Effective and Efficient Approaches to Retrieving and Using Expertise in Social Media

Reyyan Yeniterzi

CMU-LTI-15-008

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:
Jamie Callan, Chair
William Cohen
Eric Nyberg
Aditya Pal (Facebook)

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies.

For my dear parents and sister
and
the little princess Dila (2009-2015)

Abstract

The recent popularity of social media is changing the way people share and acquire knowledge. Companies started using intra-organizational social media applications in order to improve the communication and collaboration among employees. In addition to their professional use, people have been using these sites in their personal lives for information acquisition purposes, such as community question answering sites for their questions. In such environments the interactions do not always occur between users who know each other well enough to assess expertise of one another or trust the accuracy of their created content. This dissertation addresses this problem by estimating topic specific expertise scores of users which can be also used to improve the expertise related applications in social media.

Expert retrieval has been widely studied using organizational documents; however, the additional structure and information available in social media provide the opportunity to improve the developed expert finding approaches. One such difference is the availability of different types of user created content, which can be used to represent users' expertise and the information need being searched more effectively in order to retrieve an initial set of good expert candidates. The underlying social network structure constructed from the interactions among the users, such as commenting or replying, is also investigated and topic-specific authority graph construction and estimation approaches are developed in order to estimate topic-specific authorities from these graphs. Finally, the available timestamp information within social media is explored and a more dynamic expert identification approach which takes into account the recent topic-specific interest of users as well as their availability is proposed.

This available information is explored and the proposed approaches are combined in an expert identification system which consists of three parts; (1) content-based retrieval, (2) authority estimation and (3) temporal modeling. Depending on the environment and task being tested, some or all of these parts can be used to identify topic-specific experts. This proposed system is applied to two data collections, an intra-organizational blog data and a popular community question answering site's data, for three expertise estimation related tasks: identification of topic-specific expert bloggers, routing questions to users who can provide accurate and timely replies, and ranking replies based on responders' question specific expertise. Statistically significant improvements are observed in all three tasks. In addition to improving the effectiveness of expert identification applications in social media, the proposed approaches are also more efficient which makes the proposed expert finding system applicable to real time environments.

Acknowledgments

First and foremost, I would like to thank to my advisor Jamie Callan, who introduced me to the field of Information Retrieval and advised me through my PhD studies in this field. Being his student and working with him was a privilege that I am grateful for. His constructive feedbacks and continuous guiding helped me to produce this dissertation, but more importantly prepared me to the academic world. For the last 6 years, he was not just a great advisor but also the mentor who guided me through the obstacles I have encountered and prepared me for the future ones.

I am deeply grateful to my committee members, William Cohen, Eric Nyberg and Aditya Pal, who have kindly given me their valuable time and insightful feedbacks to make this thesis stronger. I express my sincerest gratitude to Ramayya Krishnan, who have supported my thesis in many ways.

I am thankful to my previous advisors, and mentors: Kemal Oflazer, Bradley Malin, Dilek Hakkani Tur, Yucel Saygin, Berrin Yanikoglu, Esra Erdem and Ugur Sezerman who have introduced me to the research world.

I am indebted to my friends and colleagues at the Carnegie Mellon University. They include but not limited to Bhavana, Derry, Yubin, Meghana, Sunayana, Jaime, Anagha, Jonathan, Le, David, Leman, Selen, Pradeep and Justin.

I am also deeply grateful to Fatma Selcen and Hakan for their close friendship and mentoring throughout my years in Pittsburgh. They together with Mehmet Ertugrul made my life in Pittsburgh very enjoyable. I would also like to thank all my other close friends for their friendship and support.

Last but most importantly, I would like to thank to my parents who have raised me as an individual who can follow her dreams and who does not give up easily. They have supported me at every stage of my life, and especially when I decided to go abroad for getting my PhD degree. This dissertation would not have been possible without the continuous motivation and support of my dear sister Suveyda. She was both there to lighten me up in my depressed moments and also there to celebrate my accomplishments. Therefore, I dedicate this dissertation to these three wonderful people in my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

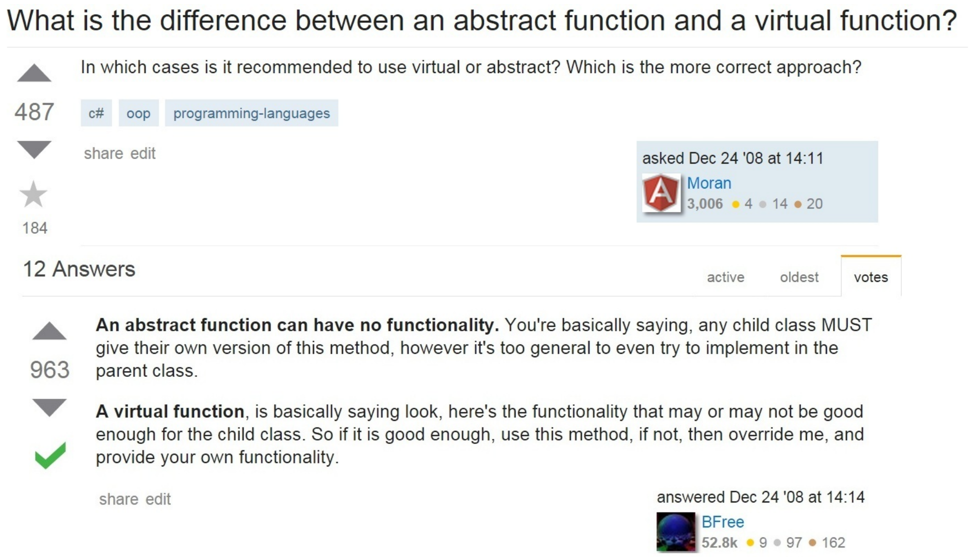
Over the last decade, social media environments where people create and share content in online communities became very popular. With the creation of different types of social media tools that work on personal computers and mobile devices, millions of people started to engage with social media in their daily lives. According to a recent survey by the Pew Internet and American Life Project [14], as of December 2012, 67% of Internet users in the United States are using social networking sites. Compared to 8% in 2005 and 46% in 2009, this change in the percentage of the users shows the increasing power of social media in community.

This growth of social media is also changing the way people obtain and exchange information in their professional and personal lives. A recent survey by McKinsey Global Institute [17] of more than 4200 companies and their employees revealed that around three quarter of the companies use social media to find new ideas and more than half of them manage their projects by communicating in these environments. The survey also reported that at least one quarter of companies use them as a resource to identify expertise within the company, where 29% of the companies use at least one social media tool for matching their employees to tasks and 26% of them assess their employees performances by using these tools.

People also benefit from social media in their personal lives in different ways. A recent survey by Morris *et al.* [65] of Facebook and Twitter users on their motivation for using their status messages to ask a question to their networks revealed that users trust their networks more than the results of search engines, especially for subjective questions like asking for an opinion or recommendation. People also use other more publicly open social media tools, like community question answering (CQA) sites, to get faster and more reliable answers to their complex or subjective questions. These services are also very popular; it has been reported that by 2009, Yahoo! Answers had more than 200 million users worldwide, with 15 million users visiting daily [85].

However, this new usage of social media for information acquisition comes with a risk that one should be aware of. Depending on the social media environment and its size, users don't always interact with users they know personally. For instance, StackOverflow, a popular domain specific CQA site, has 4.2 million users[1], programmers mostly, who don't personally know most of the other users of the site. An example question and answer from this site is presented in Figure 1.1. In this example, even though the user who asked the question doesn't know the user who answered the question, with the help of votes received from other users, the asker is convinced that this is a correct answer and accepted it as the best answer. However, users may

[1] Retrieved from http://stackexchange.com/sites?view=list#traffic on May 1, 2015



What is the difference between an abstract function and a virtual function?

487

In which cases is it recommended to use virtual or abstract? Which is the more correct approach?

c# oop programming-languages

asked Dec 24 '08 at 14:11
Moran
3,006 ● 4 ● 14 ● 20

184

12 Answers

963

An abstract function can have no functionality. You're basically saying, any child class MUST give their own version of this method, however it's too general to even try to implement in the parent class.

A virtual function, is basically saying look, here's the functionality that may or may not be good enough for the child class. So if it is good enough, use this method, if not, then override me, and provide your own functionality.

answered Dec 24 '08 at 14:14
BFree
52.8k ● 9 ● 97 ● 162

Figure 1.1: An example question and answer from StackOverflow CQA site.

not always be this lucky in receiving community feedback on reliability of content or its creator. In these situations, since users don't know each other personally and lack the knowledge and means to assess each other's expertise on the topic, it becomes hard to decide whether users have the necessary expertise on the particular topic and their topic-specific content are accurate and reliable. Because every social media user can create content without being checked on the accuracy or quality of the content, this uncertainty about the credibility of users and the content they create becomes an important problem. This problem motivates us to work on developing algorithms that will accurately assess the topic-specific expertise of users in order to improve the reliability of social media in general and effectiveness of applications which need some kind of expertise assessment of the users and their written content. This dissertation proposes to achieve these goals by developing algorithms that will effectively estimate the topic-specific expertise of users based on their user created content and interactions in the social media environment.

1.1 Retrieving Expertise in Social Media

Expert retrieval has been widely studied, especially after the introduction of the expert finding task in the Text REtrieval Conference's (TREC) Enterprise Track in 2005 [26]. The aim of this expert finding task was to produce a ranked list of experts on narrowly-defined unstructured topics by using a corpus of corporate documents. This task, which continued for four years, provided two different test collections crawled from two organizations' public-facing websites and internal emails which led to the development of many algorithms on expert retrieval.

A recent literature review, conducted by Balog *et al.*[10], provided a detailed overview of the prior research on expert finding. Among these systems, the most effective ones can be characterized by their representation of the candidate experts. *Profile-based approaches* create a model for each candidate using the associated documents of the candidate. Given a topic, standard text retrieval algorithms are used to rank these profiles based on their relevancy. On the other hand, *document-based approaches* initially retrieve topic-relevant documents, associate them with their authors, use them to determine the expertise of the authors, and finally rank the candidates based on their aggregated expertise scores. Profile-based approaches are somewhat less biased toward prolific candidates and consider all of a candidate's work, as opposed to just the work that most closely matches the query. Therefore, they are referred to as *query-independent models*, while the document-based approaches are referred to as *query-dependent models* [72]. Since document-based approaches are more topic-dependent, they outperformed profile-based approaches for enterprise documents.

Similar results can also be observed with social media collections, mainly because they don't always contain content on one specific domain, but instead contain discussions on multiple topics, such as technical subjects, hobbies, or news, concurrently. Therefore, using a query-dependent approach can result in better expertise estimates than using a query-independent approach. The success of query-dependent approaches depends on the query to be searched and the documents retrieved; therefore, the query and the indexed document structures should be carefully constructed for more effective retrievals. Different types of user created contents are available in social media environments, such as posts and comments in blogs, and questions, answers, and comments in community question answering (CQA) sites. Furthermore, some of these content types can have their own structure, such as title, body and tag fields in questions. Compared to TREC Enterprise Track's collections, social media has more rich and complex content structure. Therefore, as the first step in expert retrieval from these environments, we propose to use the available content types more effectively for better representation of users and queries to retrieve an initial good ranking of expert candidates.

In addition to the more structured content, social media environments also provide additional information that is not always available in previously studied TREC collections. One such important information social media collections have is the availability of different types of interactions among users, such as reading or commenting to each others' posts. Some of these user interactions can be an indication of more expertise of one user compared to the other one. For instance in the case of answering interaction in CQA environments, the responder has more expertise than the asker on the topic of the particular question. These interactions can be used to construct user interaction networks which can be used with network-based approaches to get an expertise ranking of users. This is actually similar to web page rankings. Instead of using the web pages and url links between them, the users and interactions between users are used to construct the network, and then similar network-based methods are applied in order to rank both the web pages and users.

Applying network-based methods to web pages returns reliable, important or popular pages, therefore these network-based approaches are referred to as *authority-based* methods. However, in expert-finding task, the phrases '*authority-based*' and '*authority-estimation*' can be confusing, since the goal of the task is to find people that are experts or authoritative, thus the word '*authority*' can be taken to mean either the task (expert-finding) or a way of accomplishing the task (network-based methods). In this dissertation '*authority estimation approach*' is used to refer to the network-based approach, '*authority network*' is used to describe the user interaction network and finally '*authority*' is used to call the top ranked nodes within these networks.

Among the TREC Enterprise Task data sets, only the W3C collection [1] contains user interactions in the form of email discussions with sender-receiver information available. However, in email communications it is not certain who is more authoritative than whom; therefore they are not very suitable for authority estimation. On the other hand, interactions in social media, such as question asking and answering, can be more informative for estimating the authority of corresponding users. Furthermore, there may be other interactions that can be indications of approval or disapproval such as votes, retweets, thumbs up/down, or like/unlike. Some other more well-defined user relations are in the form of friendships or follower/followings, which can be used to define more consistent authority estimations.

Prior work investigated the effects of some of these signals on expert finding over email, online forum collections, and community question answering sites by using topic-independent or topic-dependent authority estimation approaches. Link-based approaches, like PageRank [15], Topic-Sensitive PageRank [37] and HITS [45], that were developed for web pages, were also adapted to these environments in order to identify user authorities [20, 28, 104]. However, users are not like web pages that contain information on one topic. Users contribute to and interact with social media across multiple topics, therefore using all authority signals and applying these authority estimation approaches directly can cause topic drifts in authority estimations. In order to prevent such mistakes and improve upon the estimated content-based expertise scores, this thesis proposes more topic-specific authority network construction and estimation approaches designed for social media interactions.

Social media environments also keep timestamps for all user created content and interactions. This temporal information converts the data into a more dynamic structure which enables the construction of more dynamic expertise estimation approaches. For instance, these timestamps can be used to model a user's interest and expertise over time. Such a model will be useful for estimating more up-to-date expert candidates instead of users who have lost interest on the topic. This is especially important if information seekers are looking for experts to follow over time or to communicate with directly to resolve topic-specific problems. Especially, for expertise related tasks that require expert candidates to take an action, such as answering topic-specific questions of information seekers, the temporal models can also be useful in estimating the recent topic-specific interest and availability of users. This dissertation proposes a dynamic modeling of expertise which uses topic-relevant activities of users together with their timestamps in order to model users' expertise as well as interest changes over time.

This thesis proposes to improve upon the prior research on expert finding in several ways by using the additional and available information social media provides. As the initial step, the type and structure of the user created content and information needs are explored to identify the most effective representations to retrieve an initial good ranking of experts. Later on, the available user interactions are analyzed to construct more topic-specific authority networks and estimate more accurate authority scores. Finally, the temporal information is used to create dynamic expertise models to identify more up-to-date and available experts.

These approaches are tested on two types of social media, an intra-organizational blog collection and a community question answering collection. Topic-specific expert bloggers are identified within the first collection by applying some of these proposed approaches. This collection is a blend of enterprise and social media, therefore useful to analyze how expert finding approaches, tested in organizational documents like web pages, emails or reports, work in a more social setting within the enterprise. StackOverflow data is used as the second collection. These proposed approaches are also tested on this dataset with two expertise estimation related tasks, question routing and reply ranking.

1.2 Using Expertise in Social Media

Expert retrieval is a very useful application by itself, but it can also be a step towards improving other applications. In social media, any user can create content, therefore, the quality of any user-generated content depends on its author. The overall success of any application is also dependent on how it uses this content, based on both the content's and its author's relevancy and reliability. This thesis proposes to use the expertise of users in order to improve applications like question routing or reply ranking in community question answering (CQA) sites. Working on such tasks has another benefit which is the opportunity to test expert finding approaches with task-based evaluations. Feedback retrieved from these tasks' performances is also useful for analyzing the performance of expertise estimation.

1.2.1 Question Routing in CQA

An important problem of CQA sites is unanswered questions. In StackOverflow, around 26% of the questions have not received any replies[2]. There may be several reasons for these unanswered questions. For instance, the questions may be difficult for other users to answer, or they may not be very meaningful, or the question may not have been seen by users who can provide answers. Since these sites receive many questions (for example StackOverflow gets around 8300 questions daily[3]), it is likely that users miss questions that they can answer.

One way to reduce these unanswered questions is to route them to users who have the necessary expertise to provide accurate replies. Such a question routing mechanism does not only decrease the probability of unanswered questions, it may also decrease the average reply time for all questions, which is a desired feature in CQA sites. Receiving personalized questions which are selected based on users' expertise may also increase the engagement of these users to the system by helping them to answer more questions in less time.

This task of routing questions to the right users depends on accurate identification of question-specific experts. Given a question, this thesis proposes to estimate the expertise of users by using their content and interaction with other users. Since the success of question routing also depends on identifying candidates to reply to the routed question in a short time, temporal information is also used to develop more dynamic expertise models which are capable of finding candidates that are still interested in the particular topic and most likely available to answer.

1.2.2 Reply Ranking in CQA

Similar to web search engine rankings, in CQA sites, users also expect to see the replies ranked in a way that is most useful for them to satisfy their information need. Therefore, replies should be ranked based on their relevance to the corresponding question, accuracy in content, and quality in presenting the information. Today, many CQA sites rank replies based on the number of votes they received and place the asker's accepted (best selected) reply in the first rank. In case of a lack of these signals, some CQA systems, rank replies in preceding order of their posting time. Such a CQA portal is StackOverflow in which among the questions that are answered, around 25% of them have not received any votes to their replies, and therefore their replies are ranked by their posting times.

[2] Retrieved from http://stackexchange.com/sites?view=list#traffic on May 1, 2015
[3] Retrieved from http://stackexchange.com/sites?view=list#traffic on May 1, 2015

For such cases, this thesis proposes to use the question specific expertise and authority of the authors to provide a better ranking of replies until they receive votes. The evaluation of this task is performed with the actual votes these replies got. However, initial analysis on test sets revealed that for most of the replies the retrieved votes are biased because of the default (before receiving votes) ranking of replies. Therefore, this thesis introduces a more bias-free test set construction approach for choosing a set of questions which are less affected by the initial (default) reply ranking of the system. Later, expert retrieval approaches proposed for question routing task are also applied for more effective reply ranking.

1.3 Significance of the Research

Expert retrieval in social media is important for several reasons. The recent increase in popularity of social media is changing how information is shared and distributed in personal and professional life. Many organizations have started to employ these tools in order to improve collaboration among employees and help them to identify expertise within the organization. Even though not explicitly explored in this dissertation, the human resources departments also make use of public social media tools, like LinkedIn, StackOverflow, and Github, to identify possible passive (not actively job seeking) candidates for available positions. Therefore, organizations need effective expert retrieval tools that work on social media in order to identify expertise inside and outside the organization. Expert finding can also be used indirectly to improve some other applications in social media which require expertise estimation of the users. Such applications include routing questions or ranking replies in community question answering sites. In these applications, the topic-specific expertise of user can be used as an additional source of information to improve the overall performance of the task. Due to these reasons expertise estimation in social media is an important problem and this dissertation addresses this problem by introducing effective and efficient topic-specific expertise estimation approaches customized for social media environments [99].

Expert retrieval has been studied before in organizational settings but its applications to organizational social media have not been explored fully due to lack of available test collections. This thesis overcomes this problem by working on an international information technology firm's intra-organizational blog collection to identify topic-specific expert bloggers (employees) within the company. Having access to an organizational data also provided the privilege to reach less publicly common, more private data such as access logs of employees visits to other blog posts. This data is useful in terms of comparing the effects of explicit information, such as posts and comments, with more implicit information such as access (reading) information.

Task-based evaluations are also applied to analyze the effects of state-of-the-art and proposed approaches on expertise related tasks in CQA sites. The prior work on these tasks mostly focused on developing better algorithms to retrieve expertise in these environments, but we follow a different path and initially focus on finding better ways to represent expertise. Therefore, we try to answer the following research question:

- *RQ1: What are the most effective representations of information need and user expertise used for identifying expertise in question routing and reply ranking tasks in CQAs?*

Analyzing the available content in CQA sites revealed that user expertise and the information need for expertise search can be represented more effectively and also more efficiently. The proposed representations do not only improve the accuracy of expert search in CQA environments in general; but also show the power of representation in Information Retrieval (IR).

Identifying effective representations of information need not only includes choosing the right content type for retrieval, but also involves deciding on how to weight the query terms. Query term weighting has been explored a lot in IR community for different retrieval tasks. It is also important for expert retrieval, as some query terms can be more important than others for identification of expertise. Effective weightings of query terms are identified which further improved the accuracy of the expert identification for both question routing and reply ranking tasks. Furthermore, one type weighting shows how a generally working and accepted principle in IR, does not work in a particular expert finding task due to the differences in the underlying assumptions of retrieval. This finding is important in terms of showing how expert retrieval in social media can be rather different than other information retrieval tasks.

The interactions among users, either latent or more instantaneous interactions, are widely explored with authority-based approaches to identify topic-independent or topic-specific authoritative users. The prior work mostly focused on adapting these approaches, which are originally developed for web pages, to these environments in order to estimate authority scores of users. This dissertation analyzes these adaptations of authority-based approaches and addresses the following research question:

- *RQ2: Do the assumptions of topic-specific authority estimation approaches developed for web pages hold for user authority networks in social media? For the ones that do not, what kind of algorithmic modifications can be performed so that they hold, and is it possible to make additional assumptions and necessary modifications which can provide more effective and efficient topic-specific authority-based expertise estimations?*

Similar to content-based approaches, before focusing on the authority estimation approaches, the representations where these approaches will be applied to, more specifically the authority graphs (networks), are explored. Since expertise is estimated at the topic level, the authority graphs should also be topic-specific. Applying the topic-specific graph construction approaches developed for web documents directly to user networks may not return the expected topic-specific user networks since people are not the same as web pages. Therefore, this dissertation initially analyzes the widely used authority graph construction approaches and their effects on user networks. A more effective topic-specific authority graph construction approach is proposed which returned consistent and statistically significant improvements in accuracy [101]. Since authority estimation iterations are performed on more topic-focused sub-graphs, the proposed graphs provide significant gains in running times, which is especially important for the applicability of these approaches in real-time estimations.

In addition to constructing more topic-specific authority networks, the authority estimation approaches are also modified in order to estimate more accurate authority scores. During these adaptations, different user interactions perform differently, which motivates this dissertation to analyze the authoritative user interactions and check whether the connected nodes and the directed links among them satisfy the underlying assumptions of authority estimation approaches. Testing and analyzing different authoritative user interactions, such as reading, commenting and responding, shows the effects of the authority signal on topic-specific authority estimation, and how they can even affect the relative ranking of approaches. Necessary algorithm modifications are performed for some types of interaction types in order to improve the compatibility between the interaction and the approach. Furthermore, additional assumptions are also proposed to estimate not only the authorities but topic-specific authorities. Therefore, initially estimated topic-specific expertise scores of users, and how they can be used to estimate more topic-specific authorities are explored in detail. Experiments with different types of interactions provide a better understanding of the relationship between authority-based algorithms and used authority interactions. The effectiveness of these proposed approaches also shows the importance of necessary adaptation of algorithms based on the input type and the task.

The prior expert finding approaches did not model the dynamic aspects of users, like their topic-specific interest and expertise change over time. These aspects are especially important for expertise related tasks, in which the success of the tasks depends on some kind of future activity from the identified experts, such as posting content relevant to the particular topic, or answering a topic-specific question. Therefore, this dissertation addresses the following research question:

- *RQ3: What techniques can be used to identify more up-to-date topic-specific experts who have shown relatively more topic-specific expertise and interest in general and also recently?*

Some prior work on question routing tried to estimate the availability of users but they did not consider their recent topic-specific interest. Instead of estimating availability independent from the topic, this dissertation proposes a more dynamic expertise estimation approach which integrates the available temporal information into some of the existing state-of-the-art topic-specific expert finding approaches. The proposed approach [102] models the expertise, availability and recent topic-specific interest of users at the same time and provides statistically significant improvements over both static approaches and approaches that just estimate and use the availability of users.

Last but not the least, this dissertation analyzes the evaluation methodologies used in prior task-based expert identification work. For instance, the prior research on reply ranking in CQA sites randomly chose the test set questions and used the actual feedbacks of users using the system, such as votes, as assessment values. However, the interface of the CQA system and the behavior of users using the site may affect the number of votes replies receive independent from their accuracy and quality. Therefore, this dissertation initially analyzes the effects of the user interface of these environments and the uncontrolled behavior of the users on user feedbacks, and identifies two types of biases which affect the evaluation results of approaches [100]. In order to prevent the effects of these biases, this thesis focuses on the following question:

- *RQ4: What techniques can be used to construct less biased test collections based on the identified cases of biases?*

Even though a more bias free test set construction approach is proposed, the actual effects of the previously used biased test sets on prior research is unknown, which shows the significance of the identified biases.

1.4 Overview of Dissertation Organization

This thesis is organized as follows. Chapter 2 summarizes the prior work on expert retrieval. Chapter 3 gives an overview of the proposed expert retrieval system for social media and Chapter 4 details the dataset and experimental methodology used in experiments. Chapter 5 describes the proposed content-based approach, more specifically the proposed expertise representations for CQA environments. Chapter 6 presents the proposed authority network construction and authority estimation approaches and Chapter 7 explains the proposed temporal modeling of expertise. Chapter 8 combines the proposed content, authority and temporal approaches under one expert finding system. Finally, Chapter 9 concludes the dissertation.

Chapter 2

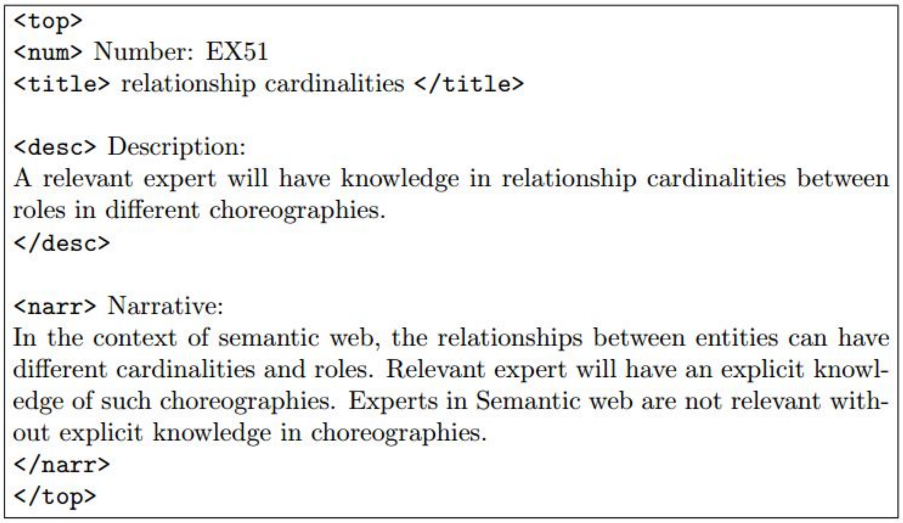
Related Work

Expert finding problem has been around for a while, and recently with the availability of community question answering sites, its applications to question routing and reply ranking tasks have received much research attention. This chapter initially covers the related work on these areas and shows in what ways this dissertation builds upon the existing work.

2.1 Related Work on Expert Retrieval

Expert retrieval has been widely studied, especially right after the launch of TREC Enterprise track in 2005, which continued for four years [4, 9, 26, 90]. This track included an expert finding task, in which the aim was to produce a ranked list of experts on narrowly-defined unstructured topics by using a corpus of corporate documents. An example topic from this track is provided in Figure 2.1. As can be seen in the figure, the TREC topics include a *title* field that consists of several key terms representing the information need. This information need is described in more detail in *description* and *narrative* fields.

This task provided a common platform for researchers to work on this problem, and led to



```
<top>
<num> Number: EX51
<title> relationship cardinalities </title>

<desc> Description:
A relevant expert will have knowledge in relationship cardinalities between
roles in different choreographies.
</desc>

<narr> Narrative:
In the context of semantic web, the relationships between entities can have
different cardinalities and roles. Relevant expert will have an explicit knowl-
edge of such choreographies. Experts in Semantic web are not relevant with-
out explicit knowledge in choreographies.
</narr>
</top>
```

Figure 2.1: An example TREC Enterprise Track topic for expert search.

development of many expert finding algorithms. These algorithms are initially characterized into several groups. In *profile-based approaches*, a candidate profile is created by using all the associated documents of the candidate, then given a topic, standard text retrieval is used to rank these profiles by their topic relevancy. *Document-based approaches* initially retrieve topic relevant documents, and for each associated candidate, a topic-specific expertise score is calculated by aggregating candidate's associated retrieved documents. *Graph-based approaches* go beyond using only the text context associated with the expert candidates and explore additional evidence of expertise by exploiting the relations between candidates and documents. *Learning-based approaches* incorporates many features and learn a parametric probability model by using training data in order to estimate expertise.

2.1.1 Document-Candidate Associations

Before getting into details of these expert finding approaches, a summary of the widely used document-candidate associations is explained in this section. Associating documents with the right set of candidates is the initial key step towards accurate expert retrieval.

Establishing connections between documents and candidates starts with identifying the set of candidates within the document. Candidates either exist in the context of document or they are explicitly mentioned in the document metadata. The former case mostly applies to email bodies as people refer to or talk about each other, project reports with names of the responsible personnel, or official papers with citations and references other similar papers and authors. Email collections, with senders and recipients specifically tagged within documents, are examples of documents with users available in metadata.

These tags mostly contain unique identifiers, therefore they provide unambiguous identification of expert candidates [6]. However, in case of lack of such metadata, *named entity recognition (NER)* can be applied to documents in order to match the existing candidate identifiers. Previous work developed rule-based methods to match the full names, last names or e-mail addresses of the candidates [8, 114]. During this matching, there is a trade of between recall and precision, choosing a strict rule-based approach increases the precision with the cost of missing some existing associations. On the other hand, using a more relaxed approach may create associations that do not exist. There is also the possibility of ambiguity which is handled by either using a set of heuristic rules to match the identifiers with the most probable candidates [114] or associating document with all the matched expert candidates [8].

After identifying the candidates within the document, there is also the step of assigning weights on the associations based on their strengths. Some prior work used a *boolean* existence approach, and assigned equal weights to all candidates that exist in the document [7]. Some other prior work used *frequency-based* approach and assigned weights based on the frequency of the candidate match within the document [7, 8]. There are also approaches that used the specific location of the candidate match within the document to assign weights [6]. The effect of weighting associations depends on the approach used; for document-based approaches, weighting the strength of associations provided limited improvements while more considerable improvements were observed with profile-based approaches.

In general, it has been observed that the optimum document-candidate association approach highly depends on the type and structure of the documents. For our research, establishing such associations is not an issue. In social media, all users are given unique user ids which are used to tag all their activities within the environment. This available metadata is used in this dissertation in order to unambiguously match any content with its author.

2.1.2 Profile-based Approach

Profile-based approach initially tries to construct a textual representation of users based on documents they are associated with, and then, these user representations are ranked based on their relevance to the given query. In other words, by using all the associated documents, these approaches construct models of candidates that can be considered as big documents, and so the problem of expert finding becomes document retrieval as ranking user profiles instead of documents.

A profile-based approach for expert retrieval was initially proposed by Balog *et al.* [8]. In this approach Bayes' Theorem was applied to estimate the probability of expert candidate, e, given the query, q, $P(e|q)$, as shown:

$$\begin{aligned} P(e|q) &= \frac{P(q|e)P(e)}{P(q)} && (2.1) \\ &\simeq P(q|e)P(e) && (2.2) \end{aligned}$$

In this equation, $P(q)$ can be ignored since it is going to be same for all expert candidates for a given query. The $P(e)$ is the probability of expert candidate (candidate prior) which is assumed to be uniform in most expert finding algorithms. Therefore, the most and sometimes the only important component in the formula becomes the $P(q|e)$ which is the probability of the query given the expert candidate.

2.1.2.1 Model 1

Balog *et al.* [8] proposed *Model 1*, a profile-based approach, in which the $P(q|e)$ is calculated by using all the associated documents of the candidate. *Model 1*, which is also known as *candidate model*, uses all the associated documents of the candidate to build a candidate language model, θ_e, representing the candidate's expertise areas. θ_e is a multinomial probability of distribution over the vocabulary terms. This model assumes that query terms, t, are independently and identically sampled which makes

$$P(q|\theta_e) = \prod_{t \in q} P(t|\theta_e)^{n(t,q)} \tag{2.3}$$

where $n(t,q)$ is the number of term t in query q. In order to prevent data sparsity problems, smoothing is applied by using $P(t)$, the probability of term t within the collection as shown:

$$P(t|\theta_e) = (1 - \lambda)P(t|e) + \lambda P(t|C) \tag{2.4}$$

where λ is the smoothing parameter, and C is the collection of all documents. Since terms are connected to candidate experts through documents, $P(t|e)$ can be represented as follows:

$$P(t|e) = \sum_{d \in D_e} P(t|d,e)P(d|e) \tag{2.5}$$

where D_e is a subset of documents with $p(d|e) > 0$[1]. Assuming that terms and candidates are conditionally independent given document d, the above equation becomes:

$$P(t|e) = \sum_{d \in D_e} P(t|d)P(d|e) \tag{2.6}$$

[1] With this condition, this summation over documents associated with expert candidate will be the same as a summation performed over all the documents within the collection.

At the end, the following model is used for estimating expertise.

$$P(q|\theta_e) = \prod_{t\in q}\left((1-\lambda)\left(\sum_{d\in D_e} P(t|d)P(d|e)\right) + \lambda P(t)\right)^{n(t,q)} \tag{2.7}$$

2.1.3 Document-based Approaches

Instead of using all documents of users and directly modeling the expertise of candidates, document-based models initially retrieve documents that best match the query and then identify expert candidates by using associations retrieved from these documents. Since this approach considers documents individually, more topic-specific expertise estimates can be performed easily by turning the focus on top ranked documents instead of using all documents. Two state-of-the-art document-based approaches are described in this section.

2.1.3.1 Model 2

Balog *et al.* [8] also proposed a document-based approach called *Model 2*, where $P(q|e)$ is calculated over documents associated with e. Unlike *Model 1*, where expert candidates are modeled directly, in *Model 2*, initially the documents are modeled and retrieved, and then expertise is estimated over the associated candidates. This model, also known as *document model*, can be formulated as follows:

$$P(q|e) = \sum_{d\in D_e} P(q|d,e)P(d|e) \tag{2.8}$$

Similar to *Model 1*, *Model 2* also assumes that query terms are independently and identically sampled, and terms and candidates are conditionally independent given the document.

$$\begin{aligned} P(q|d,e) &= \prod_{t\in q} P(t|d,e)^{n(t,d)} && (2.9)\\ &= \prod_{t\in q} P(t|d)^{n(t,d)} && (2.10)\end{aligned}$$

For each document, a document model can be also inferred with $P(t|d) \approx P(t|\theta_d)$ where

$$P(t|\theta_d) = (1-\lambda_d)P(t|d) + \lambda_d P(t|C) \tag{2.11}$$

Finally, the equation becomes:

$$P(q|e) = \sum_{d\in D_e}\left(\prod_{t\in q} P(t|\theta_d)^{n(t,d)}\right)P(d|e) \tag{2.12}$$

Initially, all documents associated with the expert candidate, as shown with $d \in D_e$, were used for both models. However, assuming all documents of a candidate are on a particular topic is not very realistic. Later experiments on topicality, performed by retrieving query relevant top n documents, and using these documents in modeling, showed improvements in *Model 2*, however did not help *Model 1* [8].

Overall, *Model 2* outperformed *Model 1* in several ways. In terms of applicability, *Model 2* is easier to set up compared to *Model 1*. *Model 1* requires separate indexes to be built, however

Model 2 can be built over any document search engine. Experiments with *Model 1* and *Model 2* also showed that *Model 2* is more robust to parameters chosen and outperforms *Model 1* in nearly all conditions [5]. However, in terms of time complexity, *Model 1* is more efficient than *Model 2* mainly due to retrieving a ranked list of candidates directly instead of first retrieving a ranked list of documents, then associating them with their corresponding expert candidates, and finally ranking the identified list of candidates based on their expertise.

2.1.3.2 Voting Models

Another set of document-based expert finding approaches are the voting models[2] which were introduced by Macdonald and Ounis [61]. These models are adaptations of data fusion methods to expert retrieval. They use standard text retrieval to retrieve documents relevant to a topic and then candidates receive a (possibly weighted) vote for each retrieved document they are associated with. Finally, expert candidates are ranked by the number of votes they receive.

Macdonald and Ounis used *Votes*, which assumes that all retrieved documents of a candidate provide an equal vote towards the expertise of the candidate, as their baseline. In other words, candidates are ranked by the number of retrieved documents they are associated with. The authors also adapted the Reciprocal Rank data fusion technique [105] to expert retrieval. In *Reciprocal Rank* (RR) method, the rank of the candidate is determined by the sum of the reciprocal ranks of the retrieved associated documents. This method favors candidates associated with many top ranked documents. Another method applied is the *CombSUM* method [33] in which the relevance score of the candidate is the sum of relevance scores of the ranked associated documents. In *CombMNZ* method [33], the sum from *CombSUM* is multiplied with the number of retrieved documents associated with the candidate to determine the final score of the candidate. *CombMNZ* favors both candidates with many relevant documents and candidates with many top ranked documents. Voting models using minimum, maximum, medium and average of relevance scores are also experimented with. Additionally, exponential variants of these methods are also proposed where the score of each document is transformed by applying the exponential function (e^{score}) as suggested by Ogilvie and Callan [69]. Applying the exponential function is necessary for retrieval systems which return the log of the actual document relevancy probability. By applying this exponential function, these negative relevance scores are converted to positive values. Applying this exponential function also boosts the scores of highly ranked documents.

2.1.3.3 Voting Models and Model 2

Under certain conditions, voting models can be considered very similar to the Balog's *Model 2*. Combining Equation 2.2 and 2.8 can be represented as

$$P(e|q) \simeq \sum_{d} P(q|d)P(d|e)P(e) \tag{2.13}$$

In Equation 2.13,

- $P(q|d)$ is the query-relevance score of document d[3], which is binary for the *Votes* method, document's reciprocal rank for the *ReciprocalRank* approach, and document's retrieval score for the *CombSUM* and *CombMNZ*.

[2] Similarity of voting models to Model 2 is explained in Section 2.1.3.3.

[3] The $P(q|d)$ should not be confused with the document relevance score from a document search engine. The relevance score (or its exponential) from document search engine is directly used only in *CombSUM* approach.

- $P(d|e)$, document-candidate association, is considered as binary in voting models.
- $P(e)$ is uniform for *Votes*, *ReciprocalRank* and *CombSUM*, and it is the number of relevant retrieved documents associated with expert candidate *e* for *CombMNZ*.

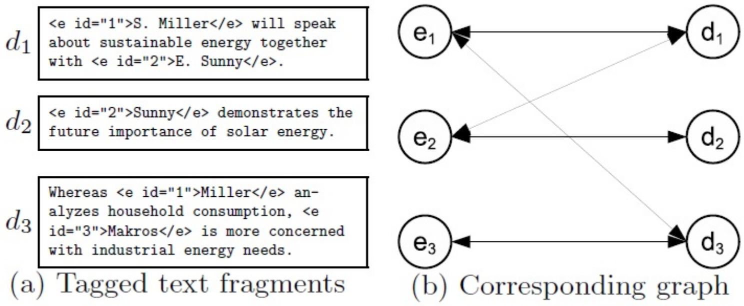
Therefore, *Model 2* can be considered as a type of voting model. Both models are built over a document search engine, and both of them use the retrieved documents for estimating expertise.

2.1.4 Graph-based Approaches

In graph-based models, entities, like candidates and documents, are represented as nodes, and relations and interactions among these entities are represented as edges connecting the nodes. These graphs consist of either all the documents and expert candidates within the whole collection or they are created by using only topic-relevant documents and candidates.

2.1.4.1 Infinite Random Walk Model

Serdyukov *et al.* [84] proposed a multi-step relevance propagation algorithm that works on topic-relevant *expertise graphs*. An example topic-relevant expertise graph is provided in Figure 2.2. In this bipartite graph, the edges are in between documents and expert candidates.



```
<e id="1">S. Miller</e> will speak
about sustainable energy together
with <e id="2">E. Sunny</e>.
```

```
<e id="2">Sunny</e> demonstrates the
future importance of solar energy.
```

```
Whereas <e id="1">Miller</e> an-
alyzes household consumption, <e
id="3">Makros</e> is more concerned
with industrial energy needs.
```

(a) Tagged text fragments (b) Corresponding graph

Figure 2.2: A sample bipartite expertise graph.

Unlike one step propagation in *Model 2* and voting models, this model proposes propagating expertise multiple times within the expertise graph based on the intuition that this is how users seek expertise. Given an expertise graph with document-document, candidate-candidate and document-candidate links, an expert seeker may do the following actions in order to find expertise.

- At any time, randomly (1) read a document or (2) consult with a candidate.
- After reading a document, (1) read one of the linked documents or (2) consult with one of the associated candidates.
- After consulting with a candidate, (1) read one of the associated documents or (2) consult with one of the connected candidates.

Based on the assumption that documents and expert candidates of the same knowledge are located close to each other in expertise graphs, Serdyukov *et al.* [84] proposed random walk algorithms for estimating expertise. In infinite random walk model, the user visits document and expert candidate nodes over and over again. During this infinite walk, in order to keep candidate in close proximity to relevant documents and candidates, jump transitions are introduced. According to the model, the probability of jumping to a document node is equal to document's probability of being relevant to the query (Equation 2.14). Similarly, the jump probability to an expert candidate node is equal to the probability of randomly choosing a retrieved document associated with the candidate (Equation 2.15). Following equations provide a formal representation for this model

$$P_{jump}(d) = P(d|q) \tag{2.14}$$

$$P_{jump}(e) = \frac{n(e, Top)}{|Top|} \tag{2.15}$$

$$P_i(d) = \lambda P_{jump}(d) + (1 - \lambda) \sum_{e \rightarrow d} P(d|e) P_{i-1}(e) \tag{2.16}$$

$$P_i(e) = \lambda P_{jump}(e) + (1 - \lambda) \sum_{d \rightarrow e} P(e|d) P_{i-1}(d) \tag{2.17}$$

where λ is the probability that at any step the user makes a jump instead of following an outgoing link. $n(e, Top)$ is the number of associated documents of expert candidate e within the retrieved $|Top|$ number of most relevant documents.

Graphs used for infinite random walk model are not strictly bipartite due to the probability of a jump to any node, but they still ignore the transitions between same type of entities like documents to documents or candidates to candidates. Users may follow document links to reach other related documents, or they can be directed to other expert candidates through their friends' or co-workers' references. Such additional links can be introduced to graphs depending on the availability of data. Serdyukov *et al.* [84] used documents' links and organizational structure to extend their expertise graph, and applied infinite random walk approach by updating Equations 2.16 and 2.17 in the following way:

$$P_i(d) = \lambda P_{jump}(d) + (1 - \lambda) \left((1 - \mu_d) \sum_{e \rightarrow d} P(d|e) P_{i-1}(e) + \mu_d \sum_{d' \rightarrow d} P(d|d^{'}) P_{i-1}(d') \right) \tag{2.18}$$

$$P_i(e) = \lambda P_{jump}(e) + (1 - \lambda) \left((1 - \mu_e) \sum_{d \rightarrow e} P(e|d) P_{i-1}(d) + \mu_e \sum_{e^{'} \rightarrow e} P(e|e^{'}) P_{i-1}(e') \right) \tag{2.19}$$

In above equations

$$P(d|d^{'}) = \frac{1}{N_{d'}} \quad \text{and} \quad P(e|e^{'}) = \frac{1}{N_{e'}} \tag{2.20}$$

where $N_{d'}$ is the number of outgoing document links from the document $d^{'}$ and $N_{e'}$ is the number of outgoing candidate links from the expert candidate $e^{'}$.

In addition to the *infinite random walk* model, *finite random walk* approach, which assumes that a user performs some predefined number of actions during expertise search instead of a non-stop process, has also been proposed.

These random walk approaches have been compared against Balog's *Model 2* [8] and *Votes* approach from Voting models [61] and shown to outperform these approaches on TREC's enterprise collection. However, their performances have not been compared to other more advance voting models. This comparison is an analysis that will be performed in the following chapters of this thesis.

2.1.4.2 Authority-based Approaches

In addition to directly using the user created content, the underlying social network structure between candidates is a valuable source for network-based methods in order to identify more influential topic-specific experts. Therefore, previous research also worked on estimating centrality and authority of users in the graph with or without using the query by applying network-based estimation approaches like PageRank [15] or HITS [45]. Before getting into details of these network-based expertise estimation approaches, which are referred to as authority-based approaches, and their application to user interaction (authority) networks, we initially describe the PageRank and HITS in order to provide some necessary background.

PageRank [15] estimates the probability of a random surfer reaching a web page. In a random walk, the surfer starts from a random page and moves to another page by either following one of the outgoing links (if there is one) or jumping to a random page. Over time, pages that are visited more are more important due to having incoming links from other important web pages. Similarly, in a user authority network, users who have been linked from other important users should be important. Based on this intuition PageRank is applied to user authority networks as shown:

$$PR(u) = \frac{1-d}{N} + d(\sum_{u_i \in IL(u)} \frac{PR(u_i)}{|OL(u_i)|}) \tag{2.21}$$

where $PR(u)$ is the PageRank score of user u, N is the number of users, $IL(u)$ is the set of users with an incoming link to user u, $PR(u_i)$ is the PageRank score of user u_i, $|OL(u_i)|$ is the number of users user u_i has an outgoing link to, and d is the damping factor for random jumps.

HITS [45] approach divides topic-related web pages into two categories; the ones that can be considered as authoritative source of information for the topic, referred to as *authority* pages, and the others called *hub* pages which contain a list of links to the topic-related authoritative pages. A good hub page points to many good authority pages, while a good authority page is linked to by many good hub pages. Similarly, for users, a good hub user knows and refers to many good authorities on a specific topic, and a good authority user is one that has been referred to by many good hub users regarding the particular topic. The authority score of user u is equal to the sum of the hub scores of users that point to user u, and hub score is equal to the sum of the authority scores of users that user u points to. Hub and authority scores of users are calculated as follows:

$$Auth(u) = \sum_{u_i \in IL(u)} Hub(u_i) \tag{2.22}$$

$$Hub(u) = \sum_{u_i \in OL(u)} Auth(u_i) \tag{2.23}$$

where $Auth(u)$ is the authority and $Hub(u)$ is the hub score of user u. $IL(u)$ is the set of users with an incoming link to user u, while $OL(u)$ is the set of users with an outgoing link from user u.

Previous work applied these PageRank or HITS algorithms to user authority networks constructed from user interactions. As different social media platforms are introduced over time, new types of interactions emerge between users which can be used to estimate authorities. Lappas et al. [95] provided a survey of authority-based expert finding approaches used on social networks.

Campbell *et al.* [20] extracted senders and receivers from topic relevant emails and built expertise graphs. They showed that applying a modified version of the HITS algorithm over these graphs improves the precision with the cost of a lower recall compared to using a simple content-based approach. Part of the email dataset used in this paper was also used by Dom *et al.* [28] in order to test HITS, PageRank and other graph-based ranking algorithms' effect on expertise ranking. According to their results PageRank performed much better than other algorithms, including HITS. Chen et al. [22] also applied PageRank and HITS to the email communication network of TREC W3C collection.

HITS and PageRank like algorithms were also applied to CQA sites with the aim of identifying expertise. Jurczyk and Agichtein [42, 43] applied HITS approach to question answer communities in order to discover authorities. Zhang *et al.* [104] applied several network-based algorithms to online help-seeking communities. The authors proposed *ExpertiseRank*, a variation of PageRank algorithm which is calculated as follows,

$$ER(u) = (1 - d) + d(\sum_{i=1}^{n} \frac{ER(u_i)}{C(u_i)}) \tag{2.24}$$

where $ER(u)$ is the *ExpertiseRank* score of user u, n is the number of users user u answered questions from, $C(u_i)$ is the number of users replied questions from user u_i and d is the damping factor for random jumps. Zhang *et al.* [104] also tested HITS algorithms and *z-score* measure which combines a user's asking and answering patterns. Their experiments revealed different results than Campbell *et al.* [20] and Dom *et al.* [28]. In their dataset, z-score, the simpler method, performed better than ExpertiseRank and HITS. After performing several simulations, the authors confirmed that the performance of graph-based algorithms highly depend on the structure of the network. For instance, they identified that their *ExpertiseRank* approach works better in networks when expertise of askers' and responders' are correlated, in other words responders are more selective on choosing challenging questions that they can answer.

One-step propagation approaches are also explored. Fu et al. [34] started with a set of experts in order to identify experts by using the associations between them. Their estimated expertise score were dependent on the expertise of the associated seed expert and the degree of association between the corresponding users. Associations like co-occurrence within a document or within an email communication (existence in any one of the from, to, cc fields) were used. They applied one-step propagation from seeds to other associated users, therefore their approach can be considered as a voting based approach which uses the co-occurrence with topic-specific experts, instead of co-occurrence in topic-specific documents.

Several prior work also used an initially estimated expertise as prior score and propagate it in graphs to identify other users. Zhang et al. [103] propagated expertise in a heterogeneous academic collaboration network in order to identify experts. An initial expertise score is calculated with profile-based approach and then a belief propagation model [31] was applied to estimate final expertise scores as shown:

$$s(v_i)^{t+1} = s(v_i)^t + \sum_{v_j \in U} \sum_{e \in R_{ji}} w((v_j, v_i), e) s(v_j)^t \tag{2.25}$$

where $s(v_i)$ is the expertise score of node v_i, U denotes all neighbors of node v_i. R_{ji} stands for all edges between v_j and v_i, $e \in R_{ji}$ is one kind of edge between nodes v_j and v_i and $w((v_j, v_i), e)$ is the corresponding propagation coefficient. In an earlier version [53], the authors also applied this approach to a homogeneous academic network. This approach is similar to other propagation algorithms with an earlier estimated expertise used as the initial (prior) score before iterations. Lu et al. [60] also applied the same algorithm to CQA answering graphs which do not only have the direct answering edges but also the latent edges which were obtained with similarity between askers' and responders' profiles.

Similarly, Karimzadehgan et al. [44] tried to identify experts within an organization, by propagating expertise from users with profiles to users who don't have profiles. After initial expertise score was estimated for a user with profile, the organizational hierarchy was used to propagate this expertise to other employees who are working closely together with the particular identified expert candidate (such as his managers or peers etc.). This approach is similar to graph smoothing as initially estimated expertise scores were locally smoothed with their neighbors scores as follows:

$$p_{smooth}(q|e_j) = \alpha p(q|e_j) + \frac{1-\alpha}{N_j} \sum_{i=1}^{N_j} p(q|e_i) \tag{2.26}$$

where $p_{smooth}(q|e_j)$ is the final expertise score of e_j, while $p(q|e_j)$ and $p(q|e_i)$ are the initial scores of e_j and e_i. α is weighting parameter and N_j is the number of e_j's neighbors. In addition to applying this algorithm for direct neighbors, the second and third degree neighbors were also experimented.

Jiao et al. [39] proposed a modified PageRank approach for online communities, in order to overcome a possible expert spamming issue caused by mutually referencing activities within a small group of users. With the proposed Weighting Reference Relationship algorithm, the transition probabilities are re-weighted by decreasing the weights of direct or indirect back-link references within a certain radius.

Seo and Croft [83] proposed two graph construction approaches. In post-based approach the individual post's nodes are linked to associated users' nodes. In thread-based approach, instead of using individual documents for posts, a thread was constructed by using all the posts, and graphs are constructed by linking these threads to associated users. Applying these approaches to emails and online discussion forums within TREC enterprise collection showed that thread-based graphs are more effective due to capturing a better understanding of the posts together with other posts within the thread.

Finding topic-specific experts in Twitter is also important in terms of identifying influential and authoritative users to follow. For this purpose, Weng *et al.* [96] proposed TwitterRank, an extension of PageRank, to identify topic-sensitive influential twitterers. The proposed algorithm uses both the topical similarity between users and the link structure (following-follower network) to identify topical authorities in microblogs. This algorithm is different than PageRank in that transitions between users are weighted with the topic specific similarity of users which has been calculated from estimated topic models. Pal and Counts [70] also proposed a non-graphical approach for estimating topical authorities in microblogs. The authors applied probabilistic clustering over a set of user features. The users within the cluster were later ranked to create a final ranking of authorities. Compared to link-based authority estimation approaches, their algorithm shown to be more efficient in terms of running time.

Additionally Noll *et al.* [68] proposed a HITS-like algorithm, called *SPEAR*, for ranking experts in a collaborative tagging system. Their method assumed that there is a mutual re-inforcement relationship between user's expertise and the quality of tagged documents, and experts (discoverers) are the ones who finds out useful resources before other users (followers) do. In this setting, experts act as hubs and receive hub scores as expertise score, and documents act as authorities and receive authority scores as quality score.

Authority estimation approaches were also applied to author networks constructed from citations. Liu et al. [57] proposed *AuthorRank*, a weighted version of PageRank approach where the weights are based on co-authorship frequency between authors. Deng et al. [27] extended the *AuthorRank* approach by using community information. They estimated *AuthorRank* for each community and scores from communities that are similar to a given query are combined to estimate the final score.

SmallBlue (aka. IBM Atlas) [29, 54, 55] is a social networking application designed to locate knowledgeable members and communities within an organization. This system combined two prior works: *ExpertiseNets* [92] and *CommunityNet* [91]. *ExpertiseNet* are dynamic graphs constructed for each user to represent particular user's expertise profile. In these graphs, exponential random graph model [88] was used to describe the relational information of expertise areas and the dynamic actor-oriented models [89] were used to describe the temporal evaluation of these expertise areas. *CommunityNet* graphs were also built for each user in order to capture the context-dependent and temporal evaluation information from email communications of the particular user. The authors proposed Content-Time-Relation algorithm which is an incremental LDA approach to model the contextual, relational and temporal information together. SmallBlue system used the network structure in order determine the shortest path from expert seeker to identified expert.

More details on other prior work on applying authority-based approaches to community question answer sites are given in Section 2.2.2. Among these authority estimation algorithms, there is not a clear winner. Their effects seem to depend on the environment and network structure. However, in general, approaches that are more topic-dependent work better than ones that are topic-independent.

2.1.5 Learning-based Approaches

A learning-based approach which incorporates many features without the need for making any more assumptions is proposed by Fang et al. [30]. The *discriminative learning framework* (DLF) considers expert retrieval to be a classification problem which treats experts as positive data, $P(r = 1|e, q)$, and inexperts as negative data, $P(r = 0|e, q)$, where $P(r|e, q)$ denotes relevancy of an expert candidate e on the given query. Given the relevance judgment $r_{q,e}$ for each training query-expert candidate pair (q, e), which is assumed to be independently generated, the conditional likelihood L of the training data is expressed as:

$$L = \prod_{q}^{Q} \prod_{e}^{E} P_\theta(r = 1|e, q)^{r_{q,e}} P_\theta(r = 0|e, q)^{1-r_{q,e}} \tag{2.27}$$

where Q is the number of queries and E is the number of expert candidates. $P(r = 1|e, q)$ is parameterized by θ, which is estimated by maximizing the above likelihood function. After finding the optimum parameters θ, $P_\theta(r = 1|e, q)$ can be computed directly for each expert candidate. Ranking candidates in descending order of this function provides a ranked list of topic-specific experts.

Similar to generative models (such as *Model 2* [8]), Fang *et al.* [30] estimates $P_\theta(r = 1|e, q)$ by aggregating document query relevance and document candidate association as follows:

$$P_\theta(r = 1|e, q) = \sum_{t=1}^{n} P(r_1 = 1|q, d_t) P(r_2 = 1|e, d_t) P(d_t) \tag{2.28}$$

where $P(d_t)$ is the prior probability of document, which is generally assumed uniform (i.e., $P(d_t) = 1/n$). $P(r_1 = 1|q, d)$ is the probability of document d to be relevant to query q, and

$P(r_2 = 1|e, q)$ is the probability of candidate expert e to be relevant to document d. These are estimated by logistic functions on a linear combination of features as shown:

$$P(r_1 = 1|q, d_t) = \sigma\left(\sum_{i=1}^{N_f} \alpha_i f_i(q, d_t)\right) \tag{2.29}$$

$$P(r_2 = 1|e, d_t) = \sigma\left(\sum_{j=1}^{N_g} \beta_j g_j(e, d_t)\right) \tag{2.30}$$

where $\sigma(x) = 1/(1 + exp(-x))$ is the standard logistic function. In above equations, f and g are two feature vectors indicating document's relevance to query and document-candidate association. The f vector consists of features like document language model, PageRank, URL length and title words, while the g vector includes features such as exact name match, email match, document structure match and proximity. Finally, α and β are the weights of these features learned from training.

This model is referred to as the *arithmetic mean discriminative* (AMD) model [30] due to taking average of $P(r = 1|q, d, e)$ with respect to documents in Equation 2.28. An alternative to this model is the *geometric mean discriminative* (GMD) model [30] which takes geometric mean instead of arithmetic mean.

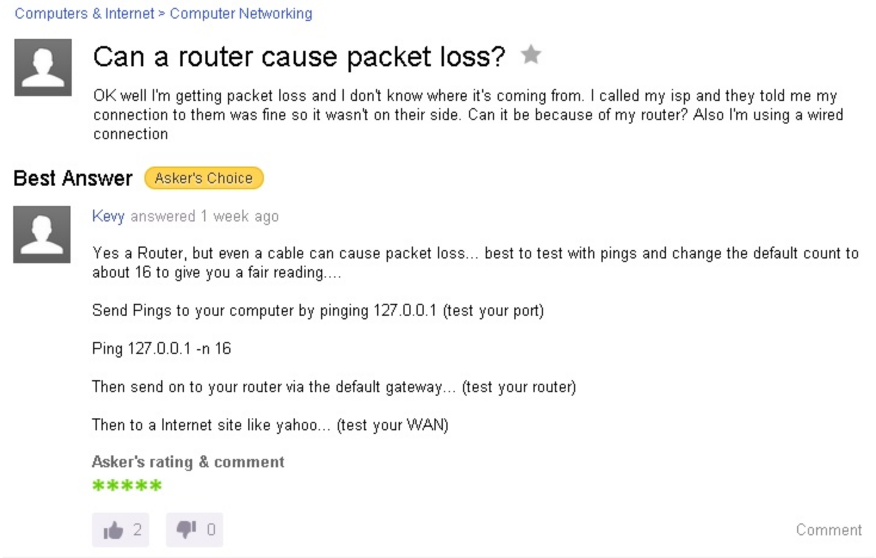
These models are similar to *Model 2* of generative models in terms of aggregating relevant documents with respect to their relevance and document-candidate associations. However, they are different in how they estimate these probabilities. Generative models use language models for estimating document relevance and several heuristics to associate candidates with document. On the other hand, discriminative models apply a logistic function to a set of document and candidate related features to estimate these probabilities. These models consider more evidence than *Model 2* in estimating expertise. However, even though to their solid theoretical ground, these models are not very effective due to the difficulty in estimating the class conditions because of lack of relevance assessments and possibly wrong assumption of $P(r = 0|e, q)$ [10].

2.2 Related Work on Expert Retrieval in CQA

Community Question Answering sites provide an online environment where information seekers can submit their questions which are answered by other members of the community. Resolving problems by getting help from possibly more knowledgeable and experienced people is a common use case of expert retrieval. Therefore, CQA sites can be considered among ideal social media types for expert identification.

2.2.1 User and Information Need Representation

In these environments the information need and user representation is quite different from previously used expert finding collections. Example question and answer threads from two commonly used CQA sites by the research community, Yahoo! Answers and StackOverflow, are given in Figure 2.3 and 2.4. In both systems the questions consist of a title and/or body. The title summarizes the main information need, while body gives the necessary details and can be left out if not needed. The Yahoo! Answers and StackOverflow sites are different with respect to how they categorize the questions. In Yahoo! Answers, questions are submitted into specific



Computers & Internet > Computer Networking

Can a router cause packet loss?

OK well I'm getting packet loss and I don't know where it's coming from. I called my isp and they told me my connection to them was fine so it wasn't on their side. Can it be because of my router? Also I'm using a wired connection

Best Answer Asker's Choice

Kevy answered 1 week ago

Yes a Router, but even a cable can cause packet loss... best to test with pings and change the default count to about 16 to give you a fair reading....

Send Pings to your computer by pinging 127.0.0.1 (test your port)

Ping 127.0.0.1 -n 16

Then send on to your router via the default gateway... (test your router)

Then to a Internet site like yahoo... (test your WAN)

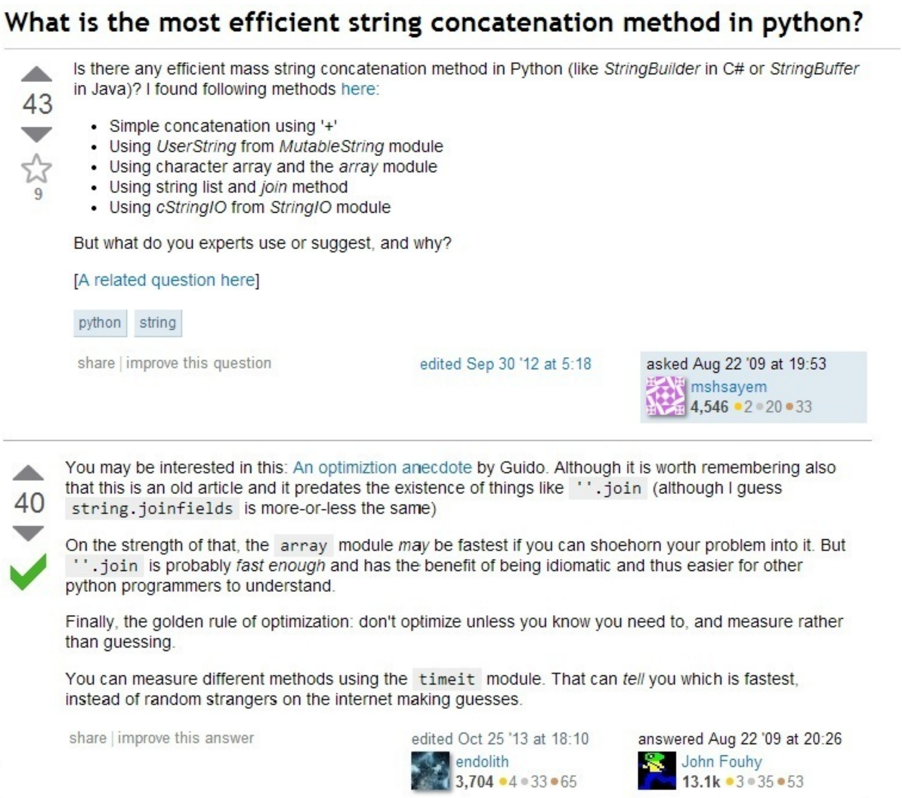
Asker's rating & comment

Figure 2.3: An example question thread from Yahoo! Answers.

and predefined categories, for instance the question in Figure 2.3 is submitted to *Computer Networking* subcategory under *Computers & Internet* category. On the other hand CQA sites like StackOverflow let users to decide up to 5 (user defined) tags to tag their questions as seen in blue boxes in Figure 2.4. Both systems allow information seekers to select an answer as best, and also let users to vote for answers, while in StackOverflow users can also vote for questions.

Different fields of information are used to represent expert candidates in these environments. Most of the prior work constructed user profiles by using either all the answers (and their corresponding questions) [51, 52, 58], or only the question texts of the replied questions [58, 73]. Additionally, some used all the questions users asked and answered [35] to represent expertise. Liu et al. [58] provided a comparison of some of these representations. They constructed user profiles by using either all (or the best) provided answers and their corresponding questions or using all (or the best) replied questions (without answers). Among these different representations, profiles created from using all the answered questions provided the most consistent results.

Depending on the CQA environment, the additional category and tag information was also used to construct more topic-specific representations. Li et al. [52] incorporated the category information of the question and built category sensitive language models of users by using questions and answers from the same (or similar) category of the particular question. Recent work also explored the effects of tags in user representation [21, 97]. Chang and Pal [21] analyzed the effects of using question tags of StackOverflow on topic modeling. The authors argued that question tags are better source for topic modeling due to their keyword like structure and common use by the community members over time. Similarly, Yang et al. [98] incorporated tags to identify topics based on the assumption that using tags within contextual content (answer bodies) can lead to discovery of better user topical interests. Yang and Manandhar [97] also assumed that tags are more representative then the question content while modeling user expertise. They used a probabilistic matrix factorization (PMF) over user-tag expertise matrix to learn the user latent feature space and the tag latent feature space. However, these works did not analyze or report whether the performance improvements are due to the proposed approaches



What is the most efficient string concatenation method in python?

Is there any efficient mass string concatenation method in Python (like *StringBuilder* in C# or *StringBuffer* in Java)? I found following methods here:

- Simple concatenation using '+'
- Using *UserString* from *MutableString* module
- Using character array and the *array* module
- Using string list and *join* method
- Using *cStringIO* from *StringIO* module

But what do you experts use or suggest, and why?

[A related question here]

python string

edited Sep 30 '12 at 5:18

asked Aug 22 '09 at 19:53
mshsayem
4,546 ●2 ●20 ●33

You may be interested in this: An optimiztion anecdote by Guido. Although it is worth remembering also that this is an old article and it predates the existence of things like `''.join` (although I guess `string.joinfields` is more-or-less the same)

On the strength of that, the `array` module *may* be fastest if you can shoehorn your problem into it. But `''.join` is probably *fast enough* and has the benefit of being idiomatic and thus easier for other python programmers to understand.

Finally, the golden rule of optimization: don't optimize unless you know you need to, and measure rather than guessing.

You can measure different methods using the `timeit` module. That can *tell* you which is fastest, instead of random strangers on the internet making guesses.

edited Oct 25 '13 at 18:10
endolith
3,704 ●4 ●33 ●65

answered Aug 22 '09 at 20:26
John Fouhy
13.1k ●3 ●35 ●53

Figure 2.4: An example question thread from StackOverflow.

(topic modeling or PMF) or results of using tags instead of other fields. This thesis expands this previous work by performing a detailed analysis of different representations of users' expertise and information need. In addition to proposing an effective representation of expertise, effective use (weighting) of terms within these representations are also exploited.

The rest of this section describes the prior approaches used for expert finding in CQA sites and how they exploit these different representations of expertise.

2.2.2 Expertise Estimation Approaches

Expert retrieval approaches described in Section 2.1 were also used in CQA communities to identify experts for a given question. Unlike TREC collections, there is not a widely distributed and used dataset for expert finding tasks in CQA sites. The prior researchers mostly constructed their own data from popular CQA sites like Yahoo! Answers or StackOverflow. Even though using their data gives more freedom to researchers, it makes it harder to compare different approaches which have been tested on different data collections.

The prior expert finding work on CQA sites can be also divided to the same categories used in Section 2.1.

2.2.2.1 Graph-based Approaches

The prior work on CQA initially applied graph-based approaches to asker-responder networks in order to identify topic-specific experts within community. As mentioned before, Zhang *et* *al.* [104] applied several network-based algorithms to online help-seeking communities. They proposed *ExpertiseRank*, which is a variation of PageRank approach, to estimate authorities based on responding interactions. They also introduced *Z-score* which is a measure that combines one's asking and responding patterns. They compared these with *Answer Count* approach, which can be also referred to as *InDegree*, and *HITS*. In their dataset, *Z-score*, the simpler method, outperformed the authority-based approaches. A detailed analysis performed on their authority network showed a skew in users' indegree distributions, which probably is the reason why authority estimation approaches could not beat basic count-based approaches. After performing several simulations, the authors confirmed that the performance of graph-based algorithms highly depend on the structure of the network.

Bouguessa et al. [13] also compared the performance of link-based authority approaches with *InDegree* approaches such as *Answer Count* or *Best Answer Count*, in estimating expertise. The authors argue that *PageRank* and *ZScore* approaches work when user interactions are originated from one topic. In case of multiple topics, these approaches may not work as expected. They argue that *HITS* is also not suitable for these environments due to *HITS* being based on mutually reinforcing relationships between users, which does not exist in these environments. We also agree with authors regarding this problem of *PageRank*, and also show that this topic-specificity is also a problem with *HITS*. In order to overcome this problem, we propose constructing more topic-specific networks.

Bouguessa et al. [13] proposed the *Best Answer Count* (based on *InDegree*) approach which reflects the choice of users on provided information being useful and indication of authority. This approach outperformed other approaches in CQA communities, and it has been widely used as a state-of-the-art baseline by much prior work. However, this dissertation shows that constructing test sets by using best reply selection information in CQA communities can return biased test sets which may be also in favor of this approach.

Jurczyk and Agichtein [42] applied *HITS* to question answer communities in order to discover authorities for certain question categories (e.g. *Science*). The *HITS* approach seems to be effective for discovering authorities in topical categories. Jie et al. [40] estimated the reputation of users on a heterogeneous network consisting of links based on actions like reply, vote, accept as best, thumps up/down etc.. They applied a weighted HITS approach with varying weights used for different types of actions.

Extensions to regular authority-based approaches were also proposed and tested in CQA data collections. For instance, Zhou et al. [107] improved upon Jurczyk and Agichtein's [42] *HITS* by incorporating question categories and using topical similarity of question categories as weights during *HITS* calculations. In another work [106, 108], the same authors also proposed a topic-sensitive generative model by combining both link analysis and topical similarity between users to identify experts in CQA communities. They proposed a topic-sensitive random surfer model, called *topic-sensitive probabilistic method* (*TSPM*), similar to *TSPR* and *TPR* approaches developed for web pages. They initially identified topics users are interested in by applying LDA-based approach to user profiles that are constructed from the questions asked and answered by users. These topics are used to estimate topic-specific authorities. The probability of user being interested to a topic is used as teleportation weight. Furthermore the topic-specific similarity between two users is used as transition weight. Their proposed approach outperformed the traditional link analysis approaches.

Zhu et al. [112, 113] proposed extending the authority network constructed from the target category with information coming from its relevant categories in order to estimate *Category*

Relevancy-based Authority Ranking (*CRAR*) of users. Content based and user interaction based category similarities were used initially to estimate the category relevancies. Then, the *Topical Random Surfer* (*TRS*) [67] model was applied to estimate multiple-category-sensitive topical authorities over this extended authority network.

Yang et al. [98] proposed *CQARank* approach, an extension of PageRank, which integrates textual content model to link structure analysis, in order to estimate user topical interest and expertise for different topics. To improve the probability of random surfer visiting user nodes with higher topical expertise and interest, they incorporated the similarity score of users for a certain topic into the transition matrix, and also similar to [106], they incorporated users' topical interest and expertise into the teleportation vector.

Three types of authority networks are widely used for estimating authorities by the prior work. These are:

- *Asker-Replier Networks* (*ARN*): Askers and all responders are used as nodes and edges are directed from askers to corresponding responders [42]. All responders are treated equally in these networks, and best answer information is not used.
- *Asker-Best Replier Networks* (*ABRN*): Askers and only the responders with best selected replies are used as nodes. Other responders whose replies are not selected as best are ignored in these networks. Edges are directed from askers to best selected answers' authors [13].
- *Competition-Based Expertise Network* (*CBEN*): This network can be also referred to as *responder-best responder* network. Only responders are used as nodes, askers are not represented in these graphs. Direction of the edges are from non-best responders towards to best selected responders [3]. There is no self-loop (edges from best responder to best responder itself) in these networks.

Among these networks the last two outperformed the first one, however this dissertation shows some cases of bias which may affect the relative ranking of these approaches that make use of best answer selection information.

2.2.2.3 Profile-based Approaches

Profile-based approaches are also widely used in CQA environments to identify experts. Most of these applied language-based modeling approaches to rank these user profiles. Liu et al. [58] was among the first ones that applied language modeling approaches to user profiles with the aim to identify a list of possible responders for a given question. They compared query-likelihood model, relevance model and cluster-based language models and observed that performances of the systems are not significantly different than each other.

Li et al. [52] incorporated the category information of the question and built category sensitive language models of users by using questions and answers from the same (or similar) category of the particular question. Using these sub-categories information in order to prevent irrelevant questions and answers to be used in expertise estimation resulted in higher accuracies with lower computational cost.

There is also a set of previous works that applied topic modeling approaches in order to explore the latent relationship between terms. Guo et al. [35] discovered latent topics in the content of questions as well as the associated answers, and users' interest. They introduced User-Question-Answer (UQA) Model, a user-centric generative model, which incorporates users' asking and responding patterns by using user-profiles that consist of all the questions they asked and answered.

Riahi et al. [75] compared word-based methods like *TFIDF* and *language models* with topic models, like *LDA* and the *Segmented Topic model* (*STM*), over user profiles are constructed from the corresponding questions of the best selected answers of users. Topic models outperformed the word-based approaches, and the *STM* approach, which allows each question within profile to have a separate distribution over topics, consistently performed better than *LDA* which groups all questions under a single topic distribution.

Yang et al. [98] proposed a probabilistic generative model, called the *Topic Expertise Model* (*TEM*), which jointly models topics and expertise. The authors used the voting information to estimate user topical expertise. For better modeling of user topical interest, in addition to the user created content (answers), they also utilized the question tags.

Chang and Pal [21] worked on the problem of collaborative question routing, in which the aim is to identify a set of users who would collaborate together (either as responding or commenting) to improve the lasting value of the question answer thread. They used answering, commenting and voting propensities of users. For estimating topic expertise they applied three topic models, spectral clustering [66] and LDA [12] on question tags and LDA on question tags and text. Applying spectral clustering on tag graphs outperformed LDA approaches.

Pedro and Karatzoglou [82] proposed a topic modeling approach which takes into account the community feedback. Their model, *RankSLDA*, extends the supervised-LDA model by considering a learning-to-rank paradigm.

2.2.2.2 Document-based Approaches

InDegree approaches like *Answer Count* and *Best Answer Count* can be considered as document-based approaches when applied to topic-specific graphs. In that respect they are very similar to *Votes* approach from *Voting Models* where each instance of document add 1 to the expertise of its corresponding author. An approach that combines these two was proposed by Chen and Nayak [23] called *Best Answer Ratio*, which is equal to the ratio of *Best Answer Count* to *Answer Count*.

Similar to document-based approaches, Zhou et al. [110] introduced thread-based models which consist of question-reply thread with different weights on question and reply (all replies are merged into one). In this model, the top *n* relevant threads are retrieved and each thread contributes to the ranking score of a user based on the association function between the user and the thread (the quality of the reply) which is calculated by using the likelihood of question to the reply. A cluster-based model, which groups threads with similar content (topic) into same clusters, and a similar thread-based model approach are applied to these. In general, thread-based models outperformed the profile-based models in their experiments.

2.2.2.4 Learning-based Approaches

Carefully constructed features were also exploited in feature-based approaches to estimate the expertise of users in CQA communities. Li and King [51] improved the language modeling approaches by using answer quality, which is calculated by using user's average answer quality from similar answers and other similar users' answer qualities. Features like number of answers, length of answer, number of up and down votes, and answerer's best answer ratio were used in estimating users' answer qualities. Using the answer quality and an estimated value of user availability improved the results of question routing.

Zhou et al. [109] proposed a classification based approach which uses global features as well as local features calculated from the category of the particular question. Features representing the question, users' previous activities, and the relationship between particular question and users were used.

Zhou et al. [111] exploited the effects of users' level, engagement and authority related features extracted from corresponding user profiles on the quality of their answers. User engagement and authority related features provided significant improvements in reply ranking.

2.3 Summary

This chapter summarized the main components of expert finding system and identified four group of approaches based on how they model expertise. In *document-based approaches* the retrieved topic relevant documents are aggregated in order to estimate the expertise score while in *profile-based approaches*, the retrieval is performed directly over the candidate profiles. *Graph-based approaches* not only use the user-created content but also explore the link structure between documents and candidates. *Learning-based approaches* train an expertise estimation model that incorporates many other features.

Document-based approaches are more effective and easier to setup over an existing search engine while *profile-based approaches* require separate index to be built. However, *profile approaches* are more efficient due to retrieving and ranking only once. *Graph-based approaches* incorporate the interaction between and among documents and users, however their effectiveness depend on the structure of the graph. Experiments on graphs extracted from TREC collections and available social media environments showed that depending on the structure of the graphs, they may not be as effective as document-based approaches or similarly counting-based approaches like *InDegree*. *Learning-based approaches* incorporate many features, however their effectiveness depend on the amount and quality of training data. In TREC collections these approaches only use the relevance assessments, which are limited; however with the availability of user-generated feedback in social media environments, these models are used more.

Overall these approaches performed very well on organizational documents and some of them have also been applied to social media such as online discussion forums, CQAs and microblogs. Prior work on CQA communities used different data collections which makes it harder to compare approaches across different papers. However, general observations suggest that using topic-specific approaches are shown to be more effective than topic-independent approaches similar to previous expert retrieval research. Approaches that use the category information or information coming from similar categories are shown to be more effective than approaches that don't use that kind of information. Approaches that also explored the best answer selection or the votes have improved performance compared to ones that did not. However, these approaches may be favored due to the possible bias in data collections that were used in testing.

Among all these models, *document-based approaches* are preferred mostly due to their effective performance over *profile-based models* for using topic relevant content of users, easy to apply over existing systems, no need for training data unlike *learning-based approaches*, and not being depended on the graphical network structure of the data.

Chapter 3

The Proposed Expert Retrieval Architecture for Social Media

This dissertation proposes a detailed architecture in which different types of evidence (to be discussed in detail later) can be turned on or off, or combined with other types of evidence for effective expert retrieval. The overview of the proposed expertise estimation architecture is presented in Figure 3.1. This work expands the prior work on expert retrieval by exploiting different types of evidence available in social media environments, which can be categorized into three: (1) different types of user-created content, (2) underlying user networks and (3) temporal metadata.

The first part of the system uses different document content types to estimate the expertise of the associated users. Users can contribute to social media sites in different ways, and some of these contributions can be a better indication of expertise. This dissertation explores available content types in these environments and identifies useful ones for more effective expertise identification.

The second part analyzes the network structure among users. Interactions between users, such as commenting or answering, can be indications of expertise. This dissertation analyzes these user interactions and their use in expert finding through applying network-based approaches to user interaction (authority) networks. Furthermore, effective ways to integrate evidence from user created content to authority estimation[1] approaches are explored in order to identify topic-specific authoritative experts.

The third part of the proposed architecture investigates the available temporal metadata in social media environments. Most user actions are timestamped in these environments. This dissertation combines this temporal evidence with content evidence in order to identify experts who are still interested in the particular topic.

Each of these parts uses different types of evidence or a combination of them in order to estimate expertise. For a given topic, the most likely ranking of expert candidates is estimated as a result of each part[2]. Depending on the social media environment, or the task expert retrieval is used for, either all or only some of these parts can be applied for the final ranking of expert

[1] As described in the Introduction chapter (Section 1.1), in this thesis '*authority estimation*' is used as the network-based method, not the expert finding task. In the literature, these network-based methods have become known as '*authority-based*' approaches, and the same terminology is also adapted throughout the remainder of the dissertation.

[2] Expertise estimated from these parts are referred to as '*content-based expertise*', '*authority-based expertise*' and '*temporal-based expertise*' especially when they are used together. In the corresponding chapters where these evidence types are explored individually, they may be referred to as only '*expertise*'.

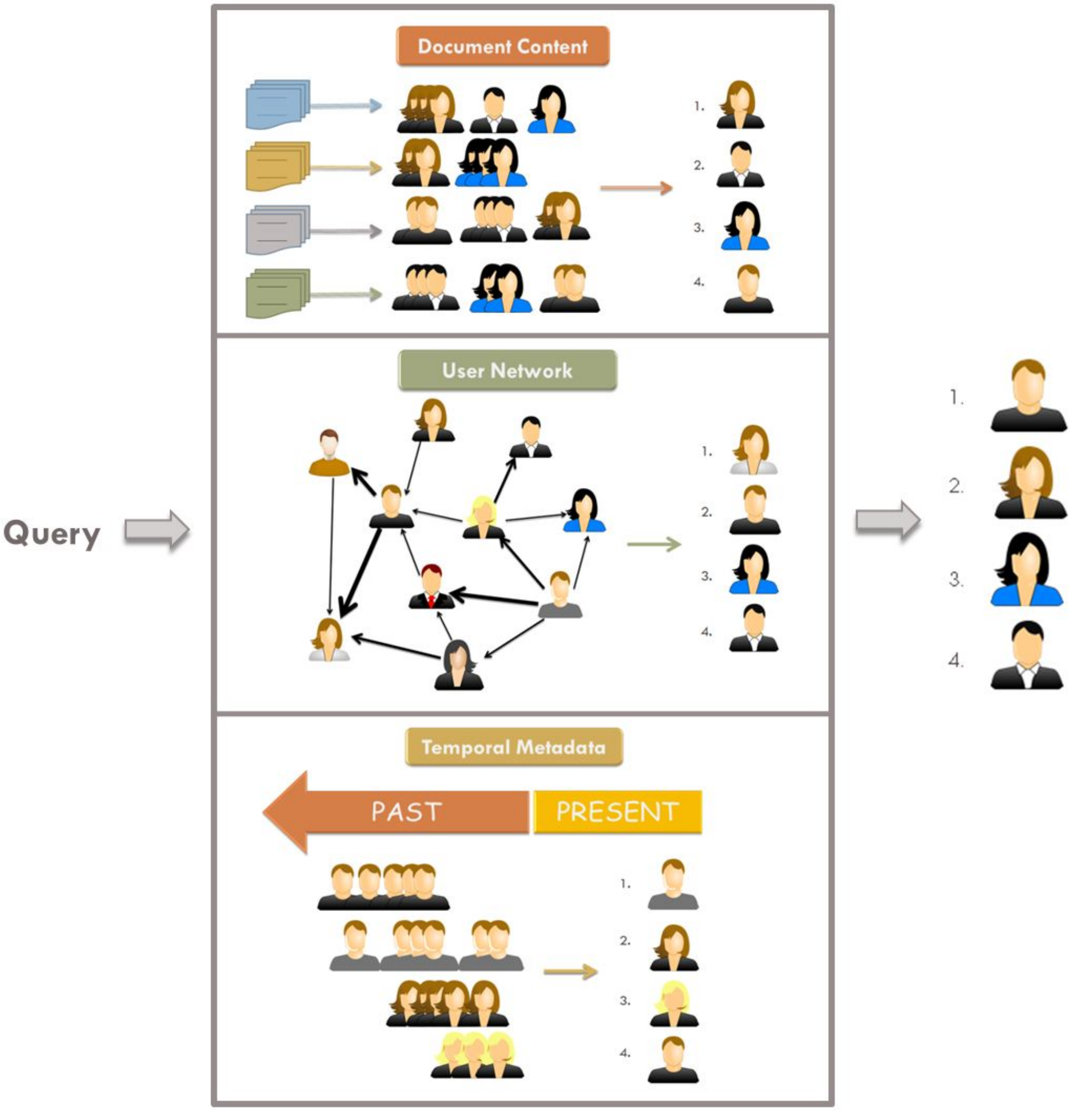
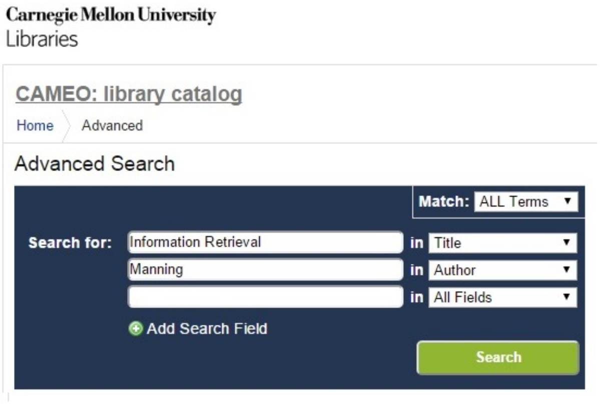


Figure 3.1: Overview of the proposed expert retrieval architecture for social media.

candidates. With experiments performed on two social media data collections for three expertise related tasks, this dissertation tries to construct a general road map for expertise retrieval in social media.

3.1 Content-based Approaches (Document Content)

Content-based retrieval of experts depends on two representations: the information need to be searched and users to be retrieved. There has been a long history of research in information retrieval on identification of the effective representation of the information need and the specific entity to be searched. Using different fields of a document to better estimate how well the query matches the document is something that has been adapted by librarians long time ago. Nowadays, library search engines as shown in Figure 3.2 and domain-specific search engines



Carnegie Mellon University
Libraries

CAMEO: library catalog

Home › Advanced

Advanced Search

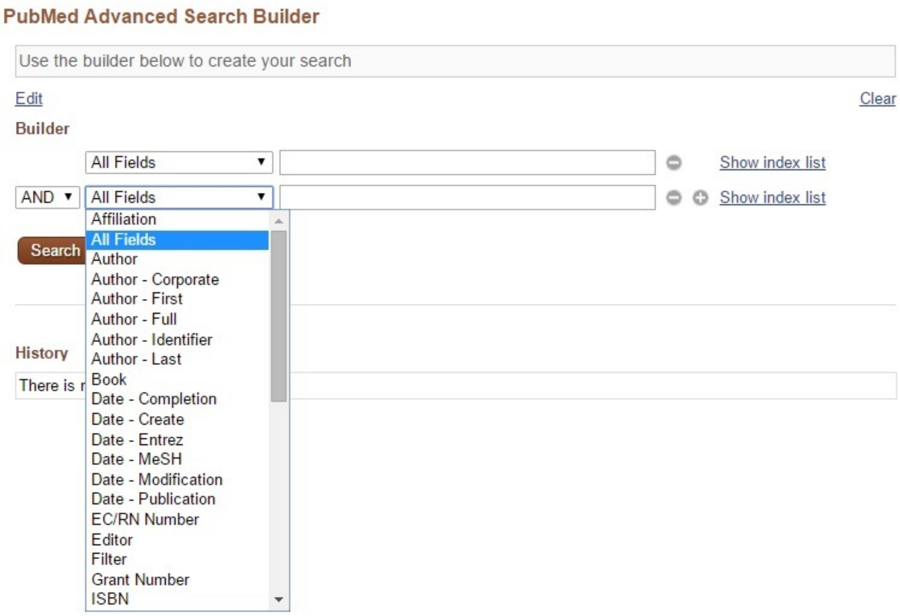
Match: ALL Terms

Search for: Information Retrieval in Title

Manning in Author

in All Fields

Figure 3.2: An example library search engine.



PubMed Advanced Search Builder

Use the builder below to create your search

Builder

All Fields

AND All Fields

History

There is n

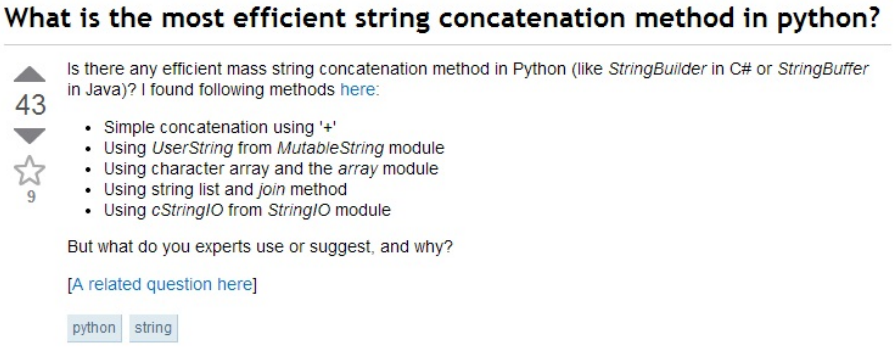
Figure 3.3: PubMed search engine as an example interface for searching different parts of the document.

(like the PubMed in Figure 3.3) enable users to search by different fields of the document.

Social media environments with different types of user-created content are also perfect environments for exploring the effective representation of the users[3]. For instance, CQA sites have different content types like questions, answers and comments. Furthermore, questions in CQAs can consist of title, body and tag fields, as shown in Figure 3.4. Identifying content types or fields which provide better representation of expertise of the associated users, are important for effective expertise estimation in these environments.

The first step in identifying these effective content types for expert retrieval is to understand

[3] In expert search problem, users are represented with the documents they are associated with. Therefore, by referring to effective representation of users, we mean effective use of the content within the associated documents.



What is the most efficient string concatenation method in python?

Is there any efficient mass string concatenation method in Python (like *StringBuilder* in C# or *StringBuffer* in Java)? I found following methods here:

- Simple concatenation using '+'
- Using *UserString* from *MutableString* module
- Using character array and the *array* module
- Using string list and *join* method
- Using *cStringIO* from *StringIO* module

But what do you experts use or suggest, and why?

[A related question here]

Figure 3.4: An example question from StackOverflow.

the underlying reason why that content is constructed and whether that content is an indication of expertise of the associated users. For instance, answers in CQAs are useful content types due to being a strong indication of expertise. Therefore, the prior work on CQAs mostly focused on answering activity as a source of expertise and mostly as the only one. However, the motivation behind commenting on a question or answer hasn't been identified yet, and whether they can be used to identify expertise to route questions to or rank replies has not been explored in detail[4]. This dissertation tries to identify the reasons why users comment on questions and answers, and then depending on the underlying motivation, it proposes to use them in expertise estimation.

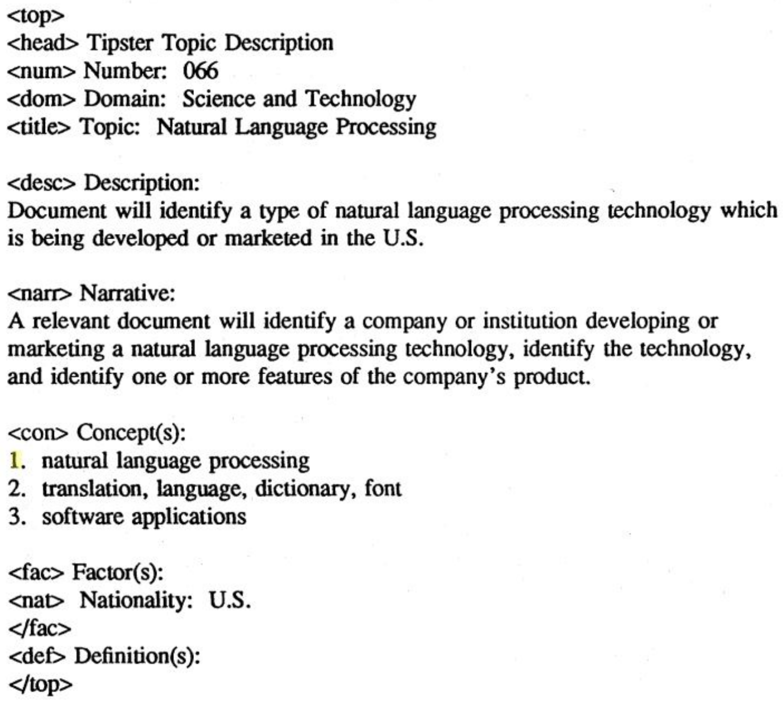
Looking at different ways of combining information from the topic to create better queries has been also studied a lot especially within the TREC community. For instance, the early TREC topics included information needs at different granularities. An example topic from TREC 1 is presented in Figure 3.5. In addition to the title field, which is somewhat similar to queries used in today's web search engines, there is also the description, narrative and concept(s) fields which provide more details regarding the underlying information need.

Questions in Community Question Answering sites also require information in different degrees of granularity (as shown in Figure 3.4) from their users in order to increase their chances of receiving timely and accurate answers. As seen in the Related Work chapter, the prior work mostly worked on developing sophisticated approaches which make use of these fields. They either used the detailed fields with the assumption that they can be better at representation, or they used the fields that best fit to their proposed approach. Only one of the earliest work by Liu et al [58] compared the use of answer bodies and question bodies for representing expertise. They did not analyze other fields within question for either representation of information need or user expertise. This dissertation analyzes all these fields in detail and addresses the following research question

- *RQ1: What are the most effective representations of information need and user expertise used for identifying expertise in question routing and reply ranking tasks in CQAs?*

Analyzing available information types for better retrieval has been applied before for other information retrieval related tasks. In early TREC challenges (TREC-1 and TREC-2), experiments

[4] Chang and Pal [21] used comments in order to identify the most compatible responders and commenters for effective collaborative question routing. They argued that some users tend to answer while some of them tend to comment, and therefore they identified separate lists of responders and commenters. Our work differs from this as we argue that comments can be also used to identify possible expert responders for question routing and reply ranking tasks.



```
<top>
<head> Tipster Topic Description
<num> Number:  066
<dom> Domain:  Science and Technology
<title> Topic:  Natural Language Processing

<desc> Description:
Document will identify a type of natural language processing technology which
is being developed or marketed in the U.S.

<narr> Narrative:
A relevant document will identify a company or institution developing or
marketing a natural language processing technology, identify the technology,
and identify one or more features of the company's product.

<con> Concept(s):
1.  natural language processing
2.  translation, language, dictionary, font
3.  software applications

<fac> Factor(s):
<nat>  Nationality:  U.S.
</fac>
<def> Definition(s):
</top>
```

Figure 3.5: An example topic from TREC-1.

performed over different representations of the topic (as seen in Figure 3.5) provided insights on the retrieval effectiveness of these fields. For instance, the *concept* field which provides a list of assorted concepts related to the topic, was identified to be very useful [48, 59], compared to more detailed fields like narrative or description. This *concept* field was removed in later TRECs to make the task more challenging [16].

Harman [36] defined this *concept* field as *"a mini-knowledge base about a topic such as a real searcher might possess"*. This dissertation also looks for a similar representation of information need for expert search in CQAs. More specifically, for a given question, we try to identify the useful fields which can be used as *a knowledge base about a question such that a real expert might possess to answer the particular question accurately*. These identified fields can be also used for effective representation of the user expertise.

Furthermore, within these searched expertise areas, not all of them can be equally important. For instance in ad-hoc document search, all terms are not equally effective in retrieval, which has been solved by applying term weighting approaches [81]. Similarly within expertise areas to be searched, some of them can be more general concepts, and even prerequisite to others. Therefore, this dissertation tries to weight these identified expertise areas based on several factors such as terms' generality, information seekers' perception, or possible expert candidates' levels of expertise.

3.2 Authority-based Approaches (User Network)

Network-based approaches (also known as authority-based approaches) have been widely used in TREC expert finding collections in order to identify experts through network-based evidence.

Social media environments with rich user interactions, like CQAs, are ideal places to apply authority-based expert finding approaches. As summarized in the Related Work chapter, prior work applied PageRank and HITS like adaptations to the available networks in these environments; however, mixed results were observed. This dissertation tries to shed light on these inconsistent behaviors of authority-based approaches by identifying several problems with the current application of these approaches and also proposes solutions to overcome these. In other words, this thesis tries to answer the following research question:

- *RQ2: Do the assumptions of topic-specific authority estimation approaches developed for web pages hold for user authority networks in social media? For the ones that do not, what kind of algorithmic modifications can be performed so that they hold, and is it possible to make additional assumptions and necessary modifications which can provide more effective and efficient topic-specific authority-based expertise estimations?*

Topic-specificity is one of the issues of user authority estimation [13]. Unlike web pages, which are mostly about a certain topic, users are interested in many topics. In terms of authority graphs, usually web pages are connected to other web pages on similar topics which may lead to construction of graphs with topical clusters of web pages with high intra and low inter topic connections. Estimating authority in these topic-independent networks may still return topic-specific authority scores within those topical clusters. However, in the case of users, the variety of their topical interests return very connected but less topically clustered graphs, which returns mixed authority scores. More topic-dependent authority networks can be constructed with HITS, however due to the same characteristics of users, these user networks may not be as topic-specific as they should be. This lack of topic-dependency in graphs can result in incorrect estimation of authority scores. Furthermore, a less topic-dependent network means more user nodes, and links among them, which increases running times of authority estimation algorithms. Since a topic-specific authority graph is constructed for each given topic and authority is estimated in real time, using larger networks may cause longer running times. Due to these reasons, this dissertation proposes a new type of HITS-like graph construction approach which returns more topic-specific user authority networks than the regular HITS user graphs. With these graphs topic-specific authority estimation can be performed more effectively and also more efficiently.

Authority-based approaches depend on propagation of authority between user nodes. Many topic-specific adaptations of authority-based approaches have been proposed, which uses initially estimated expertise scores to improve propagation of authority towards more expert user nodes. These approaches improve the performance of the authority-based approaches in estimating topic-specific expertise scores, while decreasing the randomness of the walk by increasing the probability of visiting expert users' node. We propose a different topic-specific authority estimation approach which does not affect the random walk property. In original authority estimation approaches, being connected from an authoritative node is an indication of being an authority, similarly, being connected from a topic-specific expert can be an indication of being a topic-specific expert. Based on this intuition, we propose influencing the initially estimated expertise scores of users to other connected users. Compared to other approaches which use initial expertise score of user to improve the authority score of that particular user, our proposed approach uses initial expertise score of a user to improve the authority scores of other connected users.

In addition to making the authority estimation process more topic-specific, this dissertation also analyzes specific user interactions which are commonly used to build the authority graphs. Authority estimation approaches that were initially developed for web pages and url links between them, can cause several problems when they are applied to user networks. The assumptions that hold in web graphs may not hold in some user graphs due to the user interaction used to build them. For instance, in asker-responder networks built from answering action of responders, askers, the origin of the link, are explicitly accepting their lack of knowledge or authority on the particular topic by asking the question. However, in web graphs, giving a link to another page does not have to be an indication of lack of authority of the page on that particular topic. Such a difference in the assumption of the nodes can cause problems in authority estimation approaches, like HITS, which depends on the mutually reinforcing relationship between authority and hub nodes. This dissertation analyzes these problems, and proposes necessary modifications to overcome them.

3.3 Temporal Approaches (Temporal Metadata)

TREC expert finding task aimed to identify experts for a given topic by using the available data collections. These tasks did not provide an user case scenario on what to do with the experts, therefore their evaluations of identified candidates was limited to manual assessment of their topic relevant documents. This kind of expertise assessment of browsing candidate's associated content is acceptable, if user is seeking for experts on a particular topic just to catch up with their topic-relevant content (such as their blog posts). For such a task and evaluation strategy, the prior expert finding approaches assumed the data collection to be a static one, and therefore proposed static approaches to identify expertise.

However, if expert finding is performed to communicate with the identified candidates or to follow their future topic-specific content, then static approaches may not work as expected due to not being able to model the existing and never ending change of users and their topics over time. For instance, users' interests on topics may change over time. At some point in their life, users can be experts on particular topics and may have created lots of topic related content. However, that phase of their life can be over, and they may have moved to something else. At this point, static approaches can still identify these users as experts due to their long time finished topic related activity. This will cause expertise seekers to follow or even contact with these candidates who may not be up-to-date on the particular topic. Contacting with uninterested or not up-to-date experts (or routing questions to them) may not only cause expertise seeker to lose time or receive unsatisfactory answers, but it may also cause unnecessary disruption to the identified expert candidates.

In social media environments, with the availability of timestamped user interactions, such unwanted situations can be prevented at some degree. For instance, as to be mentioned in Chapter 7, timestamps from user activities have been used by prior work [21, 51, 94] in order to estimate the availability of users for tasks like question routing. Estimating availability of users seems to improve the overall performance of question routing, however it does not help with estimating the topic-specific expertise or interest of users at a certain time. This dissertation focuses on these problems and tries to answer the following research question by proposing a temporal modeling of expertise:

- *RQ3: What techniques can be used to identify more up-to-date topic-specific experts who have shown relatively more topic-specific expertise and interest in general and also recently?*

3.4 Summary

This dissertation explores different types of evidence within social media, and proposes a more effective and efficient expert identification system. The proposed expert finding system consists of three parts, and each of these parts focus on different types of information. The first part uses the associated content of authors to effectively retrieve an initial good ranking of experts. The second part estimates authority of users from the authority networks constructed from user interactions. The final part uses timestamps to construct temporal expertise models, which not only models user's expertise but also integrates user's recent interest on the particular topic. Depending on the characteristics of social media, availability of the evidence and type of the expertise related task, some or all of these parts can be used for effective expertise estimation.

Before describing the details of these steps and proposed approaches, the datasets and the experimental methodologies used are explained in the next chapter in order to make the reader familiar with the experimented social media types and expertise related tasks.

Chapter 4

Datasets and Experimental Methodology

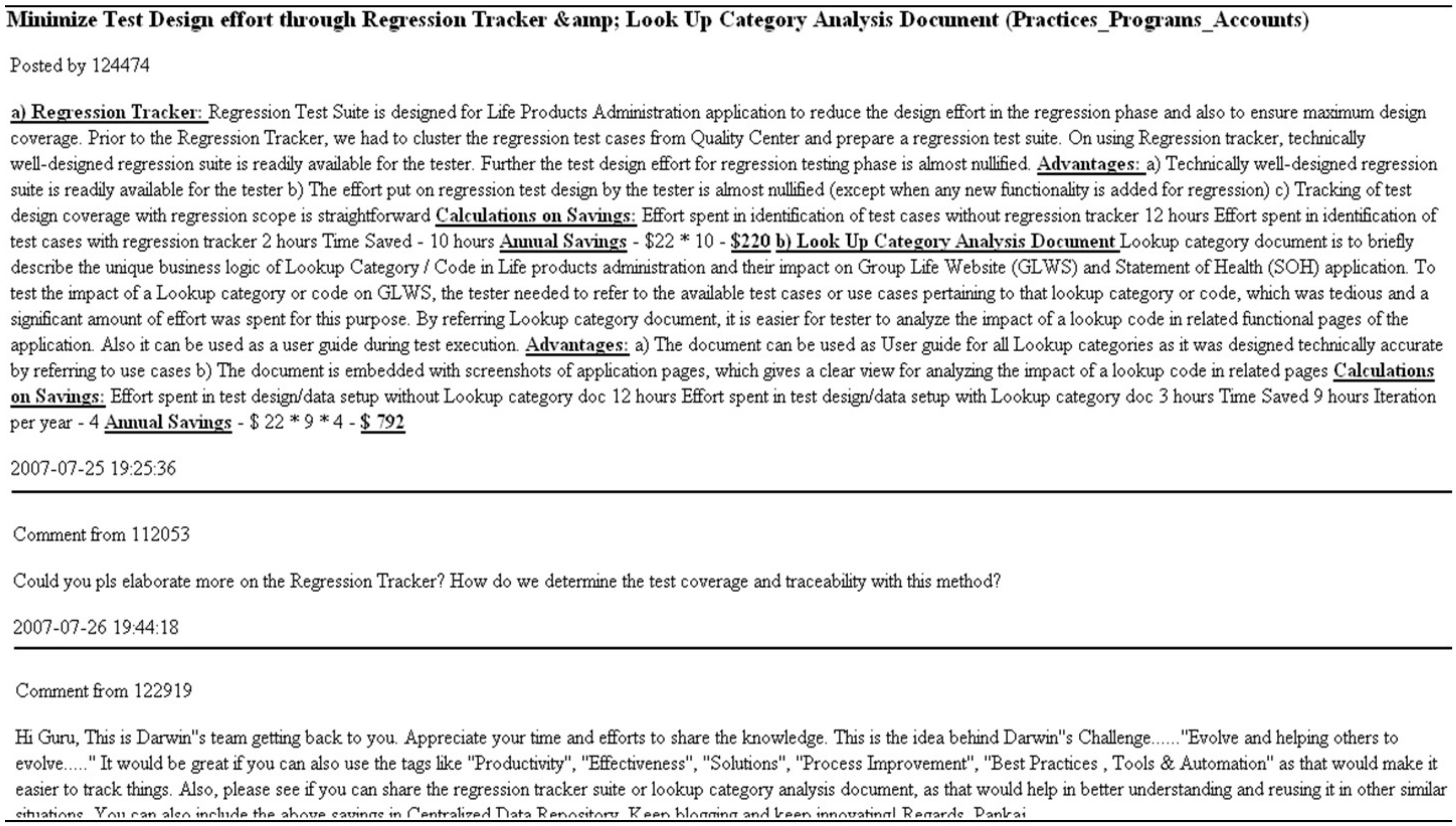
In this dissertation, two types of social media, blog and community question answering (CQA), are used to test the proposed approaches. The blog data used is an intra-organizational data collection. This dataset is just like other social media blogs with posts and comments, and it is also similar to previous expert finding TREC collections mainly due its professional domain and use within organizations among coworkers and peers. TREC's expert finding task, for a given query retrieve a rank list of expert candidates, is applied to this collection. Furthermore, similar assessment and evaluation approaches that have been used in TREC expert finding task are applied to this collection.

CQA sites provide a communication channel between information seekers and providers. They are one of the social media platforms that highly benefit from identification of expert users for a given question. Therefore, a popular CQA site with millions of users from around the world is used in this thesis as another type of social media. Due to the structure and nature of these sites, two types of expertise related tasks are chosen to test the proposed approaches. The first one is routing questions to users who have the necessary expertise on the topic of question, and the second task is ranking replies based on responders' question-specific expertise. The evaluations of these tasks are performed by using the activities and feedback of the actual users of the system.

4.1 The Corporate Blog Collection

Research on how an organization can use its internal social media for locating experts necessarily involves data that is difficult to share widely. Our research used blog and related data provided by a large multinational IT firm. This blog collection has been previously used for research [78, 79, 80]. Although the dataset is not public due to the personal and company-internal information it contains, we believe that it is typical of such datasets. The dataset characteristics are summarized below so that the dataset can be compared to other blog datasets.

The collection consists of blog data (posts and comments) and employee metadata covering a 56-month timespan. An example blog post and comments made to this post are presented in Figure 4.1. A blog post consists of a title and body, while a comment only consists of a body. Average length of the these fields are summarized in Table 4.1. All blog posts and comments are timestamped and have the author information available as seen with the unique ids. This



Minimize Test Design effort through Regression Tracker &amp; Look Up Category Analysis Document (Practices_Programs_Accounts)

Posted by 124474

a) Regression Tracker: Regression Test Suite is designed for Life Products Administration application to reduce the design effort in the regression phase and also to ensure maximum design coverage. Prior to the Regression Tracker, we had to cluster the regression test cases from Quality Center and prepare a regression test suite. On using Regression tracker, technically well-designed regression suite is readily available for the tester. Further the test design effort for regression testing phase is almost nullified. **Advantages:** a) Technically well-designed regression suite is readily available for the tester b) The effort put on regression test design by the tester is almost nullified (except when any new functionality is added for regression) c) Tracking of test design coverage with regression scope is straightforward **Calculations on Savings:** Effort spent in identification of test cases without regression tracker 12 hours Effort spent in identification of test cases with regression tracker 2 hours Time Saved - 10 hours **Annual Savings** - $22 * 10 - **$220** **b) Look Up Category Analysis Document** Lookup category document is to briefly describe the unique business logic of Lookup Category / Code in Life products administration and their impact on Group Life Website (GLWS) and Statement of Health (SOH) application. To test the impact of a Lookup category or code on GLWS, the tester needed to refer to the available test cases or use cases pertaining to that lookup category or code, which was tedious and a significant amount of effort was spent for this purpose. By referring Lookup category document, it is easier for tester to analyze the impact of a lookup code in related functional pages of the application. Also it can be used as a user guide during test execution. **Advantages:** a) The document can be used as User guide for all Lookup categories as it was designed technically accurate by referring to use cases b) The document is embedded with screenshots of application pages, which gives a clear view for analyzing the impact of a lookup code in related pages **Calculations on Savings:** Effort spent in test design/data setup without Lookup category doc 12 hours Effort spent in test design/data setup with Lookup category doc 3 hours Time Saved 9 hours Iteration per year - 4 **Annual Savings** - $ 22 * 9 * 4 - **$ 792**

2007-07-25 19:25:36

Comment from 112053

Could you pls elaborate more on the Regression Tracker? How do we determine the test coverage and traceability with this method?

2007-07-26 19:44:18

Comment from 122919

Hi Guru, This is Darwin''s team getting back to you. Appreciate your time and efforts to share the knowledge. This is the idea behind Darwin''s Challenge......''Evolve and helping others to evolve.....'' It would be great if you can also use the tags like ''Productivity'', ''Effectiveness'', ''Solutions'', ''Process Improvement'', ''Best Practices , Tools & Automation'' as that would make it easier to track things. Also, please see if you can share the regression tracker suite or lookup category analysis document, as that would help in better understanding and reusing it in other similar situations. You can also include the above savings in Centralized Data Repository. Keep blogging and keep innovating! Regards, Pankaj

Figure 4.1: An example blog post and comments from the corporate blog collection.

Field	Ave. Length
Post Title	3.94
Post Body	291.70
Comment Body	24.85

Table 4.1: Average length of fields in corporate blog collection.

# Posts	165,414
# Comments	783,356
# Employees	>100,000
# Posters	20,354
# Commenters	42,169
# Readers	92,360

Table 4.2: Statistics of the corporate blog collection.

dataset also includes access logs - which employees read which blog entries - for 44 of the 56 months. Statistics related to this dataset are summarized in Table 4.2.

Blog posts and comments are on a wide range of personal, social, and work-related topics, for example, poetry, sports, jokes, photography, self-improvement, technology, corporate functions, and testing. A single blog may contain posts on a wide variety of topics. These blogs may also contain organizational spam such as cut-and-paste from documentation or manuals due to incentives for participation that the company offered to employees when the blogs were first deployed.

Employees must login to corporate information systems; therefore users are not anonymous in this environment. All posts and comments created have the authorship information available. Only this information is used to associate posts and comments with corresponding candidates.

The access logs contain the employee ID of a blog post visitor, the date and time of the visit, the URL of the blog post visited, and the employee ID of the author of the blog post. Employees also have access to a corporate blog search engine. We were provided with this search engine's access logs, which contain queries, ids of the employees who performed the search, and timestamps of the search.

4.1.1 The Expert Blogger Finding Task

Due to the similar characteristics of this dataset with previous TREC expert finding collections, this dataset was used for a task very similar to TREC's task, which is identifying expert candidates for a given topic. Evaluation methodologies that had been used in TREC were also used for evaluating expert finding task on this blog collection.

4.1.1.1 Evaluation Data

An initial manual assessment was performed by company employees[1], but the quality (inter-rater agreement) was low. After removing possibly biased assessors and their assessed topics, the inter-rater agreement (average kappa value) was 0.38, which is interpreted as fair [2, 49] or poor

[1] The details of this prior assessment, and the results calculated with values retrieved from this assessment are presented in Appendix A.

[32] by statisticians. Therefore, a second assessment was performed. Due to data confidentiality agreement, this second assessment was performed by the author of this dissertation. In the first assessment, average number of expert candidates assessed for each topic was around 25. In the second assessment, a deeper pool was used and average number expert candidates assessed was increased to 44 on average.

40 work-related topics were created for testing. Some of these were selected from search queries in the access logs of the corporate blog search engine and the rest were created by company employees. The topics from the access logs were selected to mirror task-specific expert-seeking behavior such as 'oracle performance tuning' and 'websphere process server'. On the other hand, topics created by the employees were considerably more general like 'mainframe' and 'cloud computing'.

A sample-based approach was used to create the pool of candidate experts to be assessed. The top 10 candidates returned by several content-based expert-finding algorithms were combined to create a candidate pool. Deeper pools are desirable, of course, but an explicit goal was to produce pools small enough for an assessor to assess a query in less than an hour.

Expert candidates within the pool were anonymized and displayed in random order. For each candidate, the top 3 most topic-relevant posts or comments were displayed during assessments. Expertise was measured on a 4-point scale (not an expert, have some expertise, an expert, very expert) depending on candidate's documents.

4.1.1.2 Evaluation Metrics

In expert retrieval, the top ranked expert candidates are especially important, because the cost of a false positive in expert search is very high. Consulting with a falsely identified expert will not only be time consuming for the expertise seeker, but it will also be an unnecessary disruption for the identified expert candidate. Therefore, early *Precision@n* (*P@n*) metric was used to report the performance in TREC's expert finding task, and also used in this dissertation. *P@n* is calculated as shown for each query and then averaged over all queries.

$$P@n = \frac{|\{expert\ users\}_n \cap \{retrieved\ users\}_n|}{|\{retrieved\ users\}_n|} \tag{4.1}$$

Another metric that has been also used in TREC's expert retrieval task is the *Mean Average Precision* (*MAP*) which has been calculated as follows:

$$\begin{aligned} AP &= \frac{\sum_{k=1}^{n} (P@k\ exp(k))}{|\{expert\ users\}|} \\ MAP &= \frac{\sum_{q=1}^{|Q|} AP(q)}{|Q|} \end{aligned} \tag{4.2}$$

where *AP* is the *Average Precision* score, $|Q|$ is the number of queries, and $exp(k)$ is an indicator function equal to 1 if user k is an expert, 0 otherwise. Both *MAP* and *P@n* metrics are calculated over binary assessments values, they do not differentiate the graded relevance values. Therefore, they are calculated for different levels of expertise.

Normalized Discounted Cumulative Gain (*NDCG*) metric is also used in order to measure the graded relevance of 4-point scale assessment values. *NDCG* metric does not only consider the position of the expert candidate but also integrates the candidate's level of expertise into the calculations as shown [62]:

$$
\begin{aligned}
DCG &= expLevel_1 + \sum_{k=2}^{n} \frac{expLevel_k}{log_2 k} \\
NDCG &= \frac{DCG}{IDCG}
\end{aligned}
\tag{4.3}
$$

where $expLevel_k$ is the graded expertise level of candidate at rank k from a ranked list of candidates from 1 to n, *DCG* is the *Discounted Cumulative Gain*, and *IDCG* is the ideal *DCG*.

4.2 StackOverflow Dataset

Over the last few years, data retrieved from community question answering (CQA) sites are commonly used by the research community. The availability of content, user information, social network structures and some types of manual assessments of content through voting and best reply selection make these collections useful for many research problems. One such widely used CQA site is the StackOverflow[2]. In StackOverflow, the focus is on technical topics such as programming languages and environments, algorithms and operating systems. Users can post questions, answer questions or leave comments to both questions and answers. In most CQA sites, the question and its corresponding answers form a thread, and they are displayed together in the user interface. An example StackOverflow question thread with the question and an answer is presented in Figure 4.2.

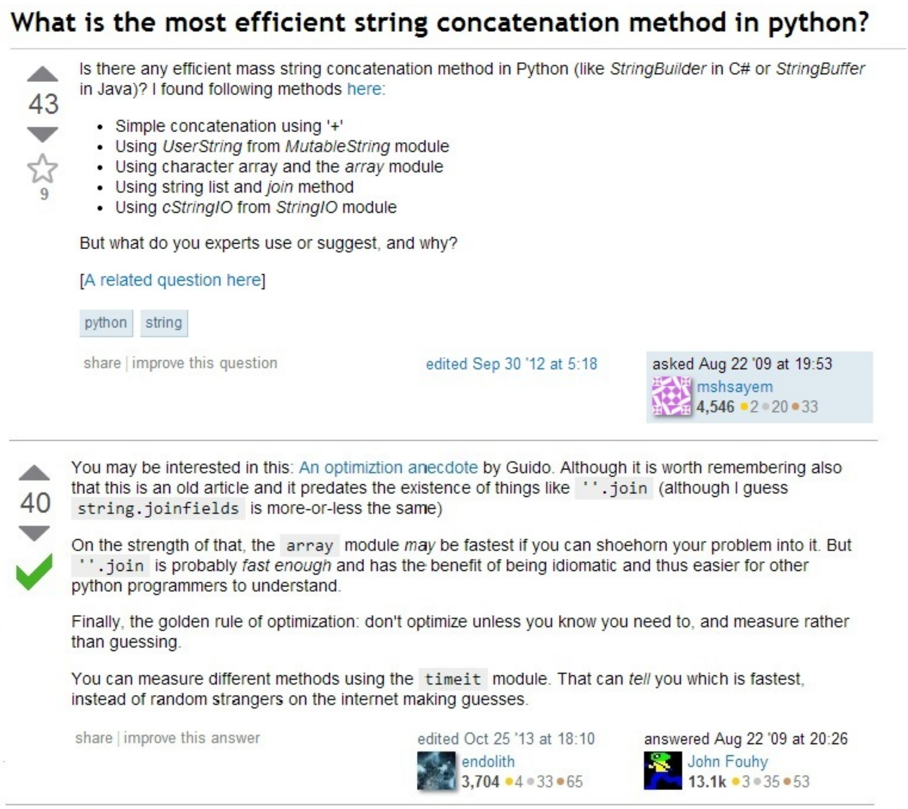
As can be seen from the example in Figure 4.2, a question consists of a title, body and tags within the blue boxes. The title consists of several important keywords, and gives the main idea of the information need. The body is the longest field which explains the information need in detail. The tags do not explain the specific information need of the question but consists of several words or phrases chosen by asker to categorize the particular question. On the other hand, an answer just consists of a body field. Both questions and answers can receive comments from other users in order to either make or ask for a clarification. The average length of these fields is presented at Table 4.3.

Questions, answers and comments in StackOverflow can receive up or down votes from other users depending on the quality, necessity or accuracy of the post. Askers can also select an answer as the best, which is shown with green check mark next to the answer. Furthermore, all posts are associated with their corresponding authors and they all are timestamped.

A public data dump of StackOverflow is used for experiments in this thesis. This collection contains all the posts (questions and answers) and comments made to these posts until May 2014. Statistics related to this collection are provided in Table 4.4.

Routing questions to users who can answer them accurately and ranking replies based on corresponding responders' question-specific expertise are two widely used tasks that require estimation of users' expertise for a given question. In this thesis, the proposed expert finding approaches are also applied and tested with respect to these tasks. Prior work used different experimental methodologies and evaluation metrics for these tasks. In the rest of this chapter, these methodologies will be explained initially and necessary improvements are proposed when necessary.

[2] http://stackoverflow.com/



What is the most efficient string concatenation method in python?

Is there any efficient mass string concatenation method in Python (like *StringBuilder* in C# or *StringBuffer* in Java)? I found following methods here:

- Simple concatenation using '+'
- Using *UserString* from *MutableString* module
- Using character array and the *array* module
- Using string list and *join* method
- Using *cStringIO* from *StringIO* module

But what do you experts use or suggest, and why?

[A related question here]

python string

edited Sep 30 '12 at 5:18

asked Aug 22 '09 at 19:53
mshsayem
4,546 ●2 ●20 ●33

You may be interested in this: An optimiztion anecdote by Guido. Although it is worth remembering also that this is an old article and it predates the existence of things like `''.join` (although I guess `string.joinfields` is more-or-less the same)

On the strength of that, the `array` module *may* be fastest if you can shoehorn your problem into it. But `''.join` is probably *fast enough* and has the benefit of being idiomatic and thus easier for other python programmers to understand.

Finally, the golden rule of optimization: don't optimize unless you know you need to, and measure rather than guessing.

You can measure different methods using the `timeit` module. That can *tell* you which is fastest, instead of random strangers on the internet making guesses.

edited Oct 25 '13 at 18:10
endolith
3,704 ●4 ●33 ●65

answered Aug 22 '09 at 20:26
John Fouhy
13.1k ●3 ●35 ●53

Figure 4.2: An example question and answer from StackOverflow.

Field	Ave. Length
Question Body	94.00
Question Title	8.51
Question Tag	2.95
Answer Body	60.94
Question Comment Body	29.68
Answer Comment Body	30.74

Table 4.3: Average length of fields in StackOverflow collection.

# Questions	7,214,697
# Answers	12,609,623
# Comments	29,226,344
# Askers	1,328,026
# Responders	869,243
# Commenters	1,055,930
# Active Users	1,721,952

Table 4.4: Statistics of the StackOverflow collection.

4.2.1 The Question Routing Task

For a given question, the question routing task returns a ranked list of users based on their relevance to the question. For this task, the top 1000 expert candidates[3] are retrieved for each question among the 869K responders of the site. Either all or some or none of the actual responders of the question are retrieved within these 1000 candidates.

In an ideal environment, evaluating this task can be performed by routing questions to these identified expert candidates, and then manual assessing the accuracy of their answers. However, due to lack of such extended manual assessments, the available data from CQA sites are being used for evaluations. For a given question all the authors of its corresponding replies are treated as relevant while all the other retrieved users who did not post an answer to the particular question are treated as irrelevant. This binary evaluation scheme, even with its flaws, was commonly used by the previous research for question routing task.

Average scores retrieved with this evaluation scheme are normally lower than the average scores from other expert retrieval research [10], due to incomplete assessments. In this task, all the highly ranked users may have the necessary knowledge and background to answer the particular question, but only the ones who actually answered the question are considered relevant while all others are assumed as irrelevant. In StackOverflow, the average reply count is 3.2, which means that for most of the questions there are only a couple of relevant users among the retrieved 1000 users. With such low number of relevant instances, it is hard to see whether the proposed approaches provide any significant changes.

In order to decrease the effects of incomplete assessments, questions with 15 replies were selected during test set construction so that questions have more users assessed as relevant on average. 50 questions with tag counts from 1 to 5 were randomly selected, and total of 250 questions were used in question routing experiments.

The success of question routing task depends on one of the identified highly ranked experts to answer the particular question, therefore the performance is reported with early precision metrics like *Precision@(5, 10, 20)*. Furthermore, the *Mean Reciprocal Rank* (*MRR*) metric is used to analyze the rank of the first identified expert candidate who can answer the question. *MRR* is calculated as follows:

$$MRR = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{expRank(q)} \tag{4.4}$$

where $expRank(q)$ is the rank of the first (top ranked) expert candidate for question q, and $|Q|$ is the number of questions.

Another metric proposed by Chang and Pal [21] is *Matching Set Count* (*MSC@n*), which reports the average number of the questions in which at least one of the users ranked within top n provided an answer to the question. The intuition behind this metric is to analyze what ratio of the questions will be answered, if questions are routed to the top ranked n candidates. *MSC@n* is calculated as shown:

$$MSC@n = \frac{1}{|Q|} \sum_{q \in Q} \mathbf{1}[\{Retrieved\ Users(q)\}_n \cap \{Expert\ Users(q)\}_n \neq \emptyset] \tag{4.5}$$

where $\mathbf{1}[cond]$ is an indicator random variable which is equal to 1 if $cond$ is true, 0 otherwise.

[3] TREC 2005 Enterprise Track [26] also asked for top ranked 1000 expert candidates to be retrieved for expert finding task.

NDCG metric is also used to show the performance in general and to give some sense of relative ranking of responders based on votes they received from their replies, which are used as graded relevance assessment scores of responders.

4.2.2 The Reply Ranking Task

Experimental setting of reply ranking task is rather different than question routing. In this setting, the aim is to rank the responders of the question based on their question-specific expertise. A use case of this task is when information seekers are confused regarding the replies received for their question. If there is no feedback retrieved from other users regarding the accuracy of the provided answers, then knowing the question-specific expertise of responders can be useful to identify the best answer, or rank them based on their author's expertise. Therefore, for this task, expertise needs to be estimated for just the corresponding responders of the given question, not for any other users.

This ranked list of responders is evaluated with respect to votes their replies received. An example StackOverflow question with replies ranked with respect to votes they received is presented in Figure 4.3. In this example, the second answer received more votes than the first answer; however it is ranked after the first one due to not being accepted as the best answer by the asker. Previous research on reply ranking in CQA sites directly used these received votes as graded relevance assessment values. Even though these available assessments values are very practical for evaluation purposes, they may not always reflect the correct assessment value of the content, due to the possible temporal or presentation bias introduced by the CQA system during voting process. These possible biases and their effects on experimental evaluation are analyzed, and a more bias free test set construction approach is proposed in the next section.

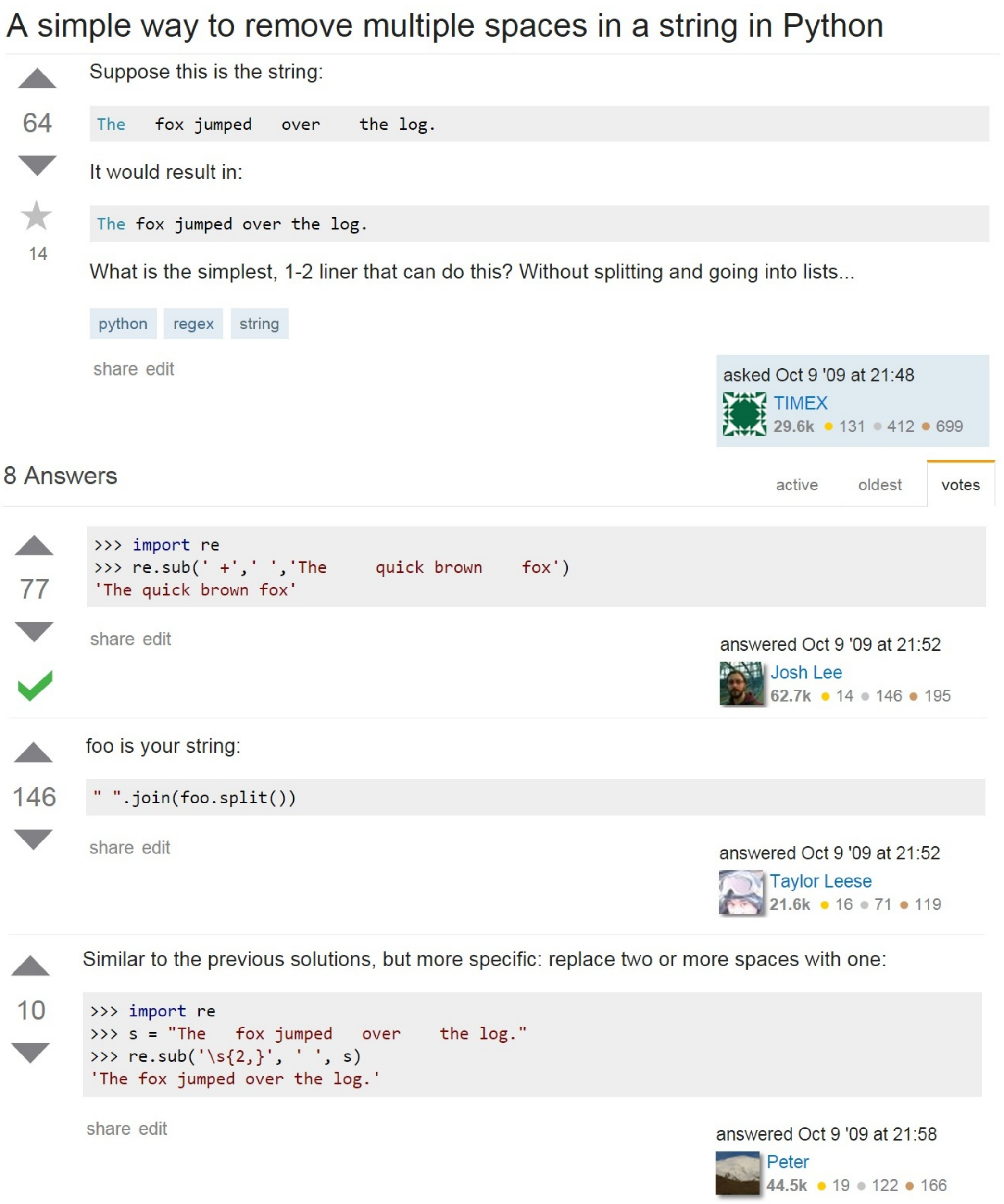
This proposed approach was used to construct the test set for reply ranking task. Questions with exactly 5 replies[4] were chosen in order to see the effects of different approaches more clearly as the relative ranking of these 5 responders change. Similar to question routing task, 50 questions with tag counts changing from 1 to 5 were chosen randomly from the bias free question collection. Overall a total of 250 questions were chosen for reply ranking experiments. These questions were selected among questions with all replies received positive votes in total. Some replies receive negative feedback from users, probably due to being wrong. Questions with such replies are not selected for test set. Furthermore, questions with the most voted reply also accepted as the best reply by the asker were chosen in order to make sure that both asker and other users agree that the same reply is the best.

Due to the graded relevance values of votes, *NDCG* metric is used to evaluate the performance. The *best answer prediction* (*BAP*) measure which is 1 if the top ranked user's reply received the highest vote, or 0 otherwise, is also used.

4.3 Data Preprocessing, Indexing and Retrieval

Both intra-organizational blog and StackOverflow data collections were initially cleaned of all the HTML tags. Stop words were also removed from both of these data sets. Furthermore,

[4] In question routing experiments, questions with 15 replies were used in order to have enough relevant users (responders) for a given question. We don't have that problem in the reply ranking task; therefore questions with 5 replies were used.



A simple way to remove multiple spaces in a string in Python

64

14

Suppose this is the string:

```
The   fox jumped   over    the log.
```

It would result in:

```
The fox jumped over the log.
```

What is the simplest, 1-2 liner that can do this? Without splitting and going into lists...

asked Oct 9 '09 at 21:48
TIMEX
29.6k ● 131 ● 412 ● 699

8 Answers

77

```
>>> import re
>>> re.sub(' +',' ','The     quick brown    fox')
'The quick brown fox'
```

answered Oct 9 '09 at 21:52
Josh Lee
62.7k ● 14 ● 146 ● 195

146

foo is your string:

```
" ".join(foo.split())
```

answered Oct 9 '09 at 21:52
Taylor Leese
21.6k ● 16 ● 71 ● 119

10

Similar to the previous solutions, but more specific: replace two or more spaces with one:

```
>>> import re
>>> s = "The   fox jumped   over    the log."
>>> re.sub('\s{2,}', ' ', s)
'The fox jumped over the log.'
```

answered Oct 9 '09 at 21:58
Peter
44.5k ● 19 ● 122 ● 166

Figure 4.3: An example question and corresponding answers with votes they received.

big code partitions within StackOverflow collection were removed. Krovetz stemming [46] was applied to these datasets.

The Indri search engine[5], which uses a retrieval model combination of language modeling and inference networks [64], is used for indexing and retrieval of expert candidates.

In StackOverflow collection, during indexing and retrieval, only the documents that were created before the time of the asked question were used in order to perform experiments in a more realistic setting.

[5] http://lemurproject.org/indri/

4.4 Parameter Optimization

Some of the proposed approaches require parameters to be set. For such experiments a parameter sweep is performed. 10-fold cross-validation is used to find the optimum parameter setting. The optimum parameter setting is identified by using the median value, and reported together with the experimental results.

4.5 Statistical Significance Tests

Two statistical significance tests; (1) randomization test and (2) sign test, are applied in order to see the effects of the proposed approaches. The randomization test is recommended for many IR tasks by Smucker et al. [87]. However, since it uses the magnitude of the difference between scores, this test may be overly influenced by outliers. On the other hand, the sign test is robust to outliers but may be influenced by many small improvements. In this thesis, both tests are used to draw safer conclusions, and results that are significant with $p < 0.05$ are presented in the tables with r (for randomization test) and s (for sign test) symbols and results which are significant with $0.05 < p < 0.1$ are presented with r' and s' symbols

4.6 Bias Analysis on CQA Vote-based Test Collections

As mentioned before, for the reply ranking task, the previous work used questions as the test queries and the votes assigned to corresponding replies as the ground truth assessments for ranking replies. No additional manual assessment was performed on replies or responders. As for test set construction, they either chose questions randomly or applied several restrictions such as choosing questions from a certain question category [3, 13] or from a certain time frame [13], or choosing questions with responders who have replied at least some number of questions [58].

However, test sets constructed with these approaches may contain several biases which may affect the reliability of the data and the experiments performed over it. Our research investigates two potential sources of bias. The first one is a temporal bias caused by voting activities which are performed before receiving all answers. Another similar bias is the presentation bias which is caused by the initial ranking of the replies, when users only look at the top ranked replies, vote for one, ignore the rest of the replies and leave the page. These biases can lead to assessments of replies without actually viewing all the replies and therefore cause construction of incompetent test sets. These issues are investigated using the StackOverflow collection.

4.6.1 Temporal Bias

In CQA sites, not all the replies are posted at the same time, and systems do not wait for some number of replies to be created but instead display replies of questions as soon as they are posted, and they also let users vote for these replies as soon as they become online. Therefore, a reply may get votes before other replies are posted. We initially analyze whether system letting users to vote for answers before receiving all the answers to the question can be a problem or not. In order to test this, the percentage of replies, votes and (best reply) accepts received within certain time periods after the question is asked are analyzed. The findings are presented in Table 4.5.

	Replies	Accepts	Votes
day 1	81.82%	50.84%	43.38%
days 2-7	5.54%	29.44%	10.33%
weeks 2-4	2.96%	9.12%	3.28%
weeks 5-52	5.79%	8.84%	15.65%

Table 4.5: Distribution of posted replies, received votes and accepted best replies over time in certain time periods starting from the time question is posted.

According to the percentages presented in Table 4.5, most of the answering (more than 80%) is performed within the day questions are asked but 18.18% of the replies are still not provided within the first 24 hours. However, half of the accepts and around 40% of the votes are given in the first day, which indicates that part of the voting and accepting activities may have been performed before all the replies are received. A more detailed analysis showed that among the given accepts and votes, 5.79% of the accepts and 10.72% of the votes were given before the last reply was provided to the question. These percentages reach to 11.10% and 17.09% respectively, when only the accepts and votes from replies of questions with multiple replies are considered[6]. In such a system, replies posted earlier have higher probability to receive more votes or accepts, since they were available to users much earlier than the others. Therefore, data collections retrieved from CQA sites which let users vote for replies before receiving all the replies can be biased towards replies that are posted earlier than the ones posted later.

4.6.2 Presentation Bias

Table 4.5 is also useful for understanding the possible effects of the presentation bias. As can be seen in the table, only around half of the votes are given within the first week a question is posted. The rest of the voting is performed much after, and even around 30% is received after a year is passed from the time question is asked. These votes are probably received from users who are directed to the question's page in CQA site by a search engine as a result of a search query. In such a web search like scenario, users may view the replies in a way like they view the web search engine results; starting from the top ranked reply they look through replies until they find what they are looking for, vote for it and leave the page without going over the remaining replies.

In such a user scenario, presentation order of the replies can affect the overall votes these replies can receive. Most of the current CQA systems use their default ranking algorithms until replies receive votes from users. For instance, StackOverflow ranks replies based on their posting time, which means that when a user clicks on a question with replies that have not been voted yet, the system displays the earlier posted replies in top ranks and the later posted replies following them in order. Such a temporal ranking, together with the only top ranked results viewing behavior of users, may lead the earliest submitted replies to receive more votes than others. In order to check this, the distribution of most voted replies are analyzed with respect to their posting times and presented in Table 4.6.

The distribution of the highest voted reply for questions with 2 to 5 replies (including ties[7]) are presented in Table 4.6. In this table, the column labeled as *first* presents the percentage of

[6] For questions with only one reply, it is not possible to receive any accepts or votes before the last reply.

[7] Therefore, the sum of all percentages in a row is not equal to 100%.

Reply Count	Percentage of Questions with Highest Voted Reply				
	First	Second	Third	Fourth	Fifth
2	65.9%	50.5%			
3	51.5%	41.7%	30.9%		
4	43.6%	34.6%	28.4%	19.9%	
5	37.9%	29.2%	24.9%	19.7%	13.7%

Table 4.6: Distribution of most voted replies (including ties) with respect to their posting times.

highest voted replies among all the first replies within that question category. According to the table, within questions with two replies, the first reply received the highest votes in around 65% of the time, while in an unbiased environment, it should have been around 50%, or distributed more to later replies, since they have the opportunity to improve on the previously posted replies. The decreasing percentage of highest voted replies from earlier replies to later replies in questions with 3-5 replies also indicates that the voting is biased towards replies that were submitted earlier. This biased distribution of votes to earlier posted replies strengthens the proposed hypothesis on presentation bias.

4.6.3 The Effects of These Biases

All these system features and user behaviors favor replies that are posted earlier. In order to check the effects of these biases on received votes, a manual assessment of best reply selection was performed on questions that have either their first or last reply received the highest votes from users. According to our hypothesis, if such voting related biases exist, then questions with first reply voted as highest should get lower agreement score between manual and voting based assessments, compared to questions with last reply voted as highest. This is mainly because, in questions where the last reply received the highest vote, even though with the default ranking of the system, at least one user went through all the replies and voted for the latest submitted reply as best. Since a similar user behavior of viewing all replies is used in manual assessments for expert finding, the agreement between manual and voting based assessments of replies should be higher in last reply voted highest questions than the first reply voted highest ones.

4.6.3.1 Manual Assessments

In the manual assessment, questions that have either the earliest or latest reply selected as best were displayed to the assessors after randomly sorting the replies. Assessors were asked to select the best reply based on its relevancy to the question, accuracy in the information provided and quality in expressing the answer. Assessors used these criteria and chose a reply as best without knowing either the original reply ID, user ID of the author, or reply posting time. In case the assessors believed that several replies are equally similar in terms of their relevance, accuracy and quality, they were given the freedom to choose multiple replies as best.

The assessments were performed by 7 volunteer assessors on randomly selected questions. A total of 300 questions with the highest voted replies equally distributed to first and last replies were used in the assessments. Assessors were given all the questions and asked to assess questions they feel comfortable with and as many of them as possible in half an hour time. In

	StackOverflow Highest Voted Reply	# Manual Agree	# Manual Disagree	Agreement Ratio
Excl.	First	16	20	0.44
Ties	Last	14	10	0.58
Incl.	First	25	24	0.51
Ties	Last	20	13	0.61

Table 4.7: Agreement ratios between manually assessed replies and StackOverflow highest voted replies.

order to increase the number of questions assessed in half an hour period, only questions with 2 to 4 replies were displayed in the assessment system.

A total of 98 questions were assessed by 7 assessors. Before performing any investigation on these assessments, a bias analysis was performed on assessors and their selections in order to make sure that they are bias free. One assessor was identified as a possible biased case where he selected the first reply more than the sum of all the other ranks he selected in randomly sorted 2 to 4 reply questions. This assessor and his assessments were removed, due to the possible bias he may have introduced to the assessments.

A total of 86 questions remained in which 24 of them were assessed by multiple assessors. Among these 24 questions, the assessors agreed on the best reply on 17 of them while had different choices for the rest of the 7 questions. Within these 7 questions, 3 of them were assessed by more than 2 assessors. The majority rule was applied to these 3 questions in order to select the more probably assessment value; however, this did not work on 4 questions which were assessed by only two assessors. In order to make the data more reliable and consistent, these 4 questions were removed from the test set which decreased the number of assessed questions to 82.

Even after removing these questions, there are still questions with multiple assessment values in the test data. These are the tie cases in which user could not select one reply as best and so selected several replies that have the same accuracy and quality. In order to analyze the results more clearly, two sets of results, one that excludes these ties (60 questions) and another one that includes those (82 questions) are provided.

The agreement ratio[8] between these manual assessments and highest voted replies are summarized in Table 4.7. In an unbiased environment, such an assessment should return similar ratios for both first and last replies that received the highest votes. But as can be seen from the Table 4.7, the match ratio of the best selected replies are higher in last reply voted highest questions than the first reply voted highest questions for both with and without tie cases. These results strengthen the proposed hypothesis of bias between the earlier replies and votes assigned to them.

[8] Cohen's kappa is a measure of inter-rater agreement for categorical items. It has been used in expert blogger finding evaluation (Section 4.1.1.1) in order to analyze the agreement ratio between assessors. However it has not been used in here, because there is only one instance in one category (best answer) and all other instances are in the same other category, which is not being the best answer. Instead of Cohen's kappa, the agreement between the best answer voted the highest by StackOverflow users and the best answer chosen by the assessor, is used to calculate the percent agreement ratio.

4.6.3.2 The Effects of These Biases on Expert Finding

In order to analyze the effects of these biases on expert finding related tasks, two commonly applied expertise estimation approaches for CQA environments were applied to test sets constructed with using only questions where either first or last replies received the highest vote.

Two widely used approaches were applied to rank the replies. In *Answer Count (AC)* [104] approach the replies are ranked by the number of topic relevant answers provided by the responder. In *Best Answer Count (BAC)* [13] approach, only the topic relevant answers that are selected as best by the person who posted the question are counted. In previous work, counting only the best selected replies (*BAC*) is shown to be more effective than counting all replies of the responder (*AC*) [3, 13].

In order to see the effects of tested approaches more clearly with respect to questions with a different number of replies, a test set of 350 questions, which consists of randomly selected 50 questions with 2 to 7 replies, was constructed. Apart from this, no other selection criteria or restriction was applied in test set construction. During experiments, the question body was used as the query, and almost all (at most 5000) query relevant replies of the responder were retrieved for each question. The results of these experiments are summarized in Table 4.8.

Reply	First Reply		Last Reply	
Count	AC	BAC	AC	BAC
2-7	0.4467	**0.4700**	**0.3367**	0.3333
2	0.6800	**0.7000**	**0.5400**	0.5200
3	0.6200	**0.6400**	**0.4400**	0.3800
4	0.4600	**0.5000**	0.3200	**0.3600**
5	0.3600	**0.4400**	0.2800	**0.3000**
6	**0.4000**	0.3800	**0.2600**	0.2400
7	0.1600	0.1600	0.1800	**0.2000**

Table 4.8: Reply ranking results for first reply and last reply test sets.

Table 4.8 presents the experimental results of applying *Answer Count* and *Best Answer Count* approaches to test sets in which only the first reply or the last reply get the highest votes. The first row presents the average best reply prediction accuracy results for all 350 questions while the rest of the rows present detailed results for each question category with different number of replies.

In Table 4.8, the *first reply* test set presents a similar behavior as reported in the previous work, *Best Answer Count* performs better than the *Answer Count* approach. This is expected because in randomly constructed test sets, the number of questions where earlier reply selected as best (or receives the highest votes) will be probabilistically higher since the distribution of most voted replies are higher in earlier submitted replies as shown in Table 4.6. Similar results are also observed in questions with changing reply counts.

Unlike the *first reply* test set, the results from the *last reply* test set is different from the previously reported results. First of all, the results in *last reply* set are much lower than the results of *first reply* set, mainly because both algorithms applied are voting based algorithms which favor active users of the environment who are most likely to answer questions much quicker than other less active users, and so have higher probability to be selected as best or voted as highest.

Secondly, the relative ranking of the approaches are also different in *first reply* and *last reply* test sets. On average the *Answer Count* approach slightly works better than the *Best Answer Count* and in terms of questions with different number of replies, there is not a consistent and clear winner. These results are different than *first reply* test set and previously reported results in the literature [3, 13] where there is a clear winner among the tested two approaches. This finding is especially important since such an inconsistency in results makes it hard to compare these two approaches and all other approaches accurately and reliably.

4.6.4 A More Bias-Free Test Set Construction Approach

In order to continue using these CQA data collections and the human assessments coming with them in expert finding related tasks, we propose a new test set construction approach which uses questions where later submitted replies selected as best or voted the highest. As mentioned before, in these type of questions, it is more probable that at least one user assesses all the replies before selecting the best one. Such an approach is more bias-free and also more similar to the construction of controlled manual assessments.

In order to see whether the test set constructed with this approach is similar to the test set constructed with manual assessments, the *Answer Count* and *Best Answer Count* approaches were applied to the manually assessed best answer prediction test set (Section 4.6.3). Among the assessed questions in this test set, only 60 of them do not include any tie conditions and have clear winners, therefore, only these were used in the experiments. The results are presented in Table 4.9.

Reply Count	# Queries	AC	BAC
2-4	60	**0.4666**	0.4333
2	13	0.6154	0.6154
3	27	**0.4074**	0.3704
4	20	**0.4500**	0.4000

Table 4.9: Using manual assessments for reply ranking experiments.

Table 4.9 contains the average results of the experiments over all 60 questions and as well as the more detailed results of questions with different reply counts. According to the table, overall the *Answer Count* approach performs better than the *Best Answer Count* algorithm. Even though the number of questions in this experiment is far lower than the number of questions in the other previously tested test sets, the overall results and the performances of approaches look similar to *last reply* test set experimental results in Table 4.8. Especially, the results of questions with 3 replies, which has the highest number of questions (27), are very similar to the same category results of *last reply* test set (with 50 questions) in Table 4.8, which are 0.4400 instead of 0.4074 in *Answer Count* and 0.3800 instead of 0.3704 in *Best Answer Count* approach.

These results strengthen the proposed idea of using questions where last reply selected as best in test set constructions, since their results are more similar to the results of manually assessed test sets. Of course one should be aware that this test set construction approach is proposed for CQA sites which use the posting time of replies in their default ranking algorithm until any user assessments (like votes) are received for replies. Similar but more customized approaches should be developed for specific CQA sites with different reply ranking preferences.

4.6.5 Summary

This section summarized some preliminary experiments on reply ranking in CQA sites. Using a bias-free test set which is created similarly to the manual assessments is important for the experimental evaluations; therefore, an initial bias analysis was performed on the currently used test set construction approaches.

In our analysis, we have identified two types of biases; temporal and presentation. It has been shown that these biases are affecting the data in a way which is changing the relative performance of the applied approaches. Such a change in the relative ranking of approaches shows the significance of these biases, and their effects on the comparison of tested approaches. Therefore, the following research question is addressed:

- *RQ4: What techniques can be used to construct less biased test collections based on the identified cases of biases?*

In order to decrease the effects of these identified biases and create less biased test collections, we proposed a test set construction approach that is customized for the tested CQA system. Since presentation and temporal biases favor earlier posted replies, we proposed selecting questions where the last posted reply received the highest votes. This is based on the intuition that these replies are better than others due to receiving the highest vote even though all the favoring towards all other replies. This proposed approach was used to create the test set which has been used in the rest of this dissertation for reply ranking experiments.

4.7 Summary

This chapter described the datasets used and the experimental methodologies applied in this thesis. Two datasets are used. The first one is an organizational social media data collection which is very similar to TREC expert finding datasets. Due to its similarity, the same expert identification task and the same assessments and evaluation approaches are applied to this dataset. The second dataset is the data dump of the StackOverflow site. Expertise related tasks like question routing and reply ranking are used to test the proposed approaches with this dataset.

The prior work used users' activities and their feedbacks to evaluate these tasks in these environments. For instance, in question routing experiments, the particular question's actual responders are assumed as relevant while the others are assumed as irrelevant. Similarly, for reply ranking task, the replies are tried to be ranked based on votes they received from users. However, our analyses revealed that the votes replies receive in CQA sites can be affected by their posting time and the presentation order of the site. It has been shown that these temporal and presentation biases cause earlier posted replies to receive more votes compared to replies received later. The experiments indicated that these biases are also affecting the relative ranking of applied approaches. Therefore, a more bias free test set construction approach which uses the questions where the latest submitted reply received the highest vote was proposed based on the intuition that with the temporal ranking of replies, at least one user went through all replies and voted for the last one as the best one.

Now that we have described the datasets and the experimental methodology, the first part of the proposed expert finding system, the content-based approaches, are described in the next chapter.

Chapter 5

Content-based Approaches

In document retrieval, research on what to search for (the query) and where to search (the documents to be searched) has received much attention for effective retrieval as explained in Section 3.1. Similarly, for the expert retrieval task, representing users and their expertise (like the documents), and information need that is to be required from the requested experts (like the query) is very important. Therefore, this dissertation initially focuses on the text-based representations of expertise.

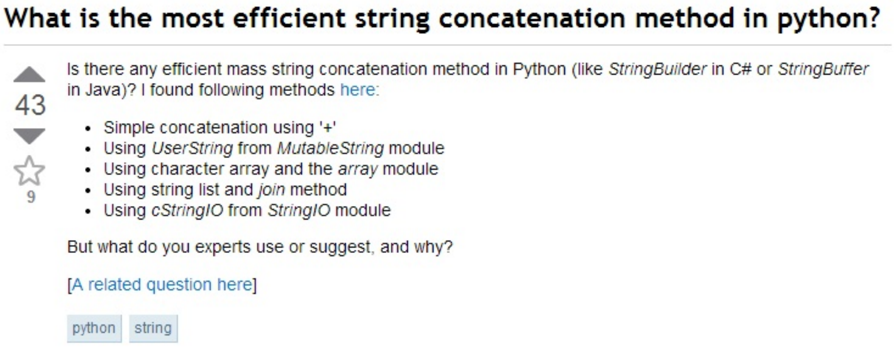
In social media environments, with the availability of different content types, our initial goal is to identify the ones that are more effective in representing and identifying expertise. In community question answering sites, finding effective content-based representations from the existing structure within questions, answers and comments is important for getting a good initial ranking of experts. This chapter analyzes these available user-created content in CQA sites and effective representation of information need and users' expertise for question routing and reply ranking task are identified as stated in the research question (RQ1).

The first part of this chapter focuses on how questions and answers can be used more effectively to represent user expertise and the information need for searching expertise. The second part of this chapter explores comments and how they can be used to improve expertise estimation.

5.1 Representation of Expertise in CQA Sites

Community question answering (CQA) sites consist of different types of user-created content. In these sites, for a given question, previously posted, possibly relevant and similar questions and replies are used to estimate the question-specific expertise of users. In this respect, the prior work mostly exploited the question as a whole with title and body to find similar user-created content with the assumption that it is the best representation of the information need. Even though the question body and title together is the most representative part of the information need, it may also be too specific for expertise search in CQA sites, given that in these environments the only way users can show their expertise on a topic is dependent on other users to ask questions on that particular topic. Therefore, using specific queries in these environments may cause some possible experts to be missed due to not being able to show their expertise on the particular question-specific detail.

This thesis proposes to address this problem by constructing more structured queries which exploit different parts of the questions, such as title and tags fields, in more detail. Title summa-



What is the most efficient string concatenation method in python?

Is there any efficient mass string concatenation method in Python (like *StringBuilder* in C# or *StringBuffer* in Java)? I found following methods here:

- Simple concatenation using '+'
- Using *UserString* from *MutableString* module
- Using character array and the *array* module
- Using string list and *join* method
- Using *cStringIO* from *StringIO* module

But what do you experts use or suggest, and why?

[A related question here]

Figure 5.1: An example question from StackOverflow.

rizes the key points of the information need, which is detailed more in the question body. On the other hand, tags are more generalized categories, which do not explain the specific information need of the question, but instead consist of some prerequisite knowledge areas that are required to answer the particular question. This thesis initially exploits these available fields in order to better represent and identify expertise in CQA sites. Furthermore, in questions, not all tags need to be equally important for identifying expertise necessary to solve the question. Some tags can be more important than others, either due to their order of selection, generality within environment or within possible responders expertise areas. Several tag weighting approaches exploiting these aspects are also analyzed with respect to their representation of the information need for expert search.

5.1.1 Representation of Information Need

Different CQA sites require different type of information during creation of a question. Almost all of them ask for title and body, but some (like Yahoo! Answers[1]) ask the question to be categorized into a predefined category, while some others (like StackOverflow[2]) ask for predefined or user defined tags to be selected for the question. An example StackOverflow question is given in Figure 5.1.

As can be seen from the example in Figure 5.1, a question consists of a title, body and tags within the blue boxes. The title consists of several important keywords from the body and gives the main idea of the information need. The tags do not explain the specific information need of the question but consists of several prerequisite knowledge areas that are required to answer the particular question. On the other hand, the body is the longest field which explains the information need in detail. However, using too much detail for finding experts may not be a good choice for two reasons. First, in CQA environments the only way users can show their expertise is through answering questions. Finding experts for a given question by using the details within the question body may not always return experts especially if the question-specific details were not asked before in previous questions. Second, using too much detail can be misleading sometimes. For instance, in the example question in Figure 5.1, the user is searching for a method in *python* and he gave example functions from *C#* and *Java*, in order to make his point clearer. When this body is used for searching similar previously posted questions and

[1] http://answers.yahoo.com/
[2] http://stackoverflow.com/

replies, these examples may cause retrieval of irrelevant documents on *C#* and *Java*. Therefore, in some cases using the body field for search and using too much information may do more harm than good.

If these information seekers look for answers to their questions through using web search engines, with the commonly used search query construction behaviors, they would probably use keywords selected from title or body of the question as their queries for finding relevant documents. The following terms, *"python effective string concatenation"*, can be used as the query. However, in an expert search engine environment, one may not focus on the title but focus only on the tags, because during expert search the information need of the user changes into something like *"Who are the most relevant responders that can answer a question on strings in python?"*. Such an information need is useful in looking for users who have the necessary knowledge and experience to answer questions related to *"string in python"*. Based on this change in the information need, the query constructed from the tag field of the question, *"python string"*, which contains the general prerequisite knowledge areas experts should have, can be more useful in identifying responders who can reply this question more accurately.

5.1.2 Representation of Users

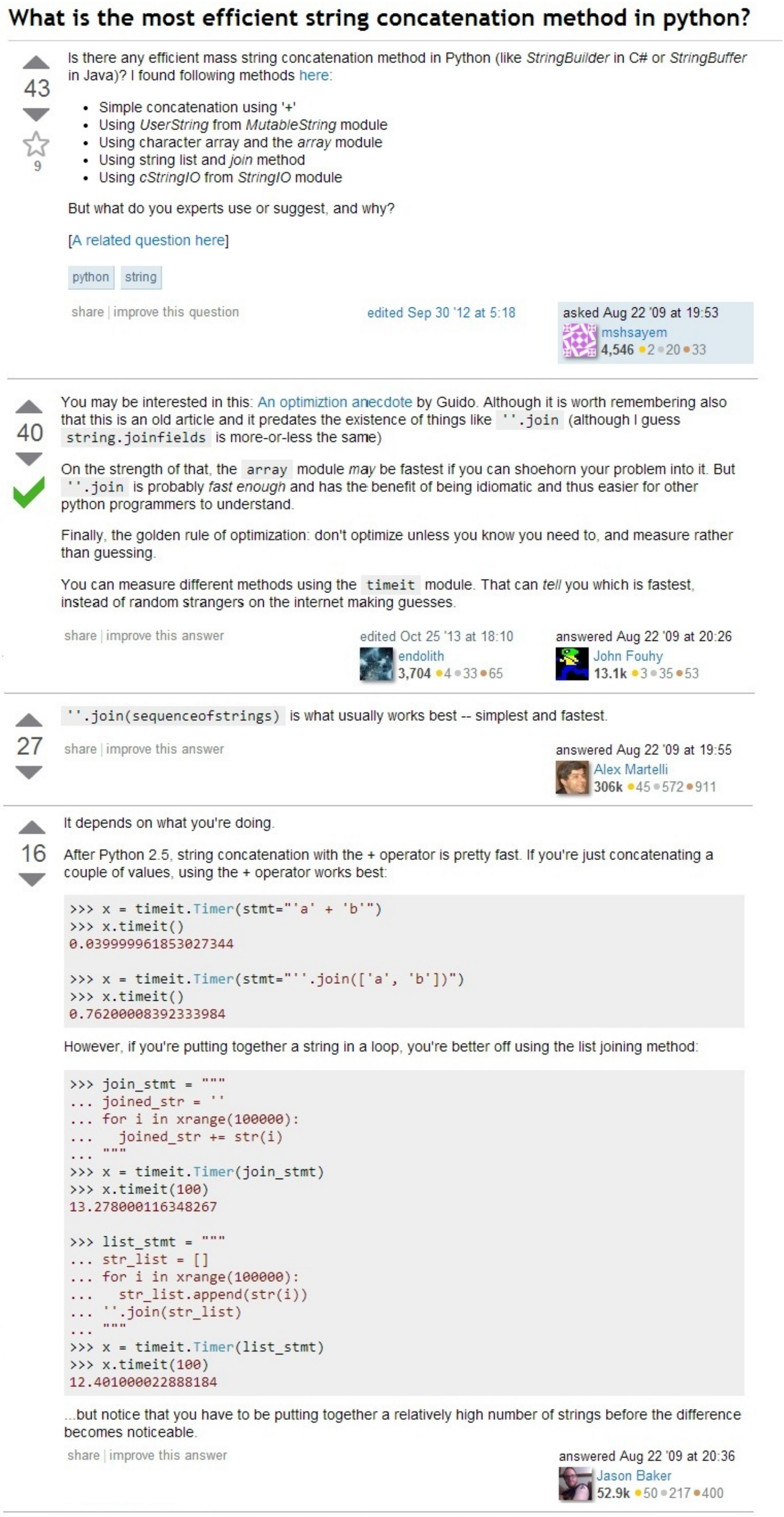
In CQA sites, the question and its corresponding answers form a thread, and they are displayed together in the user interface. An example question thread with the question and corresponding replies is presented in Figure 5.2. As it can be observed from the example replies, due to the thread-like structure, the responders do not feel the need to repeat the question in their answers; instead, they mostly provide the requested information briefly. Because of this common behavior, some key terms from the question is commonly missing from answers. For instance, the second reply in Figure 5.2 does not contain the specific terms *python* or *string*. Therefore, unlike other social media types, using only users' own created content (in this case answers) may not be the best approach to identify responders' expertise areas.

In order to address this problem, some of the prior work [58, 73] used the replied questions' text instead of the replies of the responder for building user profiles. Using the detailed question body to represent user's expertise can also mislead profiles due to the reasons and examples given in the previous section. Therefore, similar to our query construction approach, we also propose building user profiles from the tags of the replied questions.

5.1.3 Constructing Effective Structured Queries

In Sections 5.1.1 and 5.1.2, the question tags are proposed to be used to represent both the users' expertise (the document to be searched) and the main information need of a given question (the query) based on the assumption that they present a generalized category of the expertise. These proposed representations can be further improved by constructing more structured queries. In information retrieval community, much of the research have been focused on improving the representation of a given query, by weighting the query terms or expanding queries with additional terms. Similar approaches are also exploited in this dissertation in order to construct more effective queries.

Using question tags as query terms can be effective, but still all terms within question tags may not be equally important when it comes to assessing expertise of users. In fact certain tags may have more power in determining question-specific expertise. Therefore, several tag weighting heuristics are exploited in order to analyze the effects of provided tags on expertise



What is the most efficient string concatenation method in python?

Is there any efficient mass string concatenation method in Python (like *StringBuilder* in C# or *StringBuffer* in Java)? I found following methods here:

- Simple concatenation using '+'
- Using *UserString* from *MutableString* module
- Using character array and the *array* module
- Using string list and *join* method
- Using *cStringIO* from *StringIO* module

But what do you experts use or suggest, and why?

[A related question here]

edited Sep 30 '12 at 5:18

asked Aug 22 '09 at 19:53
mshsayem
4,546 ●2 ●20 ●33

You may be interested in this: An optimiztion anecdote by Guido. Although it is worth remembering also that this is an old article and it predates the existence of things like `''.join` (although I guess `string.joinfields` is more-or-less the same)

On the strength of that, the `array` module *may* be fastest if you can shoehorn your problem into it. But `''.join` is probably *fast enough* and has the benefit of being idiomatic and thus easier for other python programmers to understand.

Finally, the golden rule of optimization: don't optimize unless you know you need to, and measure rather than guessing.

You can measure different methods using the `timeit` module. That can *tell* you which is fastest, instead of random strangers on the internet making guesses.

edited Oct 25 '13 at 18:10
endolith
3,704 ●4 ●33 ●65

answered Aug 22 '09 at 20:26
John Fouhy
13.1k ●3 ●35 ●53

`''.join(sequenceofstrings)` is what usually works best -- simplest and fastest.

answered Aug 22 '09 at 19:55
Alex Martelli
306k ●45 ●572 ●911

It depends on what you're doing.

After Python 2.5, string concatenation with the + operator is pretty fast. If you're just concatenating a couple of values, using the + operator works best:

```
>>> x = timeit.Timer(stmt="'a' + 'b'")
>>> x.timeit()
0.039999961853027344

>>> x = timeit.Timer(stmt="''.join(['a', 'b'])")
>>> x.timeit()
0.76200008392333984
```

However, if you're putting together a string in a loop, you're better off using the list joining method:

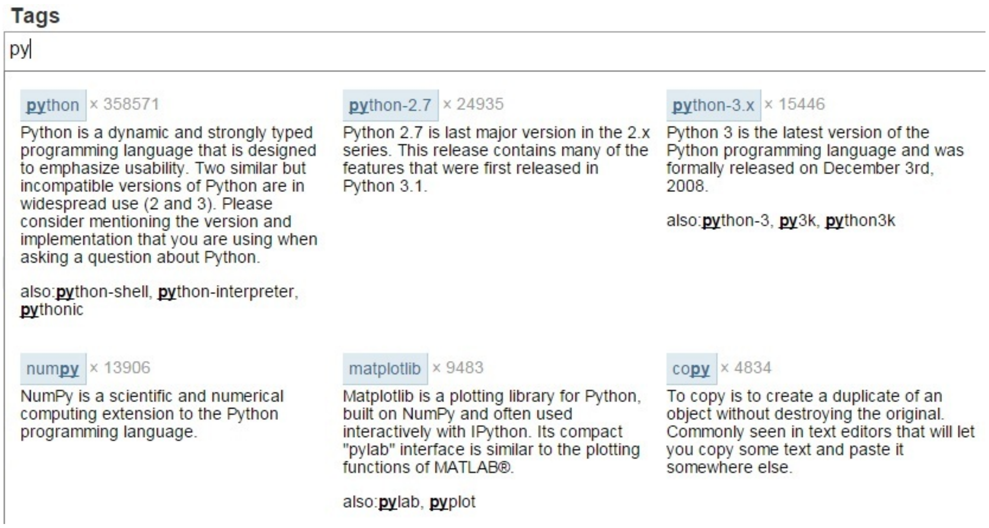
```
>>> join_stmt = """
... joined_str = ''
... for i in xrange(100000):
...   joined_str += str(i)
... """
>>> x = timeit.Timer(join_stmt)
>>> x.timeit(100)
13.278000116348267

>>> list_stmt = """
... str_list = []
... for i in xrange(100000):
...   str_list.append(str(i))
... ''.join(str_list)
... """
>>> x = timeit.Timer(list_stmt)
>>> x.timeit(100)
12.401000022888184
```

...but notice that you have to be putting together a relatively high number of strings before the difference becomes noticeable.

answered Aug 22 '09 at 20:36
Jason Baker
52.9k ●50 ●217 ●400

Figure 5.2: An example question thread from StackOverflow.



Tags

py

python × 358571

Python is a dynamic and strongly typed programming language that is designed to emphasize usability. Two similar but incompatible versions of Python are in widespread use (2 and 3). Please consider mentioning the version and implementation that you are using when asking a question about Python.

also:python-shell, python-interpreter, pythonic

python-2.7 × 24935

Python 2.7 is last major version in the 2.x series. This release contains many of the features that were first released in Python 3.1.

python-3.x × 15446

Python 3 is the latest version of the Python programming language and was formally released on December 3rd, 2008.

also:python-3, py3k, python3k

numpy × 13906

NumPy is a scientific and numerical computing extension to the Python programming language.

matplotlib × 9483

Matplotlib is a plotting library for Python, built on NumPy and often used interactively with IPython. Its compact "pylab" interface is similar to the plotting functions of MATLAB®.

also:pylab, pyplot

copy × 4834

To copy is to create a duplicate of an object without destroying the original. Commonly seen in text editors that will let you copy some text and paste it somewhere else.

Figure 5.3: An example tag suggestion box appeared in StackOverflow.

estimation. We study three types of weighting schemes which are based on different sources of information (1) the asker's information need, (2) the generality of tags over the collection and (3) the responders' expertise areas.

5.1.3.1 Tag Weighting based on Asker's Information Need

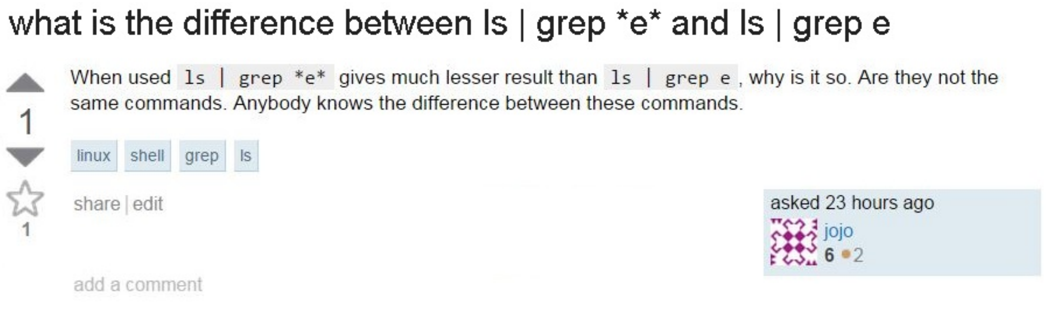
The question tags may look similar to keyword web search queries; however their formation is a little different. When users start typing tags, possible tag matches are displayed to users (as seen in Figure 5.3) in order to help them select among existing tags. Users can choose from one of the proposed tags or create their own tags. All tags are considered to be independent of each other. This gives users the freedom of assigning tags in any order they want without being restricted by any possible syntactic or commonly used ordering of terms.

However, there may exist some other kind of user motivation for the ranking of given tags. Tags are the last information that is required to post a question. Users are asked about the title and body of the question before the tags. Therefore, when it comes to selecting tags, most of the users have already described their information need in detail in the title and body fields. Unlike search engines where users start typing query terms immediately; tags, which also look like keyword queries, are constructed more carefully after detailed consideration of the information need. Additionally, unlike the title and body fields of question, there may be a limit to the number of tags that can be used for a question. For instance in StackOverflow environments, each question can have up to 5 tags. Due to these reasons, it may be the case that users start choosing tags based on their representation of the information need. In other words, more descriptive tags may have been selected earlier than other relatively less significant tags.

In order to analyze whether the order of assigned tags have any effect on the representation of the information need, this thesis proposes weighting tags based on their relative rank within other assigned tags. The following weighting is used for tags.

$$weight_{rank} = log(((N+1) - rank) + 1) \tag{5.1}$$

where N is the number of tags and $rank$ is the relative ranking of the tags. In this approach,



what is the difference between ls | grep *e* and ls | grep e

When used `ls | grep *e*` gives much lesser result than `ls | grep e` , why is it so. Are they not the same commands. Anybody knows the difference between these commands.

Figure 5.4: An example question with frequent and rare tags used together.

tags are initially ranked in reverse order (*reverse rank* = $(N + 1) - rank$), and then these ranks are logarithmically scaled. 1 is added in order to prevent giving 0 weight to the last tag. Example question in Figure 5.1 has tags "*python string*". Applying this approach to weight this query[3] returns log(3) weight for *python* and log(2) for *string* while in the original query, both tags are weighted equally.

5.1.3.2 Tag Weighting based on Term Generality over Collection

The state-of-the-art retrieval models use *term specificity* in order to weight terms according to their frequency within collection. Giving higher relevancy scores to matches of rare terms rather than frequent terms improves the retrieval performance [41]. Therefore, many retrieval models were adapted to compensate for this *term specificity* by using frequency-based measures such as the *inverse document frequency* (idf).

However, this weighting of terms according to their frequency within collection may not always be a useful feature. For instance, in CQA environments where users can show their expertise only through answering other users' question, it can be hard to find experts for very specific tags even though they exist but did not show their expertise on that particular question-specific tag. Therefore, using term specificity in these environments may not return possible expert candidates who have shown their expertise on rarely occurring terms. However, these users may have shown enough evidence of expertise on more frequently used tags, which probably represent a higher-level information need required to provide an accurate reply to the question.

An example question with both frequent and rare tags used together is given in Figure 5.4. At the same day this question was asked[4], the *document frequency* (df) of the provided tags are as follows:

$$\begin{aligned} df(linux) &= 88,465 \\ df(shell) &= 31,014 \\ df(grep) &= 6,041 \\ df(ls) &= 464 \end{aligned} \tag{5.2}$$

The question-specific tag *ls* is a relatively less used tag compared to more popular *linux* and *shell* tags. The *grep* tag is also not very frequently used. Using *term specificity* for this query gives

[3] $N = 2$, and for this specific question *python* is ranked the first while *string* is ranked the second.

[4] Snapshot was taken on November 11, 2014.

more relevance score to users who answered the previous 464 questions on *ls*, however, users who answered the 88K questions on *linux* or 31K questions on *shell* may have enough expertise to reply this particular question even though they did not answer a question tagged with *ls* or *grep*.

In order to decrease the effects of this *term specificity* weighting in different retrieval models, this dissertation proposes using *term generality* as a way to weight tags. Logarithmically scaled frequency, $log(df)$ where df stands for the *document frequency* (number of questions tagged with tag t), is proposed to be used as term weights in order to compensate for possible *term specificity* weightings within different retrieval models. Applying this weighting scheme to the original query "*python string*" returns $\log df_{python}$ and $\log df_{string}$ weights respectively. This weighting can be thought as the reverse of the idf, which is $log(N/df)$ and can be approximated as $-log(df)$ since $log(N)$ is constant. Therefore, for systems that use the standard idf in their ranking, using this *term generality* weighting disables the effects of idf.

5.1.3.3 Tag Weighting by Candidate Expertise

Pseudo (blind) relevance feedback (*PRF*), which performs a local analysis on a probabilistically query relevant part of the collection, is a commonly used approach to improve the retrieval performance. In PRF, the original query is searched within the collection and the top k ranked documents are assumed as relevant and a relevance model is computed over these documents. This approach is mostly used as a first step for query expansion where top t important terms with high relevance value that occur in these top k documents are chosen to expand the original query. PRF is also used as a reweighting approach where the constructed relevance model is used to assign weights to the original query terms.

In terms of expert retrieval, this process can be thought as using the top k retrieved expert candidate profiles[5] to find out what they are experts on in common, and either reweighting the original query terms or adding more terms (expertise areas) which are highly weighted within relevance model. In this thesis, instead of query expansion, reweighting is applied and weights estimated from PRF relevance model are used as weights of the original query terms. Analyzing the top ranked k expert candidates with respect to their common expertise level of the specific query tags may help us to identify the expected level of expertise for each tag that we should be looking for within the collection. This weighting scheme is similar to weighting based on term generality, however instead of using the whole collection, only probabilistically relevant part of the collection is used in this approach.

In this dissertation, an adaptation of Lavrenko's relevance models [50] is used for PRF[6]. In applying PRF, estimating the right value for k is important for the overall performance. Using a high k value may increase the probability of including less relevant or even irrelevant candidate profiles to the relevance feedback model. On the other hand, using very small k may also bias the feedback model towards few candidates. In this thesis, different values for k is experimented with, and the best value of k is chosen with 10-fold cross validation as described in Section 4.4.

After retrieving top k expert candidate profiles, top t terms are selected from the relevance model. If a tag from the question exists within these t terms, then the weight of the term estimated by PRF is directly used as the weight of the particular tag. For tags that are not among the top retrieved t terms, the smallest weight of the retrieved terms, which is the weight of the t_{th} term is

[5] User profiles are constructed for each responder from the tags of the questions they answered.

[6] More details of the applied PRF approach can be found at Don Metzler's PhD dissertation [63].

used as the tag weight, in order to not give those tags 0 weight. A summary of weight assignment protocol is as follows.

$$T = [\ term_1\ term_2\ \ ...\ term_t\ ],\ \ weight(term_i) >= weight(term_{i+1}) \tag{5.3}$$

$$weight_{PRF}(tag) = \begin{cases} weight(term_i), & \text{if } tag = term_i\ \ \&\ \ term_i \in T \\ weight(term_t), & \text{otherwise} \end{cases} \tag{5.4}$$

where T is a ranked list of top t terms retrieved as a result of PRF. These estimated weights, $weight_{PRF}(tag)$, are directly used as corresponding tag weights.

5.1.3.4 Summary

Overall four types of tag weightings are used. In addition to the uniform tag weighted queries, tags are weighted based on asker's information need, tag generality over the collection and candidates' expertise. In addition to these proposed representations of question tags, widely used representations of users' expertise and questions, such as question and answer bodies, are also analyzed for comparison. Two state-of-the-art expert retrieval algorithms, document and profile-based approaches, are used for retrieval with these proposed and baseline representations.

5.1.4 Identifying Experts

The prior work on expert finding in CQA sites mostly used document-based (voting-based), profile-based, or authority-based approaches. This thesis tests the proposed representation of expertise with these different models. The document and profile based approaches is tested in this section, and experiments with authority-based approaches are left to the next section. While testing these content-based approaches, only the content created before the time of the particular question is used for identifying the expert candidates.

5.1.4.1 Document-based Approaches

In CQA environments, *Answer Count* [13] approach, number of answers provided by the responder and retrieved by the query, is used as the document-based approach. *Answer Count* approach is same with the *Votes* approach from voting models [61]. Other voting models, like *Reciprocal Rank* or *CombSUM*, do not work as well as *Votes* approach in CQA sites, because only way users can show their expertise in these environments is through answering other users' unanswered questions. If a question which is very similar to the particular question is already answered accurately, then other expert candidates do not answer the question, which does not make them less of an expert on the particular question.

Two types of indexes are built for document-based approaches. The first one keeps the previously asked questions while the second one keeps the previously posted replies. For a given question, document-based approaches use the question as query to retrieve relevant documents which are later on mapped to the corresponding expert candidates. In answers index, initially the question relevant answers are retrieved, then the associated responders of the retrieved answers are used as the candidate set. On the other hand, documents retrieved from the questions index

Field	Ave. Length
Question Body	94.00
Question Title	8.51
Question Tag	2.95
Answer Body	60.94
Question Comment Body	29.68
Answer Comment Body	30.74

Table 5.1: Average length of fields in StackOverflow collection.

are actually questions that are similar to the particular question being searched. In this setting, all the replies of the retrieved questions are used and their corresponding responders are returned as possible experts. Given a question top 1000 most relevant documents[7] are retrieved and *Answer Count* approach is applied to the associated candidates.

5.1.4.2 Profile-based Approaches

For each user, a user profile (a synthetic document) is built by combining user associated content coming from different parts of CQA posts. For a given question, top relevant user profiles are retrieved and ranked based on their relevance to question as explained in Section 2.1.2.

5.1.5 Analyzing the Dataset

This proposed representation of expertise is tested on StackOverflow collection for question routing and reply ranking tasks. The details of the dataset and the experimental methodologies for these tasks are explained in Section 4.2. In this section, the StackOverflow dataset is analyzed further in order to better understand the characteristics of the collection.

The main idea of the proposed expertise representation model is to use the specific fields in questions and answers to improve both effectiveness and efficiency of the expertise estimation approaches. As shown before, there are three fields in StackOverflow questions, (1) title, (2) body and (3) tags; and one field in answers and comments, the body.

StackOverflow tags are unigram and mostly consist of a single term, but some commonly used phrases are also presented with a hyphen in between terms. For instance, in addition to *memory* tag, there is also *memory-management*, *memory-leaks* or *out-of-memory* tags and so on. Some users use these phrase tags alone, but some users prefer to use, for instance, the *memory-management* and *memory* at the same time for a question.

The average lengths of fields in StackOverflow are summarized in Table 5.1. This table supports the previously made assumptions on the length of fields (Section 5.1.1). On average the *question body* is the longest field which is expected since *question body* is the field where all details of the question are given. It is followed by the *answer body*, which is shorter probably due to thread-like presentation of the site and not including some details from *question body*. Comments made on questions and answers are in similar lengths and, have the shortest body fields. *Question title* is much shorter than *body* while it is still longer than *tag* field [8], which can

[7] This number may seem high, however the actual number of candidate experts can be less than this number due to retrieving multiple documents by the same author.

[8] The length of tag field is calculated by using the number of tags count. If the hyphened phrase tags are separated the length of the tags will be 3.82.

Tag Count	# Questions	% Questions
1	885,854	12%
2	1,866,854	26%
3	2,095,249	29%
4	1,446,812	20%
5	919,928	13%

Table 5.2: Distribution of questions with different number of tags.

contain at most 5 tags. Even though users can use up to 5 tags for their questions, the average is less which indicates that users are more selective in their tag choices. The distributions of questions with 1 to 5 tags are presented in Table 5.2.

Analysis performed on web search engine's query logs revealed that among users' web queries, around 30% of them have 1 term, 30% of them consist of 2 terms while 20% of them have 3 terms [38, 86]. The rest of them have either more than 3 terms or no terms at all. This distribution is similar to number of tags distribution in Table 5.2. The only difference is that in CQA environments users tend to use 1 more term than their usual web queries. This may be due to the intention of categorizing question into a more general concept by using an additional tag, as in the case of using *memory* tag together with the *memory-leaks* tag.

5.1.6 Experiments with Different Representations of Information Need and User Expertise

The following fields are used to represent the information need and user expertise:

- *Representation of the User Expertise:*
  - *Answer Body:* Replies of the responder are used to represent the responder's expertise.
  - *Question Title & Body:* Title and body of the questions answered by the responder are used to represent the responder's expertise.
  - *Question Title:* Title of the questions answered by the responder are used to represent the responder's expertise.
  - *Question Tag:* Tag of the questions answered by the responder are used to represent the responder's expertise.
- *Representation of the Information Need:*
  - *Body:* The question body is used as the query.
  - *Title:* The question title is used as the query.
  - *Tag:* The question tags are used as the query.

These representations are initially tested with the profile-based approach and then with the document-based approach. During these experiments uniform tag weighting (equal weights for all tags) is used.

5.1.6.1 Experimental Results of the Profile-based Approach

Profile-based experimental results of combination of these different representations are summarized in Tables 5.3 and 5.4. The first columns in tables present the user expertise representation

User	Inf.	P	P	P	MRR	MSC	MSC	MSC	NDCG
Exp.	Need	@5	@10	@20		@5	@10	@20	
Answer Body	Body	.0096	.0120	.0084	.0442	.0440	.1000	.1280	.0348
	Title	$.0216^r_s$	.0160	.0108	$.0825^r_s$	$.1000^r_s$	.1360	$.1800^r_s$	$.0518^r_s$
	Tag	.0232	.0184	$.0152^r_s$	.0743	.1000	.1520	$.2440^r_s$	$.0635^r_s$
Question Title & Body	Body	.0272	.0200	.0138	.0829	.1200	.1720	.2200	.0467
	Title	.0208	.0164	.0130	.0726	.0920	.1440	.2080	$.0570^r_s$
	Tag	.0248	.0176	.0146	$.0851^{s'}$	.1160	.1560	.2360	$.0743^r_s$
Question Title	Body	.0256	.0168	.0114	.0832	.1080	.1400	.1800	.0503
	Title	.0312	.0224	$.0196^r_s$	$.0973_s$	.1280	.1840	$.2920_s$	$.0699^r_s$
	Tag	$.0424_s$	$.0360^r_s$	$.0260^r_s$	$.1260^{r'}_s$	$.1840^r_s$	$.2720^r_s$	$.3600^r_s$	$.0951^r_s$
Question Tag	Body	.0160	.0120	.0080	.0449	.0800	.1080	.1240	.0362
	Title	.0176	.0144	$.0134^r_s$	$.0698^r_s$	.0760	$.1120^{r'}$	$.2000^r_s$	$.0609^r_s$
	Tag	$.0592^r_s$	$.0428^r_s$	$.0322^r_s$	$.1639^r_s$	$.2440^r_s$	$.3360^r_s$	$.4520^r_s$	$.1196^r_s$

Table 5.3: Question routing performance of profile-based approach with different fields used for representing user expertise and information need.

User	Inf.	NDCG	NDCG	NDCG	NDCG	NDCG	BAP
Exp.	Need	@1	@2	@3	@4	@5	
Answer Body	Body	.4778	.5720	.6580	.7293	.8043	.1920
	Title	.5112	$.6086^r$	$.6871^r$	$.7480^{r'}$	$.8205^r$	.2240
	Tag	.5371	.6179	$.6961_{s'}$	.7555	$.8265_{s'}$	.2600
Question Title & Body	Body	.5076	.6018	.6825	.7461	.8178	.2360
	Title	.5113	.6119	.6854	.7477	.8206	.2280
	Tag	.5282	.6192	$.7030_{s'}$	.7578	$.8263_{s'}$	.2520
Question Title	Body	.4910	.5761	.6583	.7258	.8075	.2040
	Title	.5233	$.6201^r_{s'}$	$.6898^r_s$	$.7545^r_{s'}$	$.8245^r$	.2400
	Tag	$.5596^{r'}_{s'}$	$.6412_{s'}$	$.7107^{r'}_s$	$.7744^r_s$	$.8362^{r'}_s$	.2640
Question Tag	Body	.4855	.5694	.6343	.7184	.8025	.2080
	Title	.5182	$.6028^{r'}$	$.6842^r_s$	$.7481^r$	$.8205^r$	.2480
	Tag	$.5603^r$	$.6398^r_s$	.7024	$.7675^r$	$.8343^r$	.2720

Table 5.4: Reply ranking performance of profile-based approach with different fields used for representing user expertise and information need.

used in indexing while the second columns show the information need representation used during querying. The rest of the columns are for different metrics used to present the corresponding experiments' results for question routing and reply ranking tasks. The statistically significant results are also indicated with r and s symbols next to the scores. r/s is used for statistically significant randomization/sign test with $p = 0.05$ while r'/s' is used for $p = 0.1$. For the same user expertise representation (first columns), the statistical significant improvements of title queries over the body queries are presented. For using tag as the information need, the results are compared with both title and body query results, and scores that are statistical significant to both representations' scores over the same user expertise are shown in the tables.

For question routing task (Table 5.3), representing users with *tags* of the questions they answered and ranking them by the *tag* field of the particular question outperformed all other representation combinations by returning a ranked list of candidates with at least one actual responder within top 5 retrieved candidates (MSC@5) 24.40% of the time. This number was only 12.80% when only title was used to represent the information need and user expertise. In reply ranking task (Table 5.4), even though the relative improvements may seem small due to re-ranking the same 5 responders with different representations, statistically significant improvements were observed when question tag is used as the query over question title or tag representation of users.

In both tables, using the *tag* field as the query provided statistically significant improvements over using the question *body* or *title* as the query with respect to different user representations. In general, using the question body, which is the most detailed part of the question, resulted with the lowest performance. The question title field, which is less detailed but probably contains the necessary key terms to express the main information need, returned the highest scores among baselines. The questions tags, which do not explain the question but instead categorize it into some prerequisite areas that are required to solve the question, outperformed both title and tag field queries and returned statistically significant improvements over the best baselines in both tasks.

With respect to comparing different user representations, it has been observed that their performances highly depend on the representation of the query used. In both tables, using question *body* as the query returned the best performance when it is applied to *question title & body* user representations, mainly due to similarity of their vocabulary and representation characteristics. Similarly, *title* queries work best with the *question title* representation of users in both tasks. For queries with *tag* field, using *question tag* for representing expertise works best with question routing task. For reply ranking task, performance of *question tag* and *question title* representations are comparable to each other, *title* is generally a little better while *tag* is better at identifying the best responder. Overall, queries work best with user representations that are in similar format, e.g. *body* query for retrieving *body* documents and *title* query for *title* documents.

To sum up, with the profile-based approach, using question tags for representing the questions works best for both question routing and reply ranking tasks. For representing the user expertise, the question tag also outperforms other representations in question routing task. In reply ranking task, both question title and tag representations of users work similarly, but still better than other representations.

5.1.6.2 Experimental Results of the Document-based Approach

The experimental results of the document-based approach with different representations are summarized in Tables 5.5 and 5.6. The first thing to notice in both tables is that compared to the profile-based approach (Tables 5.3 and 5.4), the baseline scores in the document-based approach are quite a bit higher. In both tasks, almost all baselines of the document-based approach outperformed all baselines of the profile-based approach. This difference in baselines shows that in CQA communities, for a given question using the most relevant questions or answers is a good start to estimate expertise of responders, however using all questions or replies (as in the profile-based approach) may cause topic drift which results in lower performance. Overall, these baseline results are yet another example to the outperforming performance of topic-dependent approaches over topic-independent approaches.

Interestingly, the performance difference between the document and profile based approaches gets quite low (the relative ranking is even reversed for some metrics) when the question tag

User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Answer Body	Body	.0464	.0348	.0262	.1476	.1920	.2720	.3800	.0890
	Title	.0400	.0304	.0262	.1347	.1760	.2480	.3920	.0853
	Tag	.0384	.0312	.0248	.1347	.1800	.2720	.3720	.0833
Question Title & Body	Body	.0488	.0388	.0280	.1442	.2120	.3200	.4160	.1113
	Title	.0448	.0348	.0276	.1350	.1960	.2880	.3880	.1077
	Tag	.0424	.0368	$.0314^{r}_{s'}$	.1324	.1800	.2920	.4480	.1113
Question Title	Body	.0336	.0292	.0240	.1099	.1520	.2400	.3520	.0946
	Title	.0336	.0292	.0244	.1230	.1560	.2400	.3720	.1013
	Tag	.0408	$.0368^{r'}$	$.0298^{r}_{s'}$	$.1411_{s}$	.1800	$.2960^{r'}_{s'}$	.4200	$.1130^{r}$
Question Tag	Body	.0248	.0248	.0216	.0944	.1080	.2120	.3040	.0851
	Title	$.0352^{r}_{s'}$	$.0328^{r}_{s}$	$.0268^{r}_{s}$	$.1200^{r}_{s}$	$.1520^{r}_{s}$	$.2800^{r}_{s}$	$.3920^{r}_{s}$	$.0939^{r'}$
	Tag	$.0528^{r}_{s}$	$.0444^{r}_{s}$	$.0344^{r}_{s}$	$.1623^{r}_{s}$	$.2160^{r}_{s}$	$.3400^{r'}_{s'}$	$.4680^{r}_{s}$	$.1307^{r}_{s}$

Table 5.5: Question routing performance of document-based approach with different fields used for representing user expertise and information need.

User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Answer Body	Body	.5627	.6541	.7109	.7660	.8377	.2760
	Title	.5631	.6576	.7172	.7698	.8398	.2800
	Tag	$.5935^{r'}$	.6616	.7196	.7757	.8454	$.3280^{r}_{s}$
Question Title & Body	Body	.5595	.6431	.7047	.7649	.8345	.2800
	Title	.5406	.6443	.7103	.7649	.8320	.2440
	Tag	.5661	.6509	.7142	.7725	.8383	.2920
Question Title	Body	.5527	.6363	.7063	.7597	.8323	.2800
	Title	.5652	.6371	.7155	.7687	.8361	.2720
	Tag	.5661	.6520	.7235	.7745	.8396	.2800
Question Tag	Body	.5566	.6413	.7075	.7662	.8344	.2840
	Title	.5500	.6177	.6947	.7573	.8273	.2640
	Tag	.5670	.6380	.7150	.7728	.8366	.2720

Table 5.6: Reply ranking performance of document-based approach with different fields used for representing user expertise and information need.

is used as the query to retrieve expert users which are represented with either question title or tag. This indicates that while comparing different approaches the representation used is very important, for some representations the performance difference can be significant while for others they can be in similar ranges and more comparable.

In Table 5.5, the best baseline is retrieved when question body query is searched over question body & title documents. As mentioned before, this may be due to retrieving very similar questions asked before. For user expertise representations built from body of the answers or questions, using the body of the question as query performed better than using tags, probably due to the vocabulary differences between these representations. For the rest of the user expertise representations, using tags as the information need returned statistically significant improvements over using title or body fields for querying across most metrics. Using question tags for both information need and user expertise representation outperformed all other representations and also provided statistically significant improvements over the best baseline (question body query searched over question body & title documents) across *P*@20 and *NDCG* ($p = 0.05$) and *MRR* ($p = 0.1$) metrics.

In Table 5.6, the best baseline is using question body to retrieve similar answers (first row in the table). In reply ranking task, the best performance being the one using the responders' replies is expected since the previous replies of users are important for ranking their replies to the particular question. In this setting, similar to question routing task (in Table 5.5), using tags provided improvements (even statistically significant for NDCG@1 and BAP metrics) over the best baseline which shows the effectiveness of using question tags to represent information need. In other user expertise representations, using question tags as query also provides small improvements over other query representations.

Overall, in both the profile and document-based approaches using questions tags as the information need returned statistically significant improvements over the best baseline information need representations. For question routing task, using question tags to represent user expertise also resulted in the best performance among other user expertise representations. Furthermore, with tag representation of user expertise, the profile-based approach (a topic-independent approach) returned scores that are very comparable to scores retrieved with the document-based approach (a more topic-specific approach), and even outperformed the document-based approach for some metrics in question routing task. This shows the power of effective representation in retrieval. Furthermore, increasing the performance of the profile-based approach to the level of the document-based approach is important in terms of the computational efficiency. As mentioned in related work section, one of the advantages of profile-based methods over document-based approaches is them being more time efficient. Even though efficiency is not the main focus of this dissertation, improving it (or not making it worse) is something that we care about for the applicability of the proposed ideas in real time environments.

5.1.6.3 CQA sites without Question Tags

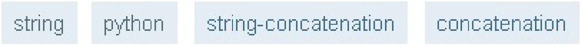
Tags are shown to be useful in representing the information need and user expertise for expert retrieval in community question answering sites. However, some CQA sites, like Yahoo! Answers[9], do not ask for question tags. For environments without tags, using the question title alone can be a solution, especially for profile-based approaches. In Tables 5.3 and 5.4, among the scores without using tags either in representing users' expertise or information need (the rows except the following ones: every third row and the last three rows), using title for representing information need and user expertise outperformed other representations with body field most of the time. Therefore, in case tags are missing, titles can be used for effective expert retrieval in these environments.

Similar to tag fields, title fields may also contain some important key terms and unlike body fields they do not contain too much detail which may prevent possible topic drift. However, most of the titles in StackOverflow are in question sentence format. They are constructed based on some language specific syntactic and semantic rules, therefore they may still contain some functional words which are not helpful for representation of information need and expertise.

One way to improve the effectiveness of title fields is through identifying tags (terms explain-

[9] Yahoo! Answers asks their users to categorize their questions under some predefined hierarchical categories.

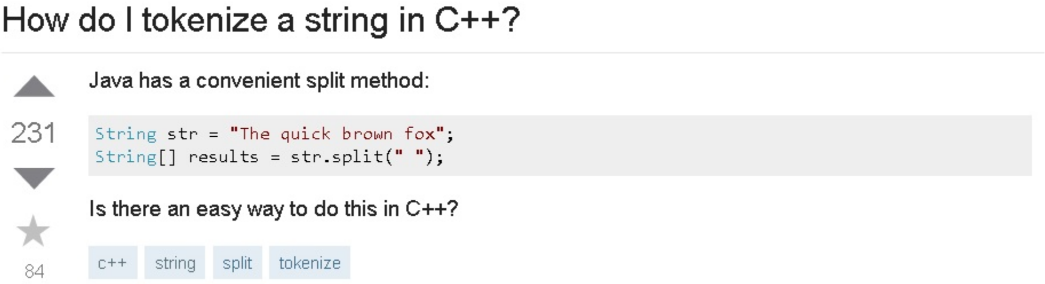
ing general knowledge areas of the information need of questions) within titles and using them as query. Tag identification is yet another task researchers have been working on [47, 74, 77, 93]. In this thesis, our focus is not to identify tags for questions. But, in order to see whether tag identification from title can be helpful for expertise estimation, a very basic matching approach is applied. All StackOverflow tags are searched within title fields, and the lexically matched ones (exact match over roots) are used as the query[10]. For instance, for the example question in Figure 5.1, the following tags are matched from the existing StackOverflow tags.



string python string-concatenation concatenation

Figure 5.5: Lexically matched and identified tags for question in Figure 5.1 from its title.

As observed, our exact matching approach matches (ordered) phrases as well as terms. The identified tags in this case are even more detailed than the user selected tags "*python*" and"*string*". Of course, this may not be the case always. For instance, for the question in Figure 5.6, the identified tags from title do not include the "*split*" tag which was selected by the user. However, 3 out of 4 tags were identified correctly with exact match approach. The effectiveness of using tags identified from question titles are compared with using titles directly (baseline) and the upper bound which is using the actual tags (user selected ones). The experimental results are presented in Tables 5.7 and 5.8, respectively for question routing and reply ranking tasks.



How do I tokenize a string in C++?

Java has a convenient split method:

```
String str = "The quick brown fox";
String[] results = str.split(" ");
```

Is there an easy way to do this in C++?

c++ string split tokenize

Figure 5.6: An example question for identification of tags from a question title.

For the question routing task, in Table 5.7, using matched tags within title as queries with the profile-based approach (the upper half of the table) provided statistically significant improvements over using titles as the query. However, the performance is still lower than using the user selected tags as the query, which means that there is still room to improve. Improvements are also observed with the document-based approach (the lower half of the table), even though they are not always statistically significant. This difference between the document and profile based approach can be explained by the differences in their retrieval models. Since the document-based approach retrieves documents initially, using only the possibly important terms from the title (and ignoring function terms which are mostly among stop words) may still return the same documents which results in similar performance. Retrieving profiles, which are concatenation of many documents, may be affected from changes in queries more than single documents.

Using extracted tags improves question routing, but does not improve reply ranking as seen in Table 5.8. For reply ranking, when user-selected tags are not available, it is best to use the title

[10] The identified tags are weighted equally during querying.

Algorithm	Information Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Profile-based	Title	.0312	.0224	.0196	.0973	.1280	.1840	.2920	.0699
	Matched Tags	$.0384^{r'}_{s}$	$.0300^{r}_{s}$	$.0242^{r}_{s}$	$.1171^{r'}_{s}$	$.1560_{s}$	$.2240^{r'}_{s}$	$.3360^{r'}_{s}$	$.0783^{r}_{s}$
	Actual Tags	.0424	.0360	.0260	.1260	.1840	.2720	.3600	.0951
Document-based	Title	.0336	.0292	.0244	.1230	.1560	.2400	.3720	.1013
	Matched Tags	.0320	.0316	$.0276^{r'}_{s'}$	.1250	.1440	.2600	.4040	.1029
	Actual Tags	.0408	.0368	.0298	.1411	.1800	.2960	.4200	.1130

Table 5.7: Question routing performance with title field used for representing user expertise, and title or identified tags within title used for representing the information need.

Algorithm	Information Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Profile-based	Title	.5233	.6201	.6898	.7545	.8245	.2400
	Matched Tags	.5077	.6124	.6814	.7483	.8192	.2080
	User Selected Tags	.5596	.6412	.7107	.7744	.8362	.2640
Document-based	Title	.5652	.6371	.7155	.7687	.8361	.2720
	Matched Tags	.5471	.6289	.7093	.7640	.8304	.2600
	User Selected Tags	.5661	.6520	.7235	.7745	.8396	.2800

Table 5.8: Reply ranking performance with title field used for representing user expertise, and title or identified tags within title used for representing the information need.

or body.

5.1.6.4 Summary

Statistically significant improvements were received with using the *tag* field in both question routing and reply ranking. These improvements show that in CQA environments using important key terms summarizing the information need or presenting prerequisite knowledge areas required to answer the question are more useful in estimating expertise rather than using the specific question itself as the query. In case *tag* field is not available, using *title* field or identified keywords from *title* can be also used for effective expert identification.

Regarding our research question on identification of effective representations of user expertise and information need for question routing and reply ranking tasks in CQAs, using questions tags is effective independent from the retrieval algorithm used.

Now that we know what to search on where, we focus on improving the retrieval accuracy by weighting the query terms with respect to the underlying information need that is required from the requested experts.

5.1.7 Experiments with Weighting the Question Tags

Overall, question tags are shown to be effective in expertise estimation for a given question. Some of these tags can be more useful than others. In order to analyze this, three tag weighting approaches are tested.

- *IN-Tag:* Tags are weighted based on askers perception (ordering) of the information need.

- *TG-Tag:* Tags are weighted by their generality (log scaled document frequency) over the collection.
- *CP-Tag:* Tags are weighted based on expertise levels of possible expert candidates. Top ranked *n* candidate profiles (*CP*) are used for re-weighting these tags.

These weighted queries are compared to the equally weighted (uniform) tag queries, which is represented with *U-Tag* in short.

Indri query language[11] is used to construct these weighted and structured queries. An example original (uniform) query and corresponding weighted queries in Indri query language are as follows.

- *U-Tag*

$$\#combine[tag](\ python\ string\ ) \tag{5.5}$$

- *IN-Tag*

$$\#weight[tag](\ \log(3)\ python\quad \log(2)\ string\ ) \tag{5.6}$$

- *TG-Tag*

$$\#weight[tag](\ (\log df_{python})\ python\quad (\log df_{string})\ string\ ) \tag{5.7}$$

- *CP-Tag*

$$\#weight[tag](\ weight_{PRF}(python)\ python\quad weight_{PRF}(string)\ string\ ) \tag{5.8}$$

In Indri query language *#combine* is a probabilistic *AND* and *#weight* is the weighted probabilistic *AND* operator. In weighted queries, a term t is preceded by its corresponding weight $weight_t$. [*tag*] is restricting the Indri search engine to use only the tag fields during retrieval. In the original query, query terms are not assigned specific weights, therefore all tags are weighted equally.

These proposed weighting approaches affect questions with multiple tags. Giving a weight to one tag question does not make any difference in expert candidate rankings. Therefore, in tag weighting experiments, 50 questions with one tag are removed from the test sets for both question routing and reply ranking. The experiments were performed over the rest of the 200 questions.

5.1.7.1 Experiments on Question Routing Task

The results of applying these weighted queries to question routing task are presented in Tables 5.9 and 5.10 respectively for the profile and document-based approaches. Combination of these different weighting approaches are also experimented with in order to see how these different weightings work together, and whether one weighting consistently outperforms the others. The optimum interpolation weights are identified by performing parameter optimization with 10-fold-cross-validation as explained in Section 4.4. The optimum weights are also presented with the interpolation results in every fifth row of the tables. For CP-Tag weighting approach, the

[11] http://www.lemurproject.org/lemur/IndriQueryLanguage.php

User	Inf.	P	P	P	MRR	MSC	MSC	MSC	NDCG
Exp.	Need	@5	@10	@20		@5	@10	@20	
Answer Body	U-Tag	.0250	.0205	.0165	.0798	.1050	.1650	.2600	.0670
	IN-Tag	.0250	.0195	.0163	.0752	.1100	.1700	.2450	.0687
	TG-Tag	.0280	.0210	.0158	.0772	.1200	.1750	.2400	.0689
	CP-Tag (30)	.0230	.0175	.0138	.0688	.1000	.1500	.2200	.0627
	$U_0/IN_0/TG_1/CP_0$	.0280	.0210	.0158	.0772	.1200	.1750	.2400	.0690
Question Title& Body	U-Tag	.0260	.0190	.0153	.0842	.1200	.1650	.2450	.0771
	IN-Tag	.0250	.0170	.0165	.0851	.1000	.1400	.2600	$.0790_{s}$
	TG-Tag	.0250	.0195	.0185	$.0882_{s'}$	.1100	.1650	$.2950^{r'}_{s'}$	.0778
	CP-Tag (10)	.0240	.0210	$.0190^{r'}$	.0916	.1050	.1700	.2950	.0756
	$U_{.8}/IN_0/TG_0/CP_{.2}$	.0300	.0200	.0165	$.0864_{s}$	.1250	.1750	.2550	$.0798_{s}$
Question Title	U-Tag	.0450	.0360	.0263	.1218	.1900	.2550	.3450	.0959
	IN-Tag	.0500	.0400	.0288	$.1362_{s}$	.2050	$.3150^{r}_{s}$	$.3950^{r'}_{s'}$	$.1000_{s}$
	TG-Tag	.0430	.0370	$.0303^{r}_{s}$	$.1495^{r}_{s}$	.1750	.2900	$.4100^{r}_{s}$	$.1016_{s}$
	CP-Tag (30)	.0450	.0330	.0260	.1171	.1800	.2500	.3650	.0979
	$U_{.1}/IN_{.7}/TG_0/CP_{.2}$	$.0550^{r}_{s'}$	.0390	$.0295_{s}$	$.1356_{s}$	.2250	$.3000^{r'}_{s'}$	$.4200^{r}_{s}$	$.1043^{r}_{s}$
Question Tag	U-Tag	.0630	.0445	.0330	.1678	.2550	.3500	.4650	.1216
	IN-Tag	.0630	.0450	.0343	$.1940^{r}_{s}$	.2650	.3550	.4800	$.1298^{r}_{s}$
	TG-Tag	.0700	$.0510^{r}_{s}$	$.0367^{r}_{s}$	$.1930^{r}_{s}$	.2850	.3750	$.5000^{r'}_{s'}$	$.1302^{r}_{s}$
	CP-Tag (90)	.0680	$.0495^{r'}$	.0360	$.1725_{s'}$	.2550	.3600	.4800	$.1313^{r}_{s}$
	$U_0/IN_0/TG_{.6}/CP_{.4}$	$.0750^{r}_{s}$	$.0515^{r}_{s}$	.0352	$.1900_{s}$	.2950	.3650	.5100	$.1344^{r}_{s}$

Table 5.9: Question routing performance of profile-based approach with weighted tags.

estimated (again with 10-fold-cross-validation) optimum number of top retrieved *n* profiles are also given within parenthesis.

In question routing task, using weighted tags improved the results in almost all representations. The improvements are smaller and less consistent in the document-based approach compared to the profile-based approach mainly because the document-based approach depends on the top retrieved *n* documents which makes it more vulnerable to tag weights. On the other hand the profile-based approach returns more consistent results.

For the profile-based approach (Table 5.9), weighting tags based on their tag generality (*TG-Tag*) over the collection generally outperformed other weighting approaches. This weighting type improved the performance to a point where a responder is retrieved within top 20 expert candidates for half of the questions (*MSC*@20 = 0.5). *TG-Tag* is followed by the *IN-Tag* which also returned statistically significant improvements in several cases. Even though *CP-Tag* weighting provided improvements in several cases, it is less consistent compared to others.

For the document-based approach (Table 5.10), as mentioned before, the improvements and relative ranking of weighting types are less consistent. For instance *CP-Tag* weighting performed worst in two representations while performed best in one representation. Similar to the profile-based approach, the *TG-Tag* weighting performs relatively better than others, and closely followed by *IN-Tag*. This behavior of *TG-Tag* can be due to users selecting and responding to questions with general tags (like *linux*) more than questions with more specific tags (like *grep* or *ls*). In other words, the *TG-Tag* weighting performing better can be specific to just the question routing task.

User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Answer Body	U-Tag	.0390	.0310	.0255	.1352	.1800	.2600	.3650	.0808
	IN-Tag	.0400	.0315	.0265	.1350	.1900	.2600	$.4100^{r'}_{s'}$	.0812
	TG-Tag	.0400	.0325	.0250	.1321	.1850	.2800	.3750	.0815
	CP-Tag (90)	.0380	.0260	.0215	.1207	.1550	.2150	.3350	.0767
	$U_0/IN_{.3}/TG_{.6}/CP_{.1}$	.0440	.0325	.0263	.1314	.2050	.2700	.3900	$.0885^{r}_{s}$
Question Title& Body	U-Tag	.0450	.0390	.0335	.1344	.1850	.3000	.4650	.1130
	IN-Tag	.0430	.0425	.0325	.1315	.1850	.3350	.4600	.1154
	TG-Tag	.0440	.0400	.0325	.1366	.1900	.3050	.4600	$.1160_{s}$
	CP-Tag (60)	.0360	.0340	.0295	.1142	.1550	.2800	.4150	.1055
	$U_{.4}/IN_{.3}/TG_{.3}/CP_0$	.0500	.0415	.0330	$.1414_{s}$	.2050	.3300	.4750	$.1186^{r}_{s}$
Question Title	U-Tag	.0420	.0380	.0308	.1392	.1800	.2950	.4300	.1119
	IN-Tag	.0450	.0360	.0323	.1450	.1850	.2850	.4450	$.1158^{r}_{s}$
	TG-Tag	.0470	.0385	.0318	$.1522^{r'}_{s'}$	.1900	.3050	.4450	$.1169^{r}$
	CP-Tag (100)	.0460	.0370	.0305	.1384	.2000	.3000	.4150	.1135
	$U_{.3}/IN_0/TG_{.5}/CP_{.2}$	$.0510^{r}_{s'}$	.0405	$.0335^{r}_{s}$	$.1565^{r}$	$.2100^{r'}_{s'}$	.3100	$.4600^{r'}_{s'}$	$.1224^{r}_{s}$
Question Tag	U-Tag	.0550	.0465	.0353	.1609	.2150	.3450	.4750	.1323
	IN-Tag	.0530	.0455	.0345	.1554	.2100	.3400	.4750	.1307
	TG-Tag	.0550	.0465	.0350	.1607	.2200	.3500	.4800	.1314
	CP-Tag (100)	.0560	.0480	.0350	.1682	.2300	.3650	$.4900^{r'}$	$.1353_{s}$
	$U_{.5}/IN_{.1}/TG_0/CP_{.4}$	.0590	$.0500^{r}_{s}$	.0360	$.1738^{r'}_{s}$	.2350	$.3800^{r}_{s}$	.4950	$.1369^{r}_{s}$

Table 5.10: Question routing performance of document-based approach with weighted tags.

5.1.7.2 Experiments on Reply Ranking Task

For reply ranking task (Tables 5.11 and 5.12), weighting tags also returned improvements over uniform tag weights. In the profile-based approach (Table 5.11), the *CP-Tag* weighting outperformed other weightings in answer body and question title and body profiles. Using *CP-Tag* weighted tags on users' answer profiles returned the best performance among all other user expertise representations and tag weightings in the profile-based approach. One reason for this can be that in reply ranking task, the replies receive votes based on accuracy and also the presentation of information. So in addition to being an expert on a particular question, being able to convey expertise is another factor that affects the ranking of replies. Users that are more explanatory in their answers have higher probability of using important key terms that are related to the general expertise area of the information need. Using these terms makes their answers more relevant to *CP-Tag* weighting of query terms. For question title and tags representations, *TG-Tag* and *IN-Tag* perform similarly.

In the document-based approach (Table 5.12), using *IN-Tag* weighting generally outperforms other weightings, probably due to selecting responders who respond to questions that are very similar to a particular question, at least with respect to the asker's perception of the information need. If answers are voted with respect to their information coverage of what has been asked, then using the asker's perception of seeked information makes more sense. Even though the improvements are small, they are consistent. Getting this consistent behavior in the document-based approach which was not the case with question routing task is not very surprising. As also mentioned previously, the document-based approach performs similarly to the profile-based approach in question routing task, while it outperforms the profile-based approach in

User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Answer Body	U-Tag	.5301	.6127	.6909	.7517	.8223	.2550
	IN-Tag	.5289	.6210	.6891	.7527	.8223	.2500
	TG-Tag	.5382	.6266	.6942	.7543	.8259	.2650
	CP-Tag (20)	$.5708^{r'}$	$.6402^{r'}_{s}$	$.7093^{r'}$	.7617	$.8346^{r'}$	.2850
	$U_0/IN_0/TG_0/CP_1$	$.5708^{r'}$	$.6402^{r'}_{s}$	$.7093^{r'}$	.7617	$.8346^{r'}$	.2850
Question Title& Body	U-Tag	.5244	.6194	.6998	.7521	.8233	.2550
	IN-Tag	.5290	.6228	.7045	.7562	.8255	.2550
	TG-Tag	.5257	.6193	.6994	.7523	.8236	.2550
	CP-Tag (50)	$.5580^{r'}$	$.6412^{r'}$	.7063	.7626	$.8332^{r'}$	.2900
	$U_0/IN_0/TG_{.1}/CP_{.9}$	$.5580^{r'}$	$.6415^{r'}$	.7093	.7614	$.8335^{r'}$	.2900
Question Title	U-Tag	.5518	.6306	.7044	.7665	.8309	.2550
	IN-Tag	.5510	.6270	$.6931^{r'}$	.7665	.8291	.2600
	TG-Tag	.5419	.6213	$.6930^{r'}$	.7616	.8263	.2400
	CP-Tag (10)	.5484	.6216	.6942	.7598	.8270	.2550
	$U_{.2}/IN_0/TG_{.6}/CP_{.2}$	.5576	.6266	.6977	.7656	.8303	.2600
Question Tag	U-Tag	.5424	.6268	.6903	.7566	.8256	.2600
	IN-Tag	.5511	.6270	.6951	.7567	.8277	.2750
	TG-Tag	$.5567^{r'}$	.6360	$.7010^{r}$	.7612	$.8308^{r}$	$.2850^{r'}_{s'}$
	CP-Tag (10)	.5573	.6309	.6948	.7611	.8296	.2800
	$U_0/IN_{.3}/TG_{.6}/CP_{.1}$	$.5632^{r}$	.6333	$.7001^{r}$	$.7595^{s'}$	$.8310^{r}$	$.2850^{r'}_{s'}$

Table 5.11: Reply ranking performance of profile-based approach with weighted tags.

reply ranking task. Therefore, the document-based approach being more consistent in the reply ranking task is not a coincidence.

These differences between tasks raise the question of whether the definition of expertise differs for different tasks. In question routing, since the task is to retrieve responders who can provide accurate answers, the definition of expertise seems more general. However, with reply ranking task, we are trying rank responders who we can get from question routing task, but elaborating more on their question-specific expertise. So while question routing task looks for more general experts to just get possibly accurate replies, reply ranking task is looking for more specific signals of expertise. This is also obvious as using question tags to represent users works very well in question routing task, while using users' own answers outperforms question tags representations of users in reply ranking task.

5.1.7.3 Summary

To sum up, weighting tags outperformed uniform weighting of tags in both the profile and document-based approaches for both tasks. These improvements show that some tags are more important than others with respect to representing the information need required to answer the particular question.

For question routing task, using *tag generality* weighting to retrieve expert candidates on generality of the information need of the question outperformed other weightings. With this tag weighting approach, the relative order of the user expertise representation stays the same,

User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Answer Body	U-Tag	.5805	.6486	.7106	.7678	.8382	.3100
	IN-Tag	.5706	.6572	.7136	.7704	.8381	.2950
	TG-Tag	.5632	.6487	.7108	.7656	.8345	.2750
	CP-Tag (20)	.5758	.6438	.7117	.7662	.8363	.2800
	$U_{.9}/IN_0/TG_0/CP_{.1}$	.5846	.6477	.7123	.7662	.8385	.3000
Question Title& Body	U-Tag	.5534	.6367	.7056	.7641	.8307	.2850
	IN-Tag	.5611	.6443	.7119	.7690	.8350	.2750
	TG-Tag	.5609	.6447	.7096	.7689	.8344	.2800
	CP-Tag (20)	.5505	.6371	.7089	.7658	.8319	.2350
	$U_0/IN_{.5}/TG_{.2}/CP_{.3}$	.5709	.6516	.7129	.7669	.8374	.2800
Question Title	U-Tag	.5477	.6381	.7122	.7646	.8309	.2650
	IN-Tag	.5605	.6455	.7195	$.7721^{r'}$	.8358	.2750
	TG-Tag	.5450	.6382	.7138	.7647	.8308	.2550
	CP-Tag (100)	.5350	.6253	.6992	.7590	.8258	.2300
	$U_0/IN_{.9}/TG_0/CP_{.1}$	.5570	.6392	.7111	.7656	.8330	.2650
Question Tag	U-Tag	.5653	.6312	.7129	.7701	.8329	.2800
	IN-Tag	.5714	$.6365^{r'}$	.7175	.7718	.8357	.2850
	TG-Tag	.5651	.6319	.7138	.7710	.8331	.2800
	CP-Tag (40)	.5526	.6333	.7078	.7638	.8302	.2700
	$U_{.2}/IN_{.6}/TG_{.2}/CP_0$	.5759	$.6389^{r}_{s}$	.7179	.7723	$.8365^{r}$	.2950

Table 5.12: Reply ranking performance of document-based approach with weighted tags.

and using questions tags to represent users outperformed all other representations. These generalized representations of information need and user expertise suggest that, for a given question, question routing task is looking for experts within a more general perspective.

Different behaviors are observed for reply ranking task. First of all the document-based approach outperforms the profile-based approach, which indicates that question-specific user evidence is more useful than more generic and prolific representations of users. Within the document-based approach, representing users with their previous answers means that in addition to relevance, the presentation power of responders, in other words how they use the related vocabulary is important. Finally, representing the information need with question tags and weighting them by asker's ordering indicates that answering very specific questions with similar underlying information need is very useful. Overall, unlike question routing, reply ranking task is looking for very question-specific expertise.

This section analyzes questions and replies, and how they can be used to represent expertise in CQAs more effectively. The following section analyzes comments similar to the analysis in this section and identifies whether they can be used to improve the accuracy of expertise estimation for question routing and reply ranking tasks. Testing the same proposed representations of expertise and weighting of query terms over comments is also useful for analyzing their consistency across different content types.

5.2 Exploring Comments for Expertise Estimation in CQA Sites

There are three types of user generated content in CQA communities; questions, replies and comments. The first two were used widely in order to identify expert users in these environments; however, comments which are yet another source of information for expertise estimation, have not been explored in detail.

Chang and Pal [21] used comments for the collaborative question routing problem. They identified a set of most compatible responders and commenters with the aim to improve the long lasting value of a question thread with posted replies and comments. Unlike routing questions to specific users, they route them to group of users, who would be willing to collaborate either by answering or commenting. They argue that users have different propensity to answer and comment; therefore, they built separate lists of responders and commenters for a given question.

Building upon this work, we analyze the StackOverflow comments, and explore how these comments can be used to improve expertise estimation performance for both question routing and reply ranking tasks.

5.2.1 Commenting in StackOverflow

In StackOverflow, both questions and replies can receive comments from users. In StackOverflow environment, users can comment on both questions and replies. There are a total of 29,222,308 comments[12]. 41% of them are made on questions while the rest of them are on replies. An example question and reply with comments is presented in Figure 5.7. Comments consist of comment content, author information and timestamp of the posting time. Similar to question and replies, users can also vote for comments as seen next to the comments. The votes received do not affect the relative ranking of comments, rather they are ordered by their posting time. This is mainly due to protect the conversational structure within comment threads. For example, in Figure 5.7, user *MeqDotNet* asked a question with posting a comment, and user *Usman Y* replied *MeqDotNet*'s question again with commenting.

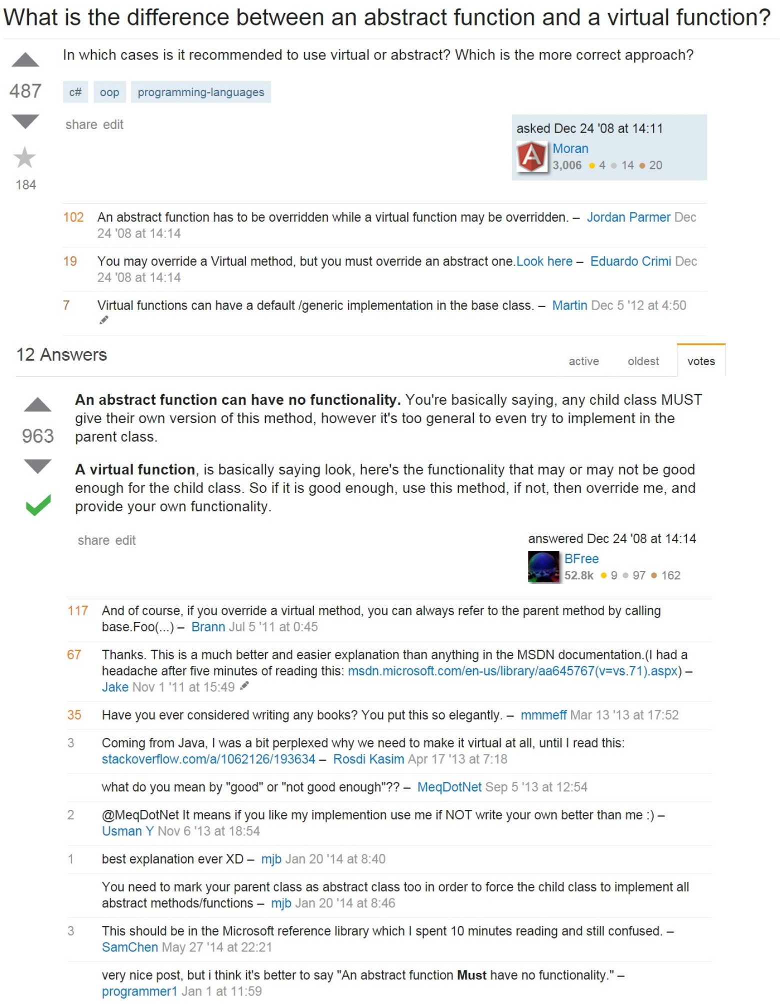
Our initial goal was to understand why users leave comments in the first place. Therefore, we manually analyzed a set of comments and grouped them into categories. Even though these categories may not cover all cases of comments, they still give some underlying understanding of comments in general.

Comments on questions can be divided mainly into two categories. The first group of comments on questions is constructed to clarify a point in question, make it more understandable and unambiguous in general. The askers can use these comments to make their question more clear. An example of this user case is provided in Figure 5.8. The left side of the figure displays the original initial posted question (Figure 5.8(a)) and edits made to the question [13] (Figures 5.8(c) and 5.8(e)). The right side of the figure shows the comments on question which led to those changes (Figures 5.8(b), 5.8(d) and 5.8(f)). As seen in the figure, after a question is posted, users use comments to ask for clarifications or make suggestions. Depending on these comments, askers may make changes to the original question. In this example, two suggestions were made by users, and asker used both of these suggestions and made the necessary edits in the question.

The second group of comments on questions is to provide answers to the question. An example to this is given in Figure 5.9. In the example, the first comment provides a correct reply

[12] There are another 4036 comments without any parents (either question or reply). These questions or replies are probably deleted from the system, but comments made on them are still available.

[13] Text in green shows the edits.



What is the difference between an abstract function and a virtual function?

487 — In which cases is it recommended to use virtual or abstract? Which is the more correct approach?

asked Dec 24 '08 at 14:11
Moran
3,006 ● 4 ● 14 ● 20

102 An abstract function has to be overridden while a virtual function may be overridden. – Jordan Parmer Dec 24 '08 at 14:14

19 You may override a Virtual method, but you must override an abstract one.Look here – Eduardo Crimi Dec 24 '08 at 14:14

7 Virtual functions can have a default /generic implementation in the base class. – Martin Dec 5 '12 at 4:50

12 Answers

963 — **An abstract function can have no functionality.** You're basically saying, any child class MUST give their own version of this method, however it's too general to even try to implement in the parent class.

A virtual function, is basically saying look, here's the functionality that may or may not be good enough for the child class. So if it is good enough, use this method, if not, then override me, and provide your own functionality.

answered Dec 24 '08 at 14:14
BFree
52.8k ● 9 ● 97 ● 162

117 And of course, if you override a virtual method, you can always refer to the parent method by calling base.Foo(...) – Brann Jul 5 '11 at 0:45

67 Thanks. This is a much better and easier explanation than anything in the MSDN documentation.(I had a headache after five minutes of reading this: msdn.microsoft.com/en-us/library/aa645767(v=vs.71).aspx) – Jake Nov 1 '11 at 15:49

35 Have you ever considered writing any books? You put this so elegantly. – mmmeff Mar 13 '13 at 17:52

3 Coming from Java, I was a bit perplexed why we need to make it virtual at all, until I read this: stackoverflow.com/a/1062126/193634 – Rosdi Kasim Apr 17 '13 at 7:18

what do you mean by "good" or "not good enough"?? – MeqDotNet Sep 5 '13 at 12:54

2 @MeqDotNet It means if you like my implemention use me if NOT write your own better than me :) – Usman Y Nov 6 '13 at 18:54

1 best explanation ever XD – mjb Jan 20 '14 at 8:40

You need to mark your parent class as abstract class too in order to force the child class to implement all abstract methods/functions – mjb Jan 20 '14 at 8:46

3 This should be in the Microsoft reference library which I spent 10 minutes reading and still confused. – SamChen May 27 '14 at 22:21

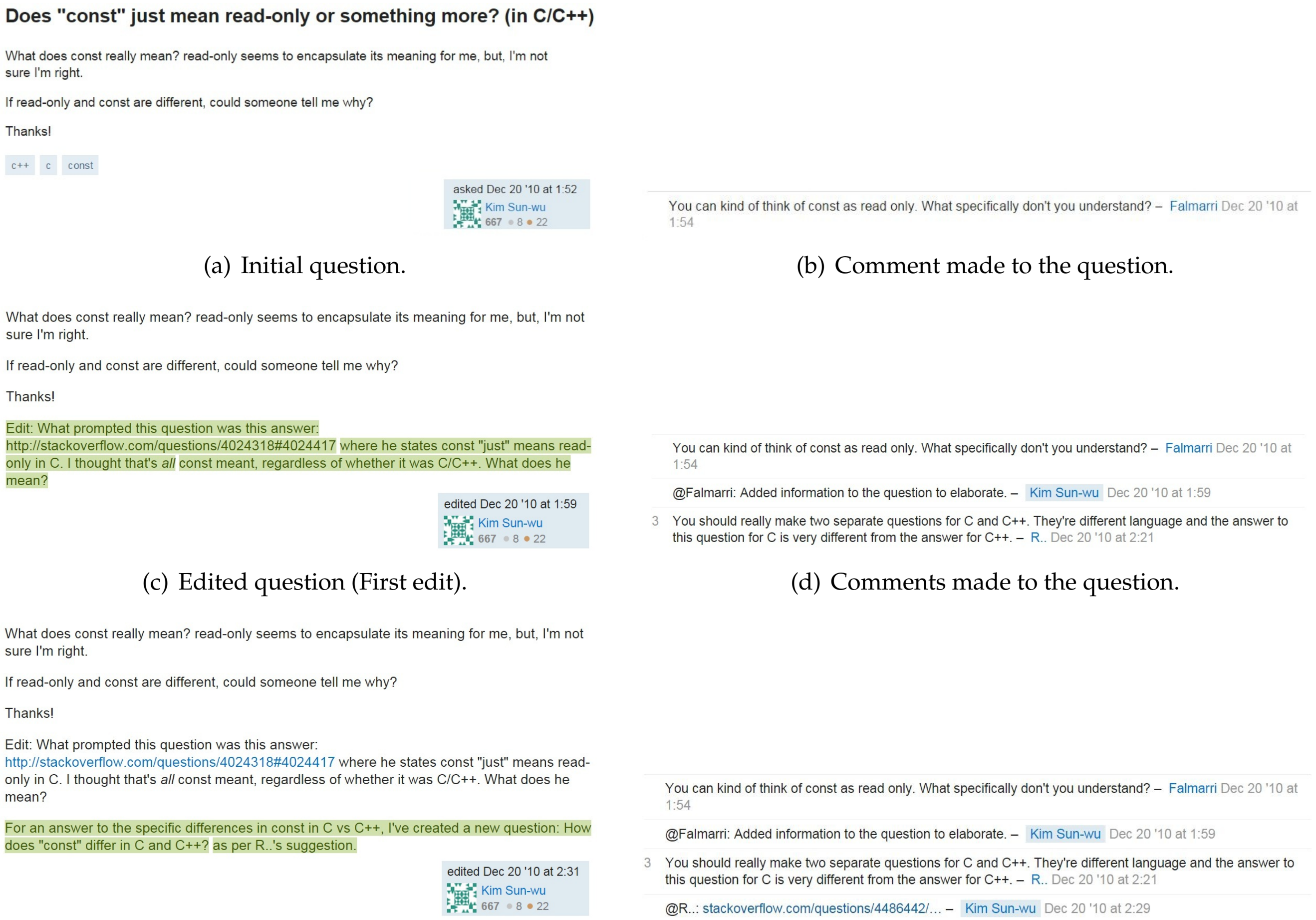
very nice post, but i think it's better to say "An abstract function **Must** have no functionality." – programmer1 Jan 1 at 11:59

Figure 5.7: An example question and reply with comments.

to the question. The first reply which is accepted as best by the asker also refers to the same comment as a correct reply. The reason why users prefer to post the answer as comment rather than as reply is unknown. One possible reason can be that user is a newly joined user who is hesitant to answer the question; however, it is not the case in this example. User *"Dieter"* is among the top 3% most reputed users in StackOverflow with more than 600 posted replies at the time he posted this comment.

Another example to this case is available in Figure 5.7. In this example, all question comments are providing a reply to the question. The first two comments (by users *Jordan* and *Eduardo*) and the first reply which was accepted as the best reply by asker and received the highest number of votes from other users were all posted at the same time. The two comments are almost the same and very brief, while the reply is more explanatory and detailed. These brief comments with replies observed in this example and also within the example in Figure 5.9 suggest that maybe users tend to leave comments when they are in hurry, or not available to give an elaborate reply.

The comments on replies can be also categorized into two groups. The first group of comments praises the answer and while the second group of comments suggests corrections or



Does "const" just mean read-only or something more? (in C/C++)

What does const really mean? read-only seems to encapsulate its meaning for me, but, I'm not sure I'm right.

If read-only and const are different, could someone tell me why?

Thanks!

c++ c const

asked Dec 20 '10 at 1:52
Kim Sun-wu
667 ● 8 ● 22

(a) Initial question.

You can kind of think of const as read only. What specifically don't you understand? – Falmarri Dec 20 '10 at 1:54

(b) Comment made to the question.

What does const really mean? read-only seems to encapsulate its meaning for me, but, I'm not sure I'm right.

If read-only and const are different, could someone tell me why?

Thanks!

Edit: What prompted this question was this answer: http://stackoverflow.com/questions/4024318#4024417 where he states const "just" means read-only in C. I thought that's *all* const meant, regardless of whether it was C/C++. What does he mean?

edited Dec 20 '10 at 1:59
Kim Sun-wu
667 ● 8 ● 22

(c) Edited question (First edit).

You can kind of think of const as read only. What specifically don't you understand? – Falmarri Dec 20 '10 at 1:54

@Falmarri: Added information to the question to elaborate. – Kim Sun-wu Dec 20 '10 at 1:59

3 You should really make two separate questions for C and C++. They're different language and the answer to this question for C is very different from the answer for C++. – R.. Dec 20 '10 at 2:21

(d) Comments made to the question.

What does const really mean? read-only seems to encapsulate its meaning for me, but, I'm not sure I'm right.

If read-only and const are different, could someone tell me why?

Thanks!

Edit: What prompted this question was this answer: http://stackoverflow.com/questions/4024318#4024417 where he states const "just" means read-only in C. I thought that's *all* const meant, regardless of whether it was C/C++. What does he mean?

For an answer to the specific differences in const in C vs C++, I've created a new question: How does "const" differ in C and C++? as per R..'s suggestion.

edited Dec 20 '10 at 2:31
Kim Sun-wu
667 ● 8 ● 22

(e) Edited question (Second edit).

You can kind of think of const as read only. What specifically don't you understand? – Falmarri Dec 20 '10 at 1:54

@Falmarri: Added information to the question to elaborate. – Kim Sun-wu Dec 20 '10 at 1:59

3 You should really make two separate questions for C and C++. They're different language and the answer to this question for C is very different from the answer for C++. – R.. Dec 20 '10 at 2:21

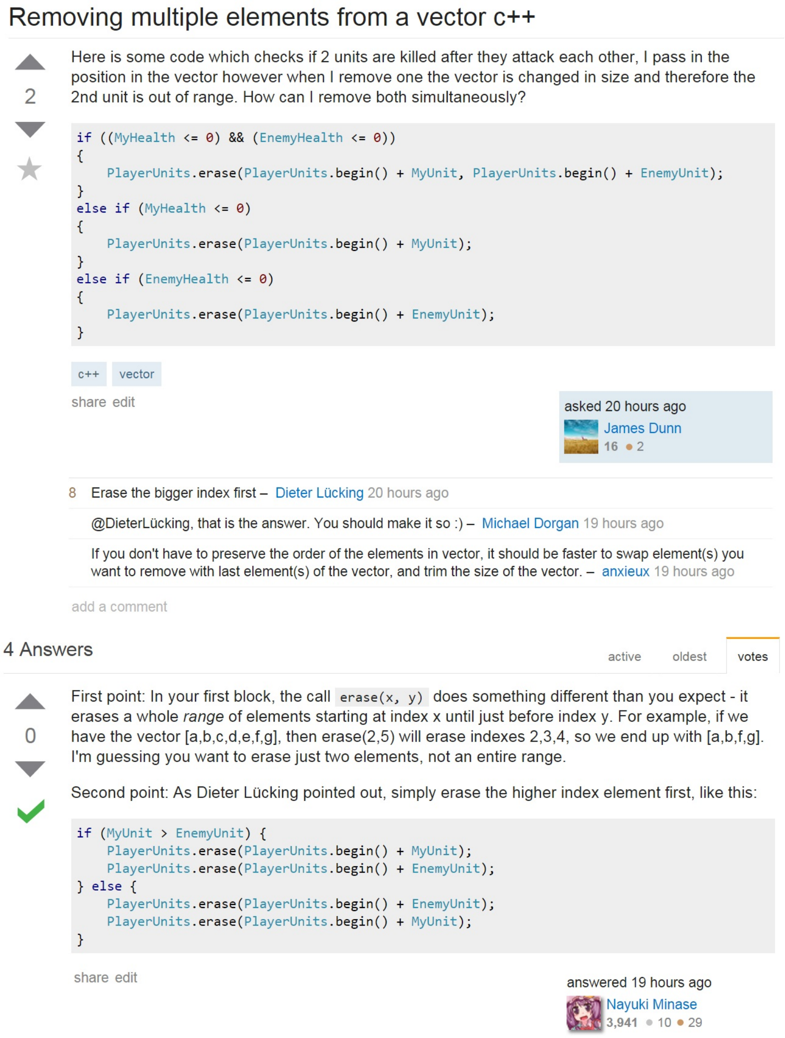
@R..: stackoverflow.com/questions/4486442/… – Kim Sun-wu Dec 20 '10 at 2:29

(f) Comments made to the question.

Figure 5.8: An example question and comments made on it in order to clarify the question.

improvements to the reply. Example to both of these cases are presented in Figure 5.10 and continued in Figure 5.11. Similar to Figure 5.8, the left side of the figure presents the original and edited replies, while the right side of the figure show the comments. Figure 5.10(a) displays the question and the initial reply. Figure 5.10(b) contains two comments; the first comment (from Oliver) can be categorized into the first category while the second comment (from Steve) is an example to second category. Steve made a suggestion which had been considered by Nawaz (the author of reply) and necessary edits were performed in Figure 5.10(c). In Figures 5.10(d), 5.11(b) and 5.11(d) we see other users making suggestions to the reply, and Figures 5.11(a) and 5.11(c) show how the author of the reply respond to these suggestions.

In Figures 5.8 and 5.11 we also see the author of question or reply communicating with commenters through comments. Getting into dialog with other commenters is yet another common case of commenting which can be observed in both questions and replies. In these situations, users may resolve an issue through back and forth commenting to each other.



Removing multiple elements from a vector c++

Here is some code which checks if 2 units are killed after they attack each other, I pass in the position in the vector however when I remove one the vector is changed in size and therefore the 2nd unit is out of range. How can I remove both simultaneously?

```
if ((MyHealth <= 0) && (EnemyHealth <= 0))
{
    PlayerUnits.erase(PlayerUnits.begin() + MyUnit, PlayerUnits.begin() + EnemyUnit);
}
else if (MyHealth <= 0)
{
    PlayerUnits.erase(PlayerUnits.begin() + MyUnit);
}
else if (EnemyHealth <= 0)
{
    PlayerUnits.erase(PlayerUnits.begin() + EnemyUnit);
}
```

asked 20 hours ago
James Dunn
16 ● 2

8 Erase the bigger index first – Dieter Lücking 20 hours ago

@DieterLücking, that is the answer. You should make it so :) – Michael Dorgan 19 hours ago

If you don't have to preserve the order of the elements in vector, it should be faster to swap element(s) you want to remove with last element(s) of the vector, and trim the size of the vector. – anxieux 19 hours ago

4 Answers

First point: In your first block, the call `erase(x, y)` does something different than you expect - it erases a whole *range* of elements starting at index x until just before index y. For example, if we have the vector [a,b,c,d,e,f,g], then erase(2,5) will erase indexes 2,3,4, so we end up with [a,b,f,g]. I'm guessing you want to erase just two elements, not an entire range.

Second point: As Dieter Lücking pointed out, simply erase the higher index element first, like this:

```
if (MyUnit > EnemyUnit) {
    PlayerUnits.erase(PlayerUnits.begin() + MyUnit);
    PlayerUnits.erase(PlayerUnits.begin() + EnemyUnit);
} else {
    PlayerUnits.erase(PlayerUnits.begin() + EnemyUnit);
    PlayerUnits.erase(PlayerUnits.begin() + MyUnit);
}
```

answered 19 hours ago
Nayuki Minase
3,941 ● 10 ● 29

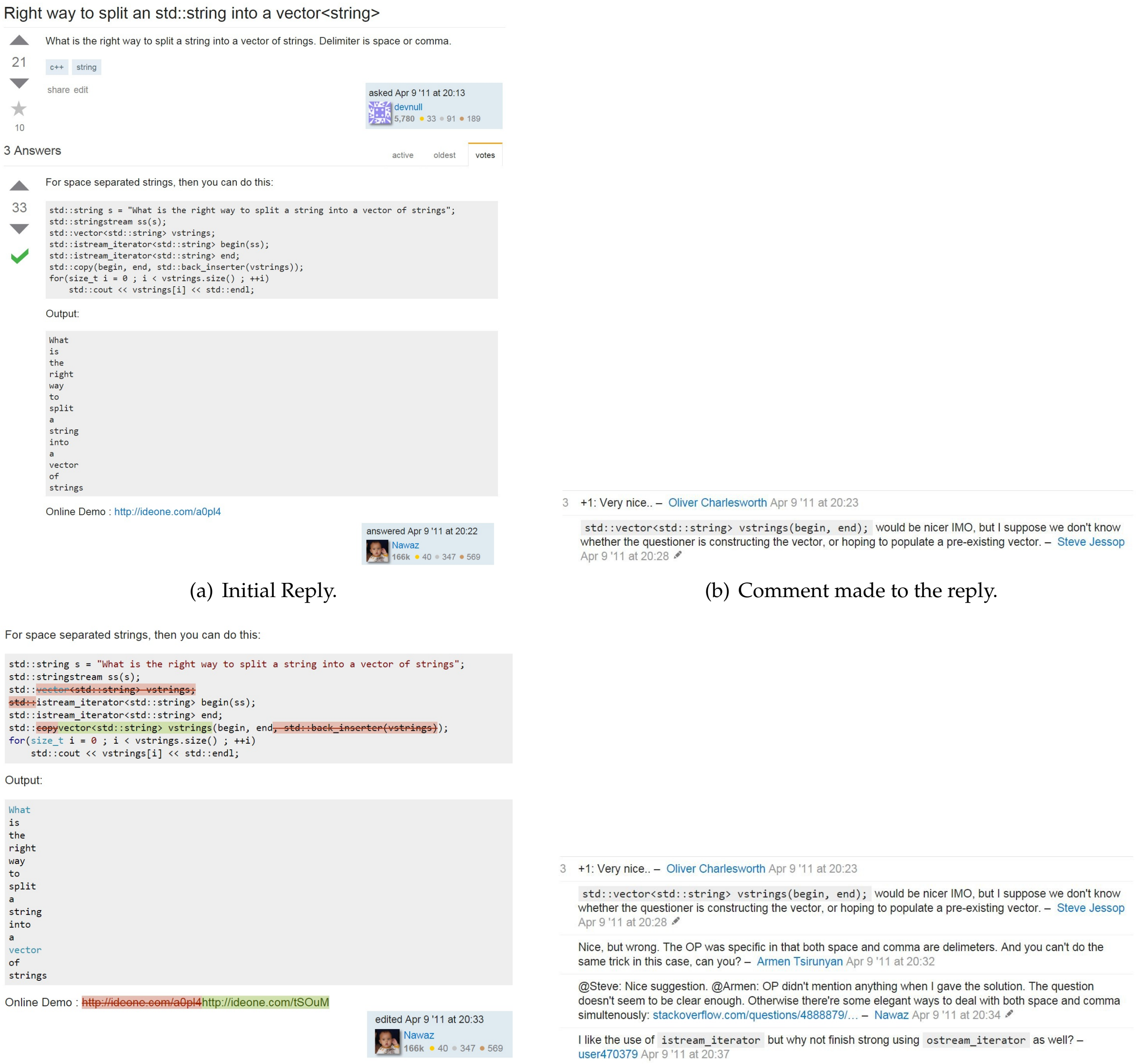
Figure 5.9: An example question and a comment which contains the answer.

5.2.2 Pre-Processing Comments

Based on these common commenting behaviors several pre-processing steps were applied in order to analyze comments and commenters more effectively. Initially, comments that were constructed by the authors of parent posts (questions or replies) were removed due to these comments being made most probably to reply to another prior comment on the particular post. Secondly, comments made on the same post by the same user were merged into one comment, bodies of these multiple comments are merged into one. These multiple comments by the same user are probably result of a discussion among commenters, and this step is performed in order to not count the comments of same user on same post multiple times.

The number of comments before and after pre-processing are presented in Table 5.13. Removing the self-made comments and merging comments of the same author decreased the number of comments drastically. However the total number of comments is still around 10.1M which is still very comparable to 7.2M questions and 12.6M replies.

The number of unique commenters before and after pre-processing is also available in Table



Right way to split an std::string into a vector<string>

21

What is the right way to split a string into a vector of strings. Delimiter is space or comma.

c++ string

share edit

10

asked Apr 9 '11 at 20:13
devnull
5,780 ● 33 ● 91 ● 189

3 Answers

active oldest votes

33

For space separated strings, then you can do this:

```
std::string s = "What is the right way to split a string into a vector of strings";
std::stringstream ss(s);
std::vector<std::string> vstrings;
std::istream_iterator<std::string> begin(ss);
std::istream_iterator<std::string> end;
std::copy(begin, end, std::back_inserter(vstrings));
for(size_t i = 0 ; i < vstrings.size() ; ++i)
    std::cout << vstrings[i] << std::endl;
```

Output:

```
What
is
the
right
way
to
split
a
string
into
a
vector
of
strings
```

Online Demo : http://ideone.com/a0pl4

answered Apr 9 '11 at 20:22
Nawaz
166k ● 40 ● 347 ● 569

(a) Initial Reply.

3 +1: Very nice.. – Oliver Charlesworth Apr 9 '11 at 20:23

`std::vector<std::string> vstrings(begin, end);` would be nicer IMO, but I suppose we don't know whether the questioner is constructing the vector, or hoping to populate a pre-existing vector. – Steve Jessop Apr 9 '11 at 20:28

(b) Comment made to the reply.

For space separated strings, then you can do this:

```
std::string s = "What is the right way to split a string into a vector of strings";
std::stringstream ss(s);
std::~~vector<std::string> vstrings;~~
~~std::~~istream_iterator<std::string> begin(ss);
std::istream_iterator<std::string> end;
std::~~copy~~vector<std::string> vstrings(begin, end~~, std::back_inserter(vstrings)~~);
for(size_t i = 0 ; i < vstrings.size() ; ++i)
    std::cout << vstrings[i] << std::endl;
```

Output:

```
What
is
the
right
way
to
split
a
string
into
a
vector
of
strings
```

Online Demo : ~~http://ideone.com/a0pl4~~http://ideone.com/tSOuM

edited Apr 9 '11 at 20:33
Nawaz
166k ● 40 ● 347 ● 569

3 +1: Very nice.. – Oliver Charlesworth Apr 9 '11 at 20:23

`std::vector<std::string> vstrings(begin, end);` would be nicer IMO, but I suppose we don't know whether the questioner is constructing the vector, or hoping to populate a pre-existing vector. – Steve Jessop Apr 9 '11 at 20:28

Nice, but wrong. The OP was specific in that both space and comma are delimeters. And you can't do the same trick in this case, can you? – Armen Tsirunyan Apr 9 '11 at 20:32

@Steve: Nice suggestion. @Armen: OP didn't mention anything when I gave the solution. The question doesn't seem to be clear enough. Otherwise there're some elegant ways to deal with both space and comma simultenously: stackoverflow.com/questions/4888879/… – Nawaz Apr 9 '11 at 20:34

I like the use of `istream_iterator` but why not finish strong using `ostream_iterator` as well? – user470379 Apr 9 '11 at 20:37

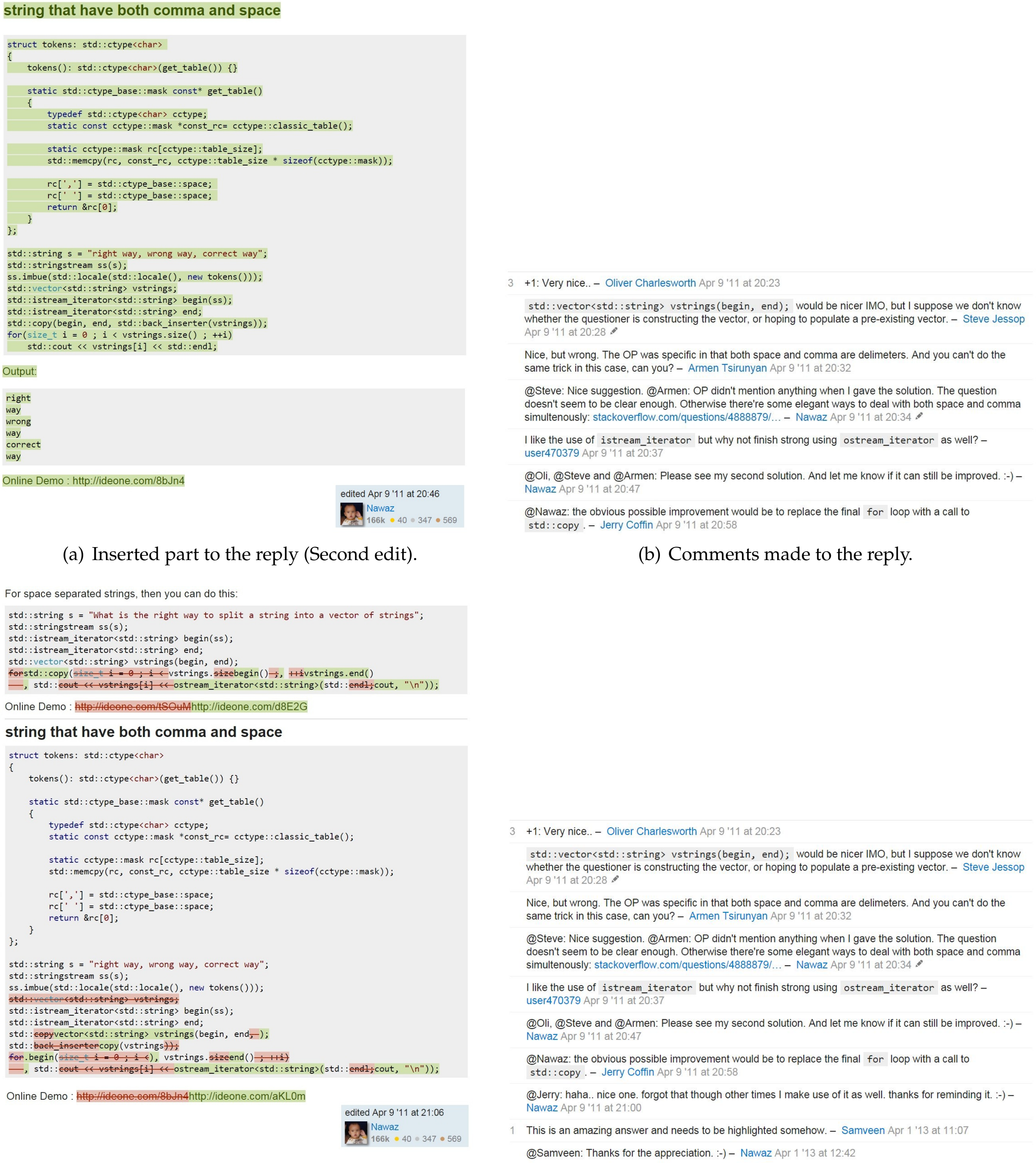
(c) Edited reply (First edit).

(d) Comments made to the reply.

Figure 5.10: Example reply and comments (1).

	Questions	Replies
# Comments before pre-processing	11,849,536	17,372,772
# Comments after pre-processing	6,486,113	3,666,301
# Unique Commenters before pre-processing	720,299	919,236
# Unique Commenters after pre-processing	201,806	204,973

Table 5.13: Commenting related statistics.



string that have both comma and space

```
struct tokens: std::ctype<char>
{
    tokens(): std::ctype<char>(get_table()) {}

    static std::ctype_base::mask const* get_table()
    {
        typedef std::ctype<char> cctype;
        static const cctype::mask *const_rc= cctype::classic_table();

        static cctype::mask rc[cctype::table_size];
        std::memcpy(rc, const_rc, cctype::table_size * sizeof(cctype::mask));

        rc[','] = std::ctype_base::space;
        rc[' '] = std::ctype_base::space;
        return &rc[0];
    }
};

std::string s = "right way, wrong way, correct way";
std::stringstream ss(s);
ss.imbue(std::locale(std::locale(), new tokens()));
std::vector<std::string> vstrings;
std::istream_iterator<std::string> begin(ss);
std::istream_iterator<std::string> end;
std::copy(begin, end, std::back_inserter(vstrings));
for(size_t i = 0 ; i < vstrings.size() ; ++i)
    std::cout << vstrings[i] << std::endl;
```

Output:

```
right
way
wrong
way
correct
way
```

Online Demo : http://ideone.com/8bJn4

edited Apr 9 '11 at 20:46
Nawaz
166k ● 40 ● 347 ● 569

(a) Inserted part to the reply (Second edit).

3 +1: Very nice.. – Oliver Charlesworth Apr 9 '11 at 20:23

`std::vector<std::string> vstrings(begin, end);` would be nicer IMO, but I suppose we don't know whether the questioner is constructing the vector, or hoping to populate a pre-existing vector. – Steve Jessop Apr 9 '11 at 20:28 ✎

Nice, but wrong. The OP was specific in that both space and comma are delimeters. And you can't do the same trick in this case, can you? – Armen Tsirunyan Apr 9 '11 at 20:32

@Steve: Nice suggestion. @Armen: OP didn't mention anything when I gave the solution. The question doesn't seem to be clear enough. Otherwise there're some elegant ways to deal with both space and comma simultenously: stackoverflow.com/questions/4888879/… – Nawaz Apr 9 '11 at 20:34 ✎

I like the use of `istream_iterator` but why not finish strong using `ostream_iterator` as well? – user470379 Apr 9 '11 at 20:37

@Oli, @Steve and @Armen: Please see my second solution. And let me know if it can still be improved. :-) – Nawaz Apr 9 '11 at 20:47

@Nawaz: the obvious possible improvement would be to replace the final `for` loop with a call to `std::copy`. – Jerry Coffin Apr 9 '11 at 20:58

(b) Comments made to the reply.

For space separated strings, then you can do this:

```
std::string s = "What is the right way to split a string into a vector of strings";
std::stringstream ss(s);
std::istream_iterator<std::string> begin(ss);
std::istream_iterator<std::string> end;
std::vector<std::string> vstrings(begin, end);
~~for~~std::copy(~~size_t i = 0 ; i < ~~vstrings.~~size~~begin()~~ ;~~, ~~++i~~vstrings.end()
~~    ~~, std::~~cout << vstrings[i] << ~~ostream_iterator<std::string>(std::~~endl;~~cout, "\n"));
```

Online Demo : ~~http://ideone.com/tSOuM~~http://ideone.com/d8E2G

string that have both comma and space

```
struct tokens: std::ctype<char>
{
    tokens(): std::ctype<char>(get_table()) {}

    static std::ctype_base::mask const* get_table()
    {
        typedef std::ctype<char> cctype;
        static const cctype::mask *const_rc= cctype::classic_table();

        static cctype::mask rc[cctype::table_size];
        std::memcpy(rc, const_rc, cctype::table_size * sizeof(cctype::mask));

        rc[','] = std::ctype_base::space;
        rc[' '] = std::ctype_base::space;
        return &rc[0];
    }
};

std::string s = "right way, wrong way, correct way";
std::stringstream ss(s);
ss.imbue(std::locale(std::locale(), new tokens()));
~~std::vector<std::string> vstrings;~~
std::istream_iterator<std::string> begin(ss);
std::istream_iterator<std::string> end;
std::~~copy~~vector<std::string> vstrings(begin, end~~, ~~);
std::~~back_inserter~~copy(vstrings~~));~~
~~for~~.begin(~~size_t i = 0 ; i <~~), vstrings.~~size~~end()~~ ; ++i)~~
~~    ~~, std::~~cout << vstrings[i] << ~~ostream_iterator<std::string>(std::~~endl;~~cout, "\n"));
```

Online Demo : ~~http://ideone.com/8bJn4~~http://ideone.com/aKL0m

edited Apr 9 '11 at 21:06
Nawaz
166k ● 40 ● 347 ● 569

(c) Edits made to the reply.

3 +1: Very nice.. – Oliver Charlesworth Apr 9 '11 at 20:23

`std::vector<std::string> vstrings(begin, end);` would be nicer IMO, but I suppose we don't know whether the questioner is constructing the vector, or hoping to populate a pre-existing vector. – Steve Jessop Apr 9 '11 at 20:28 ✎

Nice, but wrong. The OP was specific in that both space and comma are delimeters. And you can't do the same trick in this case, can you? – Armen Tsirunyan Apr 9 '11 at 20:32

@Steve: Nice suggestion. @Armen: OP didn't mention anything when I gave the solution. The question doesn't seem to be clear enough. Otherwise there're some elegant ways to deal with both space and comma simultenously: stackoverflow.com/questions/4888879/… – Nawaz Apr 9 '11 at 20:34 ✎

I like the use of `istream_iterator` but why not finish strong using `ostream_iterator` as well? – user470379 Apr 9 '11 at 20:37

@Oli, @Steve and @Armen: Please see my second solution. And let me know if it can still be improved. :-) – Nawaz Apr 9 '11 at 20:47

@Nawaz: the obvious possible improvement would be to replace the final `for` loop with a call to `std::copy`. – Jerry Coffin Apr 9 '11 at 20:58

@Jerry: haha.. nice one. forgot that though other times I make use of it as well. thanks for reminding it. :-) – Nawaz Apr 9 '11 at 21:00

1 This is an amazing answer and needs to be highlighted somehow. – Samveen Apr 1 '13 at 11:07

@Samveen: Thanks for the appreciation. :-) – Nawaz Apr 1 '13 at 12:42

(d) Comments made to the reply.

Figure 5.11: Example reply and comments (2).

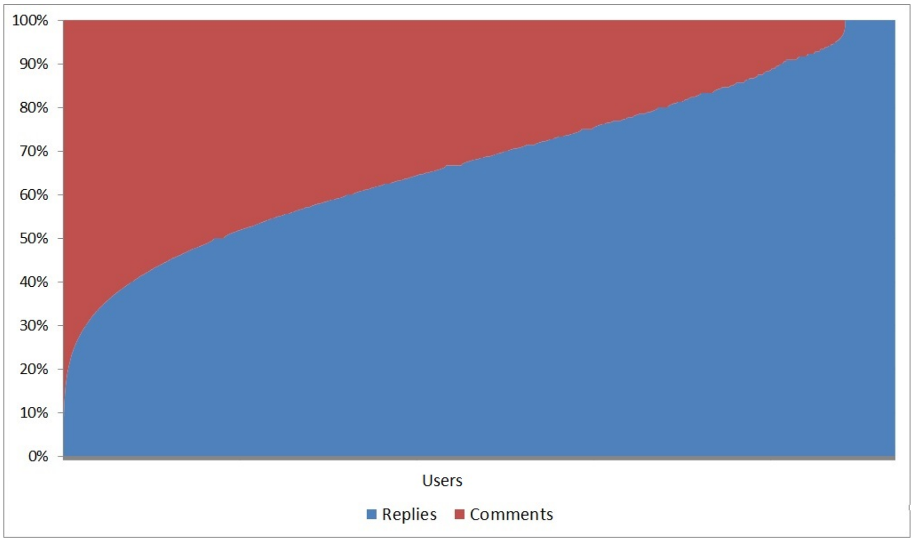


Figure 5.12: Answering and commenting activity distributions (in %) of users with at least 10 replies.

5.13. The drastic decrease in the frequency of commenters indicates that many users comment on their own posts and, therefore removed during pre-processing. However, there are still plenty of users who comment on others' posts, and among them there are users who reply to others questions. It has been found that among 869K responders, 22.5% of users also comment to questions and similarly 23% of users also comment to replies. In total 28% of responders comment either on questions or replies.

Given that there are users who only contribute very little, we also analyzed the commenting behavior among active responders. Users who replied to at least 10 questions, around 154K users, were analyzed specifically. Their answering and commenting activity distributions (in percentage) are presented in Figure 5.12. According to the figure, the percentage of users who both answer and comment are actually very high. Only around 6% of these users only answered but did not comment (right side of the figure, users with all blue). This figure shows that most of the active responders are also contributing with comments. Therefore, comments can be another source of evidence for modeling expertise of users.

5.2.3 Using Comments in Expertise Estimation

The high frequency of comments and the common use of comments by responders make them a valuable source for identifying expertise. The identified commenting behaviors in Section 5.2.1 are yet other reasons to use comments in expertise estimation. Both categories of commenting on questions, asking for or suggesting a clarification, and answering the question, can be considered as signals of expertise on the specific topic of question[14]. With respect to comments made on replies, the ones making suggestions to the reply or correcting it can be considered as positive signal of expertise; however, comments praising the reply are not as definite. Acknowledging the correctness of reply or thanking the responder can be performed by either expert users or

[14] Even though asking for clarification seems like a weak form of evidence for expertise, it is a step towards providing an accurate reply to the question.

users who find the information provided useful. However, in this dissertation, both types of comments made on replies are considered as useful information, and all comments are used to model expertise.

5.2.4 Experiments with Different Representations of Information Need and User Expertise

Since all types of commenting activities are considered as positive expertise signal as in the case of answering activities, the same document and profile-based approaches were also applied to comments in order to estimate expertise. In using replies, the previous section explored different types of user expertise and information need representations. It has been found that for question routing task, using question tags for both representing the user expertise and information need outperforms all other representation combinations in both the document and profile-based approaches (Tables 5.9 and 5.11). On the other hand, for reply ranking task, searching question tags over user representations constructed from answer bodies performed better than all other representations in both approaches (Tables 5.10 and 5.12). Therefore, in this section question tags and comment bodies (instead of answer bodies) representations of user expertise are tested with different representations of information needs.

Similar to the experiments with replies, only comments that are posted before the posting time of the particular question were used in experiments. In order to see the effects of comment types on expertise estimation more clearly, comments on questions and replies were analyzed individually. Experiments were performed for both question routing and reply ranking tasks.

5.2.4.1 Experiments on Question Routing Task

The experimental results of question routing task from the profile and document-based approaches are presented in Tables 5.14 and 5.15. For both approaches, using comments does not outperform replies (Tables 5.3 and 5.5), however they are still comparably effective. Even with using only comments either made on questions or answers, at least one actual responder can be retrieved within the top 10 ranked candidates 26.80% of the time (as seen in Table 5.15 rows 6 and 12, where question tags are used to represent user expertise and information need). This shows the effectiveness of comments as a source of evidence for expertise retrieval.

Furthermore, trends observed with replies are also observed with comments. The document-based approach started with higher baseline performances compared to the profile-based, but the profile-based approach caught the document-based approach, and for some metrics even performed better, when question tags are used for both representing user expertise and information need.

Using question tags for representing information need generally provided statistically significant improvements over other representations of information need. Only with the document-based approach when comment body was used to represent user expertise, using tags as query did not provide consistent improvements, similar to what was observed with reply bodies. As explained before this may be due to the vocabulary difference between these fields, which affects document-based approaches more since the effectiveness of the approach depends on the retrieved top *n* documents. Even though tag queries did not perform well on retrieving body fields with the document-based approach, using them to retrieve tag fields outperformed all other representations.

Parent Type	User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Question	Comment Body	Body	.0144	.0104	.0068	.0517	.0680	.1000	.1320	.0285
		Title	.0152	.0128	.0090	$.0527_{s}$	.0760	.1120	.1560	$.0374^{r}_{s}$
		Tag	.0216	.0168	$.0130^{r}_{s'}$	$.0705^{r'}_{s}$	.1040	.1520	$.2120^{r}_{s}$	$.0535^{r}_{s}$
	Question Tag	Body	.0136	.0084	.0074	.0370	.0600	.0760	.1360	.0336
		Title	.0104	.0104	.0090	$.0474_{s}$	.0440	.0960	.1520	$.0437^{r}_{s}$
		Tag	$.0328^{r}_{s}$	$.0256^{r}_{s}$	$.0198^{r}_{s}$	$.1125^{r}_{s}$	$.1520^{r}_{s}$	$.2160^{r}_{s}$	$.3000^{r}_{s}$	$.0796^{r}_{s}$
Answer	Comment Body	Body	.0144	.0096	.0072	.0507	.0680	.0920	.1200	.0314
		Title	$.0224^{r'}_{s'}$	$.0164^{r}_{s}$	$.0110^{r}_{s}$	$.0778^{r}_{s}$	$.1000^{r'}_{s'}$	$.1480^{r}_{s}$	$.1880^{r}_{s}$	$.0434^{r}_{s}$
		Tag	.0264	.0204	$.0140_{s'}$	$.0764^{r}_{s}$	.1200	.1680	.2160	$.0541^{r}_{s}$
	Question Tag	Body	.0144	.0108	.0074	.0488	.0640	.0960	.1280	.0344
		Title	.0184	.0148	$.0114^{r'}_{s}$	$.0712^{r}_{s}$	.0800	.1280	$.1920^{r}_{s}$	$.0529^{r}_{s}$
		Tag	$.0440^{r}_{s}$	$.0308^{r}_{s}$	$.0240^{r}_{s}$	$.1409^{r}_{s}$	$.1920^{r}_{s}$	$.2560^{r}_{s}$	$.3640^{r}_{s}$	$.0888^{r}_{s}$

Table 5.14: Question routing performance of profile-based approach applied to comments with different fields used for representing user expertise and information need.

Parent Type	User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Question	Comment Body	Body	.0248	.0204	.0164	.0802	.1080	.1800	.2760	.0597
		Title	.0232	.0224	.0170	.0856	.1080	.2040	.2920	.0546
		Tag	.0248	.0256	.0180	.0849	.1240	.2160	.3000	.0595
	Question Tag	Body	.0224	.0188	.0138	.0829	.1000	.1640	.2160	.0574
		Title	.0240	.0188	.0152	.0816	.1000	.1640	.2480	.0555
		Tag	$.0352^{r}_{s}$	$.0320^{r}_{s}$	$.0250^{r}_{s}$	$.1215^{r}_{s}$	$.1560^{r}_{s}$	$.2680^{r}_{s}$	$.3960^{r}_{s}$	$.0772^{r}_{s}$
Answer	Comment Body	Body	.0360	.0288	.0202	.1206	.1640	.2520	.3240	.0703
		Title	.0344	.0256	.0206	.1156	.1640	.2240	.3240	.0635
		Tag	.0384	.0268	.0192	.1193	.1800	.2280	.2960	.0667
	Question Tag	Body	.0248	.0200	.0154	.0796	.1000	.1600	.2440	.0515
		Title	.0280	.0236	.0180	.0904	.1160	.1880	.2760	.0571
		Tag	$.0408^{r}_{s}$	$.0320^{r}_{s}$	$.0256^{r}_{s}$	$.1372^{r}_{s}$	$.1680^{r}_{s}$	$.2680^{r}_{s}$	$.3960^{r}_{s}$	$.0753^{r}_{s}$

Table 5.15: Question routing performance of document-based approach applied to comments with different fields used for representing user expertise and information need.

With respect to comparing comments made on questions and answers, commenting on answers seems to be more effective in general. One possible reason for this can be that commenting on a reply can be a good indication of expertise, if the comment is constructed to improve upon the reply. For cases when the commenter's comments are accurate, the commenter can even be treated to be more expert on the topic of question compared to the corresponding reply's responder.

5.2.4.2 Experiments on Reply Ranking Task

Experimental results of reply ranking task with the profile and document-based approaches applied to comments are presented in Tables 5.16 and 5.17. Similar results observed with using replies (in Tables 5.4 and 5.6) are also observed with comments. Interestingly, with the profile-based experiments (Table 5.16), using comments made on answers returned better performance

Parent Type	User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Question	Comment Body	Body	.4569	.5549	.6425	.7136	.7960	.1760
		Title	$.4995^{r}_{s'}$	$.6060^{r}_{s}$	$.6753^{r}_{s}$	$.7382^{r}_{s}$	$.8155^{r}_{s}$	.1960
		Tag	$.5335^{r'}_{s}$	.6239	.6907	$.7563^{r}$	$.8266^{r'}$	$.2400^{r'}_{s'}$
	Question Tag	Body	.4849	.5876	.6725	.7378	.8119	.2000
		Title	$.5335^{r}_{s'}$	$.6200^{r}$	$.6944^{r'}$	$.7561^{r}$	$.8272^{r}$	$.2520^{r'}_{s'}$
		Tag	$.5677^{r'}_{s'}$	$.6443^{r'}$	.7121	$.7719^{r'}$	$.8391^{r'}$	.3000
Answer	Comment Body	Body	.4726	.5546	.6497	.7236	.8006	.1800
		Title	$.5525^{r}_{s}$	$.6282^{r}_{s}$	$.6976^{r}_{s}$	$.7583^{r}_{s}$	$.8305^{r}_{s}$	$.2760^{r}_{s}$
		Tag	.5464	.6286	.7026	.7593	.8300	.2680
	Question Tag	Body	.4926	.5824	.6675	.7335	.8099	.1960
		Title	$.5413^{r}$	$.6296^{r}$	$.6963^{r}_{s'}$	$.7586^{r}_{s'}$	$.8294^{r}$	$.2560^{r'}_{s'}$
		Tag	.5640	.6389	$.7118^{r'}$	.7654	.8353	.2760

Table 5.16: Reply ranking performance of profile-based approach applied to comments with different fields used for representing user expertise and information need.

Parent Type	User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Question	Comment Body	Body	.5506	.6319	.7007	.7580	.8317	.2800
		Title	.5483	.6433	.6995	.7572	.8311	.2560
		Tag	.5660	.6483	.7140	$.7703^{r'}$	.8391	.2840
	Question Tag	Body	.5187	.6143	.6845	.7469	.8218	.2240
		Title	.5118	.6176	.6839	.7453	.8211	.2120
		Tag	$.5562^{r}$	$.6471^{r}$	$.7113^{r}_{s}$	$.7704^{r}$	$.8373^{r}_{s}$	$.2760^{r'}_{s'}$
Answer	Comment Body	Body	.5779	.6537	.7178	.7704	.8417	.3000
		Title	.5815	.6467	.7092	.7644	.8394	.3080
		Tag	.5845	.6503	.7154	.7709	.8418	.3120
	Question Tag	Body	.5432	.6242	.6887	.7520	.8265	.2680
		Title	.5593	.6285	.6923	.7551	.8311	.2760
		Tag	.5606	.6300	.7015	.7596	.8321	.2760

Table 5.17: Reply ranking performance of document-based approach applied to comments with different fields used for representing user expertise and information need.

than using the answers itself (Table 5.4). However, using answers is still better compared to using comments made on questions. This result can be explained with one of the motivations behind commenting on answers, which is to suggest a correction or improvement. In such commenting situations, commenter is probably more expert than the responder which explains the better ranking of experts who also comments. Even though using answer comments generally work better than using question comments, the best performance is retrieved when question tags are searched over tags of the commented questions. This is yet another example of how representation can change the relative ranking of the approaches or evidence used.

In terms of the document-based approach, unlike the profile-based approach, using replies in expertise estimation performed better than using comments either on questions or answers. Answer comments working better than question comments has been explained before. The

Parent Type	User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Question	Comment Body	U-Tag	.0220	.0165	.0135	.0682	.1050	.1450	.2100	.0542
		In-Tag	.0200	.0170	.0140	.0721	.0950	.1400	.2150	$.0580^{r}_{s}$
		TG-Tag	.0250	.0170	.0140	$.0741_{s}$	.1200	.1450	.2150	$.0581^{r}_{s}$
		CP-Tag	.0180	.0140	.0120	.0594	.0800	.1200	.2000	$.0583^{r'}_{s}$
		U$_{.5}$/IN$_{0}$/TG$_{0}$/CP$_{.5}$	.0180	.0150	.0133	.0680	.0850	.1300	.2250	.0579
	Question Tag	U-Tag	.0320	.0265	.0195	.1140	.1450	.2200	.2800	.0812
		In-Tag	.0320	.0265	$.0223^{r}_{s'}$	.1126	.1450	.2200	$.3150^{r'}_{s'}$	$.0859^{r}_{s}$
		TG-Tag	.0340	$.0315^{r}_{s'}$	$.0230^{r}_{s}$	.1123	.1450	.2450	$.3250^{r}_{s}$	$.0851_{s}$
		CP-Tag	.0380	.0295	$.0238^{r}_{s}$	.1101	.1600	.2300	$.3400^{r}_{s}$	$.0864_{s}$
		U$_{.1}$/IN$_{0}$/TG$_{.9}$/CP$_{0}$	.0340	.0300	$.0228_{s}$	.1116	.1450	.2450	$.3200_{s}$	$.0847_{s}$
Answer	Comment Body	U-Tag	.0290	.0230	.0153	.0775	.1300	.1850	.2250	.0574
		In-Tag	.0280	.0220	.0155	$.0868_{s}$	.1150	.1850	.2600	.0574
		TG-Tag	.0290	.0235	.0158	$.0858_{s'}$	.1250	.1950	.2500	$.0578_{s}$
		CP-Tag	.0230	.0190	.0160	.0775	.1050	.1650	.2550	.0579
		U$_{0}$/IN$_{.1}$/TG$_{.4}$/CP$_{.5}$	.0250	.0200	.0163	.0860	.1100	.1600	.2650	$.0607'_{s}$
	Question Tag	U-Tag	.0450	.0305	.0245	.1399	.1950	.2500	.3650	.0889
		In-Tag	.0460	.0340	.0260	$.1508_{s}$	.1950	.2750	.3800	$.0950^{r}_{s}$
		TG-Tag	$.0550^{r}_{s}$	$.0380^{r}_{s}$	$.0280^{r}_{s}$	$.1589^{r'}_{s}$	.2250	$.3000^{r}_{s}$	$.4000^{r'}_{s'}$	$.0975^{r}_{s}$
		CP-Tag	.0500	$.0385^{r'}_{s}$	.0270	.1338	.2000	$.3050^{r'}_{s'}$	$.4150^{r'}_{s'}$	$.0982^{r}_{s}$
		U$_{.2}$/IN$_{.8}$/TG$_{0}$/CP$_{0}$	.0470	.0335	.0250	$.1484_{s}$	.1950	.2700	.3750	$.0942_{s}$

Table 5.18: Question routing performance of profile-based approach applied to comments with weighted tags.

reason why answer comments do not work as well as the answers with the document-based approach can be due to the low frequency of answer comments (3.6M) compared to frequency of answers (12.6M). Since the initial retrieval of documents plays a crucial role in the performance of document-based approaches, the fewer question-relevant documents retrieved can cause low performance in general. However, similar to the trend observed with using replies, question tags searched over comment bodies outperformed other representation combinations, probably due to the same reason of using actual words of the user being more informative than the tags of the commented answers' questions, while ranking responders based on their expertise.

5.2.4.3 Summary

Experiments show that comments can be used as another evidence for expertise estimation. Depending on the task and algorithm used, commenting on replies may be even more effective than the responding interaction itself, which shows the effectiveness of this content type. Furthermore, the same user expertise and information need representations are used with comments, and returned similar trends with replies. With respect to our research question this also shows the consistency of the proposed representations across different interaction types. The following section investigates how the proposed question tag weightings work over comments.

5.2.5 Experiments with Weighting the Question Tags

Experiments with weighted tags are also applied to comments in order to see whether the performance improves, and similar trends retrieved with replies are observed.

Parent Type	User Exp.	Inf. Need	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Question	Comment Body	U-Tag	.0250	.0265	.0188	.0868	.1250	.2200	.3050	.0604
		In-Tag	.0240	.0235	.0175	.0858	.1150	.2050	.2900	.0632
		TG-Tag	.0240	.0245	.0195	.0841	.1150	.2100	.3150	$.0629^{r'}_{s}$
		CP-Tag	.0220	.0225	.0168	.0745	.1050	.2000	.2650	.0616
		$U_{.8}/IN_{.1}/TG_{.1}/CP_0$	.0280	.0265	.0188	$.0884_{s'}$	.1350	.2250	.3050	$.0666_{s}$
	Question Tag	U-Tag	.0370	.0330	.0260	.1210	.1600	.2650	.4050	.0789
		In-Tag	.0340	.0325	.0250	.1166	.1500	.2650	.3900	.0761
		TG-Tag	.0370	.0335	.0258	.1221	.1600	.2700	.4050	.0780
		CP-Tag	.0350	.0290	.0233	.1182	.1550	.2500	.3750	.0769
		$U_0/IN_0/TG_1/CP_0$	.0370	.0335	.0258	.1221	.1600	.2700	.4050	.0780
Answer	Comment Body	U-Tag	.0410	.0280	.0203	.1206	.1900	.2350	.3050	.0671
		In-Tag	.0370	.0265	.0215	.1208	.1750	.2300	.3350	.0680
		TG-Tag	.0360	.0270	.0213	.1193	.1700	.2500	$.3450^{r'}_{s'}$	.0675
		CP-Tag	.0360	.0250	.0193	.1126	.1700	.2300	.3200	.0664
		$U_{.4}/IN_{.1}/TG_{.5}/CP_0$	.0410	.0280	$.0220_{s'}$	$.1232_{s}$	.1950	.2500	$.3450_{s}$	$.0732_{s}$
	Question Tag	U-Tag	.0400	.0315	.0263	.1386	.1650	.2600	.4050	.0747
		In-Tag	.0380	.0315	.0265	.1333	.1600	.2550	.4000	.0719
		TG-Tag	.0390	.0315	.0258	.1348	.1650	.2600	.3950	.0721
		CP-Tag	.0430	.0345	.0265	.1419	.1850	.2750	.4000	.0796
		$U_{.1}/IN_{.1}/TG_0/CP_{.8}$	.0440	.0350	.0270	.1412	.1900	.2800	.4100	.0803

Table 5.19: Question routing performance of document-based approach applied to comments with weighted tags.

5.2.5.1 Experiments on Question Routing Task

The experimental results of question routing task with weighted tags are presented in Tables 5.18 and 5.19 respectively for the profile and document-based approaches. In Table 5.18, with the profile-based approach, the *TG-Tag* provided consistent and sometimes statistically significant improvements in general, similar to the trend observed with replies in Table 5.9. For the document-based approach, inconsistent behaviors are observed in Table 5.19, as for some cases or metrics, weighting question tags helps, while for the rest it hurts. This inconsistent behavior of tag weighting with the document-based approach is also similar to the behaviors of replies from Table 5.10. Overall, in Table 5.19, among the individually tested weightings, only the *TG-Tag* weighting provided statistically significant improvements for some metrics and representations, which means that *TG-Tag* is also relatively more consistent than other weightings. As we mentioned before, this may be due to question routing task works better with more general representations of expertise.

5.2.5.2 Experiments on Reply Ranking Task

The results of reply ranking experiments with different types of weightings are summarized in Tables 5.20 and 5.21. Results retrieved with the profile and document-based approaches are similar to results retrieved from answering activities. For the profile-based approach (Table 5.20), the *CP-Tag* weighting works better with comment bodies, while *TG-Tag* and *IN-Tag* perform similarly in question tag representation of user expertise. For the document-based approach (Table 5.21), *IN-Tag* performs better than others for most representations except when it is used over answer comment bodies. In that specific representation *CP-Tag* performs significantly better.

Parent Type	User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Question	Comment Body	U-Tag	.5189	.6100	.6786	.7478	.8190	.2350
		In-Tag	.5004	.6029	.6721	.7421	.8141	.2150
		TG-Tag	.5090	.6051	.6737	.7437	.8160	.2200
		CP-Tag	.5174	.6117	.6808	.7482	.8189	.2350
		U$_0$/IN$_0$/TG$_1$/CP$_0$	.5090	.6051	.6737	.7437	.8160	.2200
	Question Tag	U-Tag	.5505	.6292	.7026	.7635	.8311	.2950
		In-Tag	.5527	.6253	.7000	.7630	.8296	.2950
		TG-Tag	.5622	.6300	.7027	.7654	.8326	.3100
		CP-Tag	.5595	.6301	.7005	.7617	.8313	.2850
		U$_0$/IN$_0$/TG$_{.2}$/CP$_{.8}$	.5497	.6244	.6963	.7601	.8284	.2750
Answer	Comment Body	U-Tag	.5341	.6184	.6936	.7525	.8235	.2650
		In-Tag	.5330	.6228	.6975	.7548	.8248	.2650
		TG-Tag	.5402	.6259	.7003	.7566	.8268	.2750
		CP-Tag	.5632	.6366	.7037	.7615	.8314	.3050
		U$_{.4}$/IN$_{.3}$/TG$_{.3}$/CP$_0$	.5298	.6244	.6976	.7542	.8244	.2600
	Question Tag	U-Tag	.5374	.6173	.6967	.7533	.8238	.2500
		In-Tag	.5477	$.6328^{r}_{s'}$	$.7029^{r}_{s'}$	$.7576^{r}_{s}$	$.8285^{r}_{s}$	.2650
		TG-Tag	.5437	$.6286^{r}$	.7016	.7555	$.8270^{r'}$	.2600
		CP-Tag	.5391	.6305	.6955	.7594	.8267	.2550
		U$_0$/IN$_{.1}$/TG$_{.9}$/CP$_0$	.5437	.6286	.7016	.7555	.8270	.2600

Table 5.20: Reply ranking performance of profile-based approach applied to comments with weighted tags.

The experimental results of using weighted tags over replies and comments return very similar results. This similarity is yet another indication of the consistency of the proposed weightings.

5.2.6 Summary

This section addresses research question *RQ1* by exploring comments as another resource for expert identification. It has been shown that comments can be as effective as replies and in some cases even better. Furthermore, getting similar trends from both replies and comments shows the consistency of the proposed representations of users' expertise and information needs, and weighting of query terms.

Sections 5.1 and 5.2 analyze the available representations of expertise (the content types and the structure within contents) in CQA sites. Our other data collection, the intra-organizational blog collection, consists of only the blog posts and comments, and the expert search queries consists of key terms only. In this setting, the available key term queries are used directly to search over users which are represented with their own content (posts and comments). The following section applies state-of-the-art algorithms to these representations.

5.3 Content-based Expert Retrieval in Blogs

Blogs are web documents which are written for a general audience for the purpose of sharing information. Unlike emails, they are not private but instead mostly publicly available. Unlike

Parent Type	User Exp.	Inf. Need	NDCG @1	NDCG @2	NDCG @3	NDCG @4	NDCG @5	BAP
Question	Comment Body	U-Tag	.5475	.6335	.7043	.7609	.8313	.2700
		In-Tag	.5572	.6375	$.7074_{s'}$	.7634	.8332	.2900
		TG-Tag	.5546	.6330	.7059	.7609	.8319	.2850
		CP-Tag	.5509	.6304	.6971	.7563	.8285	.2700
		$U_0/IN_{.5}/TG_0/CP_{.5}$	$.5627_{s'}$	.6362	$.7026_s$	$7605_{s'}$	$.8328_{s'}$	.2850
	Question Tag	U-Tag	.5424	.6341	.7040	.7639	.8310	.2700
		In-Tag	.5481	.6314	.7049	.7628	.8315	.2750
		TG-Tag	.5378	.6332	.7047	.7628	.8299	.2650
		CP-Tag	.5477	.6352	.7049	.7641	.8320	.2700
		$U_0/IN_0/TG_{.1}/CP_{.9}$	.5492	.6336	.7049	.7645	.8322	.2750
Answer	Comment Body	U-Tag	.5587	.6361	.7014	.7599	.8321	.2900
		In-Tag	.5457	.6300	.6961	.7550	.8279	.2750
		TG-Tag	.5535	.6347	.6975	.7571	.8297	.2800
		CP-Tag	$.5890_s$	$.6484_{s'}$	.7145	$.7686_s$	$.8387_s$	.3250
		$U_0/IN_0/TG_0/CP_1$	$.5890_s$	$.6484_{s'}$	.7145	$.7686_s$	$.8387_s$	.3250
	Question Tag	U-Tag	.5505	.6169	.6919	.7518	.8253	.2700
		In-Tag	.5582	$.6287^r$	$.7007^{r'}$	$.7567^{r'}$	$.8296^{r'}$	.2750
		TG-Tag	.5539	.6225	.6976	.7542	.8276	.2700
		CP-Tag	.5448	.6233	.6953	.7539	.8264	.2600
		$U_0/IN_1/TG_0/CP_0$	.5582	$.6287^r$	$.7007^{r'}$	$.7567^{r'}$	$.8296^{r'}$	.2750

Table 5.21: Reply ranking performance of document-based approach applied to comments with weighted tags.

the content in CQA sites (such as answers), these blog posts are independently constructed without the need for any other user inference (such as questions)[15] Furthermore, unlike emails or answers in CQAs, these blog posts are not constructed as an information exchange between a set of users, therefore they are more content free, as they can be on any topic their authors want them to be.

As for the expert retrieval queries, 40 queries, which were either selected by the company employees or selected from the company's search engine logs, were used as described in Section 4.1. They are keyword like queries and consist of several terms. 13 of them are single term queries, and the rest of them are mostly phrase like queries, such as "*virtual assistant*" or "*cloud computing*". The dataset's structure and test queries and how they are assessed are similar to TREC expert finding task data collections and test sets. Therefore, content-based expert retrieval approaches, that are described in Chapter 2 and tested on organizational documents, can be directly applied to these documents without the need for any customization.

As explained in Chapter 2, expert finding approaches can be categorized into four categories; (1) profile-based, (2) document-based, (3) graph-based and (4) learning-based. Since our test collection is limited with 40 topics, learning-based approaches are not very suitable. Instead the

[15] Post and its comments within blogs may look similar to question and its answers in CQAs, since both posts and questions are prerequisite for comments and answers. However, in terms of expertise estimation, the precondition in blogs, the blog post, is more important and useful than the comment; while for CQA sites, the precondition, the question, is an indication of lack of expertise and the answer is an indication of expertise. Therefore, showing expertise in blogs is more independent than showing expertise in CQA environments.

following methods from other three categories have been tested[16].

- Profile-based: The same profile-based approach applied to CQA sites has been also applied to the blog environment. A single profile is built for each user using all blog posts written by the user.
- Document-based: The similarity between Balog's Model 2 [8] and the Voting Models [61] have been described in Section 2.1.3. Since voting models provide more options for aggregating the documents, they have been chosen as the document-based approaches for the following experiments. *Votes*[17], *ReciprocalRank*, *CombSUM* and *CombMNZ* approaches, explained in Section 2.1.3, are applied to the bloggers of the retrieved blog posts. During retrieval only the top n documents are retrieved to identify the expert candidates for a given topic. Initial experiments performed on the data revealed that retrieving the top 1000 blog posts provides high baseline scores.
- Graph-based: The *Infinite Random Walk* (IRW) model from multi-step relevance propagation algorithms [84] (Section 2.1.4) is applied as the graph-based approach. Different λ values are tested.

The results of these experiments are summarized in Table 5.22. In the table, the first two columns present the approach and parameters used if there is any. The rest of the columns present the scores for different metrics and assessments values. In tables, *VE* is short for *very expert*, *AE* stands for *an expert* and *SE* is used for *some expertise*. As mentioned in Section 4.1.1, these are the types of assessment scores used during manual assessments to assess the candidate experts. In *VE* columns, only the candidates with *VE* assessment score are assumed as relevant while the others are treated as non-relevant. In the columns with +*AE*, candidates who get either *VE* or *AE* are treated as relevant. The final columns present results when all the candidates with scores either *VE* or *AE* or *SE* are assumed as relevant. With such an experimental evaluation, the assessment values are not graded anymore but instead they are binary; therefore metrics like P@10 and MAP are used to present the results for different assessment values. The last column summarizes the *NDCG* score which is a graded relevance metric that takes into account all relevance degrees.

According to Table 5.22, the *Reciprocal Rank* approach outperforms all the other models which suggests that highly ranked documents contribute more to the expertise of a candidate. *CombSUM*, which is using the relevance scores of the retrieved documents, follows the *Reciprocal Rank*. Among the four voting models, the *Votes* model performs the worst and *CombMNZ*, which is a mixture of *Votes* and *CombSUM*, performs in between. Even though, the *Profile-based* approach outperforms the *Votes*, it still does not beat other document-based approaches. Similarly, as reported in the previous work, the *IRW* approach outperforms the *Votes* approach, but it does not work as well as other stronger voting-based approaches. Due to its effective performance, the *Reciprocal Rank* approach was used as the content-based approach for blog collection in this thesis.

[16] Some approaches tested in here were not applied to CQA dataset, because they do not perform as well as the approaches that were tested already in previous chapters. For instance, *ReciprocalRank* outperforms other tested approaches in blog environment, however, it performs very badly in CQA environments, due to giving relatively too much value to responders associated with the top ranked questions compared to responders of other questions.

[17] *Votes* is the document-based approach that is most similar to the *Answer Count* approach used in CQA experiments

		Levels of Expertise						
Approach	Subtype / Parameters	VE		+AE		+SE		NDCG
		P@10	MAP	P@10	MAP	P@10	MAP	
Profile		.1950	.3236	.4425	.4489	.5800	.4649	.8011
Document	Votes	.1750	.2566	.3150	.3010	.4825	.3828	.4140
	ReciprocalRank	.2700	.4501	.5800	.6565	.7600	.7138	.7281
	CombSUM	.2600	.4826	.4725	.5488	.6450	.6094	.6168
	CombMNZ	.2275	.3722	.3900	.4022	.5550	.4774	.5124
IRW	λ = 0.01	.2375	.3680	.4000	.4169	.5575	.4923	.7475
	λ = 0.05	.2350	.3633	.3950	.4099	.5525	.4858	.7447
	λ = 0.10	.2350	.3477	.3925	.3999	.5475	.4781	.7380

Table 5.22: Expert ranking performance of content-based approaches in blog collection.

5.4 Summary

Expert finding in community question answering sites has been widely studied under question routing and reply ranking tasks. This chapter analyzed the representation of expertise in these environments and tried to address the following research question:

- *RQ1: What are the most effective representations of information need and user expertise used for identifying expertise in question routing and reply ranking tasks in CQAs?*

Most of the prior research used user's answers or answered questions' title and bodies to model user expertise in CQA sites. However, important details of the question can be ignored in replies due to the threadlike user interface of these systems. On the other hand, unnecessary details can be included into replies' and questions' bodies, which may be a cause for topic drift in modeling user expertise. The prior work also used the title and/or body of the question as the information need which may also cause topic drifts in expert finding. Using specific information needs directly for retrieving expert users may not be the optimum way for CQA environments. This is mainly because in these environments a specific expertise can be shown only after some users ask about that specific detail and no other user has provided an accurate answer yet. Instead, using more general categorizations of the particular information need (underlying knowledge areas required to answer the question) can be more useful for expert identification. Therefore this thesis proposes using the question tags to represent both information need and the users' expertise based on the intuition that tags consist of important key terms representing the question.

This proposed representation was tested with two state-of-the-art approaches in order to see whether its effectiveness is algorithm independent. Using tags to represent the information need provided statistically significant improvements to other representations of questions in both tasks with both approaches. This shows the effectiveness of using important and also (somewhat) more general query terms instead of using more query terms. In case tags do not exist, such as in the case with Yahoo! Answers, the question titles which are the closest thing to tags are observed to be more effective than other representations.

In user expertise representation, the best accuracy for question routing and reply ranking tasks were received with using different fields. For question routing task, using the tags of the questions users answered returned higher scores, as expected, since users may select questions based on their categories. On the other hand, for reply ranking task, responders' previous answers were more effective in representing their expertise for ranking their later answers. This also makes sense since the votes replies receive depend on the clarity and presentation of answers (more specifically the use of terms) as well as their accuracy.

Additional evidence of expertise, the comments, has been also explored. Comments have not been explored in detail with respect to their effects on expertise identification for question routing and reply ranking tasks. Therefore, we initially analyzed a group of comments manually, and identified several motivations of commenting on questions and answers. These analyses revealed that comments can be posted by users who can be as expert or even more expert than the corresponding asker and responders, which make them a useful evidence for expertise estimation. Our experiments with using comments for estimating expertise returned comparable performance to using replies. In some experimental settings for the reply ranking task, comments even outperformed replies. Such cases occur when user comments on an answer in order to suggest an improvement or correction. This can be an indication of commenter's question-specific expertise which can be even more than responder's expertise. Experiments performed on comments were also useful in terms of checking the consistency of the experimental results from using answers to estimate expertise. For both representation of information need and user expertise, comments returned similar outcomes to what was retrieved with using answers. This shows that the proposed representations and approaches are not only significant but also consistent.

With respect to the research question *RQ1*, it has been found that using question tags to represent the information need outperformed all other representations for both question routing and reply ranking tasks. In terms of user expertise representation, question tags performed the best for question routing task but body of answers or comments returned the highest accuracy in reply ranking task.

Not directly related to the research question, but an interesting observation is the effect of representations on the relative ranking of expert finding algorithms. With previous representations of expertise, the document-based approach, which is a topic-specific approach, seems to be more effective than less topic-specific approaches like the profile-based approach. But, with the question tag representation of users and information needs, the profile-based approach also performed as good as the document-based one and even performed better for some metrics. These results show the effects of representation on relative ranking of approaches. Improving the effectiveness of the profile-based approaches to the level of the document-based approaches is also important with respect to computational efficiency. Profile-based approaches, with one phase of search, are much more efficient compared to document-based approaches which have two phases of search, an initial search over the documents in collection and another search over the associated users.

In addition to identifying an important source of information for representing expertise, weighting that information is also important. Not all tags may be equally important or useful for expertise estimation. Some tags can be more important or general than others, and should affect the expertise score more than others. Therefore, this dissertation proposed three weighting schemes, which are based on askers' ordering of tags, tag generality and expertise areas of probable experts. The proposed tag weightings provided statistically significant improvements over uniform tag weightings which shows that not all tags are equally important for representing the information need.

Overall, the weighted tag queries returned the highest scores for both question routing and reply ranking tasks. For question routing task, using tag generality weighting returned the highest improvements with the profile-based approach. Tag weightings in the document-based approach returned inconsistent results due to being too much dependent on top retrieved documents. For reply ranking task, using asker's ordering of the information need returned consistent improvements with the document-based approach, while weights retrieved from probabilistically expert users' returned statistically significant improvements with the profile-based approach when applied to responders' answer bodies. These differences between question routing and reply ranking tasks may be due to differences in their definition of expertise that is being searched. Question routing task defines expertise for a given question in more general way, while reply ranking task looks for more question-specific experts, who not only provide possible accurate answers but also provide them in a descriptive way.

The effects of these weightings are especially important for showing term specificity which is a widely accepted and used term weighting in Information Retrieval, may not always work for all environments and tasks. In our case, term generality seems to be more useful for identifying experts for question routing task. Search tasks that prefer generality of terms over their specificity may prefer this term weight to be less than usual. Weighting tags experiments performed on comments also returned similar results which show the consistency of the proposed weightings.

In this chapter, the experiments performed with both the profile and document-based approaches on two expertise related tasks on CQAs demonstrated the consistent effectiveness of the proposed representations of expertise. This effective and approach-independent representation was powerful enough to change the relative ranking of approaches. Expertise representation in social media and its effects in retrieval is observed to be different than other retrieval tasks. This is also important for other expertise retrieval approaches applied to social media, and whether their widely accepted assumptions hold in these environments as expected. The following chapter analyzes the authority-based approaches, which have been originally developed for web pages, and their adaptations to user networks.

Chapter 6

Authority-based Approaches

Content-based modeling of expertise favors users who write a lot about a topic; however being prolific on a specific topic does not necessarily make one an expert. Since anyone can create content in social media, it is not enough to just write about a topic to be called an expert. Being read, voted and commented by other users, and leaving an influence on social media community regarding the particular topic is also as important. Therefore, this dissertation explores the existing user interactions in the form of user networks in order to identify influential expert users. Similar to web page graphs with web pages as nodes and url links as edges, user interaction graphs are also constructed where users are connected through user interactions. Since applying network-based approaches to these web page graphs returns authoritative web pages, over time in the literature, these network-based approaches have been referred to as the authority-based methods and web page graphs as the authority graphs. The same terminology is also used in this dissertation for the network-based (authority-based) expertise estimation methods and the user interaction (authority) networks.

Many link-based authority estimation approaches have been developed for web page graphs which are constructed by using the links among web pages. For instance PageRank [15] and HITS [45] are the two well-known authority estimation approaches for web pages. As described in Section 2.4.2, similar methods have been applied to users in social networks in order to identify authoritative experts whose posts receive many views, votes or comments. Variants of PageRank and HITS were adapted to user networks in different social media environments in order to estimate authority-based expertise scores. However, these approaches did not return consistent results across different environments and networks. Therefore, as stated in the research question *RQ2*, this dissertation initially investigates these link-based algorithms and the user authority graphs in order to identify the underlying unsatisfied assumptions. Depending on the findings, modifications either to the authority graphs or algorithms are proposed in order to improve the effectiveness and efficiency of authority-based expertise estimation approaches.

In the following sections, first, background information on how PageRank and HITS algorithms are applied to user networks in general are described. Then, the proposed more topic specific authority graphs and algorithms are presented. These proposed approaches were applied to both blog and CQA datasets. Findings from these experiments are summarized at the end of this chapter.

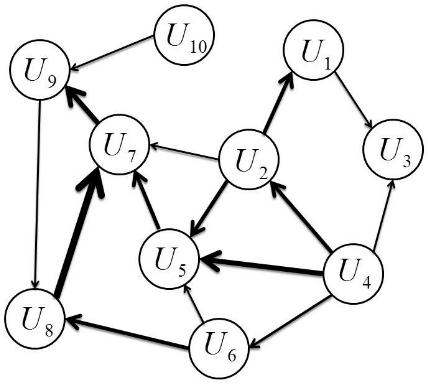


Figure 6.1: An example user authority network.

6.1 Background on User Authority Estimation

Before getting into details of approaches for estimating authority, the user authority networks should be described. Authority-based approaches use the relationship and interaction among entities to measure the influence and importance of each entity. Authority is measured over a graph in which nodes are the users, and directed edges indicate interaction between these users. These interactions can be commenting to another's post or answering question of an user who seeks information. Depending on the type of interaction, the direction of the edge can be an indication of authority. Interactions such as answering somebody's question are strong indications of question-specific authority of the responder over the asker. For such cases the direction of the edge is from asker to responder so that through authority estimation iterations, the authority score propagates more towards to these responders. An example user authority network is presented in Figure 6.1. As can be seen in the figure, the edges can also be weighted by the frequency of the interaction between connected users.

Unlike typical web page authority graphs, user authority graphs can have weighted links among nodes. It is less common to see multiple links between pairs of web pages; however interactions between users can be more frequent. In web pages, links are added more carefully, however interactions among users can be random or even accidental. Differentiating these one-time interactions from more regular and frequent interactions can be important for effective authority estimations. The thickness of the edges in graph in Figure 6.1 represents the frequency of these interactions which are used as weights in authority estimation algorithms.

6.1.1 PageRank and Topic-Sensitive PageRank

For user-authority estimation, *PageRank (PR)* [15] can be thought as the probability distribution representing the likelihood of reaching a user by randomly following authoritative links among users. Depending on the type of links (interaction they represent) a high PageRank score can be an indication of authoritative user. It has been widely applied to user networks in order to estimate users' expertise [13, 22, 28, 42, 104]. In this dissertation, the estimated *PageRank* scores are also directly used as the authority-based expertise scores of users.

PageRank is a topic-independent algorithm that considers all users and their activities over all the documents, therefore it is applied to the whole user authority network as shown in Figure 6.1. It is normally applied to unweighted web graphs for estimating authority of web pages. Its customized version for estimating authority among users in unweighted graphs is as shown:

$$PR(u) = \frac{1-d}{|U|} + d \sum_{i \in IL_u} \frac{PR(i)}{|OL_i|} \tag{6.1}$$

where $PR(u)$ is the *PageRank* score of user u, IL_u is the set of users that are linked to u (incoming links), $PR(i)$ is the *PageRank* score of user i, OL_i is the set of users that are linked from i (outgoing links), and so $|OL_i|$ is the number of outgoing links from user i. The d in Equation 6.1 refers to damping factor. The teleportation probability is uniformly distributed between all users, $1/|U|$, where $|U|$ is the number of users in the graph.

Same interaction can occur between same users multiple times over different posts in different times. Therefore, same type of interactions among users can be aggregated to determine the weight of the edge (as shown in Figure 6.1 with thickness of edges). *PageRank* scores can be calculated on these weighted graphs as shown:

$$PR(u) = \frac{1-d}{|U|} + d \left(\sum_{i \in IL_u} \frac{L(i,u)}{\sum_{j \in OL_i} L(i,j)} PR(i) \right) \tag{6.2}$$

where $PR(u)$ is the PageRank score of user u, IL_u is the set of users that are linked to u (incoming links), $L(i,u)$ is the weight of the edge (frequency of interactions) from i to u, OL_i is the set of users that are linked from i (outgoing links), $L(i,j)$ is the weight of the edge (frequency of interactions) from i to j, and $PR(i)$ is the *PageRank* score of user i.

In above equation, if the $L(i,u)$ and $L(i,j)$ weights are set to 1 as in the case of unweighted graphs where all multiple interactions between users are counted as 1, then this equation will be exactly same to Equation 6.1. In *PageRank* calculations on weighted graphs, compared to unweighted graphs, instead of just using the number of outgoing links from a node, the probability of following a link depends on the proportion of the weight of the edge to sum of weights of all the outgoing edges.

Topic-Sensitive PageRank (TSPR) [37] assumes that teleportation is possible only to users that are associated with topic-relevant content. Therefore, in topic-sensitive PageRank, unlike the regular PageRank, the teleportation probabilities are distributed uniformly among users who have created topic related content which has been retrieved as a result of searching the topic over the document collection. Instead of using a teleportation probability of $1/|U|$ for every user, the probability $1/|U_t|$ where U_t is the set of users associated with topic t, is used for users whose content have been retrieved for the particular topic. For the rest of the users, 0 is used as the teleportation probability. Both *PageRank* and *Topic-Sensitive PageRank* algorithms are applied to the whole network, which consists of all users and all interactions among them. Such a network is useful for identifying authorities in general; however identifying more topic-specific authorities can be harder. *TSPR* favors users that are associated with topic-relevant content, but it still does not differentiate whether the edges are topic-relevant or not. Therefore, more topic-specific authority networks, which focus more on topic-relevant nodes and edges, were proposed in the literature.

6.1.2 Hyperlink-Induced Topic Search (HITS)

HITS has been also used by prior research to estimate users' authority-based expertise scores [13, 20, 22, 28, 42, 43, 104]. Unlike *PageRank*, *Hyperlink-Induced Topic Search (HITS)* [45] algorithm uses a topic-specific subgraph instead of the whole graph and for each node it calculates two types of scores, *authority* and *hub*. The algorithm consists of several iterations and at each step first the *authority* and then the *hub* scores are updated. *Authority* score of a node is equal to the sum of the *hub* scores of the nodes of incoming edges. Similarly *hub* score is equal to the sum of the *authority* scores of the nodes of outgoing edges.

The default *HITS* algorithm is applied to unweighted graphs, but a customized version of *HITS* can also be applied to graphs with weighted edges as in user graphs. In such a graph, the *auth* and *hub* scores are calculated as shown:

$$Auth(u) = \sum_{i \in IL_u} L(i,u)Hub(i) \tag{6.3}$$

$$Hub(u) = \sum_{i \in OL_u} L(u,i)Auth(i) \tag{6.4}$$

where $Auth(u)$ is the authority score of user u, IL_u is the set of users that are linked to u (incoming links), $L(i,u)$ is the weight of the edge from i to u (similar to PageRank this represents the frequency of interactions), and $Hub(i)$ is the hub score of user i. Similarly, $Hub(u)$ is the hub score of user u, OL_u is the set of users user u is connected to (outgoing links), $L(u,i)$ is the weight of the edge from u to i, and $Auth(i)$ is the auth score of user i. In these equations if $L(i,u)$ and $L(u,i)$ are 1, then these will be same to HITS on unweighted graphs.

With respect to applying *HITS* to social networks, one can think of the users with high *authority* scores as the authoritative users whose content attracts the attention of many active users. Similarly, users with high *hub* scores are the active users who interact a lot with authoritative users. For instance in the blogosphere, a good *hub* is a user who reads or comments on many blog posts that also receives attention from other users, and a good *authority* is a user whose posts have been read or commented on by other users who also interact with many other users. In such a scenario, using the *HITS authority* score directly for estimating a person's authority-based expertise is a perfect fit. It can be represented as shown:

$$HITS(u) = Auth(u) \tag{6.5}$$

where $HITS(u)$ represents the *HITS* score of user u calculated either on weighted or unweighted graphs.

Kleinberg applied *HITS* to topic-specific authority sub-graphs with the aim to focus the computational effort on highly topic-relevant documents [45], instead of using all web pages as in compared to *PageRank*. Kleinberg's approach for constructing the *HITS* authority network is also used in order to construct more topic-specific user networks. Such a sub-graph is constructed by initially retrieving the top n topic-specific expert candidates, which is called the *root set*. This *root set* consists of users who have been retrieved by the content-based expert finding approaches. Later on this *root set* is expanded into a *base set* which consists of users who have interacted with these candidates in the *root set*, either by being connected to or connected from. Such a *base set* contains all the users within the *root set*. After creating this *base set*, a graph is constructed by using all the candidates within this set as nodes and existing interactions among them as edges. An example *root set*, *base set* and the constructed graph is given in Figure 6.2. Compared to the *PageRank* graph in Figure 6.1, this is a more topic-dependent authority network. As expected the *HITS* graph is not as dense as the *PageRank* graph but still contains many nodes and edges from *PageRank* graph.

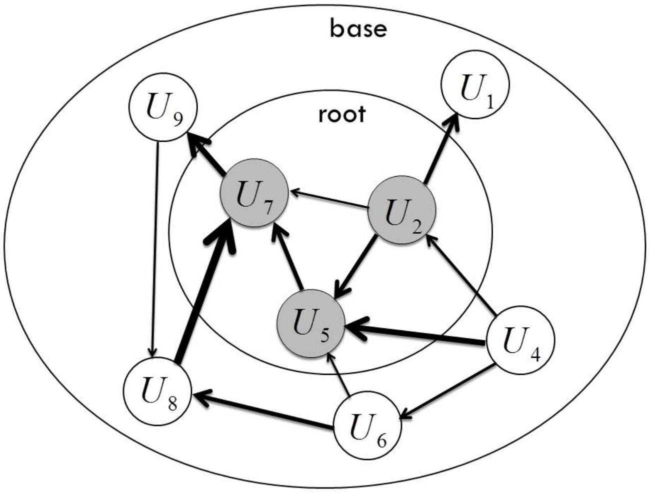


Figure 6.2: Expanding the root set (in grey) into a base set and constructing a HITS graph.

6.2 Constructing More Topic-Specific Authority Networks

The *HITS* algorithm uses *base set* nodes and edges among them to construct topic-specific subgraphs, which is well suited for web graphs in which each node (web page) is mainly about one topic. However, there are several issues that need to be considered in using HITS graphs for users and interactions among them. First, unlike web pages, users are not interested in or knowledgeable on only one topic. Instead they can be experts on or interacting with several topics, which are either related or not related to the particular topic. Because of these different types of interactions, during *base set* construction not all the inserted users and interactions between these users and the *root set* users will be topic-relevant. Since all the interactions of users are used during this expansion, the final constructed *HITS* graph will still contain many topic-irrelevant user nodes and interactions. Existence of such nodes and interactions may cause iterating the authority to topic-irrelevant users and favoring users who may be authorities on unrelated topics.

Furthermore, unlike web graphs, which are constructed by using the inlink URLs provided within pages, the user authority networks are not always constructed from explicit and intentional interactions. Depending on the social media and user interactions, there may be less intentional and sometimes even accidentally inserted edges among user nodes as a result of activities such as accessing (viewing) a post. Such non-deliberate actions cause many more edges to be included into the graph, compared to more purposefully added links among web pages. Due to these reasons, using the *PageRank* and *HITS* graphs may not be the optimum choice for estimating authority between highly connected users.

This thesis proposes constructing and using more topic-focused authority sub-graphs, called *Topic-Candidate (TC)* graphs, to estimate topic-specific authorities in user networks. The *TC* graphs are constructed by using interactions only from topic-relevant posts, rather than using interactions from all posts. These topic-related posts are identified by a document search engine built over the collection. Using the same notation used in *HITS* graphs, candidates retrieved with the content-based approach can be referred as the *root set*, while the set of users in the *root set* together with the users directly connected with them construct the *base set*. However,

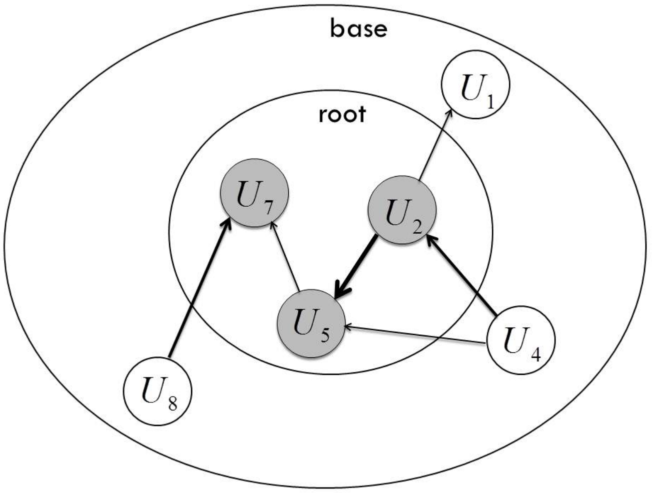


Figure 6.3: Removing topic irrelevant edges from HITS graph and constructing a Topic Candidate graph.

unlike *HITS* graphs, only users in base set that are connected to/from users in root set due to topic-relevant activities are used. All other topic-irrelevant users or interactions are ignored. An example *Topic-Candidate* graph is given in Figure 6.3. Compared to *HITS* graph in Figure 6.2, the number of nodes and edges is less, and in some cases the weight of some edges is also lower.

All the edges in Figure 6.3 are originated from topic-relevant interactions, and all the remaining edges and nodes that are missing in this figure but existed in Figure 6.2, are not related to the topic. If one takes a closer look at these graphs, they may see the topic-irrelevant interactions and nodes in Figure 6.2 that do not exist in Figure 6.3. For instance, edges $U_7 \rightarrow U_9$, $U_6 \rightarrow U_5$ and $U_2 \rightarrow U_7$ should be topic-irrelevant edges, since they don't exist in Figure 6.3. Removing edges $U_7 \rightarrow U_9$ and $U_6 \rightarrow U_5$ also caused users U_9 and U_6 to be removed, mainly because these users don't have any other topic-relevant connections to user nodes in the root set. Sometimes, only some percentage of the interactions between users can be topic-irrelevant, therefore removing these interactions do not cause these edges to be removed but only cause a decrease on their weights. Such edges with their weights decreased are $U_8 \rightarrow U_7$, $U_2 \rightarrow U_1$, $U_5 \rightarrow U_7$ and $U_4 \rightarrow U_5$.

In *PageRank* and *HITS* graphs (Figure 6.1 and 6.2), due to using all user interactions, U_7 favors U_9 with its authority. But when only the topic-relevant interactions are used, U_7 no more influences its authority to node U_9 which prevents U_9 to be estimated as an authority on the particular topic. To sum up, as can be observed in Figure 6.3, in *TC* graphs all the edges are originated from user interactions performed on topic-relevant content and they are either directed to or from one of the topic-relevant content authors (the nodes in *root set*). Possible sparsity issues of this proposed graph is discussed over the constructed authority networks.

In this section, the user authority graphs, the topic-independent PageRank and topic-dependent HITS graphs are analyzed and a new authority graph construction approach is proposed in order to construct more topic-specific user graphs. Now that the graphs are better adapted to the users, the authority estimation algorithms are analyzed in the following two sections to improve their effectiveness with the proposed graphs.

6.3 Using Expertise in Authority Estimation

The advantage of authority-based approaches, like *PageRank*, compared to more counting-based approaches, like *InLink*, is that the former ones take into account not just the number of users connected (number of incoming links) to the particular user node, but also the authority of the connected nodes. Using such approaches help to differentiate between being connected from a newbie or an authoritative user; however, they still don't take into account the topic-specific expertise of users. Since our aim is to create a ranking of users based on their topic-specific expertise, being connected from an expert should not be counted same as being connected from a topic-wise inexperienced user. Based on this assumption, more topic-specific authority-based approaches are proposed to estimate not only authoritative users but authoritative expert users.

6.3.1 Using Expertise as Influence

In the PageRank algorithm, an incoming link from an authority is much more important and effective than an incoming link from a non-authoritative user. Similarly, one can also argue that an incoming link from a topic-specific expert is much valuable than an incoming link from an inexperienced user. Based on this assumption, *ExpertiseInfluenced* authority estimation approaches are proposed which use an initially estimated topic-specific expertise score as influence of the user's node in iterations.

In *ExpertiseInfluenced* approach, initially estimated content-based expertise scores are used as weights on user nodes and integrated into the PageRank formula as shown:

$$PR_{EI}(u) = \frac{1-d}{|U|} + d\left(\sum_{i \in IL_u} \frac{L(i,u)}{\sum_{j \in OL_i} L(i,j)} PR_{EI}(i)\, E(i,t)\right) \tag{6.6}$$

where $PR_{EI}(u)$ is the expertise influenced PageRank score of user u, and $E(i,t)$ is an initially estimated content-based expertise score of user i with respect to topic t. The $E(i,t)$ score is independent from the authority estimation, and can be calculated from any type of information. Setting it equal to 1 for all users is same as the regular PageRank. In this dissertation, the scores estimated from the best content-based approach are used to set these values.

In addition to PageRank, the *ExpertiseInfluenced-HITS (HITS$_{EI}$)* approach is also proposed, which uses the initially estimated topic-specific expertise scores as node weights to influence nodes that are connected to them. These nodes' weights are influenced to other nodes as follows:

$$Auth_{EI}(u) = \sum_{i \in IL_u} L(i,u)\, Hub_{EI}(i)\, E(i,t) \tag{6.7}$$

$$Hub_{EI}(u) = \sum_{i \in OL_u} L(u,i)\, Auth_{EI}(i)\, E(i,t) \tag{6.8}$$

$$HITS_{EI}(u) = Auth_{EI}(u) \tag{6.9}$$

A similar relevancy-based HITS approach was also proposed by Bharat and Henzinger [11] in order to solve the irrelevant web pages problem in *HITS* algorithm. The authors calculated the relevance scores of web pages, and used them to regulate the influence of nodes with the aim to reduce the influence of less relevant nodes on the scores of connected nodes. In this thesis, a similar topic-based relevance approach is used to regulate the influence of users' expertise in order to improve the relative ranking of topic-specific authorities. Our user graphs are a little different than their web graphs, due to multiple interactions among users, which is less common in web page graphs.

In these approaches, the initially estimated topic-specific expertise of user is influenced to other connected users. So, these scores do not help to their owners, but instead help to other users who are linked from these users. However, these content-based expertise scores of users can be also used to improve their own authority-based expertise scores as shown in the next section.

6.3.2 Using Expertise in Teleportation

PageRank-like approaches estimate authorities through random walks over the authority graphs, as both jumping to other nodes (teleporting) and following outgoing links (transitioning) are random choices. The initially estimated expertise scores can be used to decrease the randomness in these approaches, and increase the probability of visiting topic-specific nodes. Based on this idea, as mentioned in the Related Work chapter, Zhou et al. [106] and Yang et al. [98] used some initially estimated scores in teleportation vectors to favor jumps to users who have shown more expertise than other users. In order to analyze the effects of expertise influenced authorities more clearly, these expertise teleported authorities are also tested.

ExpertiseTeleported PageRank, is calculated as shown:

$$PR_{ET}(u) = (1-d)\frac{E(u,t)}{\sum_{j}^{|U|} E(j,t)} + d\left(\sum_{i \in IL_u} \frac{L(i,u)}{\sum_{j \in OL_i} L(i,j)} PR_{ET}(i)\right) \tag{6.10}$$

where $PR_{ET}(u)$ is the expertise teleported weighted PageRank score of user u. $E(u,t)$ is the expertise score of user u on topic t. This score is normalized with respect to other users' expertise score on topic t, $(\sum_{j}^{|U|} E(j,t))$. Therefore, user u with a higher $E(u,t)$ have higher probability of being teleported to. HITS approach does not have a teleportation part, therefore there is not a *ExpertiseTeleported HITS* approach.

The effectiveness of this approach depends on the estimated expertise score $E(u,t)$. If $E(u,t)$ is estimated by an approach that uses similar information (same or similar links), then favoring users with high expertise score through teleportation does not make too much difference because they were already being favored due to their incoming links in the transition (second) part of the equation above. However, for instance, using an $E(u,t)$ score which has been estimated from blog post content of user u, on authority graphs built from reading or commenting activities, can provide observable changes in the estimated authority scores due to combining different forms of evidence.

This idea of using an initially estimated retrieval score to improve the probability of that particular nodes' visits has been used on web graphs by Richardson and Domingos [76]. They proposed an intelligent surfer model, which is guided by a probabilistic relevance model of pages for a given query. In their *query-dependent PageRank* (*QDPR*) approach, both teleporting and transiting to a page j depends on the content of page j, more specifically, the page's relevance to the given query. This approach is different from Richardson and Domingos [76] in terms of the characteristics of the graphs and how the scores are calculated for a given query. The authors calculated *QDPR* scores for query terms offline in advance. In real time, for a given query with multiple-terms, the final QDPR score is calculated by combining PageRank scores from different query terms. With the availability of *Topic-Candidate* graphs which are very effective for real time estimations; we calculated one PageRank score for a given query with multiple terms. The difference of *ExpertiseTeleported PageRank* over the work of Zhou et al. [106] and Yang et al. [98] is also the type of graph it has been applied to, which is much more topic-specific.

The most important difference between expertise-influenced and expertise-teleported PageRank and HITS is that, in the former one expertise score is propagated to other nodes, so the high expertise score of a user improves the authority scores of connected users. In the other one, high expertise score improves the authority score of that particular user by improving its chances of getting more propagation.

In addition to making additional assumptions of using content-based expertise to improve network-based expertise scores (authority scores), the adaptation of widely used authority estimation algorithms to user networks with different types of interactions are also analyzed. It has been observed that authority estimation algorithms may work fine with some user interactions, but not all.

6.4 Adapting HITS to CQA Authority Networks

Several prior work [42, 104] applied HITS to CQA environments and used the *auth* score as the authority-based expertise score of users. *Auth* score is not independent from the *hub* scores of connected nodes, on the contrary, *hub* and *auth* scores are mutually reinforcing. Getting a high authority score depends on being connected from nodes with high hub scores. Zhang et al. [104] used the assumption that *"a good hub is a user who is helped by many expert users, and a good authority (an expert) is a user who helps to many good hubs"*. However mixed results were observed, as improvements were seen in one paper [42], while HITS performed the worst in another paper [104] compared to InDegree or PageRank. Zhang et al. [104] explained this low performance of HITS with the network structure. They also mentioned the problem of HITS for experts who help to other experts. Since experts have low HITS hub scores, they propagate a low score to the experts helping them, which cause these helpful experts to receive low HITS authority scores at the end. However, no solution was proposed to solve this problem.

In this dissertation, we will focus on a different problem of HITS when it is applied to CQA authority graphs with askers. In an asker-responder graph with edges directed from askers to responders, the explanation of authority and hub users will be similar to the above description of Zhang et al. [104]. Users with high hub scores will be the ones who ask a lot of questions which are answered by users with good authority scores. So, the hub score of an asker depends on two values, (1) the number of questions one asks that were answered, and (2) the authority scores of corresponding responders. These two variables are explicitly shown in the *Hub* scores calculations as shown:

$$Hub(u) = \sum_{i \in OL_u} Auth(i) \tag{6.11}$$

The summation over outgoing links, $\sum_{i \in OL_u}$ represents the number of asked questions that were answered, and the $Auth(i)$ score represents the authority scores of the responders. As it can be seen from the equation, if both of these variables are high, then a high hub score will be generated, which will also increase the authority scores of connected responders.

The second variable (the authority scores of responders) being high is an useful signal of authority for connected responders, since it means that the asker is asking good questions which are answered by authoritative users. Since authority score of a user is equal to the sum of the hub scores of askers of the corresponding questions user had answered, answering questions of users whose other questions are also answered by authoritative users is a positive signal for particular responders' authority.

However, the first variable (the number of answered questions one asked) is not a very authoritative signal. This variable being high means that user is asking lots of questions, which is an indication of user's lack of expertise on the topic. In terms of responders, answering questions of an asker who asks lots of questions is not as useful as answering questions of another user who asks relatively fewer questions (probably less inexperienced compared to the former one). Answering questions of more experienced users is an indication of a greater expertise. Therefore, the size of OL_u is important. Right now with the original HITS algorithm, it being more returns higher hub and authority scores. However, in terms of the underlying assumption of asker-responder networks, it being low is more reasonable for the estimated authority scores in the next iteration.

This thesis proposes an adapted hub score calculation for asker-responder networks by dividing the original hub score with the number of outgoing links, $|OL_u|$, in other words the number of answered questions asked by the user u. This hub score, referred to as Hub_{CQA}, is calculated as shown:

$$Hub_{CQA}(u) = \frac{\sum_{i \in OL_u} Auth(i)}{|OL_u|} \tag{6.12}$$

Dividing the original hub score with the number of outgoing links can be though as taking an average of the authority scores of connected users. This taking the average part decreases the negative effects of asking too many questions on hub scores, which in turn affects the authority scores. With this modification, the description of hub score in asker-responder networks becomes *"a good hub is a user who is helped by expert users on average"*.

The hub score calculation is modified for asker-responder networks, but the authority score is kept as same. During authority estimations on these networks, the frequency of incoming links represents the number of answered questions. That frequency being high is useful; therefore the summation is better than taking the average. Applying this modified HITS approach to weighted asker-responder networks in CQA environments is as follows:

$$\begin{aligned} wHub_{CQA}(u) &= \frac{\sum_{i \in OL_u} L(u,i) wAuth_{CQA}(i)}{\sum_{i \in OL_u} L(u,i)} && (6.13) \\ wAuth_{CQA}(u) &= \sum_{i \in IL_u} L(i,u) wHub_{CQA}(i) && (6.14) \\ wHITS_{CQA}(u) &= wAuth_{CQA}(u) && (6.15) \end{aligned}$$

In weighted Hub_{CQA}, the total weight of the outgoing links is used during division instead of the frequency.

6.5 Experiments

In order to estimate more topic-specific authority-based expertise scores, the previous sections proposed a topic-specific user authority graph construction approach, and influence based expertise weighted authority estimation algorithms. Furthermore, a modified version of HITS is proposed for asker-to-responder networks in CQAs. All these proposed approaches are tested on both the intra-organizational blog and StackOverflow collections. The rest of this section describes the constructed authority graphs, and presents the experimental results and summarizes the findings.

6.5.1 Authority Networks

The available authoritative signals and authority networks constructed from these signals are described for each dataset.

6.5.1.1 Blog Collection

The blog collection contains two types of user interactions: reading and commenting. In this thesis, these interactions are compared in terms of their effectiveness in estimating topic-specific authority. Commenting may be viewed as a stronger form of evidence because it requires individual to take an action. Commenting information is also more generally available because most blog applications display user ids next to comments. Reading may be viewed as a weaker form of evidence because it requires less effort and may even be accidental. However, reading is much more common than commenting, which might compensate for its weaknesses and make it more useful for low-traffic situations. Typically the user ids of readers are not displayed, which makes this form of evidence somewhat unique. Most of the time this information is only available to service providers. Because it is not publicly available like comments, reading information is not widely studied by researches.

Separate graphs are created for reading and commenting interactions. Example reading and commenting activities and corresponding authority graphs are shown in Figure 6.4. Figure 6.4(a) shows commenting activities and Figure 6.4(c) presents reading activities on three blog posts[1]. Figures 6.4(b) and 6.4(d) are the corresponding authority graphs of the activities. As can be seen from the graphs, the edges are from readers or commenters to authors; if a post attracts many comments, the author benefits, not the users who participate in the discussion. These edges are weighted by the number of blog posts written by $user_i$ and read (or commented) by $user_j$. This model does not consider whether $user_j$ read a specific post once or several times; only the total number of posts that were read is important.

During constructing the *HITS* and *Topic Candidate* graphs for blog collection, the top 1000 ranked expert candidates retrieved with the best content-based approach (*Reciprocal Rank*[2]) are used as the root set. This root set is expanded into base set and the authority network is constructed with the nodes in this base set. This best content-based approach, *Reciprocal Rank*, is also used to calculate the initial expertise scores of users that are used as weights in expertise influenced and teleported authority estimations.

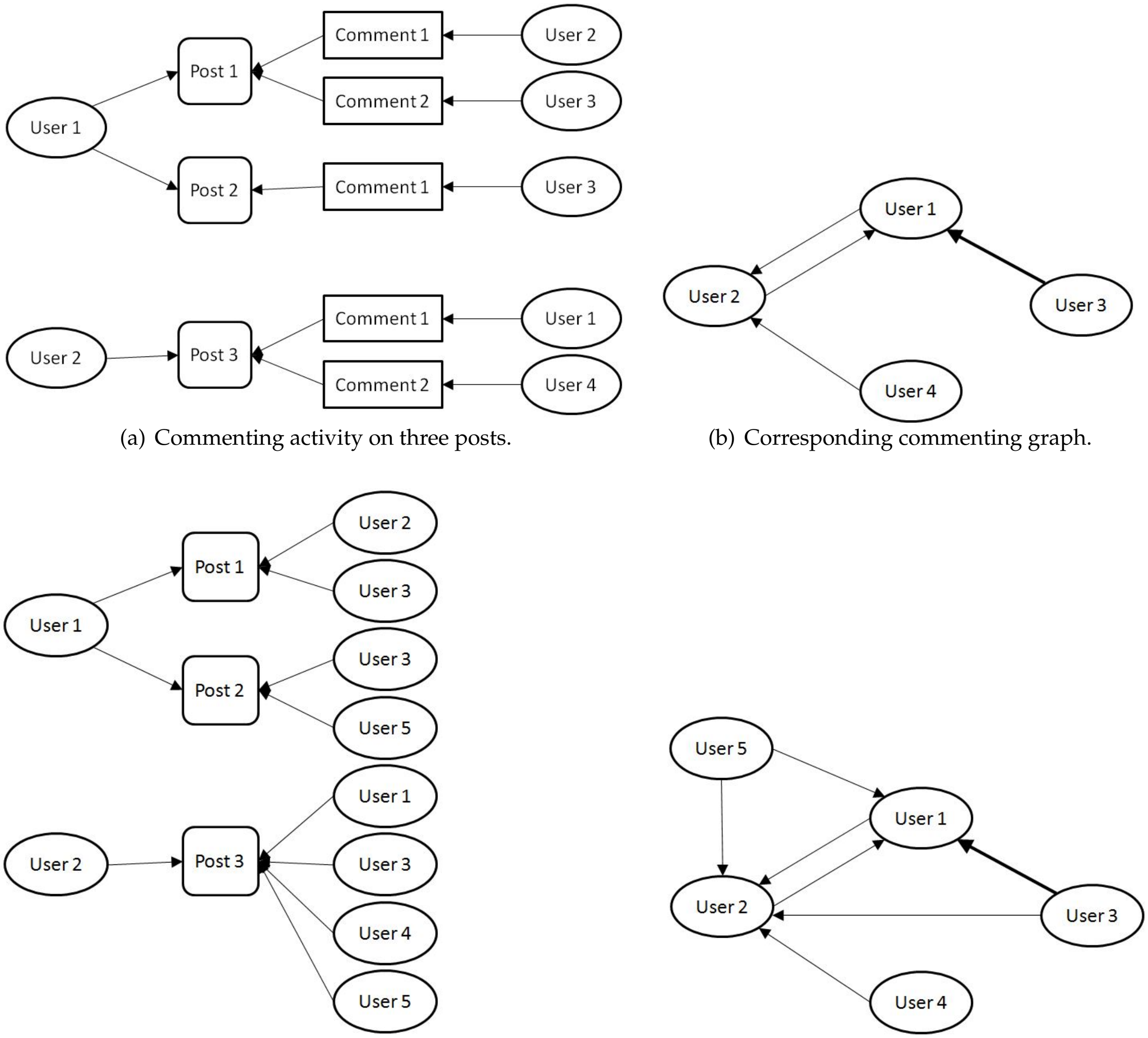
6.5.1.2 CQA Collection

In CQA environments, answering a question is a very strong indication of authority of responder over asker, since by answering questions, responders show that they are more authoritative on the topic of questions. Therefore, in an asker-responder authority network, the edges are directed from question askers to the corresponding responders. Such an answering interaction and corresponding asker-responder network is provided in Figures 6.5(a) and 6.5(b).

In Chapter 5, different representations of information need and user expertise were tried for effective content-based retrieval of experts. Among these representations using tags provided significant improvements over others, therefore tag representation is also used while constructing

[1] The users on the left (User 1 and User 2) are the authors while the users on the right are the ones reading the posts.

[2] The experimental results of applying content-based approaches to blog collection are presented in Section 5.3.



(a) Commenting activity on three posts.

(b) Corresponding commenting graph.

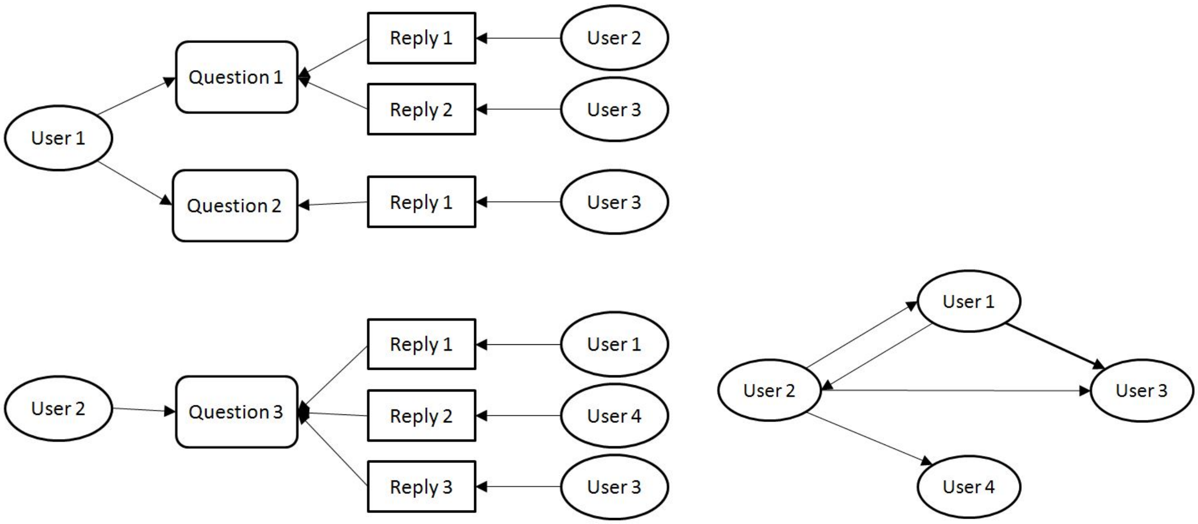
(c) Reading activity on three posts.

(d) Corresponding reading graph.

Figure 6.4: Example reading and commenting authority graphs.

HITS and *TC* graphs. For a given question and its tags, all users who have previously answered a question with any one of the corresponding tags are used as the root set.

For initial estimated expertise scores the best content-based expertise approaches that use responding activities from Chapter 5 for question routing and reply ranking tasks are used for the corresponding experiments. For question routing task, the best performing system is searching question tags (weighted by the combination of tag generality and candidate profile) over user profiles constructed from question tags of responded questions. For reply ranking task, it is applying document-based approach with question tag representations of information need (weighted by the combination of uniform and candidate profile) over answer bodies.



(a) Answering activity on three questions.

(b) Corresponding asker-responder graph.

Figure 6.5: Example asker-responder authority network.

6.5.2 Experiments with Topic-Candidate Graph

The initial set of experiments were performed to test the effectiveness of the the proposed topic-specific authority, *Topic Candidate (TC)* graph, over other commonly used authority graphs, *PageRank* (*PR*) and *HITS*. Three baseline authority estimation algorithms were applied to these three authority graphs as listed.

- PageRank (PR): Applying PR algorithm to PR (default), HITS and TC (proposed) graphs.
- Topic-Sensitive PR (TSPR): Applying TSPR algorithm to PR (default), HITS and TC graphs.
- HITS: Applying HITS algorithm to PR, HITS (default) and TC graphs.

Applying PageRank algorithm to HITS graphs, or applying HITS to PageRank graphs are not common in authority estimation. However, in order to better analyze the topic-specificity of these graphs with respect to different algorithms, possible combinations of authority graph and estimation approaches were tried.

For each PR, HITS and TC graph, the experiments were performed with weighted (by the frequency of the activity between two nodes) and unweighted graphs. The weighted graphs are presented with a 'w' symbol in the subscript, like PR_w, $HITS_w$ and TC_w, while unweighted graphs don't have any subscripts.

Usually authority-based approaches are combined with content-based approaches in some way for more effective performance. However, in the following experiments, the experimental results of authority-based approaches are presented independently without combining them with any other method, with the aim to see the effects of these authority-based methods and graphs more clearly. Even though using authority scores alone for ranking expertise may return low scores in general, it provides a better capability of comparing the proposed approaches and their effects on expertise retrieval. The combination of content-based and authority-based approaches is analyzed later in this dissertation.

		Levels of Expertise						
Algorithm	Graph	VE		+AE		+SE		NDCG
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	PR	.0100	.0080	.0200	.0115	.0450	.0159	.1247
	HITS	.0125	.0137	.0275	.0174	.0600	.0254	.1672
	TC	.0475^r_s,	.0792^r_s	.1075^r_s	.0919^r_s	.1700^r_s	.1227^r_s	.3371^r_s
TSPR	PR	.0300	.0493	.0700	.0800	.1175	.1170	.3947
	HITS	.0300	.0467	.0675	.0784	.1150	.1150	.3925
	TC	.0600^r_s	.0754^r	.1300^r_s	.1086^r_s	.2025^r_s	.1476^r_s	.3831
HITS	PR	.0100	.0092	.0175	.0079	.0400	.0133	.1068
	HITS	.0100	.0177	.0175	.0100	.0400	.0170	.1126
	TC	.0400^r_s	.0844^r_s	.1075^r_s	.0915^r_s	.1675^r_s	.1228^r_s	.3312^r_s

Table 6.1: Expert ranking performance of unweighted authority graphs constructed from reading activities in blog collection.

		Levels of Expertise						
Algorithm	Graph	VE		+AE		+SE		NDCG
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	PR	.0050	.0051	.0100	.0054	.0225	.0078	.0854
	HITS	.0075	.0139	.0250	.0180	.0625	.0292	.1859
	TC	.0425^r_s	.0827^r_s	.0950^r_s	.0886^r_s	.1525^r_s	.1137^r_s	.3333^r_s
TSPR	PR	.0400	.0947	.0925	.0985	.1525	.1328	.4167
	HITS	.0400	.0953	.0950	.0973	.1575	.1315	.4153
	TC	.0725^r_s	.1389	.1475^r_s	.1266^r	.2125^r	.1501^r	.3797
HITS	PR	.0100	.0071	.0200	.0057	.0425	.0106	.0879
	HITS	.0100	.0075	.0200	.0062	.0425	.0129	.0997
	TC	.0400^r_s	.0863^r_s	.0950^r_s	.0885^r_s	.1425^r_s	.1137^r_s	.3284^r_s

Table 6.2: Expert ranking performance of unweighted authority graphs constructed from commenting activities in blog collection.

6.5.2.1 Blog Collection

The experimental results on blog collection are summarized in Tables 6.1 and 6.2 (for unweighted graphs) and Tables 6.3 and 6.4 (for weighted graphs) respectively for reading and commenting activities. In tables, the first column presents the authority estimation algorithm and the second column shows the authority graph that was used in iterations. The rest of the columns present the scores for different metrics and assessments values. Similar to Table 5.22 from Section 5.3, results are presented for *very expert* (*VE*), *an expert* (*AE*) and *some expertise* (*SE*) users. P@10 and MAP metrics are used to present the results for these different assessment values. The last column summarizes the *NDCG* score which is a graded relevance metric that takes into account all relevance degrees.

First thing to notice in tables is the low performance of authority-based methods. Compared to content-based approaches (Table 5.22) which use user-created content of users to estimate their expertise, the low performance of authority-based expertise scores estimated from reading or commenting interactions of users are expected. Applying authority-based approaches over

Algorithm	Graph	Levels of Expertise						NDCG
		VE		+AE		+SE		
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	PR$_w$	.0100	.0081	.0200	.0112	.0475	.0165	.1243
	HITS$_w$	.0125	.0106	.0275	.0146	.0575	.0216	.1587
	TC$_w$	$.0500^{r,}_{s'}$	$.0802^{r}_{s}$	$.1075^{r}_{s}$	$.0926^{r}_{s}$	$.1775^{r}_{s}$	$.1248^{r}_{s}$	$.3395^{r}_{s}$
TSPR	PR$_w$	.0250	.0443	.0625	.0762	.1125	.1155	.3906
	HITS$_w$	.0250	.0433	.0550	.0746	.1050	.1131	.3889
	TC$_w$	$.0675^{r}_{s}$	$.0812^{r}_{s}$	$.1400^{r}_{s}$	$.1118^{r}_{s}$	$.2125^{r}_{s}$	$.1514^{r}_{s}$	.3882
HITS	PR$_w$	.0100	.0115	.0150	.0066	.0375	.0144	.0984
	HITS$_w$	.0100	.0115	.0150	.0066	.0375	.0144	.0994
	TC$_w$	$.0750^{r}_{s}$	$.1166^{r}_{s}$	$.1425^{r}_{s}$	$.1228^{r}_{s}$	$.2175^{r}_{s}$	$.1541^{r}_{s}$	$.3707^{r}_{s}$

Table 6.3: Expert ranking performance of weighted authority graphs constructed from reading activities in blog collection.

Algorithm	Graph	Levels of Expertise						NDCG
		VE		+AE		+SE		
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	PR$_w$	.0100	.0054	.0150	.0056	.0375	.0082	.0831
	HITS$_w$	.0100	.0134	.0250	.0169	.0625	.0270	.1840
	TC$_w$	$.0425^{r}_{s}$	$.0842^{r}_{s}$	$.0975^{r}_{s}$	$.0890^{r}_{s}$	$.1525^{r}_{s}$	$.1145^{r}_{s}$	$.3346^{r}_{s}$
TSPR	PR$_w$	.0350	.0940	.0875	.0939	.1475	.1287	.4121
	HITS$_w$	.0350	.0946	.0875	.0937	.1500	.1291	.4116
	TC$_w$	$.0675^{r}_{s}$	$.1333_{s'}$	$.1525^{r}_{s}$	$.1303^{r,}_{s'}$	$.2175^{r}_{s}$	$.1562^{r}$	$.3868^{r'}$
HITS	PR$_w$	.0050	.0058	.0075	.0043	.0200	.0078	.0838
	HITS$_w$	.0050	.0059	.0075	.0046	.0200	.0082	.0938
	TC$_w$	$.0650^{r}_{s}$	$.1122^{r}_{s}$	$.1375^{r}_{s}$	$.1097^{r}_{s}$	$.2150^{r}_{s}$	$.1421^{r}_{s}$	$.3643^{r}_{s}$

Table 6.4: Expert ranking performance of weighted authority graphs constructed from commenting activities in blog collection.

activities like reading or commenting may not return topic-specific experts but authorities which may be useful in improving the overall ranking of expert candidates which are identified with content-based approach. This is analyzed later in this thesis. In this chapter, in order to analyze the proposed authority-based approaches more clearly, the authority-based experiments are performed and presented individually without combining them with any other approach.

In all four tables, the baselines *PR* and *HITS* graphs perform very similar to each other for a given authority estimation approach, which may be due to the fact that *HITS* graph construction approach fail to construct topic-focused authority networks when applied to users. However, the proposed *TC* graphs provide statistically significant improvements over both graphs for different algorithms and changing assessment values. This consistent and significant improvements show the importance of working on topic-specific authority graphs. Similarly, the topic-specific *TSPR* approach performs much better than more topic-independent approaches like *PR* and *HITS* for both reading and commenting graphs. For both of these activities, applying *TSPR* algorithm to *TC* graphs returns the highest accuracy.

Comparing the results of unweighted and weighted tables (Table 6.1 with 6.3 and Table 6.2

Algorithm	Graph	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
PR	PR	.0176	.0156	.0154	.0697	.0880	.1520	.2440	.0869
	HITS	.0176	.0156	.0154	.0706	.0880	.1480	.2400	.0886
	TC	$.0512^r_s$	$.0408^r_s$	$.0304^r_s$	$.1560^r_s$	$.2240^r_s$	$.3240^r_s$	$.4520^r_s$	$.1308^r_s$
TSPR	PR	.0240	.0220	.0190	.0943	.1120	.2000	.3080	.1068
	HITS	.0240	.0228	.0190	.0940	.1120	.2080	.3080	.1061
	TC	$.0400^r_s$	$.0340^r_s$	$.0274^r_s$	$.1418^r_s$	$.1720^r_s$	$.2720^r_s$	$.4200^r_s$	$.1211^r_s$
HITS	PR	.0184	.0168	.0140	.0689	.0920	.1520	.2240	.0815
	HITS	.0184	.0168	.0140	.0687	.0920	.1520	.2240	.0809
	TC	$.0432^r_s$	$.0364^r_s$	$.0270^r_s$	$.1429^r_s$	$.1960^r_s$	$.2720^r_s$	$.3840^r_s$	$.1230^r_s$

Table 6.5: Question routing performance of unweighted authority graphs constructed from answering activities in StackOverflow collection.

Algorithm	Graph	NDCG@1	NDCG@2	NDCGP@3	NDCG@4	NDCG@5	BAP
PR	PR	.5353	.6136	.6949	.7550	.8242	.2200
	HITS	.5384	.6164	.6914	.7551	.8250	.2200
	TC	.5546	$.6408^r_s$	$.7022_s$	$.7681^r_{s'}$	$.8334^{r'}_s$	$.2640^{r'}_{s'}$
TSPR	PR	.5507	.6308	.6932	.7609	.8293	.2440
	HITS	.5507	.6279	.6922	.7601	.8287	.2440
	TC	.5486	.6417	$.7059^{r'}$	.7645	.8324	.2560
HITS	PR	.5180	.6013	.6694	.7414	.8153	.2040
	HITS	.5180	.6003	.6672	.7405	.8149	.2040
	TC	.5451	$.6361^r_s$	$.6933^r_s$	$.7586^r_s$	$.8287^r_s$	$.2440^{r'}_{s'}$

Table 6.6: Reply ranking performance of unweighted authority graphs constructed from answering activities in StackOverflow collection.

with 6.4) shows that weighting the edges with frequency of activities in *TC* graphs has little effect except for the *HITS* approach. Using weighted edges does not help so much to *PR* and *HITS* graphs, where there are more nodes and edges that are topic irrelevant than relevant. In these graphs giving more weight to frequent activities may direct authority more to topic irrelevant directions. On the other hand, in *TC* graphs, with topic-specific nodes and edges, weighting edges are actually improving the overall estimated authority of topic-specific expert candidates.

With respect to comparing the reading and commenting activities, there is not a clear winner. One may expect commenting to be more powerful since it is a more explicit form of authority signal compared to the more implicit reading signal. However, the results suggest that the higher frequency of the reading signal compensates for its weakness[3].

6.5.2.2 CQA Collection

The proposed *TC-Graph* was also applied to asker-responder networks in CQA environments in order to identify authoritative topic-specific experts for question routing and reply ranking tasks. The results of these experiments are summarized in Tables 6.5 and 6.6 for unweighted graphs and Tables 6.7 and 6.8 for weighted graphs.

[3] Statistics regarding the size of reading and commenting graphs are presented in Table 6.11. As it can be observed from the table, reading is a much more frequent than commenting in general.

Algorithm	Graph	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
PR	PR$_w$	.0168	.0152	.0152	.0711	.0840	.1480	.2400	.0867
	HITS$_w$	.0168	.0164	.0156	.0722	.0840	.1560	.2440	.0888
	TC$_w$	$.0504^r_s$	$.0400^r_s$	$.0304^r_s$	$.1574^r_s$	$.2240^r_s$	$.3240^r_s$	$.4440^r_s$	$.1313^r_s$
TSPR	PR$_w$	.0248	.0220	.0180	.0924	.1160	.2040	.3040	.1054
	HITS$_w$	.0248	.0216	.0176	.0923	.1160	.2000	.2960	.1048
	TC$_w$	$.0400^r_s$	$.0332^r_s$	$.0278^r_s$	$.1416^r_s$	$.1720^r_s$	$.2640^{r'}_{s'}$	$.4240^r_s$	$.1213^r_s$
HITS	PR$_w$	.0136	.0152	.0146	.0603	.0680	.1400	.2240	.0783
	HITS$_w$	.0136	.0156	.0148	.0610	.0680	.1440	.2280	.0783
	TC$_w$	$.0432^r_s$	$.0356^r_s$	$.0266^r_s$	$.1353^r_s$	$.1920^r_s$	$.2760^r_s$	$.3680^r_s$	$.1224^r_s$

Table 6.7: Question routing performance of weighted authority graphs constructed from answering activities in StackOverflow collection.

Algorithm	Graph	NDCG@1	NDCG@2	NDCGP@3	NDCG@4	NDCG@5	BAP
PR	PR$_w$	.5399	.6155	.6930	.7563	.8255	.2240
	HITS$_w$	.5380	.6157	.6912	.7549	.8248	.2200
	TC$_w$	.5551	$.6395^r_s$	$.7031_{s'}$	$.7681^{r'}_{s'}$	$.8333_{s'}$	.2640
TSPR	PR$_w$	.5517	.6346	.6947	.7625	.8302	.2440
	HITS$_w$	.5517	.6328	.6939	.7617	.8298	.2440
	TC$_w$	.5457	.6417	.7053	.7639	.8318	.2520
HITS	PR$_w$	.5152	.6025	.6710	.7421	.8155	.2040
	HITS$_w$	.5152	.6014	.6690	.7411	.8150	.2040
	TC$_w$	$.5630^r_s$	$.6449^r_s$	$.7009^r_s$	$.7657^r_s$	$.8347^r_s$	$.2640^r_s$

Table 6.8: Reply ranking performance of weighted authority graphs constructed from answering activities in StackOverflow collection.

In Chapter 5, document and profile-based approaches were applied to CQA data for question routing and reply ranking tasks. The results in these four tables are the authority-based approach results for the same tasks. All these three approaches use answering activity of users in order to initially rank them based on their expertise. Also, all three approaches use question tags to represent the information need and user expertise, therefore authority-based results in Tables 6.5, 6.6, 6.7 and 6.8 are very similar to profile and document-based results with uniformly weighted tag queries in Section 5.1.6.

In *TC* graph experiments, similar behaviors were observed in both question routing (Tables 6.5 and 6.7) and reply ranking (Tables 6.6 and 6.8) tasks. Similar to the results of blog experiments, using *TC* graph in authority estimation provided consistent and statistically significant improvements over the similarly behaving *PageRank* and *HITS* graphs. However, unlike the blog results, the best performance in both question routing and reply ranking tasks is observed when *PageRank* approach is applied to *TC* graph. Running *TSPR* over the *PageRank* and *HITS* graphs outperformed other approaches on the same graphs but not in *TC*.

As observed in tables, weighting the edges perform similarly with not weighting them in *PR* and *HITS* graphs. However, improvements are observed in both tasks when *HITS* approach is applied to *TC* graphs. These improvements are not as high as the improvements in reading and commenting authorities in Blog collection (relative improvements between 30-88% at P@10). Weighted and unweighted graphs are further analyzed for these collections and activities in

Activity	Graph	% Edges with Weight			
		1	2	3	> 3
Reading	PR	66.85%	14.67%	6.26%	12.22%
	HITS	65.78%	14.92%	6.44%	12.87%
	TC	95.35%	3.49%	0.74%	0.43%
Commenting	PR	72.22%	12.03%	4.85%	10.89%
	HITS	66.37%	13.06%	5.72%	14.85%
	TC	93.59%	4.72%	0.96%	0.73%

Table 6.9: Percentage of edges with varying weights in weighted graphs of Blog collection.

Task	Graph	% Edges with Weight			
		1	2	3	> 3
Question Routing	PR	94.51%	4.40%	0.72%	0.37%
	HITS	94.13%	4.67%	0.79%	0.41%
	TC	96.52%	2.90%	0.39%	0.20%
Reply Ranking	PR	94.63%	4.18%	0.74%	0.45%
	HITS	94.34%	4.39%	0.79%	0.48%
	TC	95.93%	3.20%	0.54%	0.33%

Table 6.10: Percentage of edges with varying weights in weighted graphs of StackOverflow collection.

order to better understand this difference.

6.5.2.3 Weighted vs. Unweighted Graphs

Two types of graphs, one with the edges weighted with the frequency of the activity between two users and one with not, are constructed for each activity and task. The percentages of edges with changing weights (frequency of activity) are presented in Tables 6.9 and 6.10 respectively for blog and StackOverflow collections.

As observed from the tables, the percentage of edges with weight equal to 1 (the edge is constructed due to a one time interaction between two people), can be also referred to as unweighted, is very high in CQA answering networks compared to reading and commenting authority networks in blog collection. This decrease in weighted edges is due to the low probability of users answering the same askers' questions for the second time, or even much lower as in the case of third time. On the other hand, for reading and commenting activities in blogs, users tend to follow blog posts of bloggers they find interesting, which cause many edges to be weighted.

Comparing the edge weights of different graphs for the same activity in Blog collection, *PR* and *HITS* graphs have the highest percentage of weighted edges. As expected the percentages of weighted edges are much lower in *TC* graphs, due to using only the topic-specific interactions between users, and ignoring the rest. As mentioned in Section 6.2, while constructing *TC* graphs, compared to *PR* and *HITS* graphs, we don't only remove topic irrelevant nodes but may also remove or decrease the weight of edges between two existing nodes.

Activity	Graph	# Nodes	# Edges	Running Times (ms)		
				PR	TSPR	HITS
Reading	PR	92,360	1,630,584	627,162	1,025,766	299,211
	HITS	56,909	1,480,100	406,254	991,131	266,971
	TC	6,561	9,080	264	388	3,389
Commenting	PR	42,567	214,069	37,261	57,735	56,712
	HITS	14,358	138,135	26,184	46,487	21,878
	TC	1,110	1,565	43	55	506

Table 6.11: Average number of nodes and edges in unweighted PR, HITS and TC graphs, and running times (in milliseconds) of PR, TSPR and HITS algorithms on these graphs for expert blogger ranking task.

Task	Graph	# Nodes	# Edges	Running Times (ms)		
				PR	TSPR	HITS
Question Routing	PR	136,103	1,174,619	65,670	64,564	57,497
	HITS	93,718	1,074,462	30,351	108,941	30,192
	TC	19,684	78,741	1,675	1,679	5,131
Reply Ranking	PR	347,424	2,799,378	636,985	679,697	661,286
	HITS	239,095	2,573,731	428,512	3,540,245	653,730
	TC	48,736	204,404	29,842	46,985	44,474

Table 6.12: Average number of nodes and edges in unweighted PR, HITS and TC graphs, and running times (in milliseconds) of PR, TSPR and HITS algorithms on these graphs for question routing and reply ranking task.

6.5.2.4 Efficiency Analysis

Other than the improvements in accuracy, using *Topic Candidate* graphs can also improve the running time performance of the applied authority estimation approaches due to not using the whole graph but instead using a topic-specific part of it. Tables 6.11 and 6.12 present the average number of nodes and edges within *PR*, *HITS* and *TC* graphs, and the average running time (in milliseconds) of authority-based approaches on these graphs.

The computational complexity of each PageRank iteration is $O(N+L)$, where N is the number of nodes and L is the number of links (connections or edges) between these nodes [18]. Of course, the time complexity of the overall algorithm depends on the number of iterations until convergence. The proposed TC graphs do not directly change the complexity formulation, but improves the overall running time of these algorithms due to reducing the dimensionality of the spaces which are displayed in Tables 6.11 and 6.12.

Table 6.11 presents the average statistics for reading and commenting authority graphs of blog collection, while Table 6.12 is for question routing and reply ranking tasks of StackOverflow collection[4]. A common observation in all results is the change in number of nodes and edges for different graph types. The *PR* graphs, which are topic-independent authority networks, contain the highest number of nodes and edges. *PR* graph statistics are followed by *HITS* graphs. Even though the number of nodes in *HITS* graphs are lower, the number of edges are very close to *PR*

[4] Graph statistics and running times in Table 6.11 corresponds to the experiments presented in Tables 6.1 and 6.2. Similarly Table 6.12 corresponds to Tables 6.5 and 6.6.

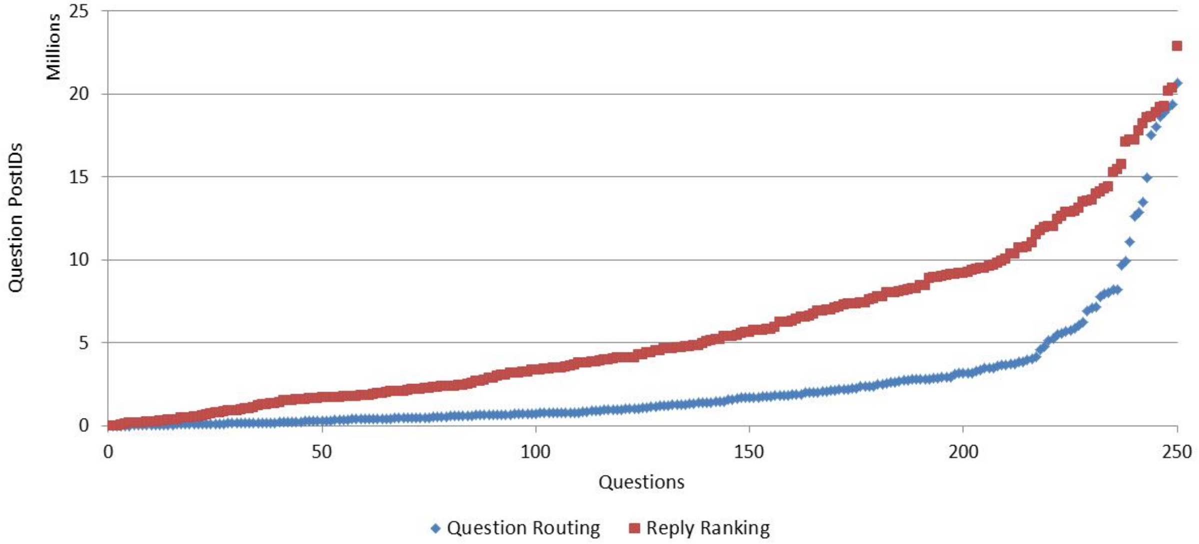


Figure 6.6: Sorted 250 question postIDs from test sets of question routing and reply ranking tasks.

graphs due to using all existing edges between two connected nodes.[5]

Compared to *PR* and *HITS* graphs, the number of nodes and edges decrease drastically in *TC* graphs. However, the graphs have still hundreds of nodes and edges. Due to this decrease in graph size, the iterations take less time and so the overall running time of the approaches also drop significantly as shown in tables. These running times show that using *Topic Candidate* graphs does not only provide more effective results but also improves the efficiency.

Similar to web search engines, for a given query, an expert blogger search engine should be also efficient in returning a ranked list of expert candidates. Likewise, for a posted question, the question routing mechanism should identify possible responders immediately. With the proposed TC graph, around a couple of seconds is required for identifying authoritative expert bloggers from reading graphs and mostly less than a second is needed for commenting graphs, compared to tens or hundreds of seconds required for *PR*[6] and *HITS* graphs. Similarly, authority estimation in answering *TC* graphs is considerably faster compared to *PR* or *HITS* graphs in StackOverflow collection.

Not directly related to efficiency but also observed from Table 6.12 is the difference between the graph sizes of question routing and reply ranking tasks. The graphs of both tasks are constructed similarly by using the previously posted questions and answers before the particular question; however the size of reply ranking graphs are more than twice the size of question routing graphs. In order to understand this, the questions in test sets were analyzed with respect to their postIDs. In StackOverflow, postIDs are incremented for each new post (question or reply), therefore a lower postID is an indication of a question or reply posted earlier. The sorted postIDs of 250 questions within test sets are presented in Figure 6.6 for question routing and reply ranking tasks. According to Figure 6.6, the randomly selected questions in reply ranking task

[5] Difference between *PR* and *HITS* graphs are mostly coming from less active user nodes with limited number of connections.

[6] Topic-independent *PR* can be estimated offline, but the topic-dependent *PR* must be estimated online for a given query.

Activity	Graph	Only Posts	Posts & Reads/Comments	Only Reads/Comments
Reading	PR	0.69%	11.25%	88.07%
	HITS	0.18%	16.80%	83.02%
	TC	2.53%	1.32%	96.15%
Commenting	PR	4.13%	21.87%	74.00%
	HITS	4.13%	43.20%	52.66%
	TC	9.38%	11.93%	78.69%

Table 6.13: Percentage of nodes with inlink and/or outlink in Blog collection.

Task	Graph	Only Answers	Asks & Answers	Only Asks
Question Routing	PR	30.31%	39.73%	29.96%
	HITS	19.93%	49.05%	31.03%
	TC	60.81%	14.12%	25.07%
Reply Ranking	PR	28.40%	30.39%	41.20%
	HITS	17.40%	38.76%	43.85%
	TC	49.92%	15.68%	34.40%

Table 6.14: Percentage of nodes with inlink and/or outlink in StackOverflow collection.

have on average higher postIDs than randomly selected questions used for question routing[7]. Using questions with higher postIDs, in other words, later post dates, causes more data to be used during graph construction which explains this difference between graph sizes.

6.5.2.5 Connectivity Analysis

The constructed *PR*, *HITS* and *TC* graphs are analyzed with respect to connectivity of nodes. Tables 6.13 and 6.14 presents the percentage of nodes with:

- only incoming edges: users who only post blog posts or answer questions (posting blog posts and answering questions is an indication of authority, therefore the direction of authority is towards these user nodes.)
- incoming and outgoing edges: users who post blog posts, read or comment to other blog posts, or both ask and answer questions
- only outgoing edges: users who only read or comment to other blog posts, or ask questions

One can make several observations from these tables. First of all, by looking at the overall percentages, the structural difference between the graphs constructed in different environments by using different activities is obvious. In Blog graphs, a huge percentage of the users are only passively contributing to the community by either reading or commenting, while the smaller rest of the users are both posting and reading/commenting to blog posts. The percentage of users who only post but never read is very low, 0.69% for *PR* and 0.18% *HITS* as shown in Table

[7] This graph shows several trends about the StackOverflow, and the number of questions and corresponding answers change over time. For instance, the randomly selected questions for question routing task used only the limitation of receiving 10 replies. The figure shows that in earlier times more StackOverflow questions received 10 or more replies on average, but over time this number decreased. There may be several reasons of this depending on the type of questions received or the answering patters of users, which are irrelevant to this dissertation and so have not been analyzed.

6.13. These may belong to managers who only post but never read other users. Similarly for the commenting activity, only around 4.13% of the users only post but not comment to others posts. On the other hand, in CQA asker-responder graphs, a high percentage of users are either only asking or only answering. This bipartite property of the graphs becomes more obvious in *TC* graphs, since with *PR* and *HITS* graphs, there may be users who only ask questions on *Java* but only answer questions on *C#*.

Continuing with the comparison of *PR*, *HITS* and *TC* graphs, in blog collection the *TC* graphs have higher relative percentage of nodes with only incoming or only outgoing edges. This is due to *TC* graphs using only the topic-specific edges during graph construction unlike others which use all edges between nodes. Comparing *PR* and *HITS*, *HITS* has the lowest percentage of only incoming or only outgoing nodes. This is due to *HITS* approach starting with a root set of nodes which are assumed to be topic-specific and then expanding the network. During this expansion, not all users can be reached, especially the ones who don't interact with any of the users in root set. Not being able to reach to these users during expansion lowers the overall percentage of nodes with only incoming or only outgoing links.

Similarly, in Table 6.14, *HITS* graphs have higher percentage of nodes with both incoming and outgoing edges compared to *PR* graphs. However, in CQA environments as graphs become more topic-specific as in the case of *TC* graphs, they also become more *bipartite-like*. In *TC* graphs, only around 15% of the nodes have both incoming and outcoming edges. In these graphs, a huge percentage of user nodes are the ones who answer questions, which is possibly due to starting the graph construction with a set of responders, and expanding it carefully by only using topic-specific edges.

6.5.2.6 Summary

Overall for all three expertise related tasks, the proposed *Topic Candidate* graphs improved both the effectiveness and efficiency of authority estimation approaches. Similar to the findings in content-based chapter (Chapter 5), the improved performance with the proposed graphs shows the power of representation, more specifically the representation of topic-specific authority over user authority networks, on expert finding. The performed analyses on these graphs also show the effects of authority-based interactions over the constructed graphs and the estimated expertise scores.

6.5.3 Experiments with Using Initially Estimated Expertise

In addition to using topic-focused graphs, an initially estimated expertise score is also proposed to be used in authority estimation approaches in order to estimate more topic-specific authority-based expertise scores. In the following experiments, *expertise influenced* and *expertise teleported* authority approaches are applied to *Topic Candidate* graphs.

6.5.3.1 Blog Collection

In experiments with blog collection the expertise scores from best content-based approach, *Reciprocal Rank*, was used as the initially estimated expertise scores in authority estimations. The experimental results are presented in Tables 6.15 and 6.16 for unweighted graphs and Tables 6.17 and 6.18 for weighted graphs respectively for reading and commenting activities. In these tables, initially the results of original algorithms, *PR*, *TSPR* and *HITS*, are displayed. These are followed

Algorithm	Levels of Expertise						NDCG
	VE		+AE		+SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.0475	.0792	.1075	.0919	.1700	.1227	.3371
PR_{Tel}	.0600	$.1285^r_s$	$.1500^r_s$	$.1552^r_s$	$.2275^r_s$	$.1962^r_s$	$.4267^r_s$
PR_{Inf}	.0575	.0867	.1200	$.1024^r$	.1900	$.1351^r_s$	$.3505^r_s$
TSPR	.0600	.0754	.1300	.1086	.2025	.1476	.3831
TSPR_{Tel}	$.0925^r_s$	$.1646^r_s$	$.2025^r_s$	$.1953^r_s$	$.2850^r_s$	$.2385^r_s$	$.4601^r_s$
TSPR_{Inf}	.0700	.0851	.1425	.1123	.2150	.1488	.3867
HITS	.0400	.0844	.1075	.0915	.1675	.1228	.3312
HITS_{Inf}	$.0950^r_s$	$.1631^r_s$	$.1800^r_s$	$.1455^r_s$	$.2675^r_s$	$.1719^r_s$	$.4051^r_s$

Table 6.15: Expert ranking performance of using initially estimated expertise in authority estimation on unweighted *Topic Candidate* graphs constructed from reading activities in blog collection.

Algorithm	Levels of Expertise						NDCG
	VE		+AE		+SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.0425	.0827	.0950	.0886	.1525	.1137	.3333
PR_{Tel}	$.0725^r_s$	$.1599^r_s$	$.1375^r_s$	$.1415^r_s$	$.1850_s$	$.1580^r_s$	$.3868^r_s$
PR_{Inf}	$.1450^r_s$	$.2619^r_s$	$.2700^r_s$	$.2166^r_s$	$.3825^r_s$	$.2362^r_s$	$.4882^r_s$
TSPR	.0725	.1389	.1475	.1266	.2125	.1501	.3797
TSPR_{Tel}	.0900	$.1922^r_s$	$.1950^r_s$	$.1721^r_s$	$.2525^{r'}$	$.1900^r_s$	$.4247^r_s$
TSPR_{Inf}	$.1650^r_s$	$.2760^r_s$	$.3075^r_s$	$.2425^r_s$	$.4225^r_s$	$.2643^r_s$	$.5075^r_s$
HITS	.0400	.0863	.0950	.0885	.1425	.1137	.3284
HITS_{Inf}	$.1075^r_s$	$.2419^r_s$	$.2050^r_s$	$.2032^r_s$	$.2800^r_s$	$.2003^r_s$	$.4686^r_s$

Table 6.16: Expert ranking performance of using initially estimated expertise in authority estimation on unweighted *Topic Candidate* graphs constructed from commenting activities in blog collection.

by the results of *expertise teleported* (*Tel*) and *expertise influenced* (*Inf*) versions of these algorithms. Since expertise teleportation is only possible in PageRank-based approaches, expertise weighted teleportation is only applied to *PR* and *TSPR* algorithms.

Using initially estimated expertise either as influence or teleportation weight in authority estimation provided improvements over the original authority estimation approach for both activity types. However, the degree of improvements of these proposed algorithms depends on the type of interaction used. Furthermore, the relative ranking of influence and teleportation weightings are also different for reading and commenting activities. For commenting interaction, both algorithms returned statistically significant improvements, but the expertise-influenced algorithms outperformed the expertise-teleported algorithms. On the other hand, for reading activity, the expertise-teleported authority algorithm performs much better than using expertise as influence.

Both algorithms working better than the original algorithms shows that using content-based expertise either to influence expertise to other users through connections, or to improve a content-wise topic-specific expert user's being visited probabilities is useful for estimating topic-specific authorities. The difference between these algorithms is due to these activities' representations

Algorithm	Levels of Expertise						NDCG
	VE		+AE		+SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.0500	.0802	.1075	.0926	.1775	.1248	.3395
PR$_{Tel}$	.0650	$.1297^r_s$	$.1500^r_s$	$.1549^r_s$	$.2350^r_s$	$.1988^r_s$	$.4298^r_s$
PR$_{Inf}$	.0625	$.0906^r$	.1225	$.1049^r_{s'}$	.1925	$.1379^r_s$	$.3554^r_s$
TSPR	.0675	.0812	.1400	.1118	.2125	.1514	.3882
TSPR$_{Tel}$	$.0925^r_{s'}$	$.1681^r_s$	$.2125^r_s$	$.1991^r_s$	$.3075^r_s$	$.2464^r_s$	$.4652^r_s$
TSPR$_{Inf}$	.0725	$.0989^r$	.1375	.1168	.2175	.1537	.3944
HITS	.0750	.1166	.1425	.1228	.2175	.1541	.3707
HITS$_{Inf}$	$.0950_{s'}$	$.1631^r$	$.1800^r_{s'}$	$.1455^r_s$	$.2675^r_s$	$.1719^r_s$	$.4051^r_s$

Table 6.17: Expert ranking performance of using initially estimated expertise in authority estimation on weighted *Topic Candidate* graphs constructed from reading activities in blog collection.

Algorithm	Levels of Expertise						NDCG
	VE		+AE		+SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.0425	.0842	.0975	.0890	.1525	.1145	.3346
PR$_{Tel}$	$.0875^r_s$	$.1674^r_s$	$.1675^r_s$	$.1546^r_s$	$.2325^r_s$	$.1737^r_s$	$.4046^r_s$
PR$_{Inf}$	$.1550^r_s$	$.2835^r_s$	$.2875^r_s$	$.2392^r_s$	$.4075^r_s$	$.2565^r_s$	$.5069^r_s$
TSPR	.0675	.1333	.1525	.1303	.2175	.1562	.3868
TSPR$_{Tel}$	$.1000^r_s$	$.2073^r_s$	$.2075^r_s$	$.1915^r_s$	$.2850^r_s$	$.2113^r_s$	$.4463^r_s$
TSPR$_{Inf}$	$.1750^r_s$	$.2975^r_s$	$.3300^r_s$	$.2696^r_s$	$.4475^r_s$	$.2907^r_s$	$.5322^r_s$
HITS	.0650	.1122	.1375	.1097	.2150	.1421	.3643
HITS$_{Inf}$	$.1075^r_s$	$.2419^r_s$	$.2050^r_s$	$.2032^r_s$	$.2800^r_s$	$.2003^r_s$	$.4686^r_s$

Table 6.18: Expert ranking performance of using initially estimated expertise in authority estimation on weighted *Topic Candidate* graphs constructed from commenting activities in blog collection.

of expertise. Commenting is a more explicit and strong form of action compared to reading. Leaving a comment to a blog post may require some kind of prior knowledge on the topic of blog posts. For instance, users may leave comments to blog posts to agree or disagree with the post, or to add their point of view, which require users to have some expertise on the topic of the post. Therefore, influencing commenters' topic-specific expertise to the author of the post improves the overall expert ranking. On the other hand, reading does not require any prior knowledge on the topic of post. Anybody can read (access to) any blog post of their choosing. Therefore, influencing the topic-specific expertise scores of readers, which can be very low, may not be as effective as using the expertise score of blog post's author as teleportation weight to improve its probability of being visited.

Overall, the *expertise influenced* authority approaches provided consistent and statistically significant improvements over original approaches, especially when applied to commenting graphs. The best performing results were obtained with *expertise influenced TSPR* and relative improvements up to 100% were observed across several metrics. These improvements show that being connected from expert users as well as the authoritative users is important in identifying authoritative topic-specific experts especially when the authority link requires the connected

Algorithm	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
PR	.0512	.0408	.0304	.1560	.2240	.3240	.4520	.1308
PR_{Tel}	.0440	.0356	.0306	.1470	.1880	.3000	.4360	.1285
PR_{Inf}	.0464	.0388	.0306	.1461	.2040	.3160	.4440	.1300
TSPR	.0400	.0340	.0274	.1418	.1720	.2720	.4200	.1211
$TSPR_{Tel}$	.0424	.0332	$.0298^{r'}_{s'}$	.1412	.1840	.2800	.4200	$.1262^{r'}_{s}$
$TSPR_{Inf}$	.0416	.0348	.0254	.1332	.1800	.2880	.3880	.1187
HITS	.0432	.0364	.0270	.1429	.1960	.2720	.3840	.1230
$HITS_{Inf}$	.0400	.0368	.0282	.1385	.1840	.2840	.3920	.1193

Table 6.19: Question routing performance of using initially estimated expertise in authority estimation on unweighted *TC* graphs constructed from answering activities in StackOverflow collection.

Algorithm	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5	BAP
PR	.5546	.6408	.7022	.7681	.8334	.2640
PR_{Tel}	.5660	.6424	.7028	.7693	.8350	.2760
PR_{Inf}	.5585	.6387	.6991	.7686	.8334	.2720
TSPR	.5486	.6417	.7059	.7645	.8324	.2560
$TSPR_{Tel}$	.5481	.6326	.6996	.7640	.8297	.2560
$TSPR_{Inf}$	.5587	.6441	.7108	.7676	.8352	$.2760^{r'}_{s'}$
HITS	.5451	.6361	.6933	.7586	.8287	.2440
$HITS_{Inf}$	.5408	.6389	.6938	.7581	.8283	.2400

Table 6.20: Reply ranking performance of using initially estimated expertise in authority estimation on unweighted *TC* graphs constructed from answering activities in StackOverflow collection.

user to have some prior topic-specific expertise. For more instant and implicit user interactions like reading, where prior topic-specific expertise is not required, *expertise influenced* authority estimations cannot beat the *expertise teleported* authority estimations but still perform better than the original unweighted approaches.

6.5.3.2 CQA Collection

The proposed weightings of expertise in authority estimations are also applied and tested for question routing and reply ranking tasks on StackOverflow collection. The results are shown in Tables 6.19 and 6.20 (unweighted graphs) and Tables 6.21 and 6.22 (weighted graphs) respectively for question routing and reply ranking tasks. Compared to improvements observed in Blog collection, the expertise weighted authority estimation approaches did not provide consistent improvements in expertise related tasks in CQA data.

Different results are observed for question routing and reply ranking tasks, and also different relative ranking of approaches are found for different authority estimation algorithms (*PR*, *TSPR* and *HITS*). For question routing task (Tables 6.19 and 6.21), using expertise in authority estimations cause drops in *PR* approach, which in its original form returns the best performing setting for question routing. Only the *expertise teleported TSPR* approach provides some improvements over *TSPR* approach. Mixed results are also observed for reply ranking task (Tables 6.20 and 6.22).

Algorithm	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
PR	.0504	.0400	.0304	.1574	.2240	.3240	.4440	.1313
PR_{Tel}	.0424	.0364	.0310	.1457	.1800	.3000	.4400	.1282
PR_{Inf}	.0448	.0384	.0304	.1452	.2000	.3080	.4400	.1301
TSPR	.0400	.0332	.0278	.1416	.1720	.2640	.4240	.1213
$TSPR_{Tel}$	.0416	.0328	$.0300^{r'}$	.1419	.1800	.2680	.4200	$.1263^{r'}_{s}$
$TSPR_{Inf}$	.0408	.0344	.0260	.1310	.1760	.2880	.3920	.1186
HITS	.0432	.0356	.0266	.1353	.1920	.2760	.3680	.1224
$HITS_{Inf}$	.0400	.0368	.0282	.1385	.1840	.2840	.3920	.1193

Table 6.21: Question routing performance of using initially estimated expertise in authority estimation on weighted *TC* graphs constructed from answering activities in StackOverflow collection.

Approach	NDCG@1	NDCG@2	NDCGP@3	NDCG@4	NDCG@5	BAP
PR	.5551	.6395	.7031	.7681	.8333	.2640
PR_{Tel}	.5659	.6412	.7033	.7691	.8349	.2760
PR_{Inf}	.5586	.6404	.7015	.7694	.8340	.2720
TSPR	.5457	.6417	.7053	.7639	.8318	.2520
$TSPR_{Tel}$	.5482	.6388	.7007	.7652	.8308	.2600
$TSPR_{Inf}$	$.5580^{r'}$	.6444	$.7130^{r'}$	$.7677^{r'}$	$.8353^{r'}$	$.2720^{r'}_{s'}$
HITS	.5630	.6449	.7009	.7657	.8347	.2640
$HITS_{Inf}$	.5408	.6389	.6938	.7581	.8283	.2400

Table 6.22: Reply ranking performance of using initially estimated expertise in authority estimation on weighted *TC* graphs constructed from answering activities in StackOverflow collection.

Even though the improvements are small, *teleportation weighted PR* and *influence weighted TSPR* approaches outperform their unweighted versions and all other weighted authority estimations.

These inconsistent results with using expertise either as influence or teleportation weight may not be surprising after all. By estimating authority for a topic-specific task, the *TSPR* approach is expected to work better than its less topic-specific version *PR*. However, the experimental results do not support this expectation, which take us back to the discussion of whether authority-based approaches are useful at all compared to more basic count-based approaches for CQA communities as discussed in Related Work (Section 2.2.2).

The prior work on estimating topic-specific expertise through authority-based approaches showed that the success of authority-based approaches highly depend on the structure of the network [104]. This structure can depend on the type of authoritative activity used. For instance, commenting and especially reading activities can be originated from users with varying levels of expertise. It is possible to see both topic-specific experts and not experts reading or commenting to another user's post on the particular topic. However, for answering activity in CQA communities, it is unexpected to see experts asking questions on the topic they answer questions of other users. Such a thing occurs rarely, probably for very difficult or ambiguous (maybe subjective) questions.

In CQAs, askers accept their lack of knowledge by asking and responders show their expertise by answering. This difference between the levels of connected users may cause the authority graph to have similar characteristics of bipartite graphs. This may even be more explicit in

Activity	Graph	Levels of Expertise						NDCG
		VE		+AE		+SE		
		P@10	MAP	P@10	MAP	P@10	MAP	
Reading	PR	.0100	.0087	.0175	.0089	.0400	.0140	.1138
	PR$_w$	.0075	.0081	.0125	.0071	.0325	.0127	.1028
	HITS	.0100	.0094	.0175	.0094	.0425	.0165	.1203
	HITS$_w$	.0075	.0082	.0125	.0073	.0325	.0137	.1085
	TC	.0500	.0805	.1175	.0944	.1800	.1265	.3404
	TC$_w$	.0525	.0915	.1175	.1079	.1875	.1409	.3585
	Best Multi-Step Propagation	.0750	.1166	.1425	.1228	.2175	.1541	.3707
Commenting	PR	.0075	.0045	.0125	.0050	.0350	.0081	.0891
	PR$_w$	.0075	.0049	.0150	.0041	.0350	.0074	.0846
	HITS	.0075	.0062	.0175	.0074	.0425	.0137	.1198
	HITS$_w$	.0075	.0055	.0150	.0051	.0375	.0090	.1101
	TC	.0425	.0914	.0975	.0918	.1525	.1185	.3383
	TC$_w$	.0650	.0975	.1325	.1014	.2000	.1309	.3538
	Best Multi-Step Propagation	.0675	.1333	.1525	.1303	.2175	.1562	.3868

Table 6.23: Expert ranking performance of *InDegree* applied to different authority graphs constructed from reading and commenting activities in blog collection.

Graph	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
PR	.0160	.0180	.0148	.0684	.0800	.1600	.2480	.0851
PR$_w$	.0168	.0168	.0142	.0735	.0840	.1440	.2440	.0860
HITS	.0176	.0176	.0146	.0689	.0880	.1600	.2480	.0853
HITS$_w$	.0176	.0164	.0140	.0767	.0880	.1440	.2400	.0869
TC	.0520	.0416	.0330	.1517	.2240	.3160	.4760	.1321
TC$_w$	.0504	.0400	.0318	.1545	.2160	.3000	.4560	.1321
Best Multi-Step Propagation	.0512	.0408	.0304	.1560	.2240	.3240	.4520	.1308

Table 6.24: Question routing performance of *InDegree* applied to different authority graphs constructed from answering activities in StackOverflow collection.

topic-specific *TC* graphs. Our analysis of graph connectivity in Table 6.14 also showed that only around 15% of the nodes in *TC* graphs have both incoming and outgoing edges. The rest of the users in these graphs either only ask or only answer to questions. With such a network structure, authority estimation approaches cannot effectively propagate authority. They behave more like one-step propagation approaches like *InDegree*.

6.5.3.3 Comparing Authority-based Approaches with *InDegree*

The prior work [13, 104] showed examples where *InDegree* approach outperforms authority-based approaches like *PageRank* and *HITS*. In order to see whether the same applies to our collections, the *InDegree* approach is also applied to them. The results are presented in Table 6.23

Graph	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5	BAP
PR	.5248	.6116	.6827	.7489	.8200	.2120
PR$_w$	.5318	.6173	.6901	.7523	.8233	.2200
HITS	.5208	.6103	.6800	.7469	.8186	.2040
HITS$_w$	.5278	.6160	.6874	.7502	.8220	.2120
TC	.5557	.6398	.6983	.7660	.8322	.2600
TC$_w$	.5575	.6407	.7019	.7654	.8328	.2640
Best Multi-Step Propagation	.5630	.6449	.7009	.7657	.8347	.2640

Table 6.25: Reply ranking performance of *InDegree* applied to different authority graphs constructed from answering activities in StackOverflow collection.

for blog collection and Tables 6.24 and 6.25 for StackOverflow collection for both unweighted and weighted[8] graphs together with the results of the best performing multi-step propagation algorithm.

In Table 6.23, for both reading and commenting activities in blog collection, the *InDegree* approach could not reach the performance of the best original authority-based (multi-step propagation) algorithm's results (Tables 6.1-6.4). However, in StackOverflow collection, the *InDegree* approach (Tables 6.24 and 6.25) performed very similarly to the multi-step propagation algorithms (Tables 6.5-6.8) applied to the same graphs for both tasks as also shown with the best performing algorithm at the last rows. Only the *TSPR* approach outperformed the *InDegree* in both *PR* and *HITS* graphs probably due to favoring users who are identified as possible expert candidates. On the other hand, in *TC* graphs *InDegree* performed very similar to *PR* approach which generally works better than other multi-step propagation algorithms. These results show that, the effectiveness of multi-step propagation algorithms depend on the structure of the network. For networks like *TC* that are in similar characteristic to bipartite graphs, one-step and multi-step propagation approaches do not make a huge difference at the end, which also explains why *expertise- influenced* or *teleported* approaches are also not making too much difference compared to their original versions when applied to asker-responder networks in StackOverflow.

In Tables 6.23, 6.24 and 6.25, the *TC* graphs provide statistically significant improvements over to both *PR* and *HITS* graphs, for all activity and task types, which means that the proposed *TC* graph is also effective with one-step propagation algorithms. Similar findings from authority-based approaches are also observed when comparing *InDegree* approach applied to weighted and unweighted graphs. Weights on topic-specific networks constructed with repetitive activities is useful, however weighted edges does not cause an observable change when used in less topic-specific networks or networks constructed from less repetitive activities.

6.5.3.4 Summary

In this section, the expertise weighted authority estimation approaches are applied to reading, commenting and answering networks. The proposed influence-based weighting is compared to unweighted and teleportation-based weighted algorithms and shown to outperform both of them with commenting graphs. This is mainly due to commenting interactions being based on some

[8] Applying *InDegree* approach to weighted graphs is very similar to *Votes* approach from voting models and *Answer Count* approach for CQA communities.

prior knowledge on the particular topic, which becomes useful during propagation of authority. On the other hand, in reading graphs the teleportation-based weighting performed better than influence-based weighting, due to reading interaction not requiring any prior expertise on the particular topic.

In terms of asker-responder networks, the expertise difference between the asking and responding interaction causes construction of bigram like graphs with low number of nodes with both incoming and outgoing connections. In such authority networks, the authority cannot be propagated to other nodes to due to lack of nodes connecting other nodes. As shown with the experimental results, in these networks multi-step propagation algorithms work very similar to one-step propagation algorithms, and using initially estimated content-based expertise either as influence or teleportation weight does not make a difference in estimated authority-based expertise score.

6.5.4 Experiments with $HITS_{CQA}$

Applying HITS to asker-responder networks in CQA sites may not return optimum results due to the inconsistencies between the activity used to construct the authority networks and the underlying assumption of the algorithm. In order to decrease the effects of these inconsistencies, an adaptation is proposed to HITS algorithm, more specifically to the calculation of hubs score. This adaptation is specifically proposed for the asker-responder networks. The corresponding experimental results are presented in Tables 6.26 and 6.27, respectively for question routing and reply ranking tasks.

Overall, using the modified $HITS_{CQA}$ returns small improvements (sometimes statistically significant) for both question routing and reply ranking tasks, except for the frequency-based weighted graphs for question routing task. Improvements are more observable in unweighted graphs, and interestingly the unweighted graphs work better in question routing task, while for reply ranking task weighted graphs work more effectively.

The improvements even though small, show the effectiveness of the proposed approach. Not seeing drastic improvements in scores can be expected, and probably due to the characteristics of users' answering choices. If responders answer questions of askers with similar expertise levels, then using either the summation or average will have similar effects in overall. This proposed algorithm may work better in networks where responders answer questions from users with changing expertise or authority levels.

In order to observe the dependency of this approach on asker-responder networks, and whether it works on other authority graphs, it was also applied to reading and commenting authority graphs from blog collection. As shown in Table 6.28, using $HITS_{CQA}$ on reading or commenting activities causes consistent drops in performance due to inconsistencies between $HITS_{CQA}$ and the underlying assumptions of these interactions. For these interactions, using the original HITS is more ideal.

6.6 Summary

In this chapter, the available social networks are exploited as another source of expertise in order to estimate effective authority-based expertise scores. The authority estimation approaches developed for web pages are analyzed within these environments and the following research question is addressed:

Algorithm	Weight	P@5	P@10	P@20	MRR	MSC@5	MSC@10	MSC@20	NDCG
HITS	✗	.0432	.0364	.0270	.1429	.1960	.2720	.3840	.1230
$HITS_{CQA}$	✗	.0456	.0396	$.0298^{r'}_{s'}$	$.1501_{s}$	.2000	$.3080^{r'}_{s'}$	$.4240^{r'}_{s'}$	$.1281^{r'}_{s'}$
HITS	✓	.0432	.0356	.0266	.1353	.1920	.2760	.3680	.1224
$HITS_{CQA}$	✓	.0424	.0352	.0282	.1389	.1800	.2600	.3960	$.1257_{s}$

Table 6.26: Question routing performance of using $HITS_{CQA}$ in authority estimation on *TC* graphs constructed from answering activities in StackOverflow collection.

Algorithm	Weight	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5	BAP
HITS	✗	.5451	.6361	.6933	.7586	.8287	.2440
$HITS_{CQA}$	✗	.5610	.6432	$.7018^{r'}$	$.7677^{r'}$	$.8345^{r}$	$.2720^{r'}_{s'}$
HITS	✓	.5630	.6449	.7009	.7657	.8347	.2640
$HITS_{CQA}$	✓	.5650	.6469	.7072	.7681	.8366	.2760

Table 6.27: Reply ranking performance of using $HITS_{CQA}$ in authority estimation on *TC* graphs constructed from replying activities in StackOverflow collection.

Activity	Algorithm	Weight	Levels of Expertise						NDCG
			VE		AE		SE		
			P@10	MAP	P@10	MAP	P@10	MAP	
Read	HITS	✗	.0400	.0844	.1075	.0915	.1675	.1228	.3312
	$HITS_{CQA}$	✗	.0350	.0821	.1000	.0893	.1575	.1167	.3271
	HITS	✓	.0750	.1166	.1425	.1228	.2175	.1541	.3707
	$HITS_{CQA}$	✓	.0600	.1062	.1250	.1132	.1925	.1443	.3585
Comment	HITS	✗	.0400	.0863	.0950	.0885	.1425	.1137	.3284
	$HITS_{CQA}$	✗	.0275	.0732	.0850	.0779	.1275	.1015	.3123
	HITS	✓	.0650	.1122	.1375	.1097	.2150	.1421	.3643
	$HITS_{CQA}$	✓	.0625	.1158	.1275	.1066	.1900	.1302	.3544

Table 6.28: Expert ranking performance of using $HITS_{CQA}$ in authority estimation on *TC* graphs constructed from reading/commenting activities in blog collection.

- *RQ2: Do the assumptions of topic-specific authority estimation approaches developed for web pages hold for user authority networks in social media? For the ones that do not, what kind of algorithmic modifications can be performed so that they hold, and is it possible to make additional assumptions and necessary modifications which can provide more effective and efficient topic-specific authority estimations?*

Authority estimation depends on two factors: the constructed authority graph and the multi-step propagation algorithm which iterates over this graph. This dissertation focused on both of these aspects and their interactions, and more topic-specific graph construction and authority estimation approaches are proposed to improve effectiveness.

For topic-specific user related tasks, constructing more topic-specific graphs is an important initial step towards effective authority-based expertise estimations. Both topic-independent PageRank and topic-dependent HITS graphs which were developed for web pages are analyzed for user networks. Compared to web pages, the users are less topically clustered and have more mixed connections due to their diverse interests. Based on these differences between web pages and users, the regular topic-dependent authority graph construction algorithm does not return the expected topical graphs for users. This dissertation proposed a graph construction approach which returns more topic-specific user authority graphs, called the *Topic Candidate* graphs. The statistically significant improvements observed with these graphs on three tasks show the general effectiveness of the proposed graph construction approach. In addition to effectiveness, the proposed *TC* graphs also provide significant gains in efficiency by lowering the running times. This is especially important, since these approaches are topic-dependent approaches which require real-time estimation of authority for each given query.

Other than topic-specificity, the authority graphs being weighted (or not) by the frequency of the activity are also analyzed. The experiments with these two types of graphs showed that weighted graphs can be more effective when the authority link between users is originated from a repetitive type of activity like reading or commenting to blog posts. If the authoritative activity is not very repetitive, as answering a certain user's questions which is not likely in current popular CQAs for most topics, then the difference between authorities estimated from unweighted and weighted graphs are not very observable and consistent.

Furthermore, the connectivities of the user authority graphs are investigated. It has been observed that in graphs with more nodes with second-degree connectivity[9], the multi-step propagation algorithms provide improvements over one-step propagation approaches. Such graphs include reading or commenting authority graphs, where the authors of the posts also interact with other users' posts. However, it is not very common to see users who both ask and answer questions on the same particular topic. Therefore, topic-specific graphs constructed from answering activities in CQAs may result in bipartite-like graphs with few second-degree connected nodes. In these asker-responder networks, one and multi-step propagation approaches return very similar results, which matched with the findings of the prior work on similar graphs. Constructing more topic-specific authority graphs for asker-responders helps but propagating authority in these networks does not improve the performance compared to one-step propagation. Since one-time propagation approaches iterates over the graph only once, they are also more efficient then multi-step approaches. As a result, connectivity of the graph is important in terms of effective and efficient estimation of authority-based expertise.

In addition to the graphs, the algorithms that iterate over these graphs are also important for effective authority estimation. Authority-based approaches developed for web-pages may not always directly fit to other entities' authority graphs with different node and interaction types. For instance, this dissertation showed how the assumption of HITS does not hold on asker responder networks, and so an adaptation is proposed which provided small but consistent improvements for this particular inconsistency.

Apart from checking the applicability and underlying assumptions of these authority based approaches on these networks, a new assumption is also proposed, which is whether being connected from topic-specific experts is an indication of being an expert. Experiments performed with using initially estimated expertise scores as influence and teleportation weights outperformed the original approaches (statistically significant most of the time) in Blog collections. The experiments showed that if the authoritative action requires the tail node of the directed edge to have some prior knowledge or interest on the particular topic, then using expertise as influence helps more than using it with teleportation. However, if such a prior knowledge or interest is not required for the insertion of authority edge, then using expertise to weight teleportation

[9] Nodes which have both incoming and outgoing links, therefore can connect the nodes of the incoming edges to the nodes of the outgoing edges

probabilities is more effective.

These proposed approaches are tested on two data collections, blog[10] and StackOverflow, with three interaction types, reading, commenting and answering. Especially working on an intra-organizational blog collection provided the opportunity to use access logs (referred to as reading information), which are not widely available to or studied by the research community. This form of information is useful due to showing more implicit and frequent interactions between users. The experiments on this collection with *reading* and *commenting* activities showed that both of them, whether explicit or implicit, are useful in estimation topic-specific authority.

Overall, experiments performed with these three interaction types on two data collections over three tasks provided a better understanding of the behaviors of authority-based approaches and their applications to user networks. These algorithms provided inconsistent behaviors across prior research. Analyzing these approaches and their underlying assumptions, and authority graph properties such as the nodes, type and frequency of connections between these nodes explained some of these inconsistent behaviors. Adaptations were proposed which improved the effectiveness, efficiency and consistency of these approaches.

[10] The authority-based expert blogger finding approaches evaluated with company employees' assessments are also available at Appendix A. The findings from those experiments are similar to the trends we observed from the results presented in this chapter.

Chapter 7

Temporal Approaches

Social media sites, like CQAs, are dynamic environments where new users join constantly, or the level of activity or interest of existing users changes over time. Classic expertise estimation approaches, which were mostly developed for static datasets, cannot effectively model changing expertise and interest levels in these sites. However, these dynamic aspects of the environment should be taken into account for more effective expert identification, especially when the identified experts are expected to take an action, like answering questions, in order to satisfy the information seekers. This thesis proposes to use the available temporal evidence in social media sites to make the existing approaches more dynamic and effective. This chapter starts with an analysis of the dynamism of these sites and then addresses research question *RQ3*.

7.1 Motivation for Temporal Modeling

CQAs are very dynamic sites. New users join these sites every day, and inactive users may become more active over time. For instance, Figure 7.1 shows how the number of responders changes over time in StackOverflow. As shown in the figure, the number of responders (possible candidate experts) increases exponentially. Widely used expert finding methods may not favor these new users with limited reply history. Instead, they promote users who were actively answering questions for a long time. However, in CQA sites, the only way users can show their expertise is through answering other users' posted questions. Unlike blog or microblog sites where users can post whatever they want and whenever they want, in CQA sites users' contributions are limited by posted questions related to their expertise. Furthermore, these questions should not have been answered before, or answered but have not been completely resolved. Depending on the number of questions that satisfy these conditions, it requires some time for users to build reputation in these sites. Having a small reply history does not make them less of an expert when it comes to answering questions accurately. Therefore, these users should also be considered as possible candidates for question routing.

Another dynamic aspect of CQA site is the degree of activity change over time. For effective question routing, questions should not only be answered accurately, but also answered within an acceptable time frame. Routing questions to inactive users can result in delays and even failures in receiving answers, therefore finding experts who can provide timely answers is also important. Cai and Chakravarthy's [19] analysis on users' answering activities in StackOverflow over monthly intervals showed considerable activity fluctuations over time. Additionally, we

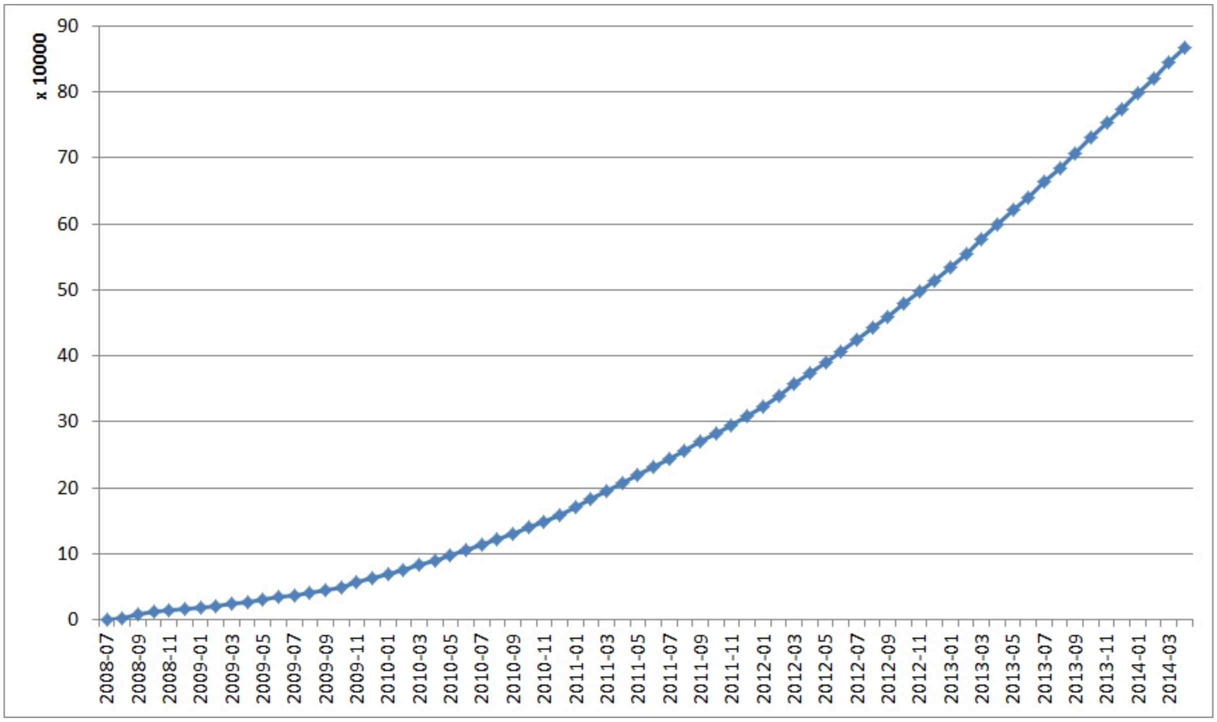


Figure 7.1: The number of repliers in StackOverflow over time.

calculated the coefficient of variation (CV)[1] of answering activities for StackOverflow data over weeks. The frequency distribution of CV values is shown in Figure 7.2. Around 90% of the active users have $CV > 1$. This means that for most users who answer on average n questions per week, the standard deviations are more than n, as they may answer more than $2n$ questions in a week and may not answer any in another week.

The change in user's interest is yet another reason why users who were answering topic relevant questions before, may not be interested in answering anymore. Cai and Chakravarthy [19] performed a correlation analysis on users' replies (words used) over time. Their analysis revealed possible topic drifts for some users. Changes in users' availabilities and interests are important temporal factors and should be considered for more effective expert retrieval for tasks like question routing.

7.2 Related Work

In order to overcome these problems, temporal information in CQA sites has been used in several ways by the prior work. Pal et al. [71] initially estimated the number of evolutionary patterns of users, and identified mainly three types of experts; experts who are consistently active in the community, experts who were initially active but later become passive, and experts who were passive initially and active later. Later on they applied SVM to the relative temporal series of number of answers and best answers in order to identify experts, and showed that estimating

[1] Coefficient of variation: The ratio of the standard deviation to the mean, $CV = \frac{\sigma}{\mu}$. It represents the measure of variation (σ) within a distribution with respect to its mean (μ). CV is very sensitive to small changes when μ is close to 0, so only the activities of users with answering $\mu >= 1$ are used.

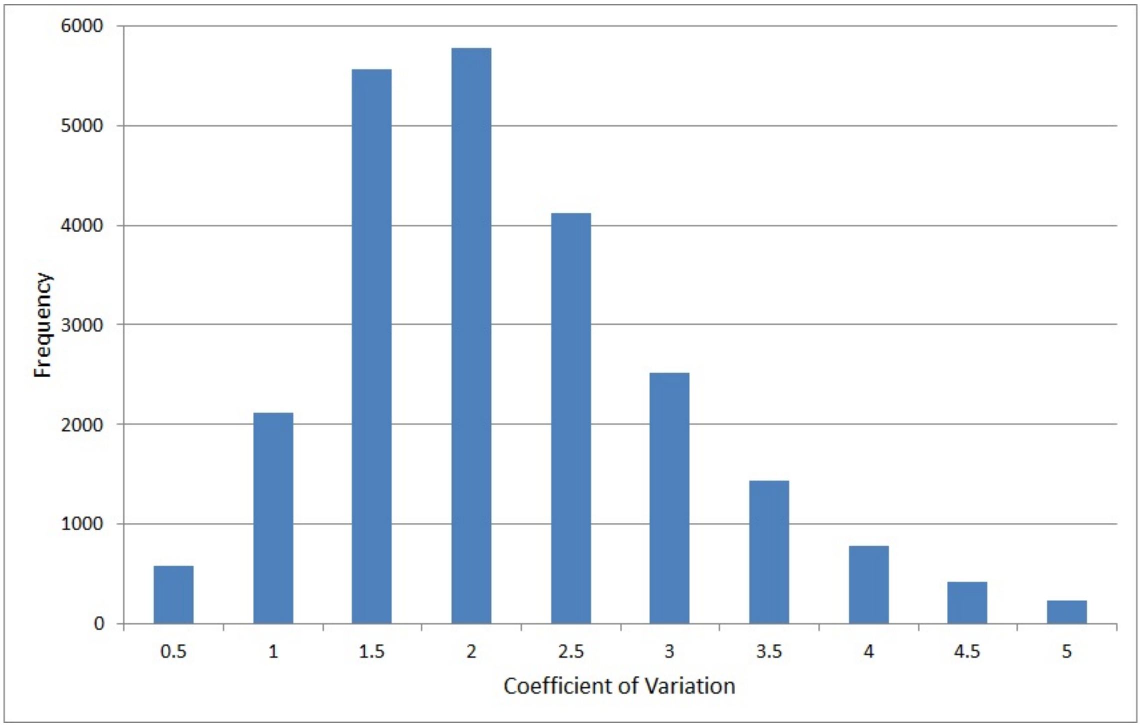


Figure 7.2: The frequency distribution of coefficient variation of users answering activities.

expertise using temporal data outperforms using static snapshot of the data. We believe that experts can be divided into these three categories in general, but we still argue that users are more complex and unique in some ways which makes it harder to use these more general categorizations of expertise for everybody.

Cai and Sharma [19] also used temporal features calculated between the period a question and all of its corresponding answers are posted, to improve answer quality prediction. Our work is different from [19] as we use temporal data before the question posting time to identify possible experts, on the other hand they used temporal features calculated after the posting time of question with the aim to capture the activity levels of users around the time question is posted to the time last reply is posted.

Some work used temporal data to estimate the availability of users. Li and King [51] applied an autoregressive model to forecast the availability of users for a given day by using previous days' activities. Sung et al. [94] estimated availability as a recency function where the replies of the user are weighted inversely proportional to their age. Chang and Pal [21] also built binary classifiers on previous *n* days of activity with different machine learning approaches. But these classifiers did not beat the simple baselines of assuming always available or using the availability status of previous day directly. Several prior work also used users' hourly activity distributions to find the specific times of the day that users are available and active at the site [21]. Additionally, Liu and Agichtein [56] analyzed the answering behaviors of users with respect to different times of the day and week. Using users' availability with respect to certain days and hours improved the performance of expertise related tasks in CQAs. Even though the improvements, this prior research considered availability as something topic-independent. However, users can be available to answer questions on some topics but not other topics, for example, not on topics that they used to answer before but not anymore. In order to estimate recent topic-specific availability and interests of users, this dissertation proposes to model the user availability together with the topic by adapting temporal discounting models.

7.3 Temporal Discounting

Temporal discounting, which is a widely studied phenomenon in economics and psychology, refers to the decrease in the subjective value of a reward as its receipt delays over time. In other words, the longer one needs to wait for a future reward, the lower its present subjective value becomes. People have a tendency to discount delayed rewards, and give more value to near future rewards. This behavior can be also observed for the past. People give more value to recent events than events that occurred a long time ago. Similarly, in today's drastically changing world, recently retrieved information is more valuable than older information. Therefore, in dynamic environments where users' activity and interest levels change over time, systems should have a tendency to give more value to recent activities and discount earlier activities especially when interacting with users in real time.

Two forms of temporal discounting functions have been used widely, exponential and hyperbolic discounting. In the exponential (*exp*) discounting model, the value of answers are exponentially discounted as time goes back. Exponential discounting can be represented as $e^{-k\Delta t}$, where k represents the parameter describing the rate of decrease and Δt is the number of time intervals passed since present time. The hyperbolic (*hyp*) discounting model is in the form $1/(1 + k\Delta t)$. The hyperbolic model shows very rapid fall initially, and then the decrease becomes more gradual as time goes back, or in other words, as Δt gets higher. Integrating temporal discounting to document-based expert finding approaches or approaches like *InDegree* is straightforward. Instead of counting all topic relevant instances equally, a discounted value depending on their time of creation (Δt) is used.

This proposed temporal approach models several types of user information together. First, since it uses all topic-relevant activities of users, it can be considered as a dynamic variant of widely used static expertise estimation approaches. Compared to static approaches, our approach will give enough credit to newly joined responders while not ignoring the earlier answers of existing ones. Second, this approach models users' interest change over time unlike prior approaches. Previous expert finding algorithms did not differentiate users who are still actively contributing to the topic vs. users who have contributed a lot in past but not anymore. In other words, we use the recent topical interests of users, which can be also referred to as their topic-specific availability, as if they are not any more interested in the topic, they are also unavailable to answer topic-specific questions, even though they may answer questions in other topics.

7.4 Temporal Modeling of Expertise

Unlike static models, temporal modeling of expertise requires a preprocessing step to construct time intervals, and assign the timestamped expertise evidence to the constructed intervals.

7.4.1 Constructing Time Intervals

Assume that t_1 represents the time of the first question posted to CQA website (or launch of the site) and t_q represents the specific time the question q is posted to the site. During identifying expert candidates who can answer question q, only the questions and replies posted within the period $[t_1, t_q]$ are used. Previous approaches mostly treat replies posted within this interval equally. However, in our proposed approach, the value of a posted reply depends on its posting time; therefore, replies are initially grouped with respect to their posting times. $[t_1, t_q]$ interval is divided into specific periods of times such as days, weeks, biweeks and months. The dates of posts are used to find their corresponding time intervals. The day interval of the first post is set as 1, $d(t_1) = 1$, while the day interval of question q is equal to 1 + the number of days passed since t_1. Week, biweek and month intervals are also calculated similarly.

7.4.2 Temporal Count-based Approaches

As expressed before, the exponential (*exp*) discounting can be represented as $e^{-k\Delta t}$, while the hyperbolic (*hyp*) discounting model is in the form $1/(1+k\Delta t)$. In these representations, k represents the parameter describing the rate of decrease and Δt is the number of time intervals passed since reply was posted. For a given question q and for any reply posted at time interval i, Δt_i is calculated as $d(t_q) - d(t_i)$ for days. It is always the case that $d(t_q) >= d(t_i)$ for any i, since only the replies posted before t_q are used. Δt_i is calculated similarly for weeks, biweeks and months.

Integrating temporal discounting to counting-based expertise calculation algorithms is straightforward. Instead of counting all instances equally, a discounted value depending on their time of creation is used. These temporal modifications are applied to two widely used approaches *Answer Count* (*AC*) [2] and *ZScore*[3]. In *AC* approach, the static expertise estimation of user u is equal to the number of replies posted by user u. On the other hand, its temporal discounted versions are as follows:

$$\begin{aligned} AC_{exp}(u) &= \sum_{i=1}^{q} R_i(u) e^{-k\Delta t_i} \\ AC_{hyp}(u) &= \sum_{i=1}^{q} \frac{R_i(u)}{1 + k\Delta t_i} \end{aligned} \tag{7.1}$$

where $R_i(u)$ is the number of replies posted by user u at interval i. Similarly for temporal *ZScore* approach, first the *ZScore* is calculated for each time interval (7.2) and then it is discounted with respect to its temporal distance from question's interval (7.3). Its formulation is as follows:

$$ZScore_i(u) = \frac{R_i(u) - Q_i(u)}{\sqrt{R_i(u) + Q_i(u)}} \tag{7.2}$$

[2] We should note that we don't switch baselines for different environments or tasks. We use either the same or very similar approaches, but refer to them with the names that are commonly used by the research community for that particular experimental setting. For instance, query dependent *Answer Count* approach is same as applying *InDegree* approach to a question-specific replying (weighted) network, which is also same as applying document-based *Votes* approach to all question relevant replies without restricting it to top n retrieved ones.

[3] ZScore has been mentioned before in Related Work chapter (Section 2.1.4.2). It combines users' asking and answering patterns. It has not been used in this dissertation before due to its lower performance compared to Answer Count approach. It has been used in this section, in order to show that not only Answer Count algorithm can be modified but other similar count-based algorithms can be modified with temporal information.

$$
\begin{aligned}
ZScore_{exp}(u) &= \sum_{i=1}^{q} ZScore_i(u)e^{-k\Delta t_i} \\
ZScore_{hyp}(u) &= \sum_{i=1}^{q} \frac{ZScore_i(u)}{1+k\Delta t_i}
\end{aligned}
\tag{7.3}
$$

where $Q_i(u)$ is the number of questions posted by user u at interval i.

7.5 Experiments

In this section, initially the static and temporal baseline approaches are presented, and then the proposed temporal discounting approach is tested.

7.5.1 Baseline Temporal Approaches

The original *AC* and *ZScore* algorithms were used as baselines for static approaches. Their question dependent versions were used for more effective performance where question tags are used to represent both the information need and user expertise. For a given question, uniformly weighted question's tags are initially searched among other previously asked questions' tags. Then the retrieved questions' responders are extracted, and for each user the number of retrieved answers and asked questions are used to calculate the *AC* and *ZScore* scores.

Two prior work on availability estimation are also used as temporal baselines. Sung et al. [94] estimated availability as a sigmoid function applied recency value which is calculated as follows:

$$
\frac{1}{1+e^{-\alpha \sum_{i=1}^{|R(u)|} \frac{1}{age(r_i)+2}}}
\tag{7.4}
$$

where $|R(u)|$ is the number of replies posted by user u at any time and $age(r_i)$ is the number of days passed since reply i is posted. Sung et al. set α as 0.1 [94]. The same value is also used in experiments in this thesis.

Chang and Pal [21] also built binary classifiers on previous n days of activity with different machine learning approaches. However, these classifiers did not beat the simple baselines of assuming always available or using the availability status of previous day directly. Using always available is same as the static approach, so the status of previous day is used as another temporal baseline in this thesis.

Sung et al. [94] used all answers of users while Chang and Pal [21] used all replies of users from a certain time frame (previous day) in order to estimate availability. Our proposed approach is different from these as we only use the particular question related replies in temporal modeling. This use of topic dependent activities is useful for modeling user's interest as well. Estimating user availability is useful only if user is still answering questions on the particular topic of question, which may not always be the case.

The following equation is used to combine the content (*AC* and *ZScore*) and availability scores of Chang and Pal, and Sung et al.:

$$
finalScore = content^{\lambda} * availability^{1-\lambda}
\tag{7.5}
$$

Min-max normalization is applied to the the first availability baseline to make its range $[0,1]$ [94]. Similarly for the second temporal baseline, the availability is either 0 or 1. Therefore,

Content Approach	Temporal Approach	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
AC		.0504	.0400	.0318	.1545	.2160	.3000	.4560	.1330
	+ Chang	.0504	.0448	.0372	$.1592_s$	.2080	$.3520^r_s$	.5120	.1339
	+ Sung	.0504	.0440	.0352	$.1572_s$	.2040	$.3520^r_s$	.5000	.1357
ZScore		.0448	.0376	.0304	.1438	.1960	.2880	.4240	.1196
	+ Chang	.0456	.0428	.0334	$.1493_s$	.1920	.3240	.4800	.1222
	+ Sung	.0464	.0400	.0308	$.1505_s$	.2000	$.3160^{r'}_{s'}$	.4320	.1224

Table 7.1: Question routing performance of static and temporal baseline approaches.

content scores are also normalized to have a similar range with availability scores. 10-fold cross-validation is used to find the optimum parameter setting for the interpolation. The optimum parameter is identified by using the median value.

7.5.2 Question Routing Experiments

The proposed temporal modeling approach is only applied to the question routing task. Reply ranking task does not benefit from this temporal modeling of expertise, mainly because with answering the particular question, responders already confirm their current expertise and interest on the topic of the question. Temporal discounting has not been applied to the expert blogger finding task either. The task did not specify identifying up-to-date experts, therefore the assessments also did not take the recency or up-to-datedness of the information provided in blog posts. Instead of identifying experts to get into contact or follow in future, this task remained as identifying experts from a static snapshot of a data collection.The TREC expert finding task also used a similar task definition and assessment methodology.

The results of static and temporal baselines are summarized in Table 7.1. Combining estimated availability with original approaches, which do not use any temporal information, provided consistent improvements across various metrics, and even statistically significant for MRR and MSC@10. Using temporal information just for estimating availability is shown to be effective.

In our proposed approaches, in addition to availability; the topic-specific interest of users and the recently joined users' activities are also modeled. The experimental results of these are presented in Tables 7.2 and 7.3 respectively for the Answer Count and ZCount temporal models for different discounting and time intervals. The first rows of the tables contain the best baseline approaches from Table 7.1, and the rest of the rows summarize the results of the proposed temporal models. Experiments on proposed approaches were initially performed with rate of decrease $k = 1$ which has been chosen arbitrarily[4]. Results that are statistically significant to static (original) and both of the temporal baselines are specified in both tables. As seen on tables, the proposed dynamic modeling of expertise approaches consistently outperform the static and temporal baselines with respect to all experimented time intervals. Some of these differences are statistically significant over all 3 baselines.

Different behaviors are observed for *exp* and *hyp* models, possibly due to the difference in their degree of decay over time. The discounting rate (weight difference between consecutive intervals) of two models are initially similar for small values of Δt, however as Δt increases,

[4] The effects of different values of k is analyzed later.

		P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
AC	+Chang	.0504	.0448	.0372	$.1592_s$	.2080	$.3520^r_s$	.5120	.1339
AC_{exp}	day	.0680	.0508	.0408	$.2043^r$	.2960	.3800	.5320	.1457
	week	.0584	.0480	.0390	.1755	.2360	.3560	.5120	.1469
	biweek	.0616	.0496	.0400	.1691	.2560	.3720	.5160	.1461
	month	.0680	.0560	.0474	$.1879^r_s$	.2720	$.4120^{r'}_{s'}$	.6000	.1561
AC_{hyp}	day	.0792	.0640	.0484	$.2170^r_s$	.3000	$.4480^r_s$	.5880	.1663
	week	.0656	.0576	.0472	$.1780_s$	.2560	$.4240^r_s$	.6000	.1557
	biweek	.0656	.0548	.0470	$.1737_{s'}$	.2560	.3960	.6080	.1540
	month	.0608	.0516	.0422	.1674	.2480	.3720	.5760	.1509

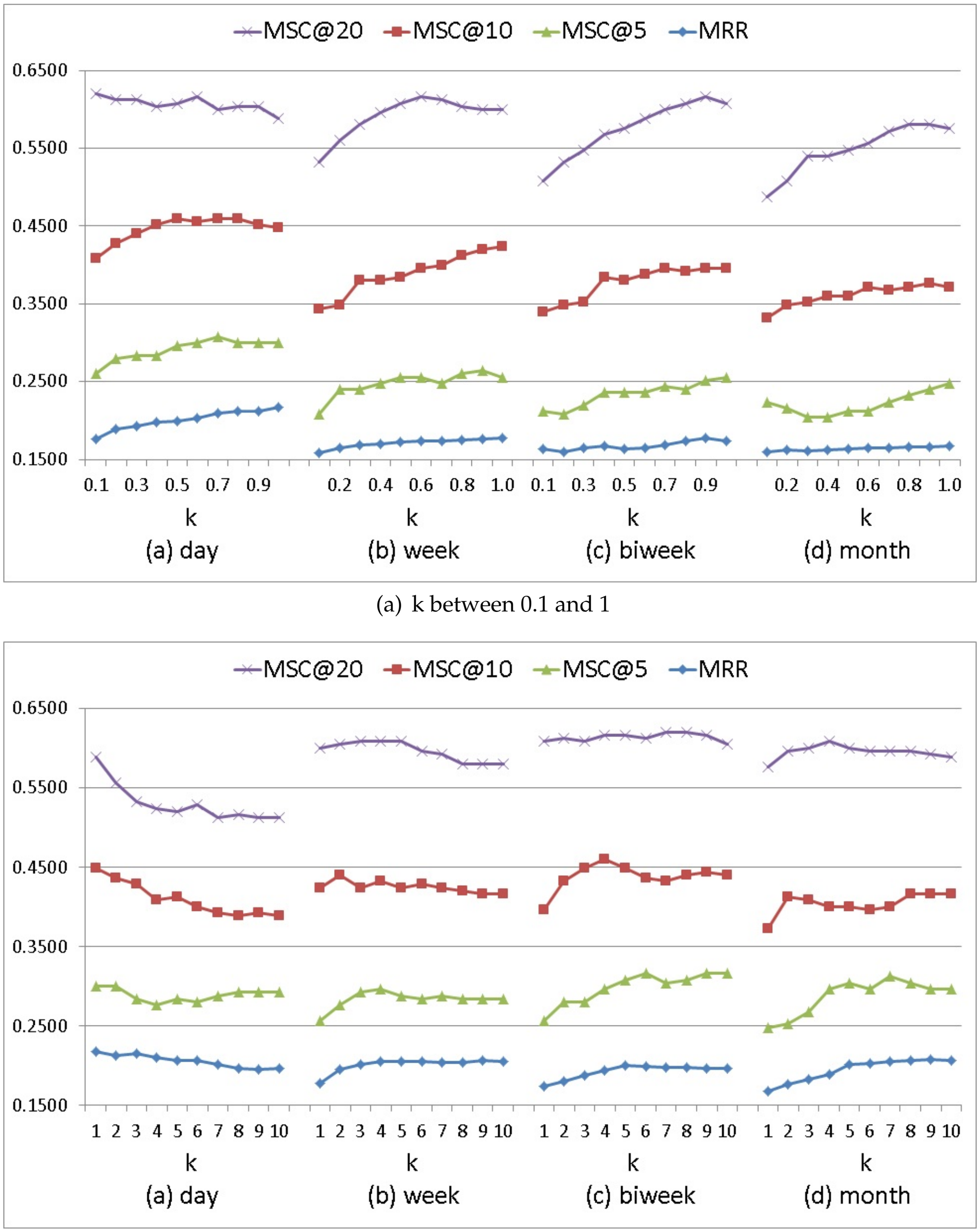
Table 7.2: Question routing performance of the proposed temporal Answer Count (AC) approach.

		P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
ZScore	+ Chang	.0456	.0428	.0334	$.1493_s$	.1920	.3240	.4800	.1222
$ZScore_{exp}$	day	.0624	.0484	.0382	$.1965^r$	.2840	.3760	.5120	.1412
	week	.0544	.0428	.0362	.1703	.2280	.3320	.4880	.1388
	biweek	.0560	.0468	.0372	.1619	.2320	.3720	.4960	.1377
	month	.0624	.0524	.0424	$.1820^r_s$	.2440	$.4040^r_s$	.5560	.1461
$ZScore_{hyp}$	day	.0680	.0604	.0438	$.2050^r_s$	.2680	$.4400^r_s$	.5640	.1555
	week	.0616	.0544	.0438	$.1787^r_s$	.2360	$.4120^r_s$	.5600	.1449
	biweek	.0592	.0520	.0414	$.1717^r_s$	.2280	$.3920^r_s$	.5640	.1431
	month	.0536	.0496	.0394	$.1664^{r'}$	.2240	$.3760^r_s$	.5360	.1385

Table 7.3: Question routing performance of the proposed temporal ZScore approach.

the drop rate exponentially increases for *exp* model, while the increase is linear for *hyp*. For instance, with $k = 1$, the weight ratios of 1st interval to the 3rd, 5th and 10th intervals are 2, 3 and 5.5 for *hyp* model, while these ratios are 7.4, 54.6 and 8103 respectively for *exp* model. This high drop rate in *exp* model causes recent intervals to receive relatively much more weight and dominate the overall score. Only the month interval, the longest time interval tested, returned consistent significant improvements with *exp* discounting; probably because activities from the most recent couple of months provide enough data to build effective user expertise and interest models. However, the same behavior doesn't apply to shorter intervals due to lack of enough information for modeling users. The day interval performs relatively better than week and biweek possibly due to its effectiveness in estimating availability of users by focusing on recent days of activity.

On the contrary to *exp* model, more consistent and statistically significant improvements are observed with *hyp* discounting, due to its smoother decay. With a smoother decrease, activities from recent intervals do not dominate the overall model. Activities from high Δt have still comparable effects on the model. In experiments with $k = 1$, shorter intervals perform better than the longer intervals; mainly because with shorter intervals, the availability and recent interest of users can be estimated more accurately. Therefore, day interval performs better than others. However, the relative ranking of these intervals also depend on the decay factor k. The decrease goes smoother over time when k is low. When k is high, the decrease between time



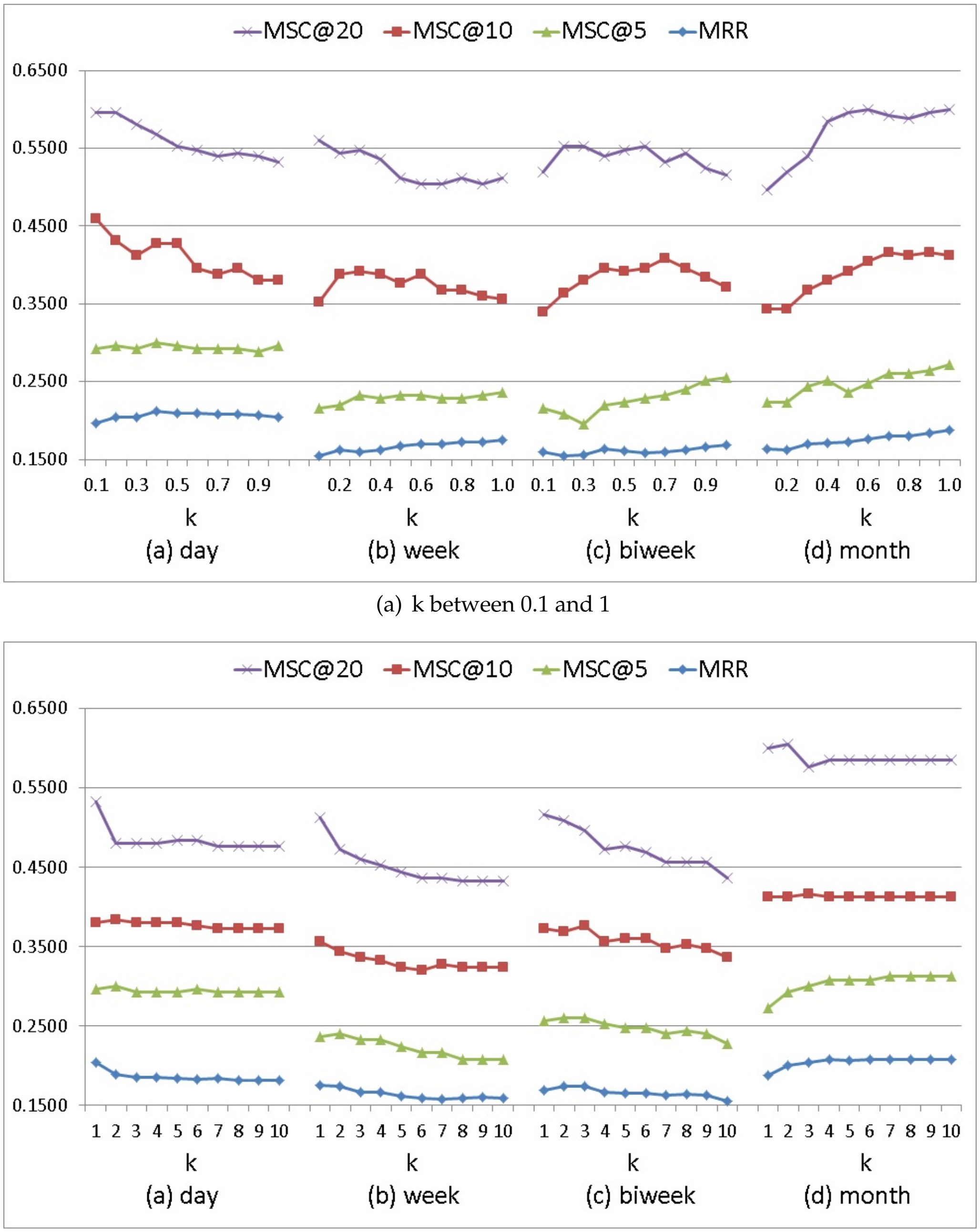
(a) k between 0.1 and 1

(b) k between 1 and 10

Figure 7.3: Question routing performance of AC_{hyp} approach with respect to different k values.

intervals becomes more drastic. In order to analyze the effects of k more clearly, the performance of proposed AC_{hyp} approach with increasing k values (from 0.1 to 1 and from 1 to 10) are presented in Figure 7.3 for different time intervals, and similarly the change of AC_{exp} with respect to k values is presented in Figure 7.4.

Several trends exist in Figure 7.3. For instance, with day intervals, the scores are highest when $k = 1$ but decrease as k gets higher values. This is because, with high values of k, the activities



(a) k between 0.1 and 1

(b) k between 1 and 10

Figure 7.4: Question routing performance of AC_{exp} approach with respect to different k values.

from small values of Δt (same day or previous day mostly) get relatively more weight in modeling expertise which negatively affects the overall ranking. On the other hand, with biweekly and monthly intervals, the performances increase as k value increases and then become more stable. This tendency towards using higher k values and giving much more value to recent biweeks and months is probably due to more effective modeling of user availability and interest in addition to expertise. Unlike days, using a couple of months activity can be enough to model users' expertise as well as their availability. Week intervals of AC_{hyp} also perform in between day, biweek and month intervals. Similar trends are also observed with $ZScore_{hyp}$.

The change of AC_{exp} with respect to varying k values is presented in Figure 7.4. Except for the MSC@20 metric, the relative change of scores is more gradual in *exp* discounting compared to *hyp* discounting (Figure 7.3) mainly because with k being equal to either 1 or 10 (or in between) will still give relatively more weight to first interval than the preceding intervals. So in both of these cases, it looks like the first interval mostly dominates the rest of the model. Overall for different time intervals, increasing the k value causes accuracy to drop, except for some metrics in month interval. For instance, the trends of varying k values in biweek intervals are different in *exp* and *hyp* discounting approaches. In *exp* discounting, using biweek intervals with high k values for identifying experts mostly returns expert candidates from the most recent biweek interval. Focusing more on the last 15 days of activity for modeling expertise is not effective, as observed from the figure.

As a result, among these approaches, the *hyp* models are more consistent and accurate than the *exp* models. Using day intervals with *hyp* discounting seems to be also more useful due to modeling and differentiating the most recent activities of users, such as their recent availabilities. In Figure 7.2, the high coefficient of variation of answering activities over weeks shows that estimating availability by using the previous week's activity may not be as effective as using the previous day. Therefore, using day interval can be better choice overall.

7.6 Summary

This chapter focuses on the dynamic aspects of CQA sites as how new users join every day or inactive users become more active over time or vice versa. There is also the change in users' topic-specific interests over time. Due to these dynamic aspects of CQAs, expertise modeling should be also more dynamic and temporal for more effective estimations. Therefore, this dissertation explores the available timestamps and addresses the following research question:

- *RQ3: What techniques can be used to identify more up-to-date topic-specific experts who have shown relatively more topic-specific expertise and interest in general and also in recently?*

Regarding this research question, this dissertation proposes adapting temporal discounting approaches from economics to expert finding task. Instead of modeling future signals to estimate the future reward, the past events are modeled to estimate users' current topic-specific expertise and interest. Two widely used counting-based approaches, *Answer Count* and *ZScore*, are modified accordingly to use the available temporal information. Several prior work that explores timestamps are also experimented with as other temporal baselines.

The experiments on the StackOverflow dataset for question routing task showed that both the proposed temporal modeling approach and the previously developed approaches that estimate and use availability of users outperform the static approach which does not use the timestamps. These consistent improvements received with the use of temporal information show the usefulness of this evidence type.

The previous temporal approaches used all user activities to estimate the availability, and then combined the availability estimates with the previously estimated topic-specific expertise score. On the other hand, our proposed approach uses temporal and content-based evidence together to construct a more dynamic model of expertise which combines the topic-specific expertise, recent topic-specific availability and interest of users. The consistent and for some metrics statistically significant improvements of the proposed temporal approach over other temporal baselines shows the effectiveness of combining temporal data with topic-specificity.

The proposed temporal discounting model depends on several factors, such as the used discounting types, length of time intervals and decay rates. These are also analyzed with respect to their effects on expertise estimation. Two discounting types are compared, and the *hyperbolic* discounting observed to be more consistent compared to the *exponential* discounting. The high drop rate of the *exponential* discounting causes activities in recent intervals to dominate the model, which returns inconsistent results. The smoother decrease in the *hyperbolic* discounting enables activities from various time intervals to affect the overall model. In terms of the length of time intervals, using shorter time intervals seems to be a better choice for differentiating very recent information, which is useful in modeling recent availability of users. As Chang and Pal [21] also showed in their experiments that using the previous day activity to estimate availability is more effective than other more complex approaches. The decay factor k is also important for the effectiveness of the model. The optimum value of k depends on interval length, in other words the size of data used in each interval. If the interval length is small (as in days) then increasing k value causes drops in effectiveness due to modeling expertise with limited amount of information coming from recent intervals. For longer intervals (like months), using higher k values and giving relatively more weight to recent activities is more effective.

The discounting property of the proposed approach enables identification of experts who regularly show expertise within environment, and also the recently joined experts or users who have shown recent topic-specific expertise. Due to these advantages, the proposed approach can be the answer to the research question. The effective performance of these proposed approaches on expert retrieval also shows that for expertise estimation related tasks which require the identified experts to take action, the dynamism of the users should be taken into account. This dissertation introduces a simplistic start to this dynamic modeling of topic-specific user expertise and interest.

Chapter 8

Combining Approaches

This chapter combines evidence from the best content, authority and temporal (if available) approaches in order to see whether a combination of evidence provides a better ranking of expertise than the individual evidence. A weighted combination of normalized scores coming from these approaches is used to get a final ranking of experts.

8.1 The Expert Blogger Finding Task

For expert blogger finding task, state-of-the-art content-based expert finding approaches were applied in Chapter 5, in order to find a ranking of expert candidates based on the content they authored. Furthermore, reading and commenting activities were explored to estimate authority of users in Chapter 6. In order to see whether authority-based evidence can improve the performance of content-based approaches, these authority scores are used to re-rerank the initially retrieved content-based expert candidates.

Due to its effective performance (in Table 5.22), the *Reciprocal Rank* approach was used as the content-based approach. For reading interaction, the best accuracy is observed with the expertise-teleported TSPR approach (Table 6.17), while for commenting interaction, the expertise-influenced TSPR approach (Table 6.18) outperformed all others. Authority scores estimated with these approaches are used in re-ranking.

The blog collection assessments contain 3 types of relevance categories, but since our aim is to rank the very expert candidates in higher ranks, a detailed analysis on the effects of re-ranking was performed only on the very expert (VE) assessed candidates, while considering all other relevance categories as irrelevant. With such an experimental evaluation, the assessment values are not graded anymore but instead they are binary. Metrics like Precision@5, MAP and MRR are used to present the results.

During re-ranking, all expert candidates from content-based approach are used, and reading and commenting authority scores of those candidates are combined to calculate the final score as shown:

$$finalScore = \lambda * contentScore + \beta * readingAuthorityScore + \theta * commentingAuthorityScore \tag{8.1}$$

where $\lambda + \beta + \theta = 1$. A parameter sweep was performed and 10-fold cross validation is applied in order to find the optimum parameter setting. In this final reranking of expert bloggers, only the content and authority-based evidence is used. Temporal information did not provide

Content/Reading/Commenting	P@5	P@10	MAP	MRR
1/0/0	.3100	.2700	.3621	.5156
0/1/0	.3350	.2700	.3514	.4887
0/0/1	.3250	.2700	.3784	.6157
0/.1/.9	.3400	.2700	.4194	.6350

Table 8.1: Expert ranking performance of combining best performing content and authority-based approaches.

any significant improvement in these experiments due to the definition of the task which is identifying expert bloggers in general. Neither in the task nor in the assessments, the recency of the information provided within the posts, or the up-to-datedness of the bloggers or their posts on the particular topic are used.

The scores after re-ranking are presented in Table 8.1. In the table, the first column shows the evidence weight used during re-ranking. The rest of the columns present the P@5, P@10, MAP and MRR scores calculated by assuming *VE* expert candidates relevant while others not. In these experiments re-ranking is performed over all expert candidates retrieved with content-based approach, but only the top ranked 10 candidates are evaluated in order to see the effect of re-ranking on top ranked results.

The top three rows present the results of using only content, only reading or only commenting authority scores in re-ranking. Using only content (1/0/0) in re-ranking returns the same ranking of experts before re-ranking, therefore it can be considered as a baseline[1]. In 0/1/0 and 0/0/1, authority scores are individually interpolated with the content-based scores. The final row, presents the results of re-ranking candidates with the optimum weight which is retrieved from 10-fold cross validation. In the table, P@10 score remains same across all experiments, which means that using authority either did not carry any expert to the top 10, or carried some but also lost some during re-ranking.

According to the table, using reading authority scores improves only the P@5, but not the others. On the other hand, commenting authority scores improve P@5, MAP and MRR scores due to it being a stronger form of evidence than reading. Weight parameters are optimized with respect to MAP score. The highest MAP score is returned with reranking with a combination of reading and commenting authority together. Overall, these results show that topic-specific authority estimated from being read and being commented are important signals of expertise which can be used to improve the best performing state-of-the art approaches.

8.2 The Question Routing Task

For question routing task, we have explored effective use of content in Chapter 5, effective use of answering activity as a signal of authority in Chapter 6, and the effects of temporal modeling in Chapter 7. In this section, the best performing approaches which use content, authority or temporal evidence are combined with using the following equation.

$$finalScore = \lambda * contentScore + \beta * authorityScore + \theta * temporalScore \tag{8.2}$$

[1] Compared to using all retrieved candidates in evaluations, using only the top 10 candidates only changes the MAP and maybe MRR but not P@5 and P@10 scores. That is the reason why '1/0/0' weighting which returns the same ranking of candidates before re-ranking, returns different MAP score but the same P@10 score in Tables 5.22 and 8.1.

Evidence Type	User Expertise	Information Need	Approach
Answer	Question Tag	TG&CP Tag	Profile-based
Question Comment	Question Tag	U-Tag	Doc-based
Answer Comment	Question Tag	TG-Tag	Profile-based

Table 8.2: Best performing content-based approaches for question routing task.

Evidence Type	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Answer	.0688	.0484	.0340	.1816	.2760	.3480	.4880	.1298
Question Comment	.0352	.0320	.0250	.1215	.1560	.2680	.3960	.0772
Answer Comment	.0520	.0368	.0268	.1560	.2160	.2960	.3920	.0957
Best Content (1/0/0)	.0688	.0484	.0340	.1816	.2760	.3480	.4880	.1298

Table 8.3: Question Routing performance of best content-based and combination of those approaches.

where $\lambda + \beta + \theta = 1$. A parameter sweep was performed and 10-fold cross validation is applied in order to find the optimum parameter setting.

Before combining content, authority and temporal evidence, we have combined the three types of content used during expertise estimation. In Chapter 5, in addition to the answers, the comments made on questions and answers have been investigated. Table 8.2 presents the best performing content-based evidence, representations and approaches for question routing. These are combined as follows:

$$contentScore = \lambda * answerScore + \beta * questionCommentScore + \theta * answerCommentScore \qquad (8.3)$$

where $\lambda + \beta + \theta = 1$. 10-fold cross validation is applied in order to find the optimum weights. In both Equations 8.2 and 8.3, the P@10 metric is optimized. The individual performances of these approaches, and the scores of the combined approach is presented in Table 8.3. The weights of the individual approaches that are used to construct the best content-based approach is also presented within parenthesis at the last row.

According to Table 8.2, using question tag to represent expertise performed the best across all content-based evidence types. Using uniform or tag generality based weightings also performed better than others. In combination of these different fields in Table 8.3, evidence coming from answers seems to dominate the evidence coming from both types of comments. Overall using answers alone returned the best content-based performance for question routing.

Table 8.4 presents the results of best content, authority, temporal and combination of these retrieved with Equation 8.2. In this table, the best content-based approach is coming from the combination in Table 8.3. The best authority approach is applying PageRank algorithm to TC graphs from Table 6.5. As for the best temporal approach, the hyperbolic discounted AnswerCount approach with $k = 1$ (from Table 7.3) is used. Among these evidence types, the authority-based one performs the worst. The best combination uses content and temporal evidence, and content evidence relatively more than temporal. The combination improves the scores mostly within top 10 and 20 ranks. Especially with this combination of evidence, for a given question around 62% of the time, a responder has been retrieved within top 20 expert candidates. Overall, these results indicate that content-based approach is very important source

Evidence Type	P @5	P @10	P @20	MRR	MSC @5	MSC @10	MSC @20	NDCG
Best Content	.0688	.0484	.0340	.1816	.2760	.3480	.4880	.1298
Best Authority	.0512	.0408	.0304	.1560	.2240	.3240	.4520	.1308
Best Temporal	.0792	.0640	.0484	.2170	.3000	.4480	.5880	.1663
Combined (.8/0/.2)	.0752	.0672	.0502	.2095	.3000	.4680	.6200	.1645

Table 8.4: Question Routing performance with best content, best authority, best temporal and combination of those approaches.

Evidence Type	User Expertise	Information Need	Approach
Answer	Answer Body	U&CP Tag	Document-based
Question Comment	Question Tag	TG-Tag	Profile-based
Answer Comment	Comment Body	CP-Tag	Document-based

Table 8.5: Best performing content-based approaches for reply ranking task.

for possible responder identification for a given question. Authority-based evidence does not add too much over to content-based evidence, mainly because both of them depend on the same answering activity. Temporal evidence also uses answering activity but the additional timestamp information helps.

8.3 The Reply Ranking Task

Similar to question routing, different types of content-based evidence is also combined with Equation 8.3, in order to find the best content-based combination for reply ranking task. The best representation for these different evidence types are presented in Table 8.5. In this table, unlike Table 8.2 for question routing, using answer and comment bodies in representing user expertise work better due to better modeling of users' topic-specific knowledge and their ability to convey this knowledge in their answers and comments.

The experimental results of these different types of content-based evidence and their combination are presented in Table 8.6. As a result of 10-fold cross-validation, combining answers and comments made on answers has been returned as the best content-based approach, which returned the highest scores across all metrics. This best content-based approach is also combined with the best authority-based approach which is applying expertise teleported PageRank to asker-responder TC graphs (from Table 6.20). The combination is performed with the following equation.

$$finalScore = \lambda * contentScore + (1 - \lambda) * authorityScore \tag{8.4}$$

The results are summarized in Table 8.7. As seen in the table, the best content-based approach works better than the authority-based approach, and so received more weight in the combination. Due to the weighting, the combination is similar to the content-based approach alone.

Approach	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5	BAP
Answer	.5968	.6609	.7208	.7748	.8456	.3200
Question Comment	.5771	.6450	.7121	.7734	.8402	.3120
Answer Comment	.6088	.6601	.7259	.7779	.8471	.3400
Best Content (.3/0/.7)	.6093	.6701	.7317	.7801	.8494	.3440

Table 8.6: Reply ranking performance with best content-based and combination of those approaches.

Approach	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5	BAP
Best Content	.6093	.6701	.7317	.7801	.8494	.3440
Best Authority	.5660	.6424	.7028	.7693	.8350	.2760
Combined (.9/.1)	.6065	.6753	.7284	.7882	.8503	.3440

Table 8.7: Reply ranking performance with best content, best authority, and combination of those approaches.

8.4 Summary

This chapter exploits combining three types of evidence: content, authority and temporal for all three expertise retrieval related tasks. For the expert blogger finding task, reading and commenting authority scores are used to rerank the initial content-based ranking of experts. Reranking with commenting which is a more explicit form of authority returns better accuracy than reranking with reading authority. However, the combination performs better than both of the individual authority signals, which shows the effectiveness of both type of authority signals.

For both question routing and reply ranking tasks, different types of content-based evidence are combined initially. In question routing task, using answers outperforms both types of comments, therefore returns the best performance alone. On the other hand, in reply ranking task, due to the useful evidence of expertise behind comments made on answers, the combination of answers and comments on answers returns the best performance.

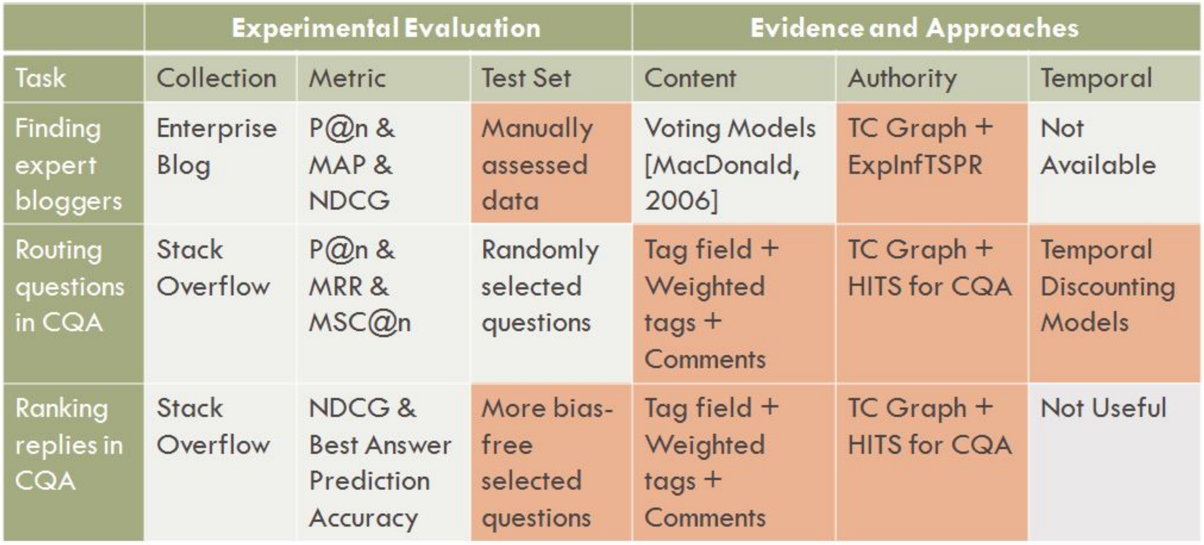
In terms of combining other types of evidence, the authority-based evidence returns the lowest performance in both tasks, and so does not improve upon other evidence types. As explained in Chapter 6, this is due to multi-propagation approaches not making too much difference in CQA answering networks. In question routing task, the use of temporal evidence together with the content-based evidence returns the best performance.

Chapter 9

Conclusion

This dissertation recognizes the increasing popularity of social media in users' professional and personal environments, and provides effective approaches to improve expertise related applications in these environments. Previously developed expert finding approaches are adapted to these environments, and they are further improved by using the available evidence social media provides; such as different user-created content types, authoritative interactions among users, and temporal information coming from the timestamped content and user interactions. The proposed expert finding system is applied to two social media collections for three expertise related tasks. An overview of the dissertation work is presented in Figure 9.1 with respect to these three expertise related social media tasks.

In the table, the cells in light orange color present the dissertation work that is described in previous chapters. For the cells with light grey color, we either used the standard and state-of-the-art approaches presented in the table, or no action was taken regarding that issue for the particular task. For each task, the collection and the metric used in experiments, and how the test set was constructed are presented initially. The evidence and approaches part of the table



	Experimental Evaluation			Evidence and Approaches		
Task	Collection	Metric	Test Set	Content	Authority	Temporal
Finding expert bloggers	Enterprise Blog	P@n & MAP & NDCG	Manually assessed data	Voting Models [MacDonald, 2006]	TC Graph + ExpInfTSPR	Not Available
Routing questions in CQA	Stack Overflow	P@n & MRR & MSC@n	Randomly selected questions	Tag field + Weighted tags + Comments	TC Graph + HITS for CQA	Temporal Discounting Models
Ranking replies in CQA	Stack Overflow	NDCG & Best Answer Prediction Accuracy	More bias-free selected questions	Tag field + Weighted tags + Comments	TC Graph + HITS for CQA	Not Useful

Figure 9.1: Overview of the dissertation work.

summarizes the identified useful evidence, and the successful content-based, authority-based and temporal approaches that are either proposed by us or adapted from the prior work.

This chapter summarizes the work presented in this dissertation, describes the main contributions and research findings, and concludes with future research directions.

9.1 Summary

This dissertation mainly focuses on exploiting available evidence of expertise in social media environments for more effective identification of expertise. These sources of expertise can be divided into three; user created content, user interactions and timestamps. Content is useful for identifying an initial set of possible experts, while interactions between these users can be used to estimate more influential and authoritative ones among them. Temporal evidence is important for estimating more up-to-date experts that are still interested in the topic. This temporal evidence is especially useful for tasks that require some kind of action from these identified experts. Depending on the environment and task being worked on, these sources of evidence are combined for more effective expert identification.

Data from two different social media environments are used in this dissertation; (1) an intra-organizational blog collection and (2) a popular community question answering site, StackOverflow. Three expertise related tasks are applied to these collections. For blog collection, the task is to find expert bloggers for a given query. For StackOverflow collection, routing questions to question-specific expert users and ranking replies based on corresponding responders' question-specific expertise tasks are worked on. Widely used metrics for these or similar tasks are used to present the experimental results. Even though experiments were specifically performed only on CQA and blog collections, we believe that some of our research findings also apply to other social media environments.

Finding expert bloggers task is very similar to TREC's expert finding task, therefore similar assessment and evaluation approaches to TREC's task are used for this task. Unbiased manual assessments were performed and used during evaluations. For question routing and reply ranking tasks, evaluation approaches that have been widely used in prior work for these particular tasks were used directly. Randomly selected questions were used as the test set for question routing task, however, for reply ranking task bias analysis performed on widely used ground truth data revealed certain types of biases which caused us to be more selective during test set construction.

9.1.1 Selecting Less Biased Questions for Reply Ranking Task

User feedbacks in CQA sites, like the number of received votes or best answer selections, are widely used by prior work in evaluations as graded relevance assessments. However, this user data collected in an uncontrolled way may contain several biases. This dissertation analyzed two specific types of biases, temporal and presentation, which are caused by the user interface of the StackOverflow and users using it. Analysis of these biases revealed that both of them favor replies that are posted earlier, and affect the relative ranking of approaches. Therefore, the following research question has been investigated: *'RQ4: What techniques can be used to construct less biased test collections based on the identified cases of biases?'*. In order to construct less biased test sets, questions and answers with user feedback received similar to manual assessments, more specifically questions where the last reply received the highest votes, are selected. The experiments showed high correlation between this less biased test set and manual assessments.

9.1.2 Content-based Approaches

Effective expert identification in any environment depends on the right representation of expertise for representing users and information needs. Therefore, this dissertation initially addressed the following research question: *'RQ1: What are the most effective representations of information need and user expertise used for identifying expertise in question routing and reply ranking tasks in CQAs?'*.

Social media environments may contain different content types with different levels of representation of expertise of corresponding users. For instance CQAs consist of questions, answers and comments. Most of the prior work focused on using detailed fields like question title and body, and answer body to represent expertise in these environments. Analyzing these different fields revealed that using very specific details to represent the information need or the user expertise is not necessarily more useful than using more general categorizations of expertise. Therefore, this dissertation proposed using question tags which looks like prerequisite knowledge areas of expertise. Using these tags for representing users' expertise areas and the required expertise to answer a given question returned statistically significant improvements in question routing task. For reply ranking task, searching question tags over previous replies of users outperformed other representations mainly due to using the content and presentation quality of replies which are positively correlated with the votes they receive.

These questions tags do not have to be equally effective for expertise estimation. Some tags can be more general or more representative of the information need searched. Therefore, an ordering or weighting among these tags are explored, and three tag weighting approaches depending on the ordering of the information need, the generality of the tag and expert candidates' expertise areas are proposed. For question routing task, using tag generality weighting provided statistically significant improvements over the best performing approach. For reply ranking task, using askers' ordering of the information need and weights retrieved from probabilistically expert candidates provided improvements.

These differences of user representation and tag weightings between tasks indicate that for question routing task, searching experts with using general representations works fine for identifying experts who can answer the question. However, for reply ranking task, combining specific information from information need and user's previous replies is more effective for identifying specific expertise scores of responders to rank their answers based on accuracy and descriptiveness.

The prior work mostly focused on using questions and answers, however, this dissertation also analyzed comments as another source of evidence of expertise. A group of comments on questions and answers were analyzed in terms of their representation of expertise, and it has been observed that most of these comments are indications of expertise, such as answering questions, or making suggestions to improve questions or answers. Using comments returned similar performance to using replies most of the time. Furthermore, for reply ranking task, using comments made on answers outperformed using answers themselves. This is probably due to comments which were posted to make a suggestion or correction of answers, which is a strong indication of expertise. The proposed expertise representation and weighting approaches were also applied to comments, and trends observed with replies were also observed with comments.

Exploring different representations of expertise provided another unexpected outcome, which is a change in the relative ordering of widely used approaches. Previous research of expert finding agreed that document-based approaches, which are topic-dependent approaches, outperform topic-independent profile-based approaches. However, using question tags to represent information need and user expertise, improved the performance of profile-based approach to the level of (and for some metrics even better than) document-based approach. With this improvement, profile-based approaches which are computationally more efficient than document-based approaches can be used without loss in effectiveness.

9.1.3 Authority-based Approaches

The underlying social networks are important for analyzing the influence of users in these environments. Users who interact with other users, or whose content grasp the interest of others, are more likely to be effective members of the community and authorities of the graph. In this dissertation, these available user networks are exploited in order to estimate network-based expertise of users (authority scores).

The commonly used asker-responder networks were tested for StackOverflow collection. The intra-organizational blog collection came with access logs (who accessed to whose post information), which is unique to organizational collections. Therefore, in addition to commenting, the widely available and explicit form of interaction, reading interactions which are more implicit, were used to estimate authoritative experts.

Prior research mostly applied authority network construction and estimation approaches that had been originally developed for web pages, to user interactions. However, users are different than web pages, therefore directly applying these approaches does not necessarily return the expected outcomes. This dissertation focuses on these algorithms and the underlying assumptions, and answers the following research question *'RQ2: Do the assumptions of topic-specific authority estimation approaches developed for web pages hold for user authority networks in social media? For the ones that do not, what kind of algorithmic modifications can be performed so that they hold, and is it possible to make additional assumptions and necessary modifications which can provide more effective and efficient topic-specific authority-based expertise estimations?'*.

The connectedness and topic diversity of users are different than web pages, therefore the topic-specific HITS graphs developed for web pages may not return topic-specific graphs for users. This dissertation proposed *Topic Candidate* (TC) graphs, which are more topic-specific adaptations of HITS web graphs. Applying these TC graphs to two datasets for three tasks returned consistent and statistically significant improvements. Furthermore, the constructed topic-focused and smaller graphs drastically decreased the running times of authority estimation algorithms, from hours to seconds in some cases.

In addition to the characteristics of the node (entity), the connections between these nodes are also important for the effectiveness of authority estimations. Some connection types do not satisfy the necessary underlying assumptions of authority estimation algorithms. For example, in asker-responders networks, a user's hub score, which directly affects the connected users' authority scores, is positively correlated with the number of questions asked by the user. However, the high frequency of asking questions in not a good indication of expertise. This dissertation tried to identify such cases of inconsistencies between algorithms and their inputs, and proposed necessary adaptations, which provided small but consistent improvements in performance.

The principle of authority estimation through propagation is that being connected from an authoritative node is an indication being authority itself. Based on this intuition, in order to identify more topic-specific expert authorities, initially calculated (mostly from content-based approaches) expertise scores of users are proposed to be used in authority estimation as influence to be propagated. Furthermore, as already proposed by the prior work, these initial expertise scores were also used as teleportation weights to improve the probability of random visits to expert users. These adaptations towards more topic-specific authority estimation worked, and both of these approaches returned statistically significant improvements when applied to blog collection. Influencing expertise approach outperformed using expertise in teleportation when the tail nodes of the directed edges have some prior topic-specific knowledge or interest. If nodes don't have prior expertise then propagating that expertise to other users does not help as much as using expertise to increase the probability of teleporting to expert users.

Using initially estimated expertise in authority estimation did not cause consistent or significant changes in question answering authority networks in StackOverflow. Analysis revealed that asker-responder *TC* graphs have similar characteristics of bipartite graphs, as for a topic, a set of users with necessary topic-specific expertise only answer questions, while another set of users with lack of topic-specific knowledge, only ask questions. It has been observed that with such authority networks, using initially estimated expertise either for influencing or teleportation does not work as intended, because even the baseline multi-step propagation approaches do not perform much better than the *InDegree*, one-step propagation approach.

9.1.4 Temporal Approaches

The prior work on expert finding mostly developed approaches that work on static (snapshots) datasets for static tasks. However, unlike most of the web documents, humans which are being retrieved in this case, are very dynamic in nature. This relatively changing state of humans becomes more important especially for expertise related tasks which require action from the identified expert candidates. For such tasks, this dynamic nature of the humans should be also taken into account for more effective performance. Therefore, this dissertation addresses the following research question '*RQ3: What techniques can be used to identify more up-to-date topic-specific experts who have shown relatively more topic-specific expertise and interest in general and also in recently?*'.

This changing and dynamic aspect of users have not been explored enough in expert finding, most probably due to lack of temporal evidence in previous data collections. However, in social media environments, all user-created content, and interactions with each other are timestamped. For tasks like question routing in CQA, previous research used these timestamps to estimate the availability of users. Even though these availability estimates did not model the topic-specific availability of users (in other words, their recent topic-specific interests), they still improved the effectiveness of the routing performance. Compared to these topic-independent temporal models, this dissertation proposed building topic-dependent temporal models.

In this regard, dynamic expertise models are proposed by adapting the temporal discounting models from economy and psychology in order to model events from past, by giving more value to recent evidence of expertise compared to older evidence. *Exponential* and *hyperbolic* discountings were integrated into the existing expert finding approaches for question routing. Both of these temporal discounting models provided statistically significant improvements compared to static approaches and their combinations with availability estimations. Further analysis revealed that the effectiveness of these temporal models depend on the rate of decay parameter, *k*, length of intervals, and the discounting type used. Overall, it has been found that for longer time intervals with enough data in each interval, giving relatively more value to recent one seems to work just fine. However, in case the interval size is short, the decay should be smoother so that data coming from earlier intervals can be also used effectively. Overall, this better weighing of older and recent evidence produces better and more up-to-date estimates of users' topic-specific expertise.

9.2 Contributions

Humans leave useful feedbacks, like best answer selections and votes, to CQA sites. These positive or negative users' feedbacks have been widely used as ground truth even though they were not constructed in controlled environments. Not being in full control of the default working principles of the system or the humans leaving feedback may result in collection of biased data. This dissertation specifically focused on two types of biases, presentation and temporal, and showed their effects. These biases which haven't been noticed before caused significant changes in the relative ranking of approaches, and raised questions on the credibility of the widely accepted research findings of the prior work on this area. These identified biases are also important for raising awareness on the risks of using ground truth data extracted from the web. Being more skeptical about such data, and looking for possible cases of bias are important. Finding such biases should not cause researchers to stop using these collections. As shown in this dissertation, identification of these biases may lead to more selective use of data which may return less biased test sets.

What to search for and where to search are two important factors in retrieval tasks. This dissertation focused on these two questions for the expert finding task in CQA sites. Available and different types of content-based evidence are explored to find effective and also approach-independent representations for both the information need (query) and user expertise. The significant improvements with the proposed representations show the power of effective content representation in expert finding. Furthermore the proposed representation improved a less effective but more efficient algorithm's accuracy to the level of a more effective algorithm. This is not only important for having an expert finding approach that works both effectively and efficiently, but also important for showing how a widely accepted relative ranking of approaches can change with different representations of information need and user expertise.

In addition to improving the content-based representation of expertise, the network-based representation of expertise has been also analyzed in this dissertation. The prior work mostly adapted previously developed web page authority estimation approaches to user interaction networks in order to estimate expertise of users. During adaptation of approaches there is value to understanding the input types and checking whether the assumptions used while developing the original approach still holds in another environment with different inputs. This dissertation initially analyzed the widely used adaptations of web page authority estimation algorithms to user interaction networks. Understanding the interaction and structure behind the graphs helped us to see the limitations of these adaptations which are probably the cause of the inconsistent results of prior work. Modifying these user interaction graphs and network-based algorithms provided consistent improvements in both accuracy and efficiency. An example to this is the construction of topic-specific graphs, which are proposed in order to make sure that authority propagates among topic related user nodes through topic related edges (user interactions). Similar to content-based representation of expertise, these graphs (network-based representations of expertise) provided statistically significant improvements in accuracy, but also decreased the computational running time drastically. This improvement in efficiency is especially important in terms of enabling real time (online) authority estimation for a given topic over a user interaction graph that is specifically constructed for the particular topic.

In addition to showing why direct adaptations based on similarity of tasks may not always work, this dissertation also showed how even the most conventional knowledge in general Information Retrieval may not return the expected outcomes with expert finding task. The term specificity weighting which has been widely studied and shown to be very effective in document retrieval in terms of improving the accuracy, returned a different outcome with expert retrieval for the question routing task. Unlike document retrieval where specific terms are more important for identifying topic-specific documents, in identifying responders for a given question, making sure that users are experts on general aspects of the requested information need is found to be more useful than trying to match users with very specific expertise areas.

Humans are dynamic in nature with changing levels of interest, expertise and availability. This dissertation proposed moving from static to dynamic modeling of expertise in order to better model expertise of these constantly changing users. The improvements received with these dynamic models show the power of temporal information in modeling expertise. Future expert retrieval approaches should make use of this temporal evidence and explore additional ways of using it. Furthermore, this use of temporal discounting towards past activities can be applied to other user modeling tasks that make use of past activity to predict present or future activity.

Overall, this dissertation provided a clearer view of expert finding in social media, and showed that for a given topic, retrieving relevant users is different than retrieving documents. We showed that the conventional wisdom of Information Retrieval, in other words the generally used and accepted approaches from document search do not necessarily work as expected for expert search in social media.

9.3 Future Work

This dissertation introduced better representations and additional evidence of expertise in CQAs, construction of more topic-specific authority graphs, better adaptations of topic-specific authority estimation approaches, and more dynamic modeling of expertise. These ideas enable new research problems, and can be extended into following directions.

9.3.1 Temporal Modeling of Topic-Specific Authority

Dynamic aspects of these environments, the changing interest and activity levels of users have been shown in this dissertation. In addition to those, there is also the changing topic-specific expertise levels of users. For instance some users start using CQA sites to ask their questions on certain topics, but over time as they learn and become more experienced, their role within the site can change as they start answering others' questions on the particular topics.

This change in user's level of expertise may not have any effect on some expertise estimation approaches. For instance, *AnswerCount* approach depends only on the number of answers posted. The number of questions asked have been used in *ZScore* approach which performs badly compared to *AnswerCount*. This shows that using previously posted questions on particular topics may not be effective. The effects of these old asking activities are minimized with the temporal discounting approach in Chapter 7.

However, approaches that depend on the role of the users, such as authority-based approach where there is a propagation of expertise from users with less topic-specific authority to users who are more authoritative, even the small changes within roles may have significant effects.

In the case of CQAs, for users who were askers before, becoming responders later on will cause those users to propagate their authority they retrieved from askers of the questions they responded to recently, to responders who had answered their questions a very long time ago.

This misdirection of authorities can be prevented by using the available timestamps of the activities, which can be either used to construct more temporal authority graphs, or used with authority estimation approaches to estimate temporal discounted authority scores. Overall, the temporal discounting models of expertise, the topic-specific authority graph (Topic Candidate graph) construction and estimations approaches proposed in this dissertation can be combined and extended to estimate more dynamic topic-specific authorities.

We believe that this is the most promising direction of research because it combines several types of evidence to estimate one expertise score. In content-based approaches, we only used the content-based evidence, but with temporal modeling of expertise we combined content-based evidence with temporal evidence which provided significant improvements in effectiveness. Similarly in proposed authority approaches, combining text-based evidence and network-based evidence returned effective and efficient improvements over using only the network-based evidence. Therefore, we believe that combining content-based, network-based and temporal evidences in one approach may be more effective than any other combination of these evidences.

9.3.2 Selective Use of Comments for Expertise Estimation

This thesis showed that comments can be useful forms of evidence to estimate expertise depending on the underlying reason they are posted. Several user cases and motivations why people post comments have been shown, such as to ask for a clarification, to make a suggestion or correction, or to praise the post. In this dissertation, all these comments are assumed to be indications of expertise, and used to model expertise.

However, not all these comments can be equally useful in indicating expertise of their authors. Suggesting a correction can be considered as a strong form of evidence, but praising a post or thanking to post's author do not always have to be performed by topic-specific experts. Categorizing comments based on their level of expertise, and using ones that are stronger form of evidence may give better estimates of expertise. For categorization, the length of the comments and the existence of sentimental or praising vocabulary can be useful. Commenting dialog between users can be also exploited to retrieve effective features. Furthermore, whether edits have been made to the post after receiving comments, and if so what kind of edits have been done, can be useful for estimating the degree of expertise behind comments. In general due to the importance of being more selective in evidence in terms of its expertise value, automatic ways of categorizing this evidence based on its indication of expertise, is an area that requires future attention.

This future work is especially high priority for reply ranking task in which comments are shown to be very useful in identifying expertise. This dissertation only analyzed comments as content-based evidence. They have not been exploited as network-based evidence, because unlike answering, commenting may not always be an indication of expertise. Using these in user interaction graphs can cause wrong propagation of authority which returns wrong estimations of authority-based expertise scores.

9.3.3 Estimating When to Use Multi-Step Propagation

Analysis and experiments performed on the proposed topic-specific asker-responder authority networks revealed that in bipartite-like graphs with most of the users either only ask (outgoing) or only answer (incoming) questions, authority may not be propagated to second or third degree nodes. This is mainly because after one step propagation, there may not be many nodes with outgoing links to follow. With such networks, multi-step propagation algorithms like PageRank perform very similar to one-step propagation approaches like InDegree in terms of effectiveness.

However, depending on the graph, these multi-step propagation algorithms may require longer running times. Even for a complete bipartite graph, at least two iterations are required in order to converge (with normalized scores the second iteration will return the same results with first one, and so the algorithm converges). However, for graphs which are not fully bipartite, convergence may require more iterations and more time, which may not be worth compared to similarly effective but more efficient InDegree approach. Even though, the proposed Topic Candidate graphs are very efficient for real time applications, iterating only once or multiple times over these graphs can have significant effects on overall running time.

The structural properties of graphs can be used to develop prediction models in order to estimate when to use more effective but less efficient multi-propagation approaches, or less effective but more efficient one-step algorithms. Features like second or third degree connectivity can be useful for this prediction. Compared to the previous two future work directions, this line of research may not seem very interesting since it does not improve the accuracy; but it is important for real-time applications of these approaches, especially when system's running time is a concern. Performing online multi-step propagation, even though with the improved efficiency, can be still time consuming and should be ignored if not necessary.

9.3.4 Being More Selective in Routing Questions

Another future work related to real time applications of expert finding is to be more selective during routing questions. Depending on the topic coverage of the system, identifying effective ways on how to not route all the questions on the same topic to the same user, but instead divide questions to experts more evenly is important.

Such a load spreading can be performed in several ways. For a given list of identified expert users for a list of questions, the temporal behaviors of users can be used to find their estimated time of answering. Routing can be performed depending on the urgency of the questions. Furthermore, the difficulty level of the question can be used to select possible responders among the identified experts. For instance, top ranked very expert users may not be disturbed for easy questions which can be answered accurately by the users at lower ranks.

This line of research can be best evaluated with a running system with real users using it. This research direction is useful and may be necessary if the system is overloaded with questions of same or similar category and the same set of users are identified as the expert users.

9.3.5 Using User Expertise as Content Reliability

Ad-hoc search is still one of the most popular applications in social media, and relevancy is the most important part of this task but not the only. Social media is a platform where any user can create and share content without being checked on the accuracy or reliability of the content.

Over time information of different quality accrues in these environments, and so identifying the reliable content becomes crucial for the success of ad-hoc search.

In such environments, expert finding can be used as a way to estimate reliability of users and their content. Since every content is linked to its user, identifying the correctness and reliability of the content becomes the problem of identifying its author's expertise on the topic of the content. These estimated expertise scores of content creators on the particular topic of the content can be used as the reliability measure to improve ad-hoc document search in social media.

Compared to other future research directions, this line of research does not try to improve the effectiveness or efficiency of expert finding systems, therefore it may not seem very high priority. However, this is another task (similar to reply ranking) that expertise estimation can be useful. Testing expert finding approaches and analyzing how they perform on this particular task is an interesting research problem which can lead to new findings in document retrieval research.

Bibliography

[1] The W3C test collection. `http://research.microsoft.com/en-us/um/people/nickcr/w3c-summary.html`, 2005. 1.1

[2] Douglas G. Altman. *Practical Statistics for Medical Research*. Chapman & Hall, 1991. 4.1.1

[3] Çiğdem Aslay, Neil O'Hare, Luca Maria Aiello, and Alejandro Jaimes. Competition-based networks for expert finding. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 1033–1036, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2034-4. 2.2.2, 4.6, 4.6.3, 4.6.3

[4] Peter Bailey, Arjen P. de Vries, Nick Craswell, and Ian Soboroff. Overview of the TREC 2007 Enterprise Track. In *Proceedings of the 16th Text REtrieval Conference*, TREC '07, Gaithersburgh, MD, USA, 2007. National Institute of Standards and Technology (NIST). 2.1

[5] Krisztian Balog. *People Search in the Enterprise*. PhD thesis, University of Amsterdam, 2008. 2.1.3

[6] Krisztian Balog and Maarten de Rijke. Finding experts and their details in e-mail corpora. In *Proceedings of the 15th International Conference on World Wide Web*, WWW '06, pages 1035–1036, New York, NY, USA, 2006. ACM. ISBN 1-59593-323-9. 2.1.1

[7] Krisztian Balog and Maarten de Rijke. Associating people and documents. In *Proceedings of the 30th European Conference on Information Retrieval*, ECIR' 08, pages 296–308, Berlin, Heidelberg, 2008. Springer-Verlag. 2.1.1

[8] Krisztian Balog, Leif Azzopardi, and Maarten de Rijke. 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*, SIGIR '06, pages 43–50, New York, NY, USA, 2006. ACM. ISBN 1-59593-369-7. 2.1.1, 2.1.2, 2.1.2, 2.1.3, 2.1.3, 2.1.4, 2.1.5, 5.3

[9] Krisztian Balog, Ian Soboroff, Paul Thomas, Nick Craswell, Arjen P. de Vries, and Peter Bailey. Overview of the TREC 2008 Enterprise Track. In *Proceedings of the 17th Text REtrieval Conference*, TREC '08, Gaithersburgh, MD, USA, 2008. National Institute of Standards and Technology (NIST). 2.1

[10] Krisztian Balog, Yi Fang, Maarten de Rijke, Pavel Serdyukov, and Luo Si. Expertise retrieval. *Foundations and Trends in Information Retrieval*, 6(2-3):127–256, 2012. 1.1, 2.1.5, 4.2.1

[11] Krishna Bharat and Monika R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 104–111, New York, NY, USA, 1998. ACM. ISBN 1-58113-015-5. 6.3.1

[12] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, March 2003. ISSN 1532-4435. 2.2.2

[13] Mohamed Bouguessa, Benoît Dumoulin, and Shengrui Wang. Identifying authoritative actors in question-answering forums: The case of Yahoo! Answers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 866–874, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-193-4. 2.2.2, 3.2, 4.6, 4.6.3, 4.6.3, 5.1.4, 6.1.1, 6.1.2, 6.5.3

[14] Joanna Brenner. Pew internet: Social networking (full detail). `http://pewinternet.org/Commentary/2012/March/Pew-Internet-Social-Networking-full-detail.aspx`, 2013. Accessed: May 2013. 1

[15] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th International Conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, 1998. Elsevier Science Publishers B. V. 1.1, 2.1.4, 6, 6.1.1

[16] Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic Query Expansion Using SMART: TREC 3. In *Proceedings of the 3rd Text REtrieval Conference*, TREC '94, Gaithersburgh, MD, USA, 1994. National Institute of Standards and Technology (NIST). 3.1

[17] Jacques Bughin, Angela Hung Byers, and Michael Chui. How social technologies are extending the organization. Technical report, The McKinsey Global Institute, 2011. 1

[18] Stefan Büttcher, Charles Clarke, and Gordon V. Cormack. *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, 2010. ISBN 0262026511, 9780262026512. 6.5.2

[19] Yuanzhe Cai and Sharma Chakravarthy. Predicting answer quality in Q/A social networks: Using temporal features. Technical report, Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX, 2011. 7.1, 7.1, 7.2

[20] Christopher S. Campbell, Paul P. Maglio, Alex Cozzi, and Byron Dom. Expertise identification using email communications. In *Proceedings of the 12th ACM Conference on Information and Knowledge Management*, CIKM '03, pages 528–531, New York, NY, USA, 2003. ACM. ISBN 1-58113-723-0. 1.1, 2.1.4, 2.1.4, 6.1.2

[21] Shuo Chang and Aditya Pal. Routing questions for collaborative answering in community question answering. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, pages 494–501, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2240-9. 2.2.1, 2.2.2, 4, 3.3, 4.2.1, 5.2, 7.2, 7.5.1, 7.6

[22] Haiqiang Chen, Huawei Shen, Jin Xiong, Songbo Tan, and Xueqi Cheng. Social network structure behind the mailing lists: ICT-IIIS at TREC 2006 Expert Finding Track. In *Proceedings of 17th Text REtrieval Conference*, TREC '06, Gaithersburgh, MD, USA, 2006. National Institute of Standards and Technology (NIST). 2.1.4, 6.1.1, 6.1.2

[23] Lin Chen and Richi Nayak. Expertise analysis in a question answer portal for author ranking. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '08, pages 134–140, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3496-1. 2.2.2

[24] J. Cohen. Weighted kappa: Nominal scale agreement provision for scaled disagreement

or partial credit. *Psychological Bulletin*, 1968. 9.3.5

[25] Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20, 1960. 9.3.5

[26] Nick Craswell, Arjen P. de Vries, and Ian Soboroff. Overview of the TREC 2005 Enterprise Track. In *Proceedings of the 14th Text REtrieval Conference*, TREC '05, Gaithersburgh, MD, USA, 2005. National Institute of Standards and Technology (NIST). 1.1, 2.1, 3

[27] Hongbo Deng, Irwin King, and Michael R. Lyu. Enhancing expertise retrieval using community-aware strategies. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 1733–1736, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-512-3. 2.1.4

[28] Byron Dom, Iris Eiron, Alex Cozzi, and Yi Zhang. Graph-based ranking algorithms for e-mail expertise analysis. In *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, DMKD '03, pages 42–48, New York, NY, USA, 2003. ACM. 1.1, 2.1.4, 2.1.4, 6.1.1, 6.1.2

[29] Kate Ehrlich, Ching-Yung Lin, and Vicky Griffiths-Fisher. Searching for experts in the enterprise: Combining text and social network analysis. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work*, GROUP '07, pages 117–126, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-845-9. 2.1.4

[30] Yi Fang, Luo Si, and Aditya P. Mathur. Discriminative models of integrating document evidence and document-candidate associations for expert search. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 683–690, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0153-4. 2.1.5, 2.1.5, 2.1.5

[31] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, October 2006. ISSN 0920-5691. 2.1.4

[32] Joseph L. Fleiss, Bruce Levin, and Myunghee Cho Paik. *Statistical Methods for Rates & Proportions*. Wiley, 3rd edition, 2003. 4.1.1

[33] Edward A. Fox and Joseph A. Shaw. Combination of multiple searches. In *Proceedings of the Third Text REtrieval Conference*, TREC '94, Gaithersburgh, MD, USA, 1994. National Institute of Standards and Technology (NIST). 2.1.3

[34] Yupeng Fu, Rongjing Xiang, Yiqun Liu, Min Zhang, and Shaoping Ma. Finding experts using social network analysis. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, WI '07, pages 77–80, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3026-5. 2.1.4

[35] Jinwen Guo, Shengliang Xu, Shenghua Bao, and Yong Yu. Tapping on the potential of Q&A community by recommending answer providers. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 921–930, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. 2.2.1, 2.2.2

[36] Donna Harman. Overview of the first TREC conference. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 36–47, New York, NY, USA, 1993. ACM. ISBN 0-89791-605-0. 3.1

[37] Taher H. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the 11th International*

Conference on World Wide Web, WWW '02, pages 517–526, New York, NY, USA, 2002. ACM. ISBN 1-58113-449-5. 1.1, 6.1.1

[38] Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs: A study and analysis of user queries on the web. *Information Processing & Management*, 36 (2):207 – 227, 2000. ISSN 0306-4573. 5.1.5

[39] Jian Jiao, Jun Yan, Haibei Zhao, and Weiguo Fan. Expertrank: An expert user ranking algorithm in online communities. In *Proceedings of the 2009 International Conference on New Trends in Information and Service Science*, pages 674–679. IEEE, 2009. ISBN 978-0-7695-3687-3. 2.1.4

[40] Shen Jie, Shen Wen, and Fan Xin. Recommending experts in Q&A communities by weighted hits algorithm. In *Proceedings of the 2009 International Forum on Information TEchnology and Applications*, IFITA'09, pages 151–154. IEEE Computer Society, 2009. ISBN 978-0-7695-3600-2. 2.2.2

[41] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972. 5.1.3

[42] Pawel Jurczyk and Eugene Agichtein. Discovering authorities in question answer communities by using link analysis. In *Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 919–922, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-803-9. 2.1.4, 2.2.2, 6.1.1, 6.1.2, 6.4

[43] Pawel Jurczyk and Eugene Agichtein. Hits on question answer portals: Exploration of link analysis for author ranking. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 845–846, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. 2.1.4, 6.1.2

[44] Maryam Karimzadehgan, Ryen W. White, and Matthew Richardson. Enhancing expert finding using organizational hierarchies. In *Proceedings of the 31th European Conference on Information Retrieval*, ECIR '09, pages 177–188, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 978-3-642-00957-0. 2.1.4

[45] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999. ISSN 0004-5411. 1.1, 2.1.4, 2.1.4, 6, 6.1.2, 6.1.2

[46] Robert Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 191–202, New York, NY, USA, 1993. ACM. ISBN 0-89791-605-0. 4.3

[47] Darren Kuo. On word prediction methods. Technical Report UCB/EECS-2011-147, EECS Department, University of California, Berkeley, Dec 2011. 5.1.6

[48] K. L. Kwok, Laszlo Grunfeld, and David D. Lewis. TREC-3 Ad-Hoc, Routing Retrieval and Thresholding Experiments using PIRCS. In *Proceedings of 3rd Text REtrieval Conference*, TREC '94, pages 247–255, Gaithersburgh, MD, USA, 1994. National Institute of Standards and Technology (NIST). 3.1

[49] J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1), 1977. 4.1.1

[50] Victor Lavrenko and W. Bruce Croft. Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information*

Retrieval, SIGIR '01, pages 120–127, New York, NY, USA, 2001. ACM. ISBN 1-58113-331-6. 5.1.3

[51] Baichuan Li and Irwin King. Routing questions to appropriate answerers in community question answering services. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1585–1588, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0099-5. 2.2.1, 2.2.2, 3.3, 7.2

[52] Baichuan Li, Irwin King, and Michael R. Lyu. Question routing in community question answering: Putting category in its place. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 2041–2044, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0717-8. 2.2.1, 2.2.2

[53] Juanzi Li, Jie Tang, Jing Zhang, Qiong Luo, Yunhao Liu, and Mingcai Hong. EOS: Expertise oriented search using social networks. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 1271–1272, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-654-7. 2.1.4

[54] Ching-Yung Lin, Kate Ehrlich, Vicky Griffiths-Fisher, and Christopher Desforges. Small-Blue: People mining for expertise search. *IEEE MultiMedia Magazine*, 15(1):78–84, 2008. 2.1.4

[55] Ching-Yung Lin, Nan Cao, Shixia Liu, Spiros Papadimitriou, Jimeng Sun, and Xifeng Yan. Smallblue: Social network analysis for expertise search and collective intelligence. In Yannis E. Ioannidis, Dik Lun Lee, and Raymond T. Ng, editors, *International Conference on Data Engineering*, pages 1483–1486. IEEE, 2009. ISBN 978-0-7695-3545-6. 2.1.4

[56] Qiaoling Liu and Eugene Agichtein. Modeling answerer behavior in collaborative question answering systems. In Paul Clough, Colum Foley, Cathal Gurrin, GarethJ.F. Jones, Wessel Kraaij, Hyowon Lee, and Vanessa Mudoch, editors, *Advances in Information Retrieval*, volume 6611 of *Lecture Notes in Computer Science*, pages 67–79. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-20160-8. 7.2

[57] Xiaoming Liu, Johan Bollen, Michael L. Nelson, and Herbert Van de Sompel. Co-authorship networks in the digital library research community. *Information Processing and Management*, 41(6):1462–1480, December 2005. ISSN 0306-4573. 2.1.4

[58] Xiaoyong Liu, W. Bruce Croft, and Matthew Koll. Finding experts in community-based question-answering services. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 315–316, New York, NY, USA, 2005. ACM. ISBN 1-59593-140-6. 2.2.1, 2.2.2, 3.1, 4.6, 5.1.2

[59] X. Allan Lu and Robert B. Keefer. Query expansion/reduction and its impact on retrieval effectiveness. In *Proceedings of 3rd Text REtrieval Conference*, TREC '94, pages 231–240, Gaithersburgh, MD, USA, 1994. National Institute of Standards and Technology (NIST). 3.1

[60] Yao Lu, Xiaojun Quan, Xingliang Ni, Wenyin Liu, and Yinlong Xu. Latent Link Analysis for Expert Finding in User-Interactive Question Answering Services. In *Proceedings of the 5th International Conference on Semantics, Knowledge and Grid*. IEEE, 2009. 2.1.4

[61] Craig Macdonald and Iadh Ounis. 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*, CIKM '06, pages 387–396, New York, NY, USA, 2006.

ACM. ISBN 1-59593-433-2. 2.1.3, 2.1.4, 5.1.4, 5.3

[62] Chris Manning. *CS 276 Information Retrieval and Web Search: Lecture Notes (Electronic Tools)*. Stanford University, Palo Alto, CA, 2013. 4.1.1

[63] Don Metzler. *Beyond Bags of Words: Effectively Modeling Dependence and Features in Information Retrieval*. PhD thesis, University of Massachusetts Amherst, 2007. 6

[64] Donald Metzler and W. Bruce Croft. Combining the language model and inference network approaches to retrieval. *Information Processing and Management*, 40(5), 2004. 4.3

[65] Meredith Ringel Morris, Jaime Teevan, and Katrina Panovich. What do people ask their social networks, and why?: A survey study of status message Q&A behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1739–1748, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. 1

[66] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856. MIT Press, 2001. 2.2.2

[67] Lan Nie, Brian D. Davison, and Xiaoguang Qi. Topical link analysis for web search. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 91–98, New York, NY, USA, 2006. ACM. ISBN 1-59593-369-7. 2.2.2

[68] Michael G. Noll, Ching-man Au Yeung, Nicholas Gibbins, Christoph Meinel, and Nigel Shadbolt. Telling experts from spammers: Expertise ranking in folksonomies. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 612–619, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-483-6. 2.1.4

[69] Paul Ogilvie and Jamie Callan. Combining document representations for known-item search. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 143–150, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. 2.1.3

[70] Aditya Pal and Scott Counts. Identifying topical authorities in microblogs. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 45–54, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0493-1. 2.1.4

[71] Aditya Pal, Shuo Chang, and Joseph A. Konstan. Evolution of experts in question answering communities. In *Proceedings of the Sixth International Conference on Weblogs and Social Media*, ICWSM '12. The AAAI Press, 2012. 7.2

[72] Desislava Petkova and W. Bruce Croft. Hierarchical language models for expert finding in enterprise corpora. In *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, ICTAI '06, pages 599–608, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2728-0. 1.1

[73] Mingcheng Qu, Guang Qiu, Xiaofei He, Cheng Zhang, Hao Wu, Jiajun Bu, and Chun Chen. Probabilistic question recommendation for question answering communities. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 1229–1230, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4. 2.2.1, 5.1.2

[74] V. Smrithi Rekha, N. Divya, and P. Sivakumar Bagavathi. A hybrid auto-tagging system for StackOverflow forum questions. In *Proceedings of the 2014 International Conference on*

Interdisciplinary Advances in Applied Computing, ICONIAAC '14, pages 56:1–56:5, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2908-8. 5.1.6

[75] Fatemeh Riahi, Zainab Zolaktaf, Mahdi Shafiei, and Evangelos Milios. Finding expert users in community question answering. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 791–798, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1230-1. 2.2.2

[76] Mathew Richardson and Pedro Domingos. The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. In *Advances in Neural Information Processing Systems 14*. MIT Press, 2002. 6.3.2

[77] Avigit K. Saha, Ripon K. Saha, and Kevin A. Schneider. A discriminative model approach for suggesting tags automatically for StackOverflow questions. In *Proceedings of the 10th Working Conference on Mining Software Repositories*, MSR '13, pages 73–76, Piscataway, NJ, USA, 2013. IEEE Press. 5.1.6

[78] Nachiketa Sahoo. *Three Essays on Enterprise Information Systems Mining for Business Intelligence*. PhD thesis, Heinz College, Carnegie Mellon University, 2009. 4.1

[79] Nachiketa Sahoo and Ramayya Krishnan. Socio-temporal analysis of conversation themes in blogs by tensor factorization. In *Proceedings of the Eighteenth Annual Workshop on Information Technologies and Systems*, WITS '08, 2008. 4.1

[80] Nachiketa Sahoo, Ramayya Krishnan, and Jamie Callan. Sampling online social networks guided by node classification. In *Proceedings of the Fourth Symposium on Statistical Challenges in Electronic Commerce Research*, SCECR '08, 2008. 4.1

[81] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513 – 523, 1988. ISSN 0306-4573. 3.1

[82] Jose San Pedro and Alexandros Karatzoglou. Question recommendation for collaborative question answering systems with RankSLDA. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, pages 193–200, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2668-1. 2.2.2

[83] Jangwon Seo and W. Bruce Croft. Thread-based expert finding. In *In Proceedings of the ACM SIGIR Workshop on Search in Social Media*, SSM '09, 2009. 2.1.4

[84] Pavel Serdyukov, Henning Rode, and Djoerd Hiemstra. Modeling multi-step relevance propagation for expert finding. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 1133–1142, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. 2.1.4, 2.1.4, 2.1.4, 5.3

[85] Chirag Shah. Effectiveness and user satisfaction in Yahoo! Answers. *First Monday*, 16(2), 2011. URL `http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/3092/2769`. 1

[86] Craig Silverstein, Hannes Marais, Monika Henzinger, and Michael Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, September 1999. ISSN 0163-5840. 5.1.5

[87] Mark D. Smucker, James Allan, and Ben Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, CIKM '07, pages 623–632, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-803-9. 4.5

[88] Tom A. B. Snijders. Markov chain monte carlo estimation of exponential random graph models. *Journal of Social Structure*, 3, 2002. 2.1.4

[89] Tom A. B. Snijders. Models for longitudinal network data. In Peter J. Carrington, John Scott, and Stanley Wasserman, editors, *Models and Methods in Social Network Analysis*, pages 215–247. Cambridge University Press, 2005. ISBN 9780521809597. 2.1.4

[90] Ian Soboroff, Arjen P. de Vries, and Nick Craswell. Overview of the TREC 2006 Enterprise Track. In *Proceedings of the 15th Text REtrieval Conference*, TREC '06, Gaithersburgh, MD, USA, 2006. National Institute of Standards and Technology (NIST). 2.1

[91] Xiaodan Song, Ching-Yung Lin, Belle L. Tseng, and Ming-Ting Sun. Modeling and predicting personal information dissemination behavior. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, pages 479–488, New York, NY, USA, 2005. ACM. ISBN 1-59593-135-X. 2.1.4

[92] Xiaodan Song, Belle L. Tseng, Ching-Yung Lin, and Ming-Ting Sun. ExpertiseNet: Relational and evolutionary expert modeling. In *Proceedings of the 10th International Conference on User Modeling*, UM '05, pages 99–108, Berlin, Heidelberg, 2005. Springer-Verlag. ISBN 3-540-27885-0, 978-3-540-27885-6. 2.1.4

[93] Clayton Stanley and Michael D Byrne. Predicting tags for StackOverflow posts. In *Proceedings of the the 12th International Conference on Cognitive Modeling*, ICCM '13, 2013. 5.1.6

[94] Juyup Sung, Jae-Gil Lee, and Uichin Lee. Booming up the long tails: Discovering potentially contributive users in community-based question answering services. In *Proceedings of the Seventh International Conference on Weblogs and Social Media*, ICWSM '13. The AAAI Press, 2013. ISBN 978-1-57735-610-3. 3.3, 7.2, 7.5.1, 7.5.1, 7.5.1

[95] Evimaria Terzi Theodoros Lappas, Kun Liu. A survey of algorithms and systems for expert location in social networks. In Charu C. Aggarwal, editor, *Social Network Data Analytics*, pages 215–241. Springer US, 2011. ISBN 978-1-4419-8461-6. 2.1.4

[96] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. TwitterRank: Finding topic-sensitive influential twitterers. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-889-6. 2.1.4

[97] Baoguo Yang and Suresh Manandhar. Tag-based expert recommendation in community question answering. In *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '14, New York, NY, USA, 2014. ACM. 2.2.1

[98] Liu Yang, Minghui Qiu, Swapna Gottipati, Feida Zhu, Jing Jiang, Huiping Sun, and Zhong Chen. Cqarank: Jointly model topics and expertise in community question answering. In *Proceedings of the 22Nd ACM International Conference on Conference on Information &#38; Knowledge Management*, CIKM '13, pages 99–108, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2263-8. 2.2.1, 2.2.2, 2.2.2, 6.3.2, 6.3.2

[99] Reyyan Yeniterzi. Effective approaches to retrieving and using expertise in social media. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (Doctoral Consortium)*, SIGIR '13, pages 1150–1150, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2034-4. 1.3

[100] Reyyan Yeniterzi and Jamie Callan. Analyzing bias in CQA-based expert finding test sets.

In *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '14, pages 967–970, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2257-7. 1.3

[101] Reyyan Yeniterzi and Jamie Callan. Constructing effective and efficient topic-specific authority networks for expert finding in social media. In *Proceedings of the ACM SIGIR First International Workshop on Social Media Retrieval and Analysis*, SoMeRA '14, pages 45–50, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3022-0. 1.3

[102] Reyyan Yeniterzi and Jamie Callan. Moving from static to dynamic modeling of expertise for question routing in CQA sites. In *Proceedings of the Ninth International Conference on Web and Social Media*, ICWSM '15, pages 702–705. The AAAI Press, 2015. 1.3

[103] Jing Zhang, Jie Tang, and Juanzi Li. Expert finding in a social network. In Ramamohanarao Kotagiri, P.Radha Krishna, Mukesh Mohania, and Ekawit Nantajeewarawat, editors, *Advances in Databases: Concepts, Systems and Applications*, volume 4443 of *Lecture Notes in Computer Science*, pages 1066–1069. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-71702-7. 2.1.4

[104] Jun Zhang, Mark S. Ackerman, and Lada Adamic. Expertise networks in online communities: Structure and algorithms. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 221–230, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-654-7. 1.1, 2.1.4, 2.1.4, 2.2.2, 4.6.3, 6.1.1, 6.1.2, 6.4, 6.5.3, 6.5.3

[105] Min Zhang, Ruihua Song, Chuan Lin, Shaoping Ma, Zhe Jiang, Yijiang Jin, Yiqun Liu, and Le Zhao. Expansion-based technologies in finding relevant and new information: THU TREC 2002 Novelty Track experiments. In *Proceedings of the 11th Text REtrieval Conference*, TREC '02, Gaithersburgh, MD, USA, 2002. National Institute of Standards and Technology (NIST). 2.1.3

[106] Guangyou Zhou, Siwei Lai, Kang Liu, and Jun Zhao. Topic-sensitive probabilistic model for expert finding in question answer communities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 1662–1666, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1156-4. 2.2.2, 6.3.2, 6.3.2

[107] Guangyou Zhou, Kang Liu, and Jun Zhao. Joint relevance and answer quality learning for question routing in community QA. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 1492–1496, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1156-4. 2.2.2

[108] Guangyou Zhou, Kang Liu, and Jun Zhao. Topical authority identification in community question answering. In Cheng-Lin Liu, Changshui Zhang, and Liang Wang, editors, *Pattern Recognition*, volume 321 of *Communications in Computer and Information Science*, pages 622–629. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-33505-1. 2.2.2

[109] Tom Chao Zhou, Michael R. Lyu, and Irwin King. A classification-based approach to question routing in community question answering. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 783–790, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1230-1. 2.2.2

[110] Yanhong Zhou, Gao Cong, Bin Cui, Christian S. Jensen, and Junjie Yao. Routing questions to the right users in online communities. In *Proceedings of the 2009 IEEE International Conference on Data Engineering*, ICDE '09, pages 700–711, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3545-6. 2.2.2

[111] Zhi-Min Zhou, Man Lan, Zheng-Yu Niu, and Yue Lu. Exploiting user profile information for answer ranking in CQA. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 767–774, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1230-1. 2.2.2

[112] Hengshu Zhu, Huanhuan Cao, Hui Xiong, Enhong Chen, and Jilei Tian. Towards expert finding by leveraging relevant categories in authority ranking. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 2221–2224, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0717-8. 2.2.2

[113] Hengshu Zhu, Enhong Chen, Hui Xiong, Huanhuan Cao, and Jilei Tian. Ranking user authority with relevant knowledge categories for expert finding. *World Wide Web*, (5): 10811107, 2013. doi: 10.1007/s11280-013-0217-5. 2.2.2

[114] Jianhan Zhu, Dawei Song, Stefan M. Rüger, Marc Eisenstadt, and Enrico Motta. The Open University at TREC 2006 Enterprise Track Expert Search Task. In *Proceedings of the Thirteenth Text REtrieval Conference*, TREC '06, Gaithersburgh, MD, USA, 2006. National Institute of Standards and Technology (NIST). 2.1.1

A. Manual Assessments Collected from Company Employees for Expert Blogger Finding Task

A set of 86 information needs were created for test set. 60 of these were selected from the search queries which were extracted from access logs. These information needs were selected to mirror task-specific expert-seeking behavior. Some example information needs are "service oriented architecture", "performance engineering", "websphere process server", "oracle performance tuning", etc. 26 information needs, which were considerably more general, were created by the company employees. Some examples to these topics are "cloud computing", "presentation skills", "mainframe", "estimation", etc.

After creating the information needs, a sample-based approach was used to create the pool of candidate experts to be assessed. Top 10 candidates were chosen from various expert finding approaches and added to the pool. An information need has 26 candidates on the average. A deeper pool was desirable but an explicit goal was to produce a pool small enough to be assessed by an untrained assessor in 15 minutes.

We developed a user-friendly assessment tool which required login access to enable tracking of assessors. After the login, the assessors were given a list of information needs. The assessment system gave assessors the freedom of choosing any topic they want to assess. However, during assessments, the assessment system provided the number of assessors working on a topic to the assessors in order to encourage them to choose topics that had not been assessed yet.

After a topic was selected, the candidate experts of the selected topic were presented in random order in order to prevent any bias towards candidates. Additionally, when two different assessors assess the same topic, they also viewed candidates in different random orders. For each candidate-topic pair, the most relevant 3 documents written by the candidate were displayed to assessors in order to help them to evaluate the topic specific expertise of the candidates. The department and position of the candidate were also displayed on upper left part of the page to provide background information on the candidate. Assessors judged expertise on a 4-point scale (not expert, some expertise, an expert, very expert). After assessing all the candidates, the assessment tool gave assessors the ability to add experts that had not been suggested by the expert retrieval tool.

The task of creating the assessment data was divided into two phases. The preliminary phase was performed with 10 assessors over 20 information needs to provide a small amount of initial data about the effort needed to get reliable assessments. In this phase, assessment of a topic took 13 minutes on the average which met our goal of 15 minutes per topic. Among the assessed 20 topics, only 4 of them were assessed by multiple assessors which was not enough to analyze the agreement between assessors and their quality. In the second phase of the assessments more overlap between assessors were tried to be obtained.

The subsequent phase aimed to create a larger dataset required for reliable measurement therefore a half day assessment workshop was organized in the company with 15 volunteer employees. A total of 52 topics were assessed and 34 of them were assessed by multiple assessors. These 34 topics were used to calculate the inter-rater agreement measures with Cohen's Kappa [25] statistics. Since the assessment scores are in 4-point scale, weighted kappa statistics [24] were used to give credit for both complete and partial agreements.

The average inter-rater agreement value for all 15 assessors and 52 topics was 0.31. A detailed analysis on individual assessors revealed that some of them are not agreeing with most of the others. Therefore kappa value 0.30 was used as a threshold to discard these bad assessors. After removing 4 bad assessors the average kappa value increased to 0.38 for the remaining 11 assessors and the number of topics decreased from 52 to 50 which was an acceptable loss given the increase in the agreements.

After removing the bad assessors there remained 28 topics assessed by one assessor and 22 topics assessed by multiple assessors. Several selection and combination methods were applied to decide the assessment scores of these 22 topics assessed by multiple assessors. In case of combination, 3 methods were tried; taking the (1) average, (2) minimum or (3) maximum of the scores. Additionally a selection was performed by selecting the scores of the assessor with the highest average kappa value. All these methods gave highly correlated results; therefore we continued the experiments by selecting the assessor with the highest average kappa value.

In order to make sure of the assessment quality, several bias analyses were performed. These are checking whether there is any significant bias towards candidates that are in the same location or department with their assessors. Another analysis was to check whether the seniority or the length of blogging of the candidates has any direct correlation with their assessment scores. A correlation check was also performed with the presentation order of the candidates during assessments and their scores. In none of these analyses, a significant bias was observed therefore the following experiments were performed over the 50 assessed topics.

Since our contribution to expert blogger finding task is specifically on authority-based approaches, the experimental results of authority-based approaches are also presented in the following tables for comparison with the results from second assessment.

Table 1 and 2 present the results of applying different authority-based approaches over different weighted graphs[1] respectively for reading and commenting. Similar trends observed in Tables 6.3 and 6.4 are also observed in these tables. The proposed Topic Candidate graphs provided consistent and statistically significant improvements over other graphs.

Tables 3 and 4 present the experimental results of using initially estimated expertise during authority estimation. Compared to Tables 6.17 and 6.18, similar trends are observed. Using expertise in teleportation worked better than using it as influence in reading authority scores. For commenting, using initially estimated expertise of users as influence returned much higher scores compared to either not using it all, or using it during teleportation as discussed in Section 6.5.3.

[1] As observed from the previous experiments, weighted graphs are more effective for repetitive activities like reading and commenting.

		Levels of Expertise						
Algorithm	Graph	VE		+AE		+SE		NDCG
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	wPR	.0920	.1255	.1200	.0624	.1420	.0455	.2234
	wHITS	.1000	.1421	.1320	.0726	.1580	.0548	.2558
	wTC	.1120	$.1733_s$	$.1860^r_s$	$.1482^r_s$	$.2540^r_s$	$.1492^r_s$	$.4034^r_s$
TSPR	wPR	.1020	.1701	.1540	.1322	.1940	.1382	.4595
	wHITS	.1000	.1691	.1520	.1304	.1920	.1370	.4583
	wTC	$.1220^r_{s'}$	$.1910_s$	$.2020^r_s$	$.1722^r_s$	$.2780^r_s$	$.1774^r_s$	.4613
HITS	wPR	.0540	.0617	.0760	.0466	.0920	.0393	.1756
	wHITS	.0540	.0618	.0760	.0466	.0920	.0393	.1764
	wTC	$.1200^r_s$	$.1954^r_s$	$.2020^r_s$	$.1775^r_s$	$.2700^r_s$	$.1802^r_s$	$.4327^r_s$

Table 1: Expert ranking performance of weighted authority graphs constructed from reading activities in blog collection.

		Levels of Expertise						
Algorithm	Graph	VE		+AE		+SE		NDCG
		P@10	MAP	P@10	MAP	P@10	MAP	
PR	wPR	.0620	.0545	.0820	.0293	.0980	.0236	.1575
	wHITS	.0820	.0950	.1060	.0563	.1280	.0508	.2622
	wTC	$.1320^r_s$	$.2171^r_s$	$.1940^r_s$	$.1686^r_s$	$.2460^r_s$	$.1622^r_s$	$.4475^r_s$
TSPR	wPR	.1280	.1925	.1840	.1516	.2400	.1612	.4797
	wHITS	.1260	.1899	.1840	.1509	.2400	.1609	.4791
	wTC	$.1580^r_{s'}$	$.2644^r_{s'}$	$.2300^r_{s'}$	$.2089^r_s$	$.2940^r_s$	$.2042^r_s$	.4954
HITS	wPR	.0200	.0283	.0380	.0286	.0560	.0253	.1544
	wHITS	.0200	.0286	.0380	.0290	.0560	.0257	.1627
	wTC	$.1720^r_s$	$.3051^r_s$	$.2560^r_s$	$.2261^r_s$	$.3320^r_s$	$.2197^r_s$	$.5075^r_s$

Table 2: Expert ranking performance of weighted authority graphs constructed from commenting activities in blog collection.

Algorithm	Levels of Expertise						NDCG
	VE		AE		SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.1120	.1733	.1860	.1482	.2540	.1492	.4034
expTelPR	.1100	$.1812_{s'}$	.1800	$.1803^{r}_{s}$	.2680	$.2052^{r}_{s}$	$.4733^{r}_{s}$
expInfPR	.1120	.1841	.1860	$.1592^{r}_{s}$	.2620	$.1632^{r}_{s}$	$.4188^{r}_{s}$
TSPR	.1220	.1910	.2020	.1722	.2780	.1774	.4613
expTelTSPR	.1200	.1941	.2000	$.2084^{r}_{s}$	.2960	$.2338^{r}_{s}$	$.4963^{r}_{s}$
expInfTSPR	.1120	.1844	.1960	.1720	.2780	.1754	.4590
HITS	.1200	.1954	.2020	.1775	.2700	.1802	.4327
expInfHITS	.1120	.2163	.1960	.1752	.2920	.1809	.4424

Table 3: Expert ranking performance of using initially estimated expertise in authority estimation on weighted *Topic Candidate* graphs constructed from reading activities in blog collection.

Algorithm	Levels of Expertise						NDCG
	VE		AE		SE		
	P@10	MAP	P@10	MAP	P@10	MAP	
PR	.1320	.2171	.1940	.1686	.2460	.1622	.4475
expTelPR	.1060	.1636	.1800	.1768	.2440	$.1863^{r}$	.4568
expInfPR	$.1740^{r}_{s}$	$.2727^{r}_{s}$	$.3080^{r}_{s}$	$.2826^{r}_{s}$	$.4240^{r}_{s}$	$.2935^{r}_{s}$	$.5634^{r}_{s}$
TSPR	.1580	.2644	.2300	.2089	.2940	.2042	.4954
expTelTSPR	.1180	.1731	.1980	.2004	.2760	.2147	.4827
expInfTSPR	$.1820^{r}$	.2706	$.3300^{r}_{s}$	$.3037^{r}_{s}$	$.4700^{r}_{s}$	$.3262^{r}_{s}$	$.5857^{r}_{s}$
HITS	.1720	.3051	.2560	.2261	.3320	.2197	.5075
expInfHITS	.1360	.2401	.2240	.2294	.3160	.2290	.5189

Table 4: Expert ranking performance of using initially estimated expertise in authority estimation on weighted *Topic Candidate* graphs constructed from commenting activities in blog collection.