2 and therefore should yield better results. However, this strategy requires a huge amount of data as well as many relations between the entities, so that the generated expertise graph has a high density as well as many edges. This is important because a well-connected graph is very beneficial to the relevance propagation. Consequently, this approach is dependent on additional domain-specific links, in contrast to the domain-independent approach followed in Model2.

This work uses the *ACL Anthology Network* dataset, which provides information about relations like collaboration, authorship and citation. Because the expertise graph can utilize all this additional information, the Multi-step Relevance Propagation approach seems very suitable and promising. The Multi-step Relevance Propagation method will serve as a baseline along with Model2 and as a foundation to this work's proposal: Expert retrieval using weighted query-based Expertise Graphs.

4 Expert Retrieval using weighted query-based Expertise Graphs

In order to understand how our proposed method works, it is necessary to consider the Multi-step Relevance Propagation method by Serdyukov et al. (2008) again and especially the Equations 3.12 and 3.13. By looking closely at the two formulas, six different probabilities influencing the result can be identified. In order to receive a more general formula, the equations can be rewritten as:

$$p_{t+1}(ca) = \lambda p_j(ca) + (1-\lambda)\Big[\big[(1-\mu_{ca}) \sum_{d \to ca} p(d \to ca) p_t(d)\big] + \big[\mu_{ca} \sum_{ca' \to ca} p(ca' \to ca) p_t(ca')\big]\Big] \tag{4.1}$$

$$p_{t+1}(d) = \lambda p_j(d) + (1-\lambda)\Big[\big[(1-\mu_{d}) \sum_{ca \to d} p(ca \to d) p_t(ca)\big] + \big[\mu_{d} \sum_{d' \to d} p(d' \to d) p_t(d')\big]\Big] \tag{4.2}$$

Besides the three parameters μ_d, μ_{ca}, λ, two jump probabilities $p_j(d)$, $p_j(ca)$ as well as four conditional probabilities $p(ca \to d)$, $p(d \to ca)$, $p(ca' \to ca)$, $p(d' \to d)$ can be identified. To understand the meaning of these six probabilities better, the probabilities are visualized in Figure 4.1.

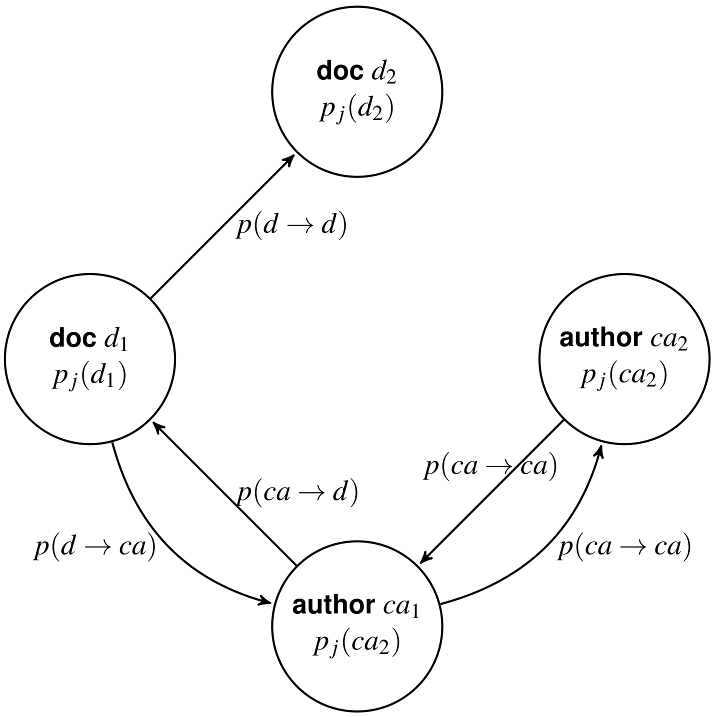

Figure 4.1: Jump probabilities & conditional probabilities in a simple expertise graph

The jump probabilities define the chance to jump to a document or author at any step while the different conditional probabilities encode the probability to follow corresponding links between entities. The approach by Serdyukov et al. (2008) is problematic in so far as all conditional probabilities are assumed to be uniform. This is only true if Equation 3.2 is used to estimate the probabilities $p(ca \rightarrow d)$, $p(d \rightarrow ca)$. The following example illustrates this issue: If a document d_1 cites two other documents d_2 and d_3, the probability to follow the citation link and examine d_2 is the same for d_3. Even though d_3 may be more relevant, for example due to more citations, the probability to examine d_2 and d_3 will stay the same.

We address this issue to improve expert finding methods with this work and, as the name suggests, our proposed method achieves this by assigning weights to the edges of query-based expertise graphs. We accomplish this by redefining the four conditional probabilities. The following sections describe how the conditional probabilities are changed.

4.1 Document-Author Relations

Imagine a user searching for expertise in a topic q of her choice. After a while, she will come across a very relevant document d written by several authors A_d. As the user is not satisfied with her search for expertise, she wants to dig deeper into the topic and therefore wants to take a look at one of the authors $ca \in A_d$. Then, she has to decide which author $ca \in A_d$ to consult. It is unreasonable that the user will just pick an author ca randomly. Consequently, a measurement for an author's relevance is necessary to rank the authors A_d. We assume the h-index is a suitable choice for determining the query-independent relevance of an author.

The h-index is defined as the highest number h of a scholar's published papers that have been cited at least h times. In other words, a scholar with an h-index of h has published h papers each of which has been cited at least h times. Thus, the h-index represents both the number of publications and the number of citations per publication. Calculations of our proposed method, however, utilize two different h-indexes. Normally, the h-index is calculated taking into account all documents written by a scholar. For this, we calculate the h-index by taking into account all publications that are present in the corpus and call it global h-index. As a consequence, the calculated global h-index for a certain scholar might be lower than for example the one listed on Google Scholar or elsewhere. Moreover, we define a second h-index called local h-index, which is calculated taking into account only the documents relevant to the user's query q. Naturally, the local h-index is often even lower than the global h-index.

Now, having both, local and global h-index, new information can be generated. By relating the local h-index with the global h-index it is possible to identify and differentiate candidate experts more precisely. For example, a candidate expert with a high global h-index but with a low local h-index has to be someone, who has written a few documents related to the user's query topic q, but also many documents concerning other topics. On the contrary, a candidate expert having a local h-index similar or equal to her global h-index must be someone, who has focused much of her work on the user's topic q. Comparing these two mentioned candidate experts, we assume the latter is the more relevant candidate expert.

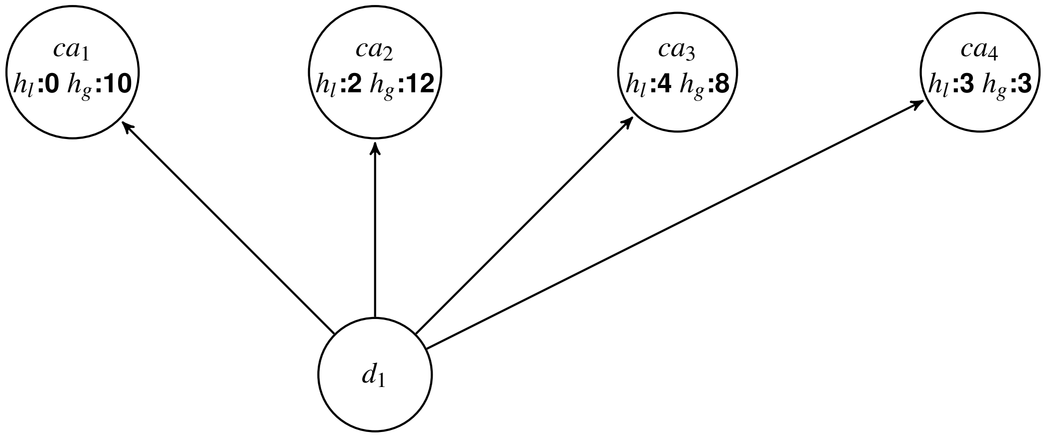

Figure 4.2: An expertise graph demonstrating document-author relations. h_g stands for the local h-index of an author; h_g stands for the global h-index of an author.

The expertise graph in Figure 4.2 shows a simple scenario demonstrating the usage of h-indexes. In this idealized expertise graph a document d_1 is written by 4 candidate experts ca_1 through ca_4. Every candidate expert has it's own local h-index h_l and global h-index h_g. Remember that the goal is to weight the edges between the document and the candidate experts. This is done by setting the local h-index in relation with the global h-index:

$$p(d \rightarrow ca) = \begin{cases} 0, & \text{if } h_l(ca) = 0 \text{ or } h_g(ca) = 0 \\ \frac{h_l(ca)}{h_g(ca)}, & \text{otherwise} \end{cases} \tag{4.3}$$

Applying this formula on the current example leads to following calculations:

$$\begin{aligned} p(d_1 \rightarrow ca_1) &= 0 \\ p(d_1 \rightarrow ca_2) &= \frac{2}{12} = \frac{1}{6} \\ p(d_1 \rightarrow ca_3) &= \frac{4}{8} = \frac{1}{2} \\ p(d_1 \rightarrow ca_4) &= \frac{3}{3} = 1 \end{aligned}$$

However, these results do not provide a helpful weighting, yet. If these scores were used as weights and are interpreted as the probability that a random walker follows this document-author link and consults the corresponding author, ca_4 would always be consulted whereas ca_1 would never be consulted. Therefore, in order to smooth the results as well as to obtain a better probability distribution that adds up to 1, the softmax function is applied on the results:

$$softmax(0, \frac{1}{6}, \frac{1}{2}, 1) = (0.15, 0.18, 0.25, 0.42) \tag{4.4}$$

4.2 Document-Document Relations

If a user searching for expertise in a topic q encountered a relevant document d' she would not only consult authors of the document d' but she would also take a closer look at the documents $D_{d'}$ cited by d'. Simply put, the searching user would not only follow document-author links but also document-document links. The same problem arises again, namely, the users has to decide which document $d \in D_{d'}$ to read. In order to solve this problem we introduce a feature called recency. We assume that a user will most likely decide to read the most recent paper cited by the document d'. A simple scenario similar to the one described above is visualized in Figure 4.3.

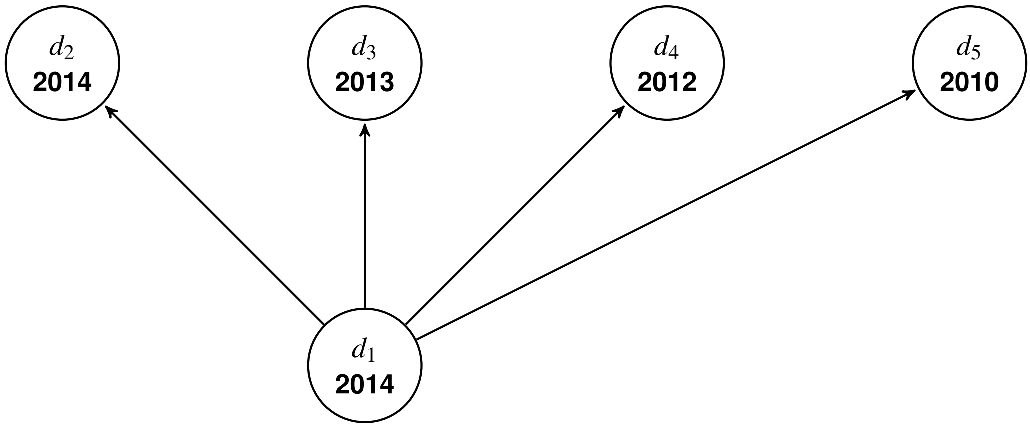

Figure 4.3: An expertise graph demonstrating document-document relations

In this expertise graph a document d_1 cites 4 other documents d_2 through d_5. It is important to note, that all documents are relevant to the query, otherwise they would not appear in the expertise graph as stated in Section 3.2.1. As each document is usually published in a certain year or can at least be associated with a date, this information can be used to calculate local and global distances. The local distance is simply defined as the difference between the publication years of two documents (4.5) while the global distance is defined as the difference of the current year and the publication year of the document. In this example, the local distance of d_3 and d_1 would be 1 and the global distance of d_3 would be 5, since it is 2018 at the time of writing this work.

$$local(d',d) = year(d') - year(d) \tag{4.5}$$

By relating the local distance to the sum of all local distances a first weighting of the document-document links can be obtained.

$$tf(d',d) = \begin{cases} 1, & \text{if } localsum(d') = 0 \\ \frac{localsum(d') - local(d',d)}{localsum(d')}, & \text{otherwise} \end{cases} \tag{4.6}$$

In Formula 4.6 $localsum(d')$ is the sum of all local distances between the document d' and the documents $d \in D_{d'}$ cited by d' and is simply defined as:

$$localsum(d') = \sum_{d' \to d} local(d',d)\,. \tag{4.7}$$

The calculation of the final weighting of the edges also takes into account the information about the global distance. It is promising to compare the global distance to the maximum global distance. The maximum global distance is dependent on the document corpus and is simply defined as the largest global distance of all documents in the corpus. For the sake of this example it is assumed that the oldest document in the corpus was written in 1965 and therefore the maximum global distance amounts to 53. The final formula for calculating the weighting of document-document relations is shown in Equation 4.8.

$$p(d' \to d) = tf(d',d) \times \frac{max}{global(d',d)} \tag{4.8}$$

Using this formula to compute the weights for the edges in the expertise graph in Figure 4.3 leads to following calculations:

$$\begin{aligned}
p(d_1 \to d_2) &= \frac{7-0}{7} \times log(\frac{53}{4}) = 2.58 \\
p(d_1 \to d_3) &= \frac{7-1}{7} \times log(\frac{53}{5}) = 2.02 \\
p(d_1 \to d_4) &= \frac{7-2}{7} \times log(\frac{53}{6}) = 1.56 \\
p(d_1 \to d_5) &= \frac{7-4}{7} \times log(\frac{53}{8}) = 0.81
\end{aligned}$$

In order to smooth the results as well as to get a probability distribution that adds up to 1 and thus to obtain a better weighting, the softmax function is applied on the results.

$$softmax(0.81, 1.56, 2.02, 2.58) = (0.08, 0.17, 0.27, 0.48) \tag{4.9}$$

4.3 Author-Author Relations

At some point an expertise seeking user will find a relevant author *ca*. Besides following document-author links, a good idea is also to examine other authors A_{ca}, that collaborated with the author *ca*. As always, the user needs to rank the authors A_{ca} according to a certain criteria. If following author-author links is interpreted as consulting authors A_{ca} that are recommended by the author *ca*, the most reasonable criteria is the number of collaborations as we assume an author would rather recommend a person with whom she has often collaborated than a person with whom she has worked only a few times. The collaboration count feature therefore defines how often two persons have worked together. In our implementation, two authors collaborated when they wrote a document together or appeared together in the list of authors of a document.

Our proposed method defines two different collaboration counts, namely, the global collaboration count and the local collaboration count. Similar to the local and global h-index, the global collaboration count indicates the number of collaborations across the whole document corpus, while the local collaboration count represents the number of collaborations in the set of documents relevant to the user's query q.

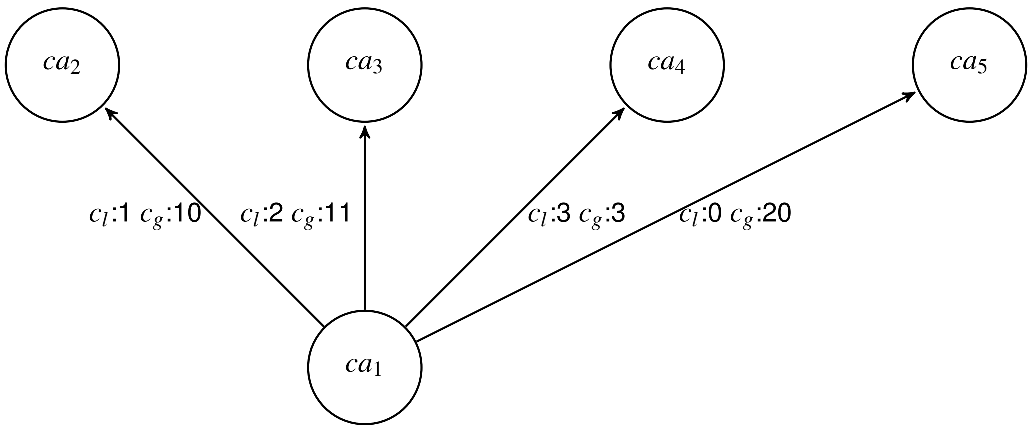

Figure 4.4: An expertise graph demonstrating author-author relations. c_l stands for local collaboration count; c_g stands for global collaboration count.

The exemplary expertise graph in Figure 4.4 shows 5 authors of which author ca_1 has collaborated with each of the authors ca_2 through ca_5. It is important to note, that this relation is bidirectional. However, the computed weights of each edge are direction-dependent. Simply put, this means that the probability that, for example, ca_1 recommends ca_2 is not necessarily the same as the probability that ca_2 recommends ca_1. Formally speaking, $p(ca_1 \rightarrow ca_2) \neq p(ca_2 \rightarrow ca_1)$.

Each edge is labeled with c_l and c_g representing the local and global collaboration count, respectively. The comparison of the number of local and global collaborations leads to new information. By relating these two collaboration counts it is possible to differentiate collaborations between authors more precisely. For example ca_1 and ca_5 have published and worked on 20 papers together, however, none of these papers were relevant to the user's query. On the contrary, ca_1 and ca_4 only worked on 3 papers together, but, in this case, all of these documents were relevant to the query. Therefore, we assume that ca_4 is a better candidate expert than ca_5. This is expressed in the following formula:

$$p(ca' \rightarrow ca) = \frac{c_l(ca', ca)}{1 + \sum_{ca' \rightarrow ca''} c_l(ca', ca'')} \times \frac{c_l(ca', ca)}{c_g(ca', ca)}. \tag{4.10}$$

Moreover, this formula includes local collaborations in proportion to the sum of all local collaborations outgoing from the author ca'. This increases the importance of authors with many local collaborations even more. The computations below show the formula applied on the expertise graph in Figure 4.4:

$$\begin{aligned} p(ca_1 \rightarrow ca_2) &= \frac{1}{7} \times \frac{1}{10} = \frac{1}{70} \\ p(ca_1 \rightarrow ca_3) &= \frac{2}{7} \times \frac{2}{11} = \frac{4}{77} \\ p(ca_1 \rightarrow ca_4) &= \frac{3}{7} \times \frac{3}{3} = \frac{3}{7} \\ p(ca_1 \rightarrow ca_5) &= \frac{0}{7} \times \frac{0}{20} = 0 \end{aligned} \tag{4.11}$$

There is one problem with this formula, though. Even though ca_5 has zero local collaborations with ca_1 meaning they did not work together on a document regarding the user's query topic, they still worked together 20 times. Moreover, ca_5 is relevant to the user's query, otherwise ca_5 would not appear in the expertise graph as explained in Section 3.2.1. Consequently, ca_5 could still be a good candidate expert. Therefore, assigning a zero probability to follow the author-author link between ca_1 and ca_5 seems unreasonable. For that reason as well as to obtain a probability distribution that adds up to 1 the softmax function is applied on the result set.

$$softmax(0, \frac{1}{70}, \frac{4}{77}, \frac{3}{7}) = (0.22, 0.22, 0.23, 0.33) \tag{4.12}$$

4.4 Remaining Relations

Out of the six identified probabilities mentioned at the beginning of this chapter, our proposed method alters three conditional probabilities, namely the document-author, document-document and author-author conditional probability. These conditional probabilities were modified to improve the initial method proposed by Serdyukov et al. (2008) by assigning non-uniform weightings to the expertise graph.

As the jump probabilities were not uniform in the initial approach and they work well, there was no need to change them. Hence, the jump probabilities of our proposed method remain the same as in Equations 3.14 and 3.15.

Unfortunately, no further easily accessible information was available in the corpus that could help to further differentiate and rank documents. For this reason the last remaining relation, the author-document relation, was not changed. In this scenario a user searching for expertise in a certain query topic q comes across a candidate expert ca, who seems to be very relevant to the topic q and has written several documents D_{ca}. Again, to decide which document $d \in D_{ca}$ to read, a ranking of these documents is necessary. The first idea that comes to mind to rank these documents is to utilize the relevance of a document regarding to the query. However, ranking documents based on their relevance to the query is basically the idea of Balog et al. (2012) and the computation is shown in Equation 3.6. Moreover, document relevance scores are already used as jump probabilities, as described in the paragraph above. Another simple idea is to rank the documents based on their number of incoming citations, but this information is mainly represented in the expertise graph as the document-document relations. It would also be possible to rank the documents according to the venue where they were released. Further ideas regarding the weighting of author-document relations are discussed in Chapter 6.

5 Evaluation

5.1 Experimental Setup

5.1.1 ACL Anthology Network

In order to evaluate the expert finding methods explained in Chapter 3 and Chapter 4, a proper dataset is needed. As mentioned in the problem statement, the goal is to evaluate and test these methods on more extended domains such as citation networks, since most expert finding methods already perform well on simple domains like organizations and universities.

The *ACL*[1] *Anthology* is a digital archive of conference and journal papers about natural language processing and computational linguistics. While its primary purpose was to serve as a reference repository of research results, Bird et al. (2008) believed that these data should also be used as a platform for research. Therefore, they developed the *ACL Anthology Reference Corpus*[2] (ARC) to encourage other researchers to use it as a testbed for research experiments. This corpus includes all papers published in ACL and related organization as well as the Computational Linguistics journal.

However, ACL ARC has a major limitation as it is just a collection of papers. Thus, although it may be sufficient for a number of experiments, there is simply not enough information available for more complex experiments. Thus, even though it may still be sufficient for a number of experiments, more advanced experiments could be done if more information was available. This problem was addressed by Radev et al. (2013). Due to the lack of any citation information or any statistics about the productivity of the various researchers, Radev et al. (2013) developed the *ACL Anthology Network*[3] (AAN). AAN provides paper citation, author citation and collaboration networks of the articles included in the ACL ARC as well as rankings of papers and authors based on their centrality statistics in the citation and collaboration networks. Moreover, the citing sentences associated with each citation link are provided and various statistics about individual authors and papers are maintained. "For each author, AAN includes number of papers, collaborators, author and paper citations, and known affiliations as well as h-index, citations over time, and collaboration graph. Moreover, AAN includes paper metadata such as title, venue, session, year, authors, incoming and outgoing citations, citing sentences, keywords, bibtex item and so forth" (Radev et al. 2013). In order to achieve this, more than 30 students from the University of Michigan's CLAIR Group helped cleaning the data, creating the dataset, the networks, the statistics etc. A lot of manual labor went into the task of cleaning the data such as matching references from papers to other papers in the ACL ARC, fixing the issue of wrong or incomplete author names and multiple author identities, removing duplicate papers, correcting incorrect titles, wrong years and/or venues in several papers citation sections and so on. As a result, the AAN in its current state is a clean and curated corps with many additional information about authors and papers.

1. Association of Computational Linguistics
2. `https://acl-arc.comp.nus.edu.sg/`
3. `http://tangra.cs.yale.edu/newaan/`

In its current version the *ACL Anthology Network* includes more than 23,000 papers, more than 18,000 authors as well as 124,857 paper citations and 142,450 author collaborations. An important criterion for us was that the networks have a high density, since most graph-based algorithms work best on dense graphs. Being a clean dataset with well-connected networks as well as many further information about authors and papers, this corpus is suitable for the tasks of expert retrieval.

5.1.2 The Expert Finder Application

We have developed our own expert finding system for the evaluation, which operates on the data of the *ACL Anthology Network* for the reasons mentioned in the Section 5.1.1.

This tool allows the use of various methods to identify experts. Of course, it is only possible to find experts in the field of computational linguistics, as a corpus of scholarly publications about this particular field is used. However, since the methods used for expert finding do not depend directly on the data, it is theoretically possible to find experts in every field of interest, if the system is provided with the corresponding data. These methods include, among others, the two baselines mentioned in Chapter 3, Model2 by Balog et al. (2012) and Multi-step Relevance Propagation (RP) by Serdyukov et al. (2008), as well as our Weighted Relevance Propagation (WRP) that was explained in Chapter 4.

As typical for expert finding systems, after selecting the desired expert finding method, our tool enables the user to enter a query and it returns a ranked list of experts as well as additional information like number of incoming citations or h-index. In Figure 5.1 a mockup screenshot of the front page of our expert finding system is shown. The user is presented a large search box for entering her query as well as the ranked list of experts. Note that the images and the keywords for each expert have manually been edited since the automatic extraction of this is beyond the scope of this work but might be implemented as future work. The keywords shall briefly show the profile of an expert, so that the user can decide even better, which expert she wants to contact.

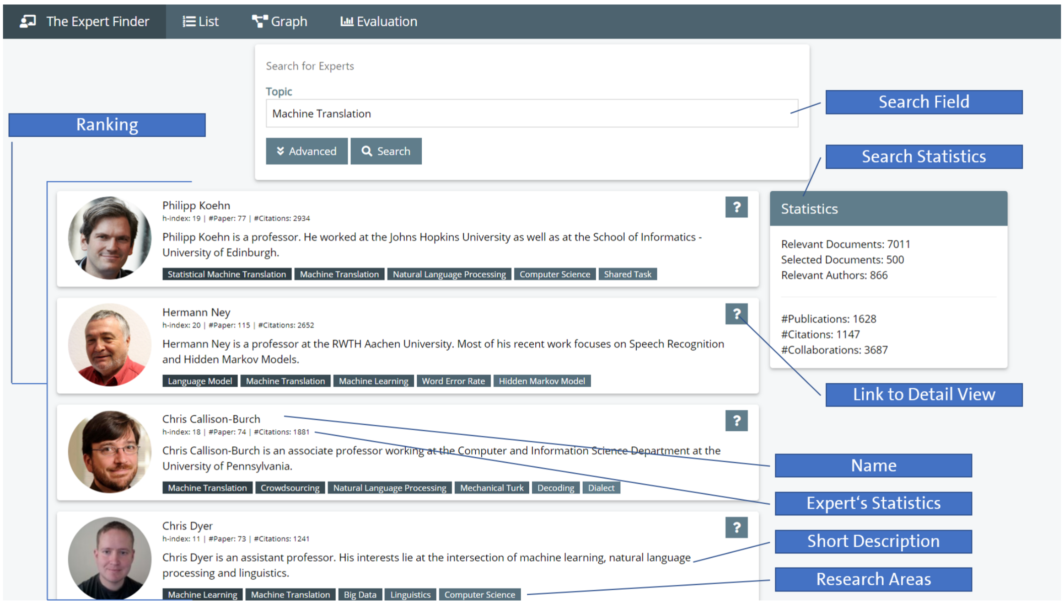

Figure 5.1: The Expert Finder - Front Page.

Moreover, our tool allows the user to take a look at the expertise graph underlying the expert finding method. This is especially interesting for the RP method and our WRP method. In Figure 5.2 the graph view of our expert finding system is shown. This view visualizes the query-based expertise graph and thus allows a look into the data and an even better understanding of the results.

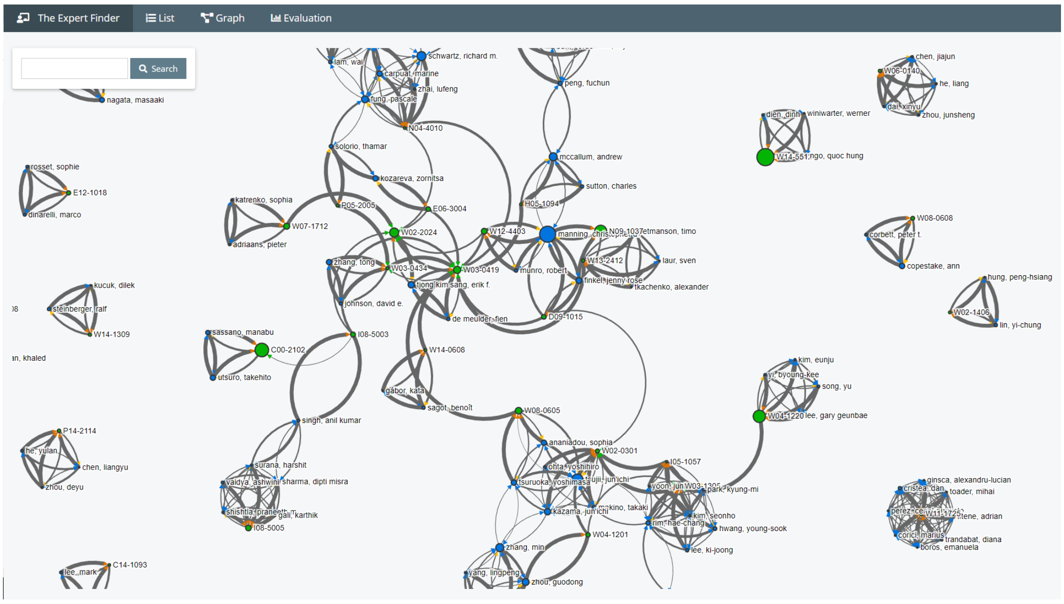

Figure 5.2: The Expert Finder - Graph View. The blue nodes represent authors, the green nodes represent documents. The size of a node is proportional to the document's or author's relevance. The edges represent relations between the entities and have different coloured arrowheads dependent on the relation type. Blue stands for collaboration, green stands for citation, yellow and orange stand for authorship. The size of the edges are proportional to their weight assigned by the Weighted Relevance Propagation method.

The last view offered by our own expert finding system is particularly important for the next section of this work, the discussion of the ranking results of various expert finding methods. This evaluation view applies every implemented expert finding method to identify experts in the given query. Then, it visualizes the resulting ranked lists of every method next to each other in a table. With the help of this view, it is possible to compare the different expert finding methods qualitatively with each other. Thus, it is easy to identify differences in the ranking of experts. An example is shown in Figure 5.3.

The detail view of a certain expert (cf. Figure 5.4) is another helpful feature of our tool to evaluate expert finding methods. Among other things, this view shows the calculated *expert score*, which is used to rank the experts, as well as the documents relevant to the query, in which the expert was involved. These relevant documents are important in that they feed into the calculation of the expert score. These documents are more or less the evidence for the experts' expertise. Depending on the expert finding method, these relevant documents are sufficient for the calculation of the final expert score, as for example in Model2 by Balog et al. (2012). In other methods, as for example in the RP method proposed by Serdyukov et al. (2008), the relevant documents are a part of the calculation whereas other factors play a more important role. Since Model2 uses only the documents as evidence for the expertise, the sum of the document relevance scores of an author represents the final *expert score*.

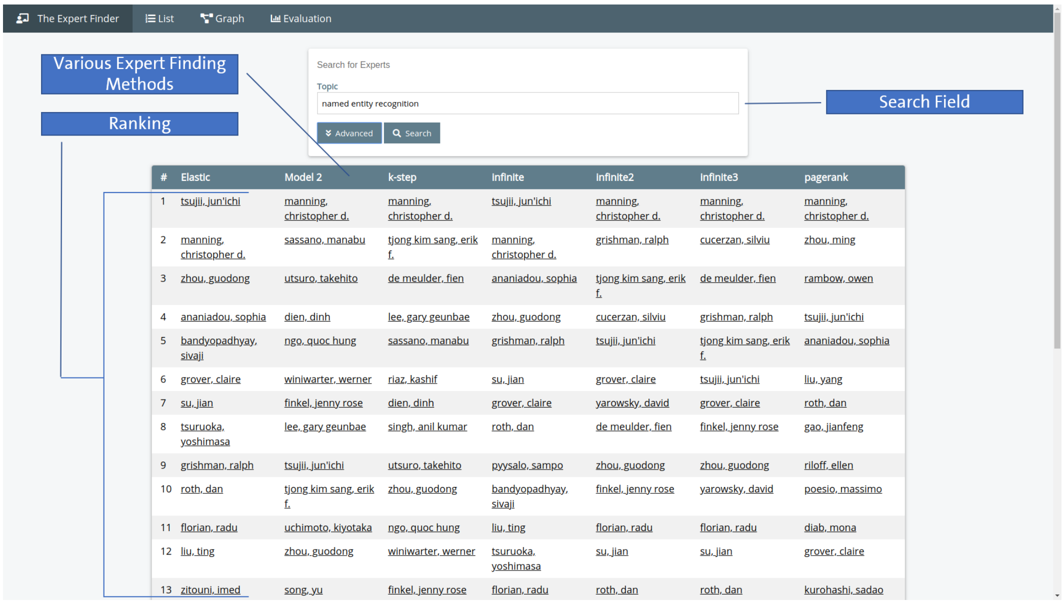

Figure 5.3: The Expert Finder - Evaluation View. Seven different methods rank experts in the topic "Named Entity Recognition".

This expert finding system is the basis for the following evaluation of the different expert finding methods. In the further course of this chapter, the two baseline methods Model2 and Multi-step Relevance Propagation as well as our own approach are compared in detail. Both advantages and disadvantages of the methods are shown as well as detailed explanations for the different ranking results. All insights concerning differences in the ranking of experts among the methods are obtained by means of this expert finding system. To reproduce and to validate our evaluation results, the expert finding tool is available at `http://ltdemos.informatik.uni-hamburg.de/xpertfinder/ui`. The source code is open source and available at `https://github.com/uhh-lt/xpertfinder`.

5.2 Results

5.2.1 Parameters

In order to evaluate the three methods Model2, Multi-step Relevance Propagation (RP) and our own approach, the Weighted Relevance Propagation (WRP), a query topic is necessary. The authors of the AAN (Radev et al. 2013) also performed a simple experiment on expert finding in the areas "Summarization", "Machine Translation" and "Dependency Parsing". They chose these three areas because they are some of the most active areas in Natural Language Processing. For the same reason we decided to evaluate the different expert finding methods on the research area "Dependency Parsing".

As mentioned in the beginning of Chapter 4, both the RP and our WRP have three parameters μ_d, μ_{ca}, λ that have to be adjusted. A very important parameter is λ since it defines the impact of the jump probabilities. The jump probabilities are defined in Equations 3.14 and 3.15 and are the same for RP and WRP. The jump probabilities can be described as prior knowledge about the documents and candidate experts. The prior knowledge of documents is encoded as a document's relevance to the query while the prior knowledge of candidate experts is influenced

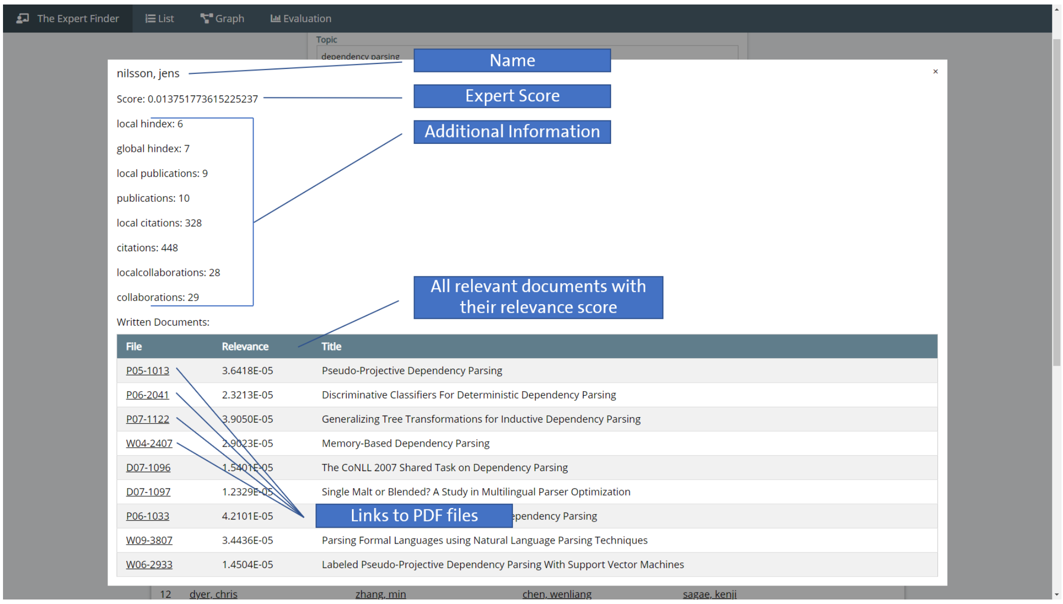

Figure 5.4: The Expert Finder - Detail View. It shows the details of the top expert in "Named Entity Recognition" according to Model2.

by the number of relevant document written by them. Especially the prior knowledge about candidate experts is important information for the ranking as it prevents candidate experts, who have written very few relevant documents, from getting a high ranking. Therefore, this information can not be ignored and setting λ to zero would not be helpful. However, giving λ a high value would not be helpful either, as this would lead to the pior knowledge dominating the ranking process. After some testing, we decided to set λ to 0.1. The other two parameters μ_d, μ_{ca} were set to 0.5. They define which conditional probabilities are more important. In other words, these two parameters define whether a user searching for expertise would more likely follow document-document links, document-author links or author-author links. Since we assume for now that these links are all equally important in a citation network, we decided set the parameters to 0.5. Moreover, our WRP approach dynamically assigns weights to the relations between entities. Setting the parameters to 0.5 was the best choice to not further influence the automatic weighting of relationships between entities.

5.2.2 Ranking Comparison

Table 5.1 shows the top 10 experts in "Dependency Parsing" that were identified using the three methods Model2, RP and WRP. The tables A.1 and A.2 in the appendix show the top 60 experts. The experts listed in the second column named GlobalCitations are ranked with a simple heuristic, which is the same as the one Radev et al. (2013) used in their small experiment to find top authors in several research areas. Simply put, all authors who have written about the query topic, which is in this case "Dependency Parsing", are ranked by their total incoming citation count. As Radev et al. (2013) did their experiments in 2009 whereas we use the current AAN release, our results using this heuristic differ from their results. Moreover, they filter the documents by matching the paper's title with the topic, while we find the documents by matching the paper's content with the topic. Table 5.2 shows these differences.

Rank	GlobalCitations	Model2	RP	WRP
1	**manning, c. d. (5)**	**nivre, joakim (47)**	**nivre, joakim (47)**	**nivre, joakim (47)**
2	klein, dan (1)	mcdonald, ryan (19)	mcdonald, ryan (19)	mcdonald, ryan (19)
3	collins, michael john (9)	**chen, wenliang (16)**	nilsson, jens (9)	nilsson, jens (9)
4	marcus, mitchell p. (1)	sagae, kenji (10)	**pereira, fernando (4)**	**hall, johan (10)**
5	pereira, fernando (4)	zhang, yue (17)	**hall, johan (10)**	**pereira, fernando (4)**
6	jurafsky, daniel (3)	hall, johan (10)	zhang, yue (17)	zhang, yue (17)
7	johnson, mark (4)	tsujii, jun'ichi (7)	oflazer, kemal (4)	carreras, xavier (8)
8	mcdonald, ryan (19)	nilsson, jens (9)	goldberg, yoav (10)	goldberg, yoav (10)
9	hovy, eduard (1)	liu, ting (10)	buchholz, sabine (1)	buchholz, sabine (1)
10	**nivre, joakim (47)**	miyao, yusuke (10)	carreras, xavier (8)	sagae, kenji (10)
...				
14	smith, noah a. (12)	goldberg, yoav (10)	riedel, sebastian (5)	**chen, wenliang (16)**
...				
19	lin, dekang (2)	bohnet, bernd (9)	**chen, wenliang (16)**	hall, keith (6)
...				
46	habash, nizar (5)	kurohashi, sadao (5)	tjong kim sang (3)	**manning, c. d. (5)**
47	riedel, sebastian (5)	farkas, richárd (6)	**manning, c. d. (5)**	huang, liang (5)

Table 5.1: Top 10 experts in "Dependency Parsing" using different expert finding methods. Important differences between the methods are highlighted. The number of relevant documents is indicated in parentheses.
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Taking another look at Table 5.1 reveals that all three expert finding methods agree on Joakim Nivre and Ryan McDonald being the first and second ranked expert in "Dependency Parsing". This differs alot from the simple heuristic, where they were ranked 10th and 8th respectively. Instead, Christopher D. Manning and Dan Klein are the top experts. Dan Klein does not even appear in the Top 60 ranking of the three expert finding methods, which is simply because he has written just one paper with not many citations that mentions dependency parsing and therefore is no good candidate expert. Christopher D. Manning, however, is ranked at least 46th and 47th by the relevance propagation methods (both the RP method as well as our WRP method are meant by this term). This can be explained as he is a well-connected scholar with a high h-index as well as with 5 publications about "Dependency Parsing". But since he is active in various research areas, he should not be considered as a top expert. Model2 ranks him even lower, which is due to the fact that he has a low publication count in the query topic and this method tends, more than the other methods, to rank people higher the more relevant documents they have written. Joakim Nivre being in the first place is undeniable. He is involved in 48 papers about "Dependency Parsing", while Ryan McDonald has written 19 papers about this topic.

Investigating the rankings of the three expert finding methods, it can be said that RP and WRP are similar in their ranking, while Model2 differs from these relevance propagation methods. A good example is Wenliang Chen, who is ranked 3rd by Model2, 19th by RP and 14th by WRP. Since this scholar was involved in 16 papers about "Dependency Parsing" Model2 ranks Wenliang Chen very high. However, at least in the AAN corpus, Wenliang Chen is not well-connected as he has only 48 incoming citations. For that reason, even though he has dedicated most of his work to "Dependency Parsing", he is ranked so low by the relevance propagation methods. This dedication is captured by our WRP method though, rating him 5 ranks higher than the RP method. Jens Nilsson, who is ranked 3rd by the relevance propagation methods, is another good example to show the differences in the rankings of the three expert finding methods. In the AAN corpus, he has published 9 papers regarding "Dependency Parsing", what accounts for rank 8

Rank	GlobalCitations	GlobalCitations(AAN)
1	manning, christopher d.	mcdonald, ryan
2	klein, dan	nivre, joakim
3	collins, michael john	pereira, fernando
4	marcus, mitchell p.	nilsson, jens
5	pereira, fernando	hall, johan
6	jurafsky, daniel	eisner, jason m.
7	johnson, mark	crammer, koby
8	mcdonald, ryan	riedel, sebastian
9	hovy, eduard	ribarov, kiril
10	nivre, joakim	hajič, jan

Table 5.2: Comparison of the GlobalCitations heuristics. Different filtering techniques and datasets account for the differences between our results (left) and Radev et al. (2013) results (right).

determined by Model2. But, his 9 papers were cited 341 times in the context of "Dependency Parsing". This contributes to him being a great candidate expert and explains his high ranking by the relevance propagation methods. By taking more information into account, especially information about the connectedness of a scholar, the relevance propagation methods clearly outperform Model2. A similar example is Fernando Pereira, who is ranked 4th and 5th by the relevance propagation methods and 41th by Model2. Again, his low ranking by Model2 is due to his low publication count of 4 documents regarding "Dependency Parsing". However, as this 4 publications were cited 411 times in the context of "Dependency Parsing" he is a very important candidate expert deserving the high rank computed by the relevance propagation methods.

Even though the resulting expert rankings of RP and WRP are very similar, it is worth examining these methods in detail. As stated above, WRP is able to capture the dedication of scholars and to rank them accordingly. One example for this was Wenliang Chen. This dedication is mainly captured by the comparison of the local h-index and the global h-index as described in Section 4.1. Sometimes this can tip the balance in the ranking as for example in the case of Fernando Pereira and Johan Hall. They are ranked 4th and 5th by the relevance propagation methods. Our WRP method decided that Johan Hall has to be the better candidate. Even though Fernando Pereira is cited 411 times in the context of "Dependency Parsing", whereas Johan Hall has only 283 incoming citations, WRP decided in favor of Johan Hall. It is interesting to understand what led to this decision. Fernando Pereira is a scholar active in many research areas. In total he has 45 publications listed in the AAN corpus of which only 4 were dedicated to "Dependency Parsing". Moreover, the fact that he is active in various research areas is reflected by his h-index as he has a local h-index of 4 and a global h-index of 17. Being a relevant scholar, putting him in the 5th place is fine and reasonable. In contrast to him, Johan Hall has 10 out of 10 publications listed in the AAN corpus that are related to "Dependency Parsing" as well as a local h-index of 6 and a global h-index of 7. Placing him above Fernando Pereira perfectly reflects our subjective opinion that an expert is a person, who has focused on one topic rather than being involved in many topics.

Generally speaking, the GlobalCitations heuristic gives a good first impression of several important scholars in the queried topic. However, this method is not reliable. Dan Klein is ranked the second best expert in "Dependency Parsing", even though he is by no means an expert

Rank	GlobalCitations	Model2	RP	WRP
1	och, franz josef	koehn, philipp	koehn, philipp	koehn, philipp
2	manning, christopher d.	ney, hermann	ney, hermann	ney, hermann
3	koehn, philipp	callison-burch, chris	callison-burch, chris	callison-burch, chris
4	ney, hermann	sumita, eiichiro	hoang, hieu	hoang, hieu
5	collins, michael john	wang, haifeng	dyer, chris	dyer, chris
6	marcu, daniel	wu, hua	zens, richard	zens, richard
7	knight, kevin	huck, matthias	och, franz josef	sumita, eiichiro
8	callison-burch, chris	waibel, alex	sumita, eiichiro	monz, christof
9	roukos, salim	freitag, markus	monz, christof	och, franz josef
10	jurafsky, daniel	peitz, stephan	waibel, alex	waibel, alex

Table 5.3: Top 10 experts in "Machine Translation" using different expert finding methods. Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

in this research area. In contrast to that, Model2 is able to rank the candidate experts much better. The candidate experts are ranked dependent on the document's relevance to the queried topic, which is measured by a simple language model. This method yields already good results, however it is strongly biased towards authors that have written or are involved in many documents about the queried topic. Table 5.1 shows the number of relevant documents of each author in parentheses to reinforce this statement. This is also partly true for the relevance propagation methods. This bias is fine though, because in most cases, scholars who have written many papers about the topic in question can be considered as experts. But, the relevance propagation methods are able to differentiate the experts better by considering their connectedness and the connectedness of their documents by exploiting links between authors and documents. This means, that the relevance of a document or an author is further influenced by factors like number of incoming citations or collaborations with other important authors. Thus, for example scholars with few highly cited papers are also considered as experts. It can be said that the RP method already covers the most important experts, while our proposed WRP method adds some fine-tuning on top of that. By taking into account more information like h-index, publication date of documents and number of collaborations, the method can adjust the ranking of the candidate experts even better and more precisely. However, often this additional information has no major impact on the top 10 experts. In Table 5.3 the top 10 experts in "Machine Translation" are shown. Very similar reasons as described in the paragraphs above are responsible for the order of the experts in this ranking.

5.2.3 Feature Analysis

In this section, our proposed Weighted Relevance Propagation (WRP) method will be analyzed in more detail. For this purpose, the influence of each modified conditional probability on the expert ranking is evaluated individually. As described in Chapter 4, WRP modifies three conditional probabilities: links between authors are weighted utilizing information about the number of collaborations, links between documents are weighted using topicality and links between documents and authors are weighted by taking the h-index into account. Normally, all three conditional probabilities are used to calculate the final expert ranking. However, in order to obtain more detailed insights into our proposed method, it makes sense to analyze the impact of each feature one by one. Table 5.4 shows the top experts in "Dependency Parsing" retrieved with Multi-step Relevance Propagation (RP) without any modifications as well as with Weighted Relevance Propagation (WRP) using either collaboration count (WRP-Collaboration), recency (WRP-Recency) or h-index (WRP-Hindex) as the only modification to the standard RP method by Serdyukov et al. (2008).

Rank	RP	WRP-Collaboration	WRP-Recency	WRP-Hindex
1	nivre, joakim	nivre, joakim	nivre, joakim	nivre, joakim
2	mcdonald, ryan	mcdonald, ryan	mcdonald, ryan	mcdonald, ryan
3	nilsson, jens	nilsson, jens	nilsson, jens	nilsson, jens
4	**pereira, fernando**	pereira, fernando	pereira, fernando	**hall, johan**
5	**hall, johan**	hall, johan	hall, johan	**pereira, fernando**
6	zhang, yue	zhang, yue	zhang, yue	zhang, yue
7	**oflazer, kemal**	buchholz, sabine	goldberg, yoav	buchholz, sabine
8	goldberg, yoav	oflazer, kemal	carreras, xavier	goldberg, yoav
9	buchholz, sabine	goldberg, yoav	buchholz, sabine	carreras, xavier
10	carreras, xavier	marsi, erwin	smith, noah a.	oflazer, kemal
11	smith, noah a.	carreras, xavier	**collins, michael john**	smith, noah a.
12	marsi, erwin	smith, noah a.	bohnet, bernd	sagae, kenji
13	sagae, kenji	sagae, kenji	sagae, kenji	**eryiğit, gülşen**
14	riedel, sebastian	collins, michael john	marsi, erwin	marsi, erwin
15	bohnet, bernd	bohnet, bernd	riedel, sebastian	chen, wenliang
16	**collins, michael john**	riedel, sebastian	kübler, sandra	bohnet, bernd
17	**eryiğit, gülşen**	eryiğit, gülşen	chen, wenliang	riedel, sebastian
18	kübler, sandra	chen, wenliang	hall, keith	kübler, sandra
19	chen, wenliang	kübler, sandra	eryiğit, gülşen	scholz, mario
20	kuhlmann, marco	hall, keith	kuhlmann, marco	collins, michael john
21	hall, keith	kuhlmann, marco	satta, giorgio	hall, keith
22	scholz, mario	scholz, mario	**oflazer, kemal**	kuhlmann, marco

Table 5.4: Analysis of the Weighted Relevance Propagation (WRP) method's features. Important differences in the ranking between a feature and the Relevance Propagation (RP) method are highlighted.
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Taking a first look at the Table 5.4 reveals that the impact of each modification on its own is very small. Especially in the upper ranks, the three features make no difference. The first three candidate experts are the same for all methods. The main reason for this is the fact that the top experts often have so much evidence for their expertise, for example in the number of incoming citations or in the number of written documents, which outweighs information about h-indexes, number of collaborations or topicality. Nevertheless, this information can make big differences in the lower rankings.

Particularly the recendcy feature has great impact on the ranking of quite a few scholars. Remember that the intention behind the implementation of recency is to boost the ranking of scholars that have worked with the queried topic recently and to drop scholars that have not worked with the topic for a long time. Kemal Oflazer is ranked 7th by RP while he is is ranked 22th by WRP-Recency. This is very reasonable since Kemal Oflazer has written four papers about "Dependency Parsing" of which the oldest document was released in 1999 while the most recent document was published in 2008. At the time of writing this work the most recent paper is from ten years ago. For that reason the WRP-Recency dropped Kemal Oflazer 15 ranks. In contrast to this case, Micheal John Collins is ranked 16th by RP, whearas he is ranked 11th by WRP-Recency. He was involved in 9 publications about "Dependency Parsing" that were published in the years from 2007 to 2012. Therefore, he is definitely a candidate expert that has done more recent work regarding the query topic "Dependency Parsing" and deserves a higher rank.

Comparing the local h-index with the global h-index to capture the dedication of a scholar also turns out to work very well. It is the only modification to the original RP method that had such a big impact that the 4th and 5th place were swapped in the expert ranking. Johan Hall and Fernando Pereira, ranked 4th and 5th by RP-Hindex respectively, were already discussed before. Johan Hall is, simply put, a very focused scholar, who has dedicated all of his work towards "Dependency Parsing". On the contrary, Fernando Pereira is a diversified scholar, who has worked on different topics like Machine Learning, Information Retrieval, Artificial Intelligence and Natural Langugage Understanding. This is also reflected by their h-indexes. Fernando Pereira has a global h-index of 17 and a local h-index of 4, while Johan Hall has a global h-index of 7 and a local h-index of 6. Another scholar, who also benefits from this method, is Gülşen Eryiğit. She is ranked 17th by RP and 13th by WRP-Hindex, because, even though she worked on various topics, her most cited papers deal with "Dependency Parsing". As these documents account for her global and local h-index of 4, her rank increased by 4 using WRP-Hindex.

Apparently, the collaboration count feature has no big impact on the expert ranking. The intention of this feature was to measure the strength of collaborations between two authors. The idea was that an author would rather redirect a user searching for expertise to an author with whom she has worked many times than anybody else. However, as it can be seen in Table 5.4, WRP-Collaboration does not make any big differences in the expert ranking. There are very little changes in the ranking, mostly two scholars swap places. Even though the idea behind this feature sounds reasonable it does not prove itself in practice. One reason for this could be the implementation of this modification. By taking a look at the Formula 4.10 again as well as the exemplary calculations (4.11) and the results after applying the softmax function (4.12), it is noticeable that the resulting weights for the edges are nearly equal. Therefore, the weights are not that helpful to prioritize candidate experts.

To sum up this section, recency as well as h-index prove to be good features to further differentiate and rank candidate experts, whereas collaboration count is not that useful. Recency is a great feature to ensure that only scholars who recently dealt with the queried topic are ranked high, while h-index is a good feature to capture dedication. The h-index feature allows scholars to rank high, even though they do not have many publications or a large number of citations but instead focus on a specific topic. The collaboration count feature could not prove to be that useful.

5.2.4 Concluding Comparison

In this section, the three expert finding methods are compared in a more general way. First, the ranked lists of candidate experts determined by the three methods were discussed extensively. Then, the main components of our proposed Weighted Relevance Propagation method were analyzed in detail. Lastly, a comparison of all discussed expert finding methods will conclude this chapter.

All three methods Model2, Multi-step Relevance Propagation (RP) and our Weighted Relevance Propagation (WRP) have one fact in common. They all use documents, which are associated with candidate experts, as evidence of the candidates' expertise. Therefore, all three methods need to determine the relevance of a document regarding a user's query. Model2 was the first approach to introduce a simple language model to measure this relevance. In a simple implementation, the relevance of a document could be calculated using solely the frequency of the query terms in the document. Model2 uses exclusively this document relevance to determine the relevance of a candidate expert by adding up the relevance of all documents written by this candidate expert. In contrast to that, RP and WRP use this information only as a part of the calculations, namely, as document priors. Nevertheless, all three methods utilize the relevance of the documents. Therefore, all methods, even though Model2 is more effected, are biased towards candidate experts that are involved in many documents. Moreover, RP and WRP also use the simple Formula 3.14 for the calculation of candidate priors. This formula basically implies: the more documents that are relevant to the query a candidate has written, the better. But, this bias is fine in most cases, as an expert can be reasonably described as someone, who has dealt with a certain topic very often.

Model2 uses solely the document relevance to identify good candidate experts. Thus, the language model is the only information that influences the expert ranking. It is completely irrelevant whether, for example in an extended domain like a citation network, the document was cited often. In a simple implementation, it basically comes down to the frequency of the query terms mentioned in the document. Model2 does not rely on any further information, but the documents written by the candidate experts. Therefore, Model2 is a very generic expert finding strategy that can be used in any domain. It yields good results and is easy to implement, especially on top of an existing document index. However, this method heavily relies on the fact, that an expert has to write and to be involved in many documents about a certain topic to be deemed an expert for this topic.

This is not necessarily true for the relevance propagation methods RP and WRP. These two methods both focus on the relations between documents and authors. Hence, even though a candidate expert was not involved in many documents about the topic in question, if the candidate and/or her documents are well-connected, she is considered as an expert. In general, one can say about RP: the more connected a candidate and/or her documents, the better. And this is even better if the documents and candidates the candidate is connected to are also well-connected. Utilizing additional information about links between documents, authors and maybe other entities (dependant on the domain, other entities might be helpful for the ranking of experts) RP can yield better results than Model2. In addition, RP is suitable and applicable for many extended domains like social networks or citation networks. However, needing information about relations between entities, RP is not as general and generic as Model2.

The same holds true for our proposed WRP method. It is again beneficial to a candidate expert to be well-connected and it is even better again if the entities connected to the candidate experts are also well-connected. However, the importance of an expert is not only measured by connectedness, document priors like document relevance and candidate priors like number of written relevant documents, but the importance is also influenced by other factors. These other factors and their influence on the ranking of experts were already discussed in Section 5.2.3. To quickly sum up, our developed WRP method shows that RP can be further refined by utilizing more information. Of course, the information that can be used depends on the availability and on the domain. Obviously, the need for more information about candidate experts, documents and possibly other entities restricts WRP all the more to extended domains where much information is available. That being said, the features used by us, recency, h-index and collaboration count, are just a few suggestions to enhance the RP method. We want to conclude this chapter by saying that it is definitely worth trying to include more information in the expertise retrieval process as the expert ranking will definitely benefit.

6 Conclusion and Future Work

Expertise is a remarkable resource. Even though expertise is difficult to measure as experts vary in experience, and their expertise is continually changing, it is a significant challenge to keep track of people's expertise. Identifying experts on specific topics can be of great value to many applications. Especially for an organization, the effective utilization of expertise is key. Managing the expertise of employees, so that experts in specific knowledge areas can be found may reduce costs and facilitate better solutions.

Organizations can make great benefits of expert identification. As a consequence, most work regarding expert finding has focused on simple domains like organizations and their intranets. Even though many expert retrieval methods have been developed, little work has been done on extended, complex domains such as bibliographic networks, hierarchical organizations and social networks.

Motivated by these facts, this work concerned itself with the task of expert finding on citation networks. One of the goals was to examine the effectiveness of standard expert retrieval methods on extended domains. As extended domains provide besides the content of the documents other important information like citations and temporal behavior, it was also an important goal to identify effective features that can help with building a high-quality expert search engine. Lastly, we aimed at utilizing these identified features to develop our own expert retrieval method.

The evaluation revealed that already simple expert retrieval methods like Model2 by Balog et al. (2012) can yield good results. The more advanced Multi-step Relevance Propagation approach by Serdyukov et al. (2008) was easily able to outperform Model2. However, this was expected, since we decided for the Multi-step Relevance Propagation method with the intention of having a standard expert retrieval method, that fits the citation network well. All in all, the evaluation shows that the standard expert finding methods are robust to extended domains like citation networks and that they appear to be generalizable to other domains.

Furthermore, three different features, that can enhance expert retrieval results, were identified and discussed in this work. These three features were used to develop the Weighted Relevance Propagation method. Compared to the two standard baseline approaches, this method is able to improve the results of expert finding by adjusting the impact of h-index, recency and collaboration count. However, further investigation unveiled that only two out of three features, namely recency and h-index, have a noticeable impact on the results while collaboration count does not account for better performance.

A useful next step would be further feature engineering. Dependent on the domain, plenty of different information is available besides the content of the documents. The AAN for example provides the venue for each document, which could be easily incorporated in the documents prior probability, since certain venues are more relevant than others. Moreover, the AAN associates each scholar with her workplace or university. This information could be also integrated in the calculation of the prior knowledge of a candidate expert. However, both suggestions require manual labor in order to rank and weight the different venues and universities. In addition to that, crawling the web for more information might be useful. In our case, the real h-index of a scholar might be helpful.

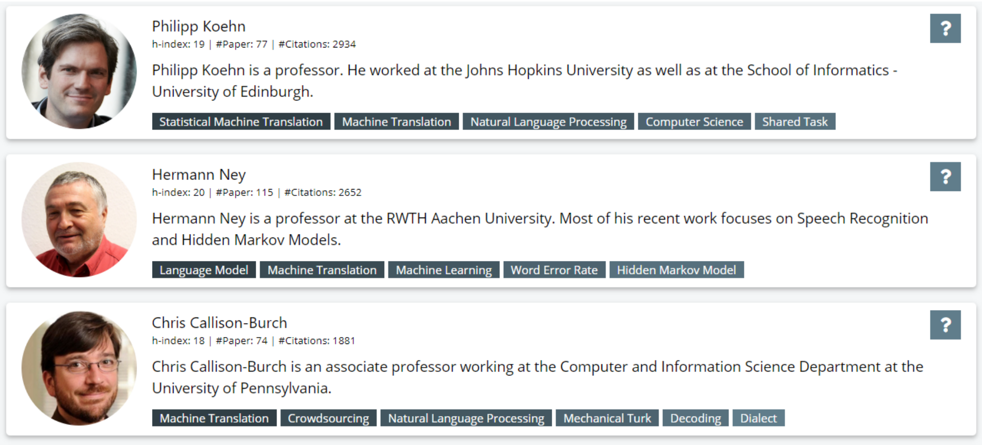

Figure 6.1: Keyword / tag lists to represent expert profiles

Simple expert finding systems, like the tool briefly discussed in Section 5.1.2 that was developed to evaluate different expert retrieval methods, solve the task of expert finding simply by prompting the user to input her query topic and then presenting the user a list of the top ranked experts. This is already very helpful. Moreover, by displaying additional statistics about the experts, like h-index and total incoming citations, expert finding tools can help the user to differentiate experts from each other even better. Furthermore, linking the ranked experts to corresponding GoogleScholar profiles or Wikipedia pages would easily improve the results and utility of expert finding systems. However, the next step to further guide and support a user in her search for expertise would be to create profiles for each expert automatically. An expert profile basically turns the task of expert finding around: given a certain expert, in which fields of knowledge has the expert the most expertise? A simple expert profile could be a list of keywords or tags describing an expert's knowledge. This profile could be integrated and displayed in the result list of an expert finding system easily. Figure 6.1 shows an exemplary result list with simple expert profiles as keyword lists. Implementing these simple expert profiles could already enhance the user experience a lot. Especially making the tags/keywords clickable and in this way enabling the user to quickly explore related knowledge areas would greatly enhance the search for experts. More advanced expert profiles could associate each knowledge area of a candidate expert with a score determining the expert's relevance in this certain knowledge area. In order to present the user even more information about an expert, this score could be calculated per year and displayed as a graph in the expert profile. In this way the activity of an expert could be tracked.

As described in Section 3.2, the Multi-step Relevance Propagation method calculates scores for candidate experts as well as documents. For the task of expert finding, the document score is just an intermediate result, whereas the expert score is used to create the final ranked expert list. However, the scores of the documents could be utilized to create sophisticated applications. By including features like recency, which was described in Section 4.2, the top documents ranked by the document score should be both the most relevant ones as well as the most recent ones given a certain query. Using these top documents as input to an advanced text summarization tool, a first draft of a review paper about the given query could be created automatically. In this way, an automatic review paper creation tool could be build. Alternatively, a system that automatically summarizes top documents for a given topic could be developed, that helps users who want to gain first insights in the current state of knowledge.

Bibliography

Bailey, Peter, Arjen P De Vries, Nick Craswell, and Ian Soboroff (2007). *Overview of the TREC 2007 Enterprise Track.* In: *Proceedings of the Sixteenth Text REtrieval Conference*. Gaithersburg, MD, USA.

Balog, Krisztian, Leif Azzopardi, and Maarten de Rijke (2006). *Formal Models for Expert Finding in Enterprise Corpora*. In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, WA, USA, 43–50.

– (2009). *A language modeling framework for expert finding*. In: *Information Processing & Management* 45.1, 1–19.

Balog, Krisztian, Toine Bogers, Leif Azzopardi, Maarten de Rijke, and Antal van den Bosch (2007a). *Broad Expertise Retrieval in Sparse Data Environments*. In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Amsterdam, The Netherlands, 551–558.

Balog, Krisztian and Maarten De Rijke (2007b). *Determining Expert Profiles (with an Application to Expert Finding)*. In: *Proceedings of the 20th International Joint Conference on Artifical Intelligence*. Hyderabad, India, 2657–2662.

Balog, Krisztian, Yi Fang, Maarten de Rijke, Pavel Serdyukov, and Luo Si (2012). *Expertise Retrieval*. In: *Foundations and Trends in Information Retrieval* 6.2-3, 127–256.

Balog, Krisztian and Maarten de Rijke (2007c). *Finding Similar Experts*. In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Amsterdam, The Netherlands, 821–822.

Balog, Krisztian, Paul Thomas, Nick Craswell, Ian Soboroff, Peter Bailey, and Arjen P De Vries (2008). *Overview of the TREC 2008 enterprise track*. In: *Proceedings of the Eighteenth Text REtrieval Conference*. Gaithersburg, MD, USA.

Becerra-Fernandez, Irma (2000). *Facilitating the Online Search of Experts at NASA using Expert Seeker People-Finder.* In: *Proceedings of the Third International Conference on Practical Aspects of Knowledge Management*. Basel, Switzerland, 23–29.

Bird, Steven, Robert Dale, Bonnie J. Dorr, Bryan Gibson, Mark Thomas Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir R. Radev, and Yee Fan Tan (2008). *The acl anthology reference corpus: A reference dataset for bibliographic research in computational linguistics*. In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation*. Marrakech, Morocco, 1755–1759.

Campbell, Christopher S., Paul P. Maglio, Alex Cozzi, and Byron Dom (2003). *Expertise Identification Using Email Communications*. In: *Proceedings of the Twelfth ACM International Conference on Information and Knowledge Management*. New Orleans, LA, USA, 528–531.

Chakraborty, J., G. Thopugunta, and S. Bansal (2018). *Data Extraction and Integration for Scholar Recommendation System*. In: *Proceedings of the 12th International Conference on Semantic Computing*. Laguna Hills, CA, USA, 397–402.

Craswell, Nick, Arjen P De Vries, and Ian Soboroff (2005). *Overview of the TREC 2005 Enterprise Track.* In: *Proceedings of the Fourteenth Text REtrieval Conference*. Gaithersburg, MD, USA, 199–205.

Craswell, Nick, David Hawking, Anne-Marie Vercoustre, and Peter Wilkins (2001). *P@noptic expert: Searching for experts not just for documents*. In: *Ausweb Poster Proceedings*. Queensland, Australia, 21–25.

Davenport, Thomas (1997a). *Knowledge management at Microsoft*. Tech. rep. Microsoft, Redmond, WA, USA.

– (1997b). *Ten principles of knowledge management and four case studies*. In: *Knowledge and Process Management* 4.3, 187–208.

Demartini, Gianluca, Julien Gaugaz, and Wolfgang Nejdl (2009). *A Vector Space Model for Ranking Entities and Its Application to Expert Search*. In: *Proceedings of the 31th European Conference on Information Retrieval Research*. Toulouse, France, 189–201.

Deng, H., I. King, and M. R. Lyu (2008). *Formal Models for Expert Finding on DBLP Bibliography Data*. In: *Proceedings of the Eighth International Conference on Data Mining*. Pisa, Italy, 163–172.

Fang, Hui and ChengXiang Zhai (2007). *Probabilistic models for expert finding*. In: *of the 29th European Conference on Information Retrieval Research*. Rome, Italy, 418–430.

Hashemi, Seyyed Hadi, Mahmood Neshati, and Hamid Beigy (2013). *Expertise Retrieval in Bibliographic Network: A Topic Dominance Learning Approach*. In: *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. San Francisco, CA, USA, 1117–1126.

Hawking, David (2004). *Challenges in Enterprise Search*. In: *Proceedings of the 15th Australasian Database Conference*. Dunedin, New Zealand, 15–24.

Himmel, Wolfgang, Ulrich Reincke, and Hans Wilhelm Michelmann (2009). *Text Mining and Natural Language Processing Approaches for Automatic Categorization of Lay Requests to Web-Based Expert Forums*. In: *Journal of Medical Internet Research* 11.3.

Hotho, Andreas, Andreas Nürnberger, and Gerhard Paaß (2005). *A brief survey of text mining.* In: *Journal for computational linguistics and language technology* 20.1, 19–62.

Jurczyk, Pawel and Eugene Agichtein (2007). *Discovering Authorities in Question Answer Communities by Using Link Analysis*. In: *Proceedings of the Sixteenth ACM International Conference on Information and Knowledge Management*. Lisbon, Portugal, 919–922.

Karimzadehgan, Maryam, Ryen W. White, and Matthew Richardson (2009). *Enhancing Expert Finding Using Organizational Hierarchies*. In: *Proceedings of the 31th European Conference on Information Retrieval Research*. Toulouse, France, 177–188.

Lappas, Theodoros, Kun Liu, and Evimaria Terzi (2009). *Finding a Team of Experts in Social Networks*. In: *Proceedings of the 15th International Conference on Knowledge Discovery and Data Mining*. Paris, France, 467–476.

Macdonald, Craig and Iadh Ounis (2006). *Voting for Candidates: Adapting Data Fusion Techniques for an Expert Search Task*. In: *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*. Arlington, VA, USA, 387–396.

Mockus, Audris and James D. Herbsleb (2002). *Expertise Browser: A Quantitative Approach to Identifying Expertise*. In: *Proceedings of the 24th International Conference on Software Engineering*. Orlando, FL, USA, 503–512.

Neshati, Mahmood, Hamid Beigy, and Djoerd Hiemstra (2012). *Multi-aspect Group Formation Using Facility Location Analysis*. In: *Proceedings of the Seventeenth Australasian Document Computing Symposium*. Dunedin, New Zealand, 62–71.

Petkova, D. and W. Bruce Croft (2006). *Hierarchical Language Models for Expert Finding in Enterprise Corpora*. In: *Proceedings of the 18th International Conference on Tools with Artificial Intelligence*. Arlington, VA, USA, 599–608.

Radev, Dragomir R., Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara (2013). *The ACL anthology network corpus*. In: *Language Resources and Evaluation* 47.4, 919–944.

Serdyukov, Pavel, Henning Rode, and Djoerd Hiemstra (2008). *Modeling Multi-step Relevance Propagation for Expert Finding*. In: *Proceedings of the 17th ACM International Conference on Information and Knowledge Management*. Napa Valley, CA, USA, 1133–1142.

Soboroff, Ian, Arjen P. De Vries, and Nick Craswell (2006). *Overview of the TREC 2006 Enterprise Track.* In: *Proceedings of the Fifteenth Text REtrieval Conference*. Gaithersburg, MD, USA.

Tang, Jie, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su (2008). *ArnetMiner: Extraction and Mining of Academic Social Networks*. In: *Proceedings of the 14th International Conference on Knowledge Discovery and Data Mining*. Las Vegas, NV, USA, 990–998.

Van Gysel, Christophe, Maarten de Rijke, and Marcel Worring (2016). *Unsupervised, Efficient and Semantic Expertise Retrieval*. In: *Proceedings of the 25th International Conference on World Wide Web*. Montreal, Quebec, Canada, 1069–1079.

Wang, B., X. Chen, H. Mamitsuka, and S. Zhu (2015). *BMExpert: Mining MEDLINE for Finding Experts in Biomedical Domains Based on Language Model*. In: *Transactions on Computational Biology and Bioinformatics* 12.6, 1286–1294.

Zhang, Jun, Mark S. Ackerman, and Lada Adamic (2007). *Expertise Networks in Online Communities: Structure and Algorithms*. In: *Proceedings of the 16th International Conference on World Wide Web*. Banff, Alberta, Canada, 221–230.

Zhu, Jianhan, Dawei Song, Stefan Rüger, Marc Eisenstadt, and Enrico Motta (2006). *The Open University at TREC 2006 Enterprise Track ExpertSearch Task*. In: *Proceedings of the Fifteenth Text REtrieval Conference*. Gaithersburg, MD, USA.

Appendix

Rank	GlobalCitations	Model2	RP	WRP
1	manning, c. d.	nivre, joakim	nivre, joakim	nivre, joakim
2	klein, dan	mcdonald, ryan	mcdonald, ryan	mcdonald, ryan
3	collins, michael john	chen, wenliang	nilsson, jens	nilsson, jens
4	marcus, mitchell p.	sagae, kenji	pereira, fernando	hall, johan
5	pereira, fernando	zhang, yue	hall, johan	pereira, fernando
6	jurafsky, daniel	hall, johan	zhang, yue	zhang, yue
7	johnson, mark	tsujii, jun'ichi	oflazer, kemal	carreras, xavier
8	mcdonald, ryan	nilsson, jens	goldberg, yoav	goldberg, yoav
9	hovy, eduard	liu, ting	buchholz, sabine	buchholz, sabine
10	nivre, joakim	miyao, yusuke	carreras, xavier	sagae, kenji
11	dyer, chris	che, wanxiang	smith, noah a.	smith, noah a.
12	palmer, martha	zhang, min	marsi, erwin	marsi, erwin
13	yarowsky, david	smith, noah a.	sagae, kenji	bohnet, bernd
14	smith, noah a.	goldberg, yoav	riedel, sebastian	chen, wenliang
15	joshi, aravind k.	gómez-rodríguez	bohnet, bernd	eryiğit, gülşen
16	tsujii, jun'ichi	candito, marie	collins, michael john	collins, michael john
17	barzilay, regina	weir, david	eryiğit, gülşen	riedel, sebastian
18	roth, dan	carroll, john	kübler, sandra	kübler, sandra
19	lin, dekang	bohnet, bernd	chen, wenliang	hall, keith
20	eisner, jason m.	kuhlmann, marco	kuhlmann, marco	kuhlmann, marco
21	clark, stephen	choi, jinho d.	hall, keith	satta, giorgio
22	huang, liang	matsubara, shigeki	scholz, mario	miyao, yusuke
23	petrov, slav	li, zhenghua	satta, giorgio	oflazer, kemal
24	liu, qun	tsarfaty, reut	miyao, yusuke	scholz, mario
25	carroll, john	collins, michael john	hajič, jan	liu, ting
26	matsumoto, yuji	liu, qun	petrov, slav	koo, terry
27	rambow, owen	inagaki, yasuyoshi	liu, ting	petrov, slav
28	galley, michel	seddah, djamé	tsarfaty, reut	tsarfaty, reut
29	màrquez, lluís	zong, chengqing	crammer, koby	gómez-rodríguez
30	curran, james r.	wang, zhiguo	tsujii, jun'ichi	attardi, giuseppe

Table A.1: Top 60 experts in "Dependency Parsing" - Part 1.
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
31	lavie, alon	kübler, sandra	gojenola, koldo	seeker, wolfgang
32	surdeanu, mihai	gojenola, koldo	gómez-rodríguez	gojenola, koldo
33	agirre, eneko	matsumoto, yuji	habash, nizar	candito, marie
34	carreras, xavier	foster, jennifer	candito, marie	smith, david a.
35	zhang, min	ambati, bharat ram	menzel, wolfgang	tsujii, jun'ichi
36	steedman, mark	satta, giorgio	zhang, min	che, wanxiang
37	hajič, jan	carreras, xavier	attardi, giuseppe	seddah, djamé
38	riezler, stefan	zhang, meishan	van den bosch, antal	surdeanu, mihai
39	mccallum, andrew	seeker, wolfgang	surdeanu, mihai	menzel, wolfgang
40	xue, nianwen	kuhn, jonas	smith, david a.	zhang, min
41	nilsson, jens	pereira, fernando	seeker, wolfgang	habash, nizar
42	cherry, colin	ohno, tomohiro	che, wanxiang	farkas, richárd
43	baldwin, timothy	versley, yannick	koo, terry	ribarov, kiril
44	zhou, guodong	oflazer, kemal	seddah, djamé	hajič, jan
45	liu, yang	sun, weiwei	farkas, richárd	choi, jinho d.
46	habash, nizar	kurohashi, sadao	tjong kim sang	manning, c. d.
47	riedel, sebastian	farkas, richárd	manning, c. d.	huang, liang
48	tjong kim sang	kazama, jun'ichi	liu, qun	marton, yuval
49	daumé iii, hal	jun, zhao	zhang, hao	liu, qun
50	hall, johan	eryiğit, gülşen	huang, liang	zhang, hao
51	cohn, trevor	zhao, hai	choi, jinho d.	rush, alexander m.
52	buchholz, sabine	uchimoto, kiyotaka	matsumoto, yuji	jaakkola, tommi
53	schütze, hinrich	kawahara, daisuke	clark, stephen	crammer, koby
54	blunsom, philip	zhang, yi	marton, yuval	tjong kim sang
55	koo, terry	sangal, rajeev	zhang, yi	van den bosch, antal
56	shen, libin	husain, samar	foster, jennifer	zhang, yi
57	zhou, ming	asahara, masayuki	rush, alexander m.	matsumoto, yuji
58	smith, david a.	søgaard, anders	kawahara, daisuke	foster, jennifer
59	sarkar, anoop	torisawa, kentaro	ribarov, kiril	globerson, amir
60	gao, jianfeng	wang, rui	liu, yang	clark, stephen

Table A.2: Top 60 experts in "Dependency Parsing" - Part 2.
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
1	och, franz josef	koehn, philipp	koehn, philipp	koehn, philipp
2	manning, c. d.	ney, hermann	ney, hermann	ney, hermann
3	koehn, philipp	callison-burch, chris	callison-burch, chris	callison-burch, chris
4	ney, hermann	sumita, eiichiro	hoang, hieu	hoang, hieu
5	collins, michael john	wang, haifeng	dyer, chris	dyer, chris
6	marcu, daniel	wu, hua	zens, richard	zens, richard
7	knight, kevin	huck, matthias	och, franz josef	sumita, eiichiro
8	callison-burch, chris	waibel, alex	sumita, eiichiro	monz, christof
9	roukos, salim	freitag, markus	monz, christof	och, franz josef
10	jurafsky, daniel	peitz, stephan	waibel, alex	waibel, alex
11	gildea, daniel	wu, dekai	byrne, william	carpuat, marine
12	dyer, chris	carpuat, marine	federico, marcello	federico, marcello
13	yarowsky, david	gispert, adrià de	huck, matthias	huck, matthias
14	papineni, kishore	banches, rafael e.	vogel, stephan	byrne, william
15	smith, noah a.	niehues, jan	carpuat, marine	vogel, stephan
16	tsujii, jun'ichi	liu, qun	liu, qun	gispert, adrià de
17	ward, todd	dyer, chris	gispert, adrià de	freitag, markus
18	mckeown, kathleen r.	fonollosa, josé a. r.	freitag, markus	liu, qun
19	wiebe, janyce	watanabe, taro	bojar, ondřej	bojar, ondřej
20	zens, richard	herrmann, teresa	wu, dekai	niehues, jan
21	ng, hwee tou	lambert, patrik	niehues, jan	bertoldi, nicola
22	resnik, philip	mariño, josée b.	peitz, stephan	wu, dekai
23	hoang, hieu	vogel, stephan	bertoldi, nicola	peitz, stephan
24	federico, marcello	bicici, ergun	birch, alexandra	herrmann, teresa
25	birch, alexandra	och, franz josef	herrmann, teresa	mariño, josée b.
26	bojar, ondřej	costa-jussà, marta ruiz	lambert, patrik	fonollosa, josé a. r.
27	bertoldi, nicola	monz, christof	fonollosa, josé a. r.	lambert, patrik
28	mihalcea, rada	hoang, hieu	banches, rafael e.	birch, alexandra
29	cowan, brooke	lo, chi-kiu	watanabe, taro	way, andy
30	shen, wade	wuebker, joern	mariño, josée b.	banches, rafael e.

Table A.3: Top 30 experts in "Machine Translation".
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
1	och, franz josef	manning, c. d.	tjong kim sang, erik f.	tjong kim sang, erik f.
2	manning, c. d.	sassano, manabu	manning, c. d.	manning, c. d.
3	klein, dan	utsuro, takehito	cucerzan, silviu	cucerzan, silviu
4	ney, hermann	dien, dinh	tsujii, jun'ichi	tsujii, jun'ichi
5	collins, michael john	ngo, quoc hung	grishman, ralph	florian, radu
6	knight, kevin	winiwarter, werner	yarowsky, david	grover, claire
7	pereira, fernando	finkel, jenny rose	florian, radu	zhou, guodong
8	roukos, salim	lee, gary geunbae	grover, claire	de meulder, fien
9	johnson, mark	tjong kim sang, erik f.	zhou, guodong	zhang, tong
10	charniak, eugene	tsujii, jun'ichi	finkel, jenny rose	yarowsky, david
11	yarowsky, david	uchimoto, kiyotaka	mccallum, andrew	grishman, ralph
12	smith, noah a.	zhou, guodong	de meulder, fien	finkel, jenny rose
13	tsujii, jun'ichi	yi, byoung-kee	zhang, tong	mccallum, andrew
14	ng, hwee tou	song, yu	su, jian	su, jian
15	roth, dan	kim, eunju	ng, hwee tou	chieu, hai leong
16	grishman, ralph	tsuruoka, yoshimasa	li, wei	li, wei
17	lin, dekang	rosset, sophie	tsuruoka, yoshimasa	tsuruoka, yoshimasa
18	clark, stephen	ekbal, asif	zhou, ming	ng, hwee tou
19	lin, chin-yew	shishtla, praneeth m.	jing, hongyan	jing, hongyan
20	cardie, claire	bandyopadhyay, sivaji	chieu, hai leong	kazama, jun'ichi
21	toutanova, kristina	su, jian	ananiadou, sophia	zhou, ming
22	petrov, slav	ananiadou, sophia	kazama, jun'ichi	roth, dan
23	weischedel, ralph m.	jun, zhao	ittycheriah, abraham	ittycheriah, abraham
24	wu, dekai	yu, xiaofeng	cucchiarelli, alessandro	ananiadou, sophia
25	riloff, ellen	de meulder, fien	velardi, paola	curran, james r.
26	màrquez, lluís	cucerzan, silviu	curran, james r.	cucchiarelli, alessandro
27	osborne, miles	žabokrtský, zdeněk	roth, dan	nguyen, huy
28	curran, james r.	kravalova, jana	zitouni, imed	zitouni, imed
29	schwartz, richard m.	etzioni, oren	osborne, miles	hachey, ben
30	carreras, xavier	gali, karthik	klein, dan	luo, xiaoqiang

Table A.4: Top 30 experts in "Named Entity Recognition".
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
1	manning, c. d.	mohammad, saif	lee, lillian	pang, bo
2	klein, dan	salameh, mohammad	pang, bo	lee, lillian
3	pereira, fernando	qin, bing	wiebe, janyce	wiebe, janyce
4	mcdonald, ryan	liu, ting	wilson, theresa	cardie, claire
5	hovy, eduard	zhou, ming	cardie, claire	wilson, theresa
6	lapata, mirella	wei, furu	hoffmann, paul	hoffmann, paul
7	yarowsky, david	tang, duyu	barbosa, luciano	barbosa, luciano
8	smith, noah a.	wan, xiaojun	tan, chenhao	tan, chenhao
9	lee, lillian	bhattacharyya, pushpak	mihalcea, rada	mihalcea, rada
10	tsujii, jun'ichi	montoyo, andrés	choi, yejin	feng, junlan
11	barzilay, regina	gutiérrez, yoan	mcdonald, ryan	montoyo, andrés
12	mckeown, kathleen r.	balahur, alexandra	feng, junlan	choi, yejin
13	wiebe, janyce	wiebe, janyce	pereira, fernando	mcdonald, ryan
14	resnik, philip	zhu, xiaodan	zhou, ming	wan, xiaojun
15	mihalcea, rada	steinberger, josef	diab, mona	zhou, ming
16	huang, liang	loukachevitch, natalia	montoyo, andrés	pereira, fernando
17	cardie, claire	chetviorkin, ilia	barzilay, regina	diab, mona
18	liu, qun	wilson, theresa	wan, xiaojun	bhattacharyya, pushpak
19	carroll, john	cardie, claire	bhattacharyya, pushpak	liu, bing
20	matsumoto, yuji	yang, jiang	manning, c. d.	mohammad, saif
21	rambow, owen	hou, min	radev, dragomir r.	manning, c. d.
22	pang, bo	elchuri, harsha	mohammad, saif	barzilay, regina
23	riloff, ellen	palanisamy, prabu	liu, bing	radev, dragomir r.
24	curran, james r.	yadav, vineet	dredze, mark	deng, lingjia
25	surdeanu, mihai	skiena, steven	riloff, ellen	dredze, mark
26	ng, andrew y.	chen, yanqing	blitzer, john	tsur, oren
27	haghighi, aria	bandyopadhyay, sivaji	banea, carmen	stoyanov, veselin
28	wilson, theresa	stein, benno	somasundaran, swapna	banea, carmen
29	radev, dragomir r.	das, amitava	stoyanov, veselin	balahur, alexandra
30	moschitti, alessandro	engels, gregor	deng, lingjia	andreevskaia, alina

Table A.5: Top 30 experts in "Sentiment Analysis".
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
1	klein, dan	liu, yang	lin, chin-yew	lin, chin-yew
2	ney, hermann	radev, dragomir r.	hovy, eduard	radev, dragomir r.
3	marcu, daniel	nenkova, ani	radev, dragomir r.	hovy, eduard
4	callison-burch, chris	mckeown, kathleen r.	mckeown, kathleen r.	mckeown, kathleen r.
5	jurafsky, daniel	hovy, eduard	nenkova, ani	nenkova, ani
6	charniak, eugene	carenini, giuseppe	liu, yang	liu, yang
7	mcdonald, ryan	wan, xiaojun	barzilay, regina	conroy, john m.
8	hovy, eduard	hirao, tsutomu	jing, hongyan	barzilay, regina
9	lapata, mirella	li, wenjie	cardie, claire	mihalcea, rada
10	smith, noah a.	lin, chin-yew	mihalcea, rada	zhou, liang
11	lee, lillian	cardie, claire	saggion, horacio	saggion, horacio
12	joshi, aravind k.	li, chen	zhou, liang	wan, xiaojun
13	tsujii, jun'ichi	saggion, horacio	wan, xiaojun	carenini, giuseppe
14	barzilay, regina	liu, fei	carenini, giuseppe	jing, hongyan
15	mckeown, kathleen r.	murray, gabriel	conroy, john m.	cardie, claire
16	ng, hwee tou	okumura, manabu	li, wenjie	li, wenjie
17	grishman, ralph	mihalcea, rada	lapata, mirella	lapata, mirella
18	brill, eric	lu, qin	firmin hand, therese	hirao, tsutomu
19	mihalcea, rada	conroy, john m.	hirao, tsutomu	firmin hand, therese
20	shen, wade	ng, raymond t.	okumura, manabu	okumura, manabu
21	lin, chin-yew	wang, lu	mani, inderjeet	murray, gabriel
22	cardie, claire	louis, annie	kan, min-yen	kan, min-yen
23	matsumoto, yuji	penn, gerald	klavans, judith l.	ng, raymond t.
24	galley, michel	weng, fuliang	drummey, kevin w.	filatova, elena
25	schwartz, richard m.	li, tao	donaway, robert l.	klavans, judith l.
26	surdeanu, mihai	dang, hoa trang	mather, laura a.	louis, annie
27	haghighi, aria	lapalme, guy	ng, raymond t.	hatzivassiloglou, vasileios
28	radev, dragomir r.	isozaki, hideki	okurowski, mary ellen	vanderwende, lucy
29	hatzivassiloglou, vasileios	litvak, marina	murray, gabriel	mani, inderjeet
30	riezler, stefan	maskey, sameer	grishman, ralph	liu, fei

Table A.6: Top 30 experts in "Summarization".
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Rank	GlobalCitations	Model2	RP	WRP
1	manning, c. d.	mihalcea, rada	yarowsky, david	yarowsky, david
2	koehn, philipp	liu, ting	ng, hwee tou	ng, hwee tou
3	pereira, fernando	ng, hwee tou	mihalcea, rada	mihalcea, rada
4	jurafsky, daniel	navigli, roberto	ide, nancy m.	navigli, roberto
5	hovy, eduard	che, wanxiang	resnik, philip	ide, nancy m.
6	chiang, david	bhattacharyya, pushpak	navigli, roberto	resnik, philip
7	palmer, martha	pedersen, ted	agirre, eneko	agirre, eneko
8	lapata, mirella	mccarthy, diana	pedersen, ted	pedersen, ted
9	yarowsky, david	stevenson, mark	dagan, ido	rigau, german
10	lee, lillian	diab, mona	rigau, german	dagan, ido
11	tsujii, jun'ichi	agirre, eneko	palmer, martha	palmer, martha
12	wiebe, janyce	donghong, ji	stevenson, mark	stevenson, mark
13	ng, hwee tou	kim, gilchang	veronis, jean	carpuat, marine
14	roth, dan	li, yongqiang	carpuat, marine	mccarthy, diana
15	resnik, philip	martinez, david	wiebe, janyce	wiebe, janyce
16	lin, dekang	wiebe, janyce	mccarthy, diana	diab, mona
17	brill, eric	guo, weiwei	diab, mona	wu, dekai
18	mihalcea, rada	ide, nancy m.	wu, dekai	veronis, jean
19	dagan, ido	baldwin, timothy	bhattacharyya, pushpak	bhattacharyya, pushpak
20	hearst, marti a.	lapata, mirella	wilks, yorick	lee, hian beng
21	carroll, john	patwardhan, siddharth	schütze, hinrich	wilks, yorick
22	matsumoto, yuji	resnik, philip	liu, ting	schütze, hinrich
23	wu, dekai	hoste, véronique	lee, hian beng	liu, ting
24	màrquez, lluís	lefever, els	martinez, david	strapparava, carlo
25	agirre, eneko	mohammad, saif	strapparava, carlo	martinez, david
26	ng, andrew y.	khapra, mitesh m.	tufiš, dan	chan, yee seng
27	carreras, xavier	chang, jason s.	chan, yee seng	tufiš, dan
28	radev, dragomir r.	li, sheng	hovy, eduard	zhong, zhi
29	zhang, min	chan, yee seng	dorr, bonnie jean	soroa, aitor
30	pantel, patrick	lee, hyun ah	soroa, aitor	hovy, eduard

Table A.7: Top 30 experts in "Word Sense Disambiguation".
Parameter settings: $\lambda = 0.1$; $\mu_d = 0.5$; $\mu_{ca} = 0.5$; top-documents = 500

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit im Bachelorstudiengang Informatik selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel - insbesondere keine im Quellenverzeichnis nicht benannten Quellen - benutzt habe. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht. Ich versichere weiterhin, dass ich die Arbeit vorher nicht in einem anderen Prüfungsverfahren eingereicht habe und die eingereichte schriftliche Fassung der auf dem elektronischen Speichermedium entspricht.

Unterschrift: *Ort, Datum:*

Erklärung zur Veröffentlichung

"Ich bin damit einverstanden, dass meine Abschlussarbeit in den Bestand der Fachbereichsbibliothek eingestellt wird."

Unterschrift: *Ort, Datum:*