EXTRACTING USERS IN COMMUNITY QUESTION-ANSWERING IN PARTICULAR CONTEXTS

by

LONG T. LE

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

Written under the direction of

Professor Chirag Shah

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2017

ABSTRACT OF THE DISSERTATION

Extracting Users in Community Question-Answering in Particular Contexts

by Long T. Le

Dissertation Director: Professor Chirag Shah

Community Question-Answering (CQA) services, such as Yahoo! Answers, Stack Overflow and Brainly, have become important sources of seeking and sharing information. Online users use CQA to look for information and share knowledge on topics ranging from arts to travel. The questions posted on CQA sites often rely on the wisdom of the crowd, or the idea that the best answer could come from a culmination of several answers by different people with varying expertise and opinions. Given that CQA is a user-driven service, user experience becomes an important aspect, affecting the activeness and even the survival of the site. In this work, we are interested in studying the behavior of the users who participate in CQA. Specifically, we wish to understand how different types of users could be identified based on their behaviors on a CQA-specific problem at hand. A user's behavior depends on their particular context. For example, when we say that Alice is a "good user," the interpretation of her behavior actually rests on the context in which it occurs. She might be a good user in the whole community, a good user for a specific topic, a good user for a particular question or a good user for a particular answer. In this dissertation, we will study and extract users in different levels of granularity. Users are the main driving force in CQA and understanding them allows us to know the current state of their respective sites. The findings in this dissertation will be useful in identifying specific CQA user types.

Preface

Parts of the dissertation are based on previous works published by the author in [62], [61] and [63].

Acknowledgements

First, I want to express my gratitude to my research advisor, Professor Chirag Shah, for his advice, support and guidance. He helped me overcome difficulties and gave me critical comments. I thank him for his guidance in the face of unexpected obstacles. Without his help, I would not have finished this dissertation.

I want to thank my committee members, Professor Amélie Marian, Professor Thu D. Nguyen and Professor Panos Ipeirotis for their comments and suggestions on my work. Their time and input are much appreciated, and they truly improved my research. Special thanks go to Profesor Thu D. Nguyen, who served as my academic advisor and provided support and encouragement throughout my time at Rutgers. I thank Professor Ahmed Elgammal and Professor Eric Allender for being members of my qualifying exam committee. Professor William Steiger also provided a great deal of sound advice. I also want to thank Prof. Tina Eliassi-Rad for advising me in my early stage of my Ph.D. program.

I would like to thank my friends in the Department of Computer Science and the InfoSeeking Lab for creating an integral academic environment. Especially, I want to thank Erik Choi for a great collaboration. I want to thank Priya Govindan for discussing my work. I also want to thanks Diana Floegel for helping with some proof-reading.

I want to thank my parents, Vinh Le and Kien Nguyen, and my brother, Hoang Le, for their love and support. Last, but not least, I want to thank my wife, Lan Hoang, for her love, encouragement and support. She always stood by me on the long road to this dissertation.

Dedication

To my family.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Human beings have always sought information from one another. Over time, people invented books, libraries and mass media to more efficiently assimilate, store and share information. More recently, the Internet and the World Wide Web (WWW) have become critical and ubiquitous information tools that have changed the way people share and seek information. Many online resources on the WWW serve as some of the largest publicly available digital libraries. As the number of new resources for communication and information technologies have rapidly increased over the past few decades [64], users have adopted various types of online information sources such as web-search engines, Wikis, forums, blogs and community question-answering (CQA) services.

In the early 2000s, web-search engines such as Google and Yahoo Search revolutionized access to web-pages by presenting users with relevant resources based on their queries. Searchers often begin the process of information-seeking by submitting simple queries to search engines. However, results from search engines either do not give satisfactory answers for complex questions, or do not provide askers with needed personalized information. Web-search services only made it easier to access existing pages on the WWW. Though their creation was a breakthrough in information access, web-search engines still lacked the capability to answer more nuanced questions that required both knowledge and expertise.

Community Question-Answering (CQA) services, such as Yahoo! Answers and Stack Overflow, have become important sources for seeking and sharing information. Online users use CQA to look for information and share knowledge on topics ranging from arts to travel. The questions posted on CQA sites provide personalized answers that often rely on the *Wisdom of the Crowd*, or the idea that everyone knows something [115]. In other words, the best answer

could come from a culmination of several answers by different people with varying opinions and levels of expertise. A CQA's main goal is to provide good answers for newly posted questions. Users in CQA are a part of an online community. They can contribute to the community by asking questions, giving answers and voting for the highest quality posts.

CQA provides a complementary approach to Automatic Question-Answering (AQA) and search engines [49], [67] ,[16]. AQA provides an automatic way to answer questions through complex natural language processing. Thus, the user can quickly obtain answers in an AQA system. Unfortunately, AQA might not perform well in a complex query or diversity domain. As compared to AQA, CQA also provides users with a more personalized experience. CQA encourages collaboration and sharing of knowledge among users. Thus, CQA users can get better comprehensive answers, which are superior in quality of content and opinion when compared to those generated by AQA.

Information seeking is the human instinct that drives people to discover what's necessary to satisfy some goal. Since the development of the Internet, CQA provides a unique platform for users to seek and share knowledge. The CQA lies between a search engine and social network as described in Figure 1.1. Search engines collect information from the whole Internet, while finding information through social networks such as Facebook or Twitter draws upon "friend" connections. CQA attracts a broader audience than social media without search engines' impersonal shortcomings. In this dissertation, we study CQAs' users in light of their ability to facilitate information seeking and sharing in a supportive environment.

Type	Search Engines	CQA (Crowd-Sourcing)	Social Network
Method	Query	Question	Question
Source	Websites	Unknown People	Know People

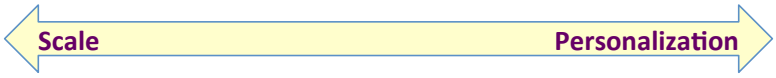


Figure 1.1: Comparing different information seeking approaches. There is trade-off between the scale level and personalization. CQA lies between the general search engines and social search.

Given that CQA is a user-driven service, user experience becomes important, as it affects the activeness and even the survival of the site. In this dissertation, we are interested in studying the behavior of users who participate in CQA as askers and answerers. Specifically, we wish to understand how different types of users could be identified based on their behaviors with respect to a CQA-specific problem at hand. The users' behaviors are evaluated in a particular context. The context might be *whole community*, *topic* or *question*. For example, a user might be good at answering a question in Java but bad at answering a question in Python. Thus, we are going to develop a framework to retrieve a user in a given context. Figure 1.2 describes the problem of finding correct users in a particular context. This is difficult to achieve because any given context has a different granularity; it might be the whole community, or a particular topic or even in a particular question.

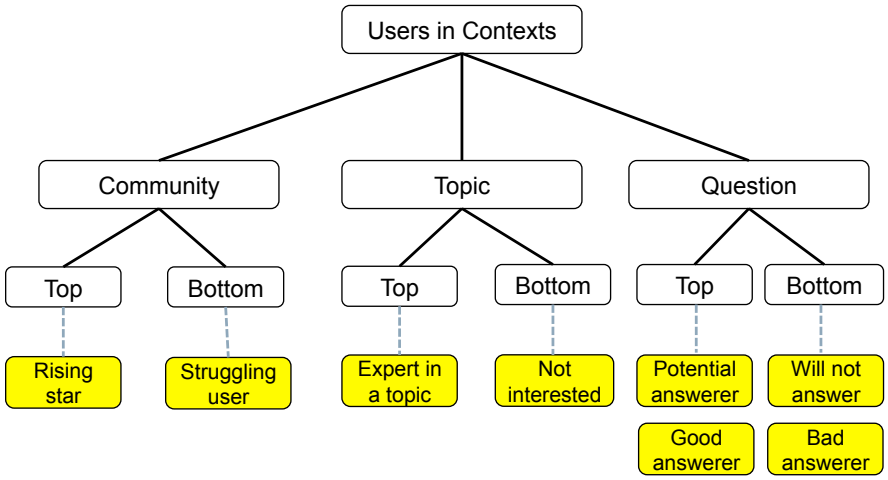


Figure 1.2: CQA users in particular contexts. The contexts can be very general such as the whole community or very specific such as a particular question/answer.

Since CQA is community-based, some special types of users include those who generally give answers, top contributors and those who provide good or bad content. Answerers are important because they provide the content that satisfies askers and also attract other online users. A small fraction of users contribute heavily in CQA; identifying these "rising star" users can help CQA community administrators adequately support them. If CQAs can somehow encourage a small percentage of low profile users to become more active, the return value will be significant. There are some other kinds of users in CQA such as churners and spammers,

but this dissertation will concentrate on the users classes in Figure 1.2. Such contributors represent different segments of content-providers within CQA services. In particular, rising stars represent top contributors, struggling users want to contribute to the community but fail to generate sufficient content and general answerers' contributions may vary.

Observing and detecting users in the early stages of their contributions is difficult but rewarding. Unlike social media participants, CQA users do not have rich social connections in which they must know their peers' information. Furthermore, CQA users do not disclose information about themselves via rich user profiles. Because CQA users are more elusive–and therefore more difficult to characterize–than social media users, detecting their potential in its early stages is very important. For example, Anderson et al. [6], [7] examined educational forums and found that incentivizing students produced increases in forum engagement. Furthermore, detecting early-stage struggling users is particularly important in the realm of education since educators can effectively intervene and assist if they know which students are struggling. Early detection of struggling users ensures that we have enough resources and time to intervene [59].

Struggling and excelling users are special cases within their respective communities, making their detection highly rewarding. These special users also get more attention in face-to-face situations. For example, the best and weakest students in the classroom should attract most of the attention from the instructor. Identifying different types of users is crucial since the users reflect the health of their communities. For example, if there are a lot of users giving bad answers or a disproportionate number of struggling users, the quality of the content will annoy the community and affect its sustainability.

1.2 Problem Definitions

In order to make it easy to follow, Table 1.1 lists the notation used in this dissertation. Let U be the set of users in CQA, Q be the set of questions, A be the set of answers, and the set of posts $P = A \cup Q$. The set I denotes the list of assessments between users such as voting for the best answer or deleting low quality contents.

The problems we address in this dissertation are:

- Looking for potential answerer: Finding the correct answerer could reduce an asker's

Symbol	Definition
$U = \{u_1, u_1, ..., u_n\}$	Set of users
$Q = \{q_1, q_2, ..., q_{m1}\}$	Set of questions
$A = \{a_1, a_2, ..., a_{m2}\}$	Set of answers
$I = \{i_1, i_2, ..., i_{m3}\}$	Set of assessments

Table 1.1: Notations used in the dissertation.

waiting time and also increase the chance that a question would be answered. We examined characteristics in two popular CQA sites (Stack Overflow and Yahoo! Answers) and proposed a method, QRec, to identify the potential answerers by constructing user profiles and activity levels.

Formal definition: Given a set of users $U = \{u_1, u_2, ..., u_n\}$, and list of posts $P = Q \cup A$,$Q = \{q_1, q_2, ..., q_{m1}\}$, $A = \{a_1, a_2, ..., a_{m2}\}$, let A_u be the list of questions answered by user u. For arbitrary $q \in Q$, find the set U_a st: $\{U_a \subset U$, $|U| = k$, and $Pr(\exists u \in U_a : u\ answers\ q)$ is maximized$\}$.

- Good and bad answerers in a particular question: Since most contents in CQA are generated by users who actively seek and share information with other users, the content quality is a critical factor to the success of the community. Therefore, assessing the quality of posts in CQA is a critical task in order to develop an information seeking environment where users receive reliable and helpful information for their educational information needs.

  Formal definition: Given a set of users $U = \{u_1, u_1, ..., u_n\}$, a set of posts $P = Q \cup A$, Q, and a set of interactions $I = \{i_1, i_2, ..., i_{m3}\}$ (such as giving thanks, making friends). For arbitrary answer $a \in A$, predict whether a will be $deleted$ or $approved$?

- Struggling users: we examine the proportion of active users who are struggling with the community. We consider struggling users to be those who have the majority of their answering contributions deleted.

  Formal definition: Given a set of users $U = \{u_1, u_2, ..., u_n\}$, and list of posts $P = \{p_1, p_2, ..., p_m\}$. Let P_u be the list of posts written by user u. Let StU, NoU, AsU be

struggling users and normal users . For arbitrary $u_i \in U$, find whether $u_i \in StU$ *or* $u_i \in NoU$?

- Finding rising stars: In CQA, there is typically a small fraction of users that provides high-quality posts and earns a very high reputation from the community. These top contributors are critical to the community since they drive the development of the site and attract the traffic from Internet users. Identifying these individuals could be highly valuable, but this is not an easy task. In this work, we attempt to perform this analysis by extracting different sets of users' features to predict their future contributions.

  Formal definition: Given a set of users $U = \{u_1, u_2, ..., u_n\}$, and list of first T posts of users $P = \{p_1, p_2, ..., p_m\}$. Let RSU, OU be the set of rising stars users, and ordinary users. For arbitrary $u_i \in U$, find whether $u_i \in RSU$?

In general, our research questions articulate the user behaviors in CQA. Users are the main driving force in CQA and understanding these users allows us to know the current state of their respective sites. For example, a lot of rising stars demonstrate a site's potential, while a lot of struggling users will hurt its community. The results from this research will help us understand users and their contributions in any online community where people are seeking, sharing and creating information. Furthermore, the findings can also help us to develop a framework to discover different types of users in different online communities. Furthermore, findings could likely help the CQA sites to improve users' experiences and, ultimately, to serve users better. We will apply different data mining approaches and the framework will leverage the Personal, Community, Textual and Contextual Features to extract the particular type of users.

1.3 Thesis Statement

Community Question-Answering (CQA) sites are becoming increasingly popular services where people find and exchange information. The primary driving force behind CQA, without a doubt, is the online community of users and visitors that embody diverse roles within each CQA site. It is, therefore, imperative that we understand these users and the behaviors that guide how they address various issues in CQA. However, unlike most social networks where users create

rich profiles and make explicit connections with other users in the community, CQA user profiles and inter-user connections may be non-existent or, at best, missing crucial information. In this dissertation, we will explore ways to construct models of CQA users in order to address problems that involve identifying specific set(s) of users. Examples of such users include users who could potentially answer a given question, participants who are going to be rising stars, and users who are likely to struggle with the community in the future. To address these questions, we will apply various machine learning and data mining techniques. The finding in this dissertation helps us identifying different types of users in online communities.

1.4 Thesis Organization

This dissertation proposes and evaluates the efficacy of a method for finding different types of users in CQA. The finding shows that it is possible to detect many types of users, especially in the early stage. The Chapter 2 discusses the background of this dissertation. In Chapter 3, we examine the possibility of finding the potential answerers when a new question is posted based on users' past history. Then, Chapter 4 examines the quality of answers and detects the users who fail to generate good quality content. Results showed that Community Features and Textual Features are more robust in detecting these activities. Chapter 5 studies the struggling CQA users who fail to generate good content. In contrast to struggling users, there is a small number of users who will contribute heavily in quality and quantity to the site. The Chapter 6 detects the rising stars in their early stage. Finally, Chapter 7 presents the conclusion and future works.

Chapter 2

Background

2.1 Online Q&A

Q&A services could be broadly classified as either online or face-to-face interactions, with traditional reference service in libraries being an example of the latter. While online Q&A usually refers to user-generated answers, there are examples of systems that do automatic extractions of answers, such as Ask.com. Within human-driven Q&A services, two types are prominent - vertical and horizontal. The former is an online Q&A service that is focused on a specific topic. Examples of vertical Q&A, also referred to as online forum, include StackOverflow [85] for programmers and PRIUSchat [94] for Toyota Prius owners. The four types of online Q&A fall under horizontal Q&A, which means they typically cover a broad range of topics instead of being organized around just one topic. To make this classification more precise, community-based, collaborative and social Q&A can be placed under peer-based services, separate from expert-based Q&A. Since the early 2000s, online CQA has become a popular source of information seeking. In fact, in the U.S., the number of visits to CQA services more than doubled after each year during the middle of 2000'.

2.2 Community Q&A (CQA)

How people share knowledge and communicate is an important research topic. In the last few decades, the Internet has changed the way people communicate and exchange ideas. CQA exemplifies this change in how people share their knowledge. In CQA, users post their questions in order to receive answers from anyone who can supply a correct answer or pertinent information. These platforms provide an unprecedented opportunity for users to enrich their minds and share knowledge. One popular way to ask a question online is through CQA, where users often

desire more personalized answers. CQA also takes advantage of *Wisdom of the Crowd*, or the idea that everyone knows something [115]. Users can contribute to the community by asking questions, giving answers, and voting for high-quality posts.

Several works have looked at user interest and motivation for participating in CQA [93, 77, 124, 90]. Adamic et al. [1] studied the impact of CQA. In [1], the authors analyzed questions and clustered them based on their contents. The results showed that many users only participate in a narrow topic area while some users can participate in a wide range of topics. Authors also showed that it was possible to predict the best answer by using basic features such as the length of answer and the history of answerers.

Shah et al. [104] reviewed the major challenges and research questions that must be addressed. CQA is designed to support users who want to ask and answer questions. After receiving the answers for a posted question, the asker may judge the quality of answers by themselves or let the community judge the answers. There are a lot of interesting problems within CQA such as what motivates users engaging in CQA, user expectations and why people spend time and effort to help others within CQA sites. Shah et al. [103] compare CQA and virtual reference to identify differences in users' expectations and perceptions. By understanding and identifying these behaviors, challenges, expectations and perceptions within the context of CQA, we can more accurately highlight potential strategies for matching question askers with question answerers.

Yahoo! Answers is a forerunner of CQA [131]. It is a general-purpose Q&A site, which accepts any question as long as it does not violate the site's guidelines. The site allows any of its users to post questions and answers. Besides general CQA such as Yahoo! Answers, different narrow topic CQA services were developed. Next, we discuss focused CQA and educational CQA.

Focused CQA: CQAs that have successfully supported Q&A that pertain to specific subjects such as programming, mathematics and music. Stack Overflow, for instance, is a popular site that attracts million of programmers. All questions in Stack Overflow must be related to programming. Within the site, users can demonstrate their expertise through knowledge sharing activities. In fact, many programmers are hired due to their high contribution to Stack Overflow [117]. Le et al. [61] showed that there is a small number of top contributors in Stack

Overflow and it is possible to detect these users in their early stages. Yao et al. [136] studied the long-term impact of site contents and demonstrated that many answers with lasting significance could be considered technical knowledge for many programmers.

CQA for Online Learning: In recent years, online learning has collapsed time and space [26], which allows users to access information and resources for educational purposes any time and from anywhere. As online learning grows in popularity, a variety of new online information sources have emerged and are utilized in order to satisfy users' educational information needs. For example, researchers have conducted empirical investigations of social media (e.g., Facebook, Twitter, etc.) in order to understand the effectiveness of higher education [119]. Khan Academy has become a popular online educational video site that has more than 200 million viewers as well as approximately 45 million unique monthly visitors [82]. Additionally, even though most CQAs are mainly focused on either general topics (e.g., Yahoo! Answers, WikiAnswers, and so on) and/or professional topics (e.g, Stack Overflow, etc.) to seek and share information, new CQAs have emerged to help students participate in question-answering interactions that share educational information for online learning. Some small-scale CQA tools were developed to support small groups of university students [8], [110]. Examples of large educational CQAs include Chegg [21], Piazza, [92], and Brainly [14]. Brainly specializes in online learning for students (i.e, middle school, high school) through asking and answering activities in 16 main school subjects (e.g., English, Mathematics, Biology, Physics, etc.) [23].

2.3 Q&A in Social Networks

Integrating question-answering into social network platforms is a new trend. Several recent works studied question-answering in social network configurations. People can easily pose questions to their friends on social networks due to the popularity of sites such as Facebook or Twitter. Morris et al. [77] investigated this kind of activity and found that users might prefer to ask questions in social networks due to response time and truthfulness. Panovich et al. [89] showed that strong tie-strength provides better answers in social networks. Wang et al. [125] studied Quora and demonstrated that the heterogeneity of its social network affects the quality of Quora's site. Fang et al. [37] integrated different social network signals to enhance

the efficacy of CQA. The work showed that different social network signals could improve the accuracy of finding the best answers. Such research is limited, however, due to the difficulty of collecting personal information from social networks such as Facebook or Twitter because of privacy concerns. Recently, some CQA sites such as Brainly introduced social media-like friendship features to encourage information exchange between users [23], [52], [62].

2.4 Searchers Become Askers: Understanding the Transition

Some previous works tried to understand users' transition from web searchers to CQA askers. Liu et al. [69] did a large scale study on this topic. The work studied different research questions including (i) When do searches turn to CQA? (ii) How do search queries relate to the associated questions posted on CQA sites? and (iii) How do searchers behave after transferring to the CQA site? The datasets used in this study include the *search-only* section and *search-ask* section. The question in *search-ask* is longer than the *search-only* question and contains more stop words. The *search-ask* queries are more likely to be unique. Interestingly, search engine result pages often return content from Yahoo! Answers. Researchers discovered that searchers are not as patient as regular askers. The asking action is often preceded by viewing CQA pages. Some works also use search to improve web question answering [2], [3]. Dror et al. [33] proposed a new algorithm to construct an asker's potential CQA question from a few keywords when they're struggling struggling to find the answer. Such built questions were shown to be meaningful and grammatically correct. Liu et al. [70] proposed a taxonomy of CQA answers and applied automatic techniques to summarize answers. Their study found that many questions might have multiple best answers.

Anderson et al. [4] studied a marked shift from Q&A to a community-driven knowledge creation process whose end product can be useful for a long period. The research used data from Stack Overflow. There are two main tasks in this study: predicting the question's long term value, and predicting when the question has been sufficiently answered. In [122], [43], the authors studied complex queries and found that traditional web search does not satisfy information needs. They developed a framework to find the queries with CQA intent. The findings in these works provided a foundation to integrate automatic web search and social

search.

2.5 Ranking in CQA

CQA has become an important source of information due to its scope and its ability to attract widespread participation. Many questions in Yahoo! Answers and Stack Overflow appear in other search engines, such as Google or Bing. Searching content in CQA sites is a useful task and recent research has proposed different models to rank information in CQA [11], [130], [126], [141]. Bian et al. [11] proposed a framework to rank posts in CQA based on both the structure and content of CQA archives. Wang et al. [126] assumed that answers are connected to questions via different latent links, and used these characteristics to rank community answers. Zhang et al. [139] used a network-based approach to develop a scalable ranking algorithm. Link-based ranking algorithms are also popular in CQA [53], [79], [138].

Due to the popularity of CQA in information retrieval, improving the search relevance in CQA is also an important topic [141], [129], [17]. Dror et al. [32] represented the user and the question as a vector of multi-channel features, which included social signals and required hundreds of features. By surveying different ranking methods and evaluating question and answer relevance, we can better understand how ranking can assist with CQA question and answer relationships. Gollapalli et al. [40] rank the users with a graph-based method, but it is not clear how to rank the answers based on the users. The neural network is also deployed to rank the content in CQA [51], [97]. Neural network shows the potential capability of analysing complex content generated by users, and it performs better than feature-based methods.

2.6 Question and Answer Quality in CQA

The quality of questions and answers in CQA is important since quality affects the users' experiences. The better the users' experiences, the more actively they participate in the site. Shah et al. [105] studied the quality of answers in CQA. In CQA, a question may receive multiple answers and the community selects which is the best. Li et al. [65], [9], [99] predict the quality of questions in CQA. Different types of features, especially textual features, are used to predict

CQA content quality [46], [11], [65]. In [100], authors showed that there is a high correlation between the quality of questions and answers. Thus, the users need to ask a good quality questions to receive high-quality answers.

Examining the quality of answers can be divided into three types of problems: *(i)* finding the best answer, *(ii)* ranking the answers, and *(iii)* measuring the quality of answers. For example, Shah and Pomerantz [105] used 13 different criteria to look for the best answers in Yahoo! Answers. Ranking answers is a useful task when a question receives multiples answers. These works focus more on the similarity between an answer and a question [114]. Surynato et al. [116] utilized the expertise of an asker and an answerer to rank the answers. In this work, the authors also recognized that different users are experts in different subjects and used this understanding to rank the answers. Recent work also showed the potential of using graphs to rank users [40], but it is not clear how to rank answers based on users.

The most popular type of problem focuses on regression-related problems, such as predicting how many answers a question will get or how much community interest a post can elicit. Researchers are interested in predicting whether certain questions in CQA will be answered and how many answers a question will receive [133, 34]. This research used features such as asker history, the question length and the question category to predict the question's answerability. Shah et al. [106] studied why some questions remain unanswered in CQA. Particularly, their work explored why fact-based questions often fail to attract an answer. Momeni et al. [76] applied machine learning to judge the quality of comments in online communities, revealing that social context is a useful feature. Yao et al. [136] examined the long-term effect of the posts in Stack Overflow by developing a new scalable regression model. Dalip et al. [29] tried to reduce the number of features in collaborative content, however, the number of reductions was not significant. Furthermore, applying feature selection can solve the issue with many features, such as over-fitting.

Developing the capability to detect an untruthful contribution is also an important task [91, 118]. Pelleg et al. [91] studied truthfulness in CQA sites. This research examined whether a user provides truthful information about themselves on CQA sites. They found that the askers generally provide accurate personal information, even when they post sensitive questions. Tan et al. [118] proposed a new method called CQAL to automatically predict the quality of a

post in a social knowledge base. In a social knowledge base, such as Wikipedia or Freebase, users can edit articles. These contributions might contain inaccurate information and reduce the user experience with social knowledge pages. Some signals that prove useful in making this determination are user contribution history, the features of each triple and user expertise. Liu et al. [68] predicted user satisfaction with CQA sites. Determining question and answer quality will ultimately prove essential when matching and analyzing questioning and answering behaviors in CQA.

Our work in Chapter 4 is close to measuring the quality of answers. This research uses past question-answering interactions and current question and/or answering activities in order to automatically predict the quality of new answers. The framework incorporates different groups of features including personal features, community features, textual features and contextual features.

2.7 Expert Finding

Finding the expert on a topic is useful in crowd-sourcing because experts can provide high quality content in a short period [87], [40]. In [127], [128], the authors proposed a method to find the expert in a small community with several thousands of users. Linked analysis is useful in finding the expert in online community [53]. Dror et al. [32] considered question recommendation as a binary classifier by extracting numerous features from questions and data. Zhang et al. [140] used a social network to identify experts by proposing a propagation algorithm. Yarosh et al. [137] studied the process of selecting an expert from a list of recommended users. Qu et al. [98] used the topics similarity to find potential answerers, which is not robust enough.

In our work, we show that using others' signals can improve the ability to find a potential expert. Recent works used external information to improve the capacity to find an expert. For example, Srba et al. [111] collected personal data from personal sites or social networks to improve the efficacy of recommending answerers, but this approach is not practical due to the privacy issue and the missed personal information in the CQA. Yan et al. [132] used 3-way tensor decomposition to recommend the answerers, but that method is not scalable for a large dataset with million of questions and users. Another approach is collaborative answering

[88] where, rather than find top individual answerers, the question is sent to a group of similar users who would collaborate to create high-value content. The approach proposed by Luo et al. [72] differs from the above-described methods in several aspects; for example, they studied commercial Q&A which is different from CQA. Their work considers user motivation and expertise, which is not available in public CQA. Furthermore, users possess different motivations when answering a question in a working environment as opposed to a public community.

2.8 User Behavior in Online Communities

An online community's success greatly depends on its users because community content and activities are primarily based on users' activities. Thus, user behavior attracts a great deal of research interest. A community's size–especially the ratio of answered to unanswered questions–is a large component of its overall effectiveness. [127, 128]. Dumais et al. [36] studied users' behaviors based their activity logs and suggested suitable methods to design the system to collect data. They also discussed the challenges when using the real logs of users' activities.

Users make diverse contributions in CQA. Le and Shah [61] showed that a small number of users contribute heavily to their community and are crucial to its health. This observation motivated these researchers to develop a framework in which they integrated various features to detect top contributors in their early stages of site use. Contrastingly, "the lurker" is prevalent in many online communities. Gong et al. [41] profiled lurkers in an online social network. Compared to other users, lurkers maintain far fewer social connections. The work found that popular global events will break lurkers' silence. It is not clear why these users are lurkers or why active users become lurkers.

The ability to detect an untruthful contribution is also an important task [91, 118]. Pelleg et al. [91] studied truthfulness in CQA sites. This research examined whether users provide truthful information about themselves on CQA sites, and found that askers generally provide accurate personal information, even when they post sensitive questions. Tan et al. [118] proposed a new method called CQAL to automatically predict the quality of a post in a social knowledge base. In a social knowledge base, such as Wikipedia or Freebase, users can edit articles. These contributions might contain inaccurate information and detract from other users'

experiences with these sites. Some signals that help determine truthfulness include user contribution history, the features of each subject and user expertise. Shingla and Krause [109] showed that users are concerned about their privacy when participating in an online community. The authors proposed a framework to maximize utility while preserving the users' privacy. Liu et al. [68] predicted users' satisfaction with CQA sites. Determining question and answer quality will ultimately prove essential when matching and analyzing questioning and answering behaviors in CQA.

A small number of users does not behave properly in online communities. These users might use hate speech [81], [75]. In current online communities, crowd-sourcing platforms rely on community members to flag such users. Kayes et al. [56], [55] show that crowd-sourcing monitoring works quite accurately in detecting these actions but the the system relies too much on human assessment. A supervised classification that uses different linguistic features such as length and politeness performs moderately well. Studies have also examined the controversial conversations that take place on social media [39], [28], [27]. These works also use the graph structure of social interaction to detect the controversial conversation.

All communities want to attract new members, but existing members who leave a community are a prevalent issue. Understanding when and why users leave a particular community provides an overview of the community's health [84], [80], [101]. These works used user behavior *ego-nets* to make behavior and activity pattern predictions. Pal et al. [86] provided an overview of user evolution in CQA, which included various users' in-site activities and their effects. Antisocial users also indicate community health. Cheng et al. [22] examined antisocial behavior in online posting. Research shows that it is significantly different from other kinds of behavior. For example, antisocial users compose their posts differently from others, or display a more negative sentiment. Deleting these users forms a bimodal distribution which is very high or very low. User experience with online activity also attracts interest from the research community [83], [58]. Scholars showed that user engagement is complex. Furthermore, click patterns, dwell times and keyboard actions correlate with user engagement. Mao et al. [74] predicted users' engagement with volunteer crowd-sourcing. The study showed that the average number of tasks was the strongest predictor of future engagement with online crowd-sourcing.

2.8.1 User Engagement with Online Communities

Gong et al. [41] profiled lurkers; they examined who lurkers are and what they think. Lurkers maintain fewer social connections than other users. The work found that popular global events will break lurkers' silence, but could not determine why lurkers behave as they do nor why active users sometimes become lurkers. The work also did not define different types of lurkers. Cheng et al. [22] examined antisocial behavior in online posting. Their research demonstrated significant differences between antisocial and typical behaviors. For example, antisocial posts contain different wording and more negative sentiment than other content. The deletion of these users forms a bimodal distribution. Cheng et al. [22] showed that it is possible to detect antisocial users after only a few posts based on features such as post content, community content and moderator action. Sun et al. [112] reviewed lurkers' behavior in their communities, and found that individual, commitment, quality and online community factors may all contribute. There are several reasons lurkers may appear in online communities, including environmental, personal, relational and security factors. Understanding these behaviors helps administrators promote healthy user activity. Shoji et al. [107] studied the life span of users in CQA. Their analysis revealed that long-lasting askers tend to seek information across a wide topical range, while long-lasting answerers tend to answer within a narrower range of topics.

Brien et al. [83] studied user engagement in a reading environment and found that a complex correlation exists between students' comprehension and their degree of engagement. Thomas et al. [120] also showed that users' engagement with online forms is complex. They claim that click patterns, dwell times are keyboard actions are correlated with the user engagement. Engagement is very important in online education, and thus has attracted a great deal of recent research [19], [71], [48]. Anderson et al. [7] studied user behavior within online courses. This work classified users into three groups: users who only watch videos, users who only try to complete the assignment and users who balance between these activities. They also examined how badges may affect online students' productivity. Badges were shown to be a powerful tool for encouraging engagement. Qiu et al. [96] analyzed the key factors that affect users' engagement in massive open online course (MOOCs). They observed that female students are more likely to ask questions in a non-science course forum, bachelors ask more questions, and

both forum activities and effective learning are significant predictors of users' achievement in a course. The work also proposed a latent dynamic factor graph model to predict students' success in online courses. Halavais et al. [44] examined the effect of general badges in CQA. Their results demonstrated various badges' respective social influences. For example, general badges are related to the life of the site while tags badges are affected by social factors. Kayes et al. [55] investigate the cultures of users in different countries based on CQA. The work shows that natural cultures differ in multiple aspects including as temporal, privacy and individual factors. In particular, users from countries with faster-paced environments exhibit clearer temporal actions in CQA. Futhermore, users contribute more if they are from individualistic countries such as western countries, but such uses are also more concerned about privacy. Unfortunately, this work does not consider whether the technology available to these countries might be a hidden factor.

2.8.2 User Similarity in Social Media

Anderson et al. [5] lists the characteristics of similar users in social media. Two aspects of similarities are considered: similarity of interest and similarity of social ties. The work shows that it is possible to predict voting outcomes based on the similarities of users who show up to provide evaluations. Many underlying phenomena are affected by users' similarities. Han et al. [45] determined the interest of the users in social media. They found that there is homophily in users' interests such as demographic information or friends. Based on that observation, it is possible to predict similar users' potential interests. The study was limited, however, when evaluating popular interests such as favorite movies or sports. A more detailed study of various types of interests–such as health or politics–will help us better understand the effects of online social influences. Many underlying phenomena are affected by users similarities.

2.8.3 Evolution of Users in CQA

Pal et al. [86] studied the temporal and dynamics activities of CQA users. There are a few types of community experts: some experts are consistently active, some experts are initially active but then passive, and some experts are initially passive but later active contributors. The temporal features show strong prediction value when classifying these experts. In [30], the

authors examined the change in linguistics in forums. This work found that all members died "old". The work built a framework to track users' linguistic changes during their lifetimes. The life of a user can be summarized in cycles: the early cycle is the period during which a user is receptive to the community–this makes-up about about one third of their lifespan; from this point, the language gap between the user and their community increases until they leave. As the community's language evolves, users who are unreceptive to such changes will leave. This work may be limited in that it does not consider CQA's information gain purpose. Users who join a community want to learn about something, and they might leave once they are satisfied with their knowledge. In another scenario, a user may leave after a change in lifestyle; for example, obtaining an new job may make them too busy for CQA participating.

Kairam et al. [54] investigated two types of social community growth: growth through diffusion and growth through other means. Diffusion growth is defined as the addition of new members who have existing social ties with current members, while non-diffusion growth comes from new members with no such relationships. The work extracts three groups of features including growth, connectivity and structure to predict community longevity. The results showed that past growth can predict short term success while the network structure can predict long term sustainability. For example, the larger the largest clique, the faster the community will expand over two years. The work does not study the interactions between groups in the same category, such as occurrences in which members from one group switch to another similar group.

Zhu et al. [142] predicted users' activity levels in online social networks. Their algorithms include three main components: personalization, dynamic modelling and social regularization. Personalization means that different users exhibit different behaviors. Dynamic modelling captures the decay of behavior by penalizing the weight of past behaviors. Finally, social regularization considers the social influence between users. Researchers then solved the optimization framework based on these three factors to predict users' social network activity. Pudipeddi et al. [95] studied the churn behaviour in a popular CQA site. Given the first T posts or the posts in the first T weeks, the work predicted whether the users would leave the community. The work demonstrated that temporal features are most useful in predicting churn behavior. This

study holds particular relevance because detecting churn in CQA is more challenging than doing so in social networks, as the former lacks user interactions such as friendship. Oentaryo et al. [84] studied the churn behaviour in social networks by using collective classification. In conventional classification, entities are considered separately, such as independent and identically distributed (i.i.d). In social networks, there are different dependencies between instances. Collective classification considers the dependencies among the entities. Dror et al. [35] studied new users who were going to leave the community. The work extracted answer features, question features, gratification features and demographics features. Random forest and SVM outperformed other methods while the number of answers demonstrated the highest predictive power. The work, however, did not consider all users in the community when determining who would leave. Richter et al [101] quantified the user churn in mobile networks. The existing method treated each user as a single entity (i.i.d) and used a lot of key performance indicators (KPI) to measure user satisfaction. Such an approach, however, may not be timely enough to re-attract users. The framework quantified social connections, partitioned the network into connected components and trained classification based on each user cluster's KPI.

It is clear from this brief literature review that a great deal of CQA sites' success and sustainability depends on content quality, community engagement and user satisfaction. In general, these factors hinge on the users who participate in the communities. Our work specifically concentrates on how user behaviors can increase site quality. The research presented here is based on a large scale study of popular CQA sites–Yahoo! Answers, Stack Overflow and Brainly. We hope to build upon preexisting work by analyzing methods and proposing strategies that help detect different types of users.

Chapter 3

Finding Potential Answerers in CQA

3.1 Motivation and Problem Definition

In this Chapter, we discuss how to find the right answerers in CQA. Although responses on CQA sites are obviously slower than information retrieved by a search engine, one of the most frustrating aspects of CQAs occurs when an asker's posted question does not receive a reasonable answer or remains unanswered. CQA sites could improve users' experience by identifying potential answerers and routing appropriate questions to them. Finding potential answerers increases the chance that a question is answered or answered quickly. In this Chapter, we predict the potential answerers based on question content and user profiles. Our approach builds user profiles based on past activity. When a new question is posted, the proposed method computes scores between the question and all user profiles to find the potential answerers. We conduct extensive experimental evaluations on two popular CQA sites–Yahoo! Answers and Stack Overflow–to show the effectiveness of our algorithm. These experiments are based on nearly 2.3 million questions from Yahoo! Answers and more than 1.3 million questions from Stack Overflow. The results show that our technique is able to predict a small group of 1000 users from which at least one user will answer the question with a probability higher than 50% in both CQA sites. Further analysis indicates that topic interest and activity level can improve the accuracy of our approach. In essence, we provide evidence that our approach leads to a new framework for retrieving specific types of people in CQA communities.

This is an important problem because recommending possible answerers could reduce an asker's wait time or increase the likelihood that their question is answered. Finding the potential answerers, however, is a difficult problem due to the diversity of the users and content in CQA. Next, we will describe our approach to solve the problem.

Problem Definition

The problem is concerned with identifying potential answerers within a CQA community when a new question is posted to that CQA service. Given new questions, we want to find the top-K potential users who will be willing to give corresponding answers. Below is the formal definition:

Formal definition:

Given:

- a set of users $U = \{u_1, u_1, ..., u_n\}$
- a set of posts $P = Q \cup A$,
  Q is the set of questions $Q = \{q_1, q_2, ..., q_{m1}\}$, and
  A is the set of answers $A = \{a_1, a_2, ..., a_{m2}\}$
- A_u be the list of questions answered by user u

Task: For arbitrary $q \in Q$, find the set U_a st: $\{U_a \subset U$, $|U| = k$, and $Pr(\exists u \in U_a : u\ answered\ q)$ is maximized$\}$.

3.2 Method

We propose a framework to predict answerers based on a posting's history and features. The first step is building a user profile based on a user's past activities. Then, given a new question q, we compute the score between q and all user profiles. A higher score indicates a higher chance that a user will answer a certain question. Our method includes the following features:

- Similarity between question content and user profile
- Similarity between question topics and user expertise topics
- Similarity between asker and answerer in information network
- User's activity level

3.2.1 Constructing a User Profile

We can implicitly or explicitly build a user profile. Online users might provide short descriptions about themselves, such as *"Software developer who spent some time in the C++ world but now lives in Eclipse developing java apps"*. From their self-declared profile, we know that this user has expertise in C++, Java and Eclipse. Unfortunately, explicitly constructed user profiles have two limitations. First, many users do not have self-declared profiles, or if they do, their description may not be complete. Second, many users do not consistently update their profile with current information. In [134], authors build the user profile based on user opinion, such as likes/dislikes in online reviews. Implicitly inferring the user profile is a better method. We do this based on the list of questions a user has answered. A user's profile is the concatenation of all the questions they have answered. Figure 3.1 gives an example of user profile.

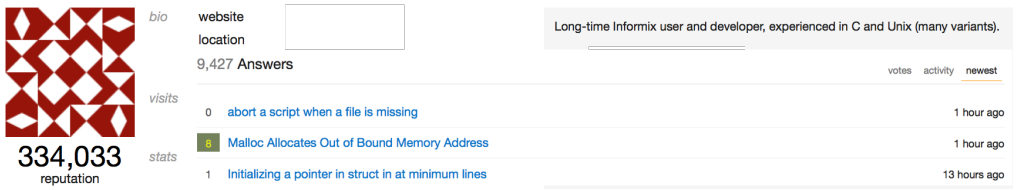


Figure 3.1: An example of user profile on Stack Overflow. User profile for the work reported here is constructed from self introduction or inferred from list of questions answered.

3.2.2 Computing Similarity Between Question and User Profile

Given all user profiles and a new question q, we measure the similarity between q and all user profiles. In order to measure the similarity, we treat each user profile or question as a document. A corpus of documents is built from all user profiles and questions. The next step is computing the *tf-idf* of each document in this corpus [73]. The *tf-idf* value of a word increases linearly with the number of times that word appears in a document, but decreases by the frequency of the word in the corpus. After computing *tf-idf*, each document d is represented by a vector $\vec{\mathbf{v}_\mathbf{d}}$. Similarity between a user and a question is measured by the *cosine similarity* between corresponding vectors. The cosine similarity between vectors $\vec{\mathbf{a}}$ and $\vec{\mathbf{b}}$ is described in Equation 3.2. Cosine similarity is used widely due to its simplicity and efficiency. The cosine similarity

in the formula is between 0 and 1 since the *tf-idf* only returns non-negative values. The value of 1 indicates an exact match while 0 indicates no relevance.

$$tf\_idf_{t,d} \;=\; tf_{t,d} \times idf_t \;=\; tf_{t,d} \times log(\frac{N}{df_t}) \tag{3.1}$$

$$\cos(\vec{\mathbf{a}}, \vec{\mathbf{b}}) = \frac{\vec{\mathbf{a}}\vec{\mathbf{b}}}{\|\vec{\mathbf{a}}\|\|\vec{\mathbf{b}}\|} = \frac{\sum_{i=1}^{n} \mathbf{a}_i \mathbf{b}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{a}_i)^2}\sqrt{\sum_{i=1}^{n} (\mathbf{b}_i)^2}} \tag{3.2}$$

3.2.3 Inferring Users' Topic Interests

Many users do not specify their topics of expertise. However, we can explicitly infer users' respective expertise topics. This idea is similar to inferring a user profile. Our framework collects the topics of all questions answered by a particular user u. User expertise topics of user u is the concatenation of topics of all questions they answered. By using the content that a user has answered, we can use topic modeling techniques such as *latent Dirichlet allocation (LDA)* to find the topics of each question [13]. LDA is a generative model. Each document in LDA is considered a mixture of different topics. Figure 3.2 describes the generative process of LDA. The only observed variable is *W*; the rest are latent variables. The process of generating the topics for each document is as follows: *(i)* Choose α and β as the parameters of the Dirichlet prior to the per document topic distributions and per topic word distribution respectively *(ii)* Choose a topic from θ distribution *(iii)* Pick up a word w from multinomial distribution. Repeat this process for all documents.

In order to match the users' topics of interest with question topics, we apply *tf-idf* again. We treat each topic inferred by LDA as a term. Each user or question is represented as a document in which the document's "terms" are its topics. Then, we compute the *tf-idf* of each document in this corpus. The last step is to compute the topic similarity between a user and a question by applying cosine similarity.

3.2.4 Similarity in Information Network

In popular CQAs such as Stack Overflow and Yahoo! Answers, user friendships are not as explicit as those found on social networking sites. We construct the graph $G = (V, E)$. The list

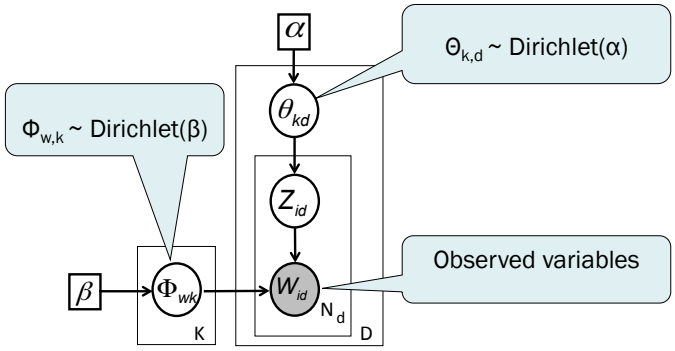


Figure 3.2: Plate notation of finding topics of documents in LDA.

of nodes, $V = \{u_1, u_2, ..., u_n\}$, is the set of users in the community. There is an edge $e \in E$ between u_i and u_j if user u_j answered a question posted by u_i. A popular method to measure the closeness between two nodes is using Random Walk with Restart (RWR). RWR provides a good relevance measure of two nodes in a graph. There are two main components in RWR starting at a *seed node* i: (a) a random walk to a neighbor is performed, and (b) at any step, there is a small probability c of jumping back to the seed node. Given a seed node i in a graph, the relevance between node i and j is computed as Equation 3.3, where $\vec{d}$ is starting vector. The relevance score is the j^{th} element of vector $\vec{r_i}$.

$$\vec{r_i} = (1 - c)G\vec{r_i} + c\vec{d} \tag{3.3}$$

A traditional method to compute the RWR is the power iteration method, which repeats Equation 3.3 until convergence. Due to the large size of our graph, we applied fast RWR, as proposed in [121]. The basic idea of fast RWR is partitioning the graph in smaller communities. These communities connect to each other through the bridge edges. Then, the RWR score can be combined based on the small communities and bridges. Ganu et al. [38] examined the measure of the similarity between users in a small forum by using continuous posts in a thread.

3.2.5 Activity Level of User

Active users are more likely to answer new questions. Section 3.3 also shows that a small fraction of users contribute most of the content in CQA. Users in Yahoo! Answers are awarded points when answering questions. Stack Overflow users also earn reputation value by answering questions. In general, users who give a greater number of answers can earn a higher reputation in Stack Overflow or higher scores in Yahoo! Answers. The majority of CQA sites have some metrics that measure the user activity level. In cases where there is no such metric, we can use the number of posts as the user activity level.

3.2.6 Summary of Our Framework to Find Potential Answerers

First, we explain how to compute the score between a user and a question. Given a question q asked by asker a and given an arbitrary user u in the community, the score between u and q is calculated, as in Equation 3.4. The score is the product between the similarity and the $log(activity\_level)$. We use the log of activity level due to the power law distribution of user activity level. In CQA, the user's activity level follows the power law distribution (shown in Section 3.3).

$$score(u,q) = log(AL(u)) \times [\alpha_0 + \alpha_1 * sim\_C(u,q) + \alpha_2 * sim\_T(u,q) + \alpha_3 * sim\_U(u,akr)] \quad (3.4)$$

where:

- $AL(u)$ is the activity level of user u
- $sim\_C(u,q)$ is the similarity between question content and user profile
- $sim\_T(u,q)$ is the similarity between question topics and user expertise topics
- $sim\_U(u,akr)$ is the similarity between user u and asker akr as in equation 3.3
- α: parameter controls the different weighting of these similarity values.

In order to find the parameter α, we used answerers' past answering history. During the observation period, we calculated the real score of each pair of users and questions. If user u is the n^{th} person who answers a question, the real score $y = 1/n$. Otherwise, we assign score 0 if this person does not answer said question. In general, the sooner a user answers a question, the higher the score value they should receive. Thus, the parameter α can be inferred as:

$$\alpha^* = \arg\min_{\alpha} \sum_{i=1}^{m} [y_i - log(AL) \sum_{i=0}^{3} \alpha_i \times sim_i] \tag{3.5}$$

where y_i is the actual output of answering activity in the observation period, m is the number of trainings and sim_i is the similarity value including similarity of content, topics and information network. The problem in Equation 3.5 can be converted to a standard linear regression problem by dividing by $log(AL)$. Then, we can easily calculate the parameter α_i.

Next, we explain our algorithm that finds potential answerers. Algorithm 1 describes our *QRec* algorithm. The first step of our algorithm constructs the user profiles. Given a new question q, *QRec* calculates the scores between q and each user. The list of potential answerers is comprised of users who have top scores. We need to build a user's profile once (Line 2 in Algorithm 1), and apply it to multiple questions. In a real application, the user might change their interest. In such a case, we can update their profile, but this is only needed after a long period (i.e, after a few months). Steps 9 and 10 are expensive, but can be computed off-line. Furthermore, we applied Fast RWR for this large information network . Another issue is finding the topics for new questions. In Step 6 of Algorithm 1, we consider each question and user profile as a document. When a new question appears, online LDA [50] can be applied to quickly find the topic of the new question without training the whole corpus again.Thus, our ranking method is scalable and can be easily applied to large datasets.

3.3 Datasets and Characterization of the Data

3.3.1 Data Description

We used data from two popular CQA sites: Yahoo! Answers and Stack Overflow. Yahoo! Answers is a general purpose CQA site while Stack Overflow is a focused CQA that hosts programming-related questions.

Algorithm 1 *QRec* algorithm

Input:

- A set of $users_i$, i =1, ..., n.
- A question q
- Size of possible answerers k

Output: The list of k users most likely to answer q

1: **for** $i = 1 : n$ **do**
2: Construct the user profile based on self-declared profile and answering history
3: **end for**

//Compute the *tf-idf* of raw content

4: Construct the RAW content corpus, each document is a user profile or a question
5: Compute the *tf-idf* of each document in RAW content corpus

//Compute the *tf-idf* of hidden topics

6: Infer the topics of interest expressed by each user profile and the hidden topics of the question q by applying LDA.
7: Construct the TOPICS corpus, where each term represents a hidden topic. Each document represents the topic interests of a user or hidden topics of a question.
8: Compute the *tf-idf* of each document in the TOPICS corpus

//Construct the information network and compute user similarity

9: $G = (V, E)$. The list of nodes: $V = \{u_1, u_2, ..., u_n\}$, are the set of users in the community. $\exists\, e \in E$ between u_i and u_j if user u_j answered a question posted by u_i
10: Compute $sim\_U(u_i, u_j$ as Equation 3.3)

//Compute scores between each users and question

11: **for** $i = 1 : n$ **do**
12: Compute scores between $user_i$ and q (as in Equation 3.4)
13: **end for**
14: **return** $topK$: list of k users with top scores

Yahoo! Answers ("answers.yahoo.com") is a forerunner of CQA. It is a general-purpose Q&A site, which accepts any question as long as it does not violate the site's guidelines. The site allows any of its users to post questions and answers. Each question in Yahoo! Answers is assigned to a particular category. A user in Yahoo! Answers might have many different types of interaction on the site, such as sharing, discussion, advice and polling [1]. In order to encourage user participation, the site awards points to users. The points determine the a user's level; the levels range from 1 to 7.

Stack Overflow ("stackoverflow.com") is another CQA specifically focused on the programming field. Stack Overflow only accepts questions and answers related to programming. Similar to Yahoo! Answers, users in Stack Overflow can engage in a wide range of activities that include upvoting and downvoting posts, or offering a bounty to a question to attract an answerer. Users in Stack Overflow earn reputations by providing high quality questions and answers. For example, a user's reputation increases if their question/answer is upvoted or their answer is accepted. In contrast, a user's reputation is diminished when their question/answer is downvoted or marked as spam. Each question is assigned tags based on its content. The tags can be considered the question topic. Since Stack Overflow is a focused site, the questions are normally difficult. Stack Overflow's administrator and community carefully manage its content. For example, duplicate questions or non-useful questions will be merged or removed to maintain the site's high quality.

Table 3.1 lists the types of actions and how they affected a user's score. Stack Overflow uses the term "reputation," while Yahoo! Answers uses the term "point." In general, Stack Overflow has a stricter policy to maintain high quality posts, while Yahoo! Answers focuses on increasing the amount of time users spend within the site.

Table 3.2 describes some statistics of the dataset used in our experiment. We crawled the questions and answers in Yahoo! Answers while the data dump of Stack Overflow is available publicly[1].

[1] `https://archive.org/details/stackexchange`

Table 3.1: Score system in Stack Overflow and Yahoo! Answer.

Stack Overflow	**Change in reputation**	**# Yahoo! Answers**	**Change in point**
Answer is upvoted	+10	Join Yahoo! Answers	+100 (one time)
Question is upvoted	+5	Ask question	-5
Answer is accepted	+15	Choose best answer	+3
Answer is downvoted	-2	Answer a question	+2
Question is downvoted	-2	Log in Yahoo! Answers	1
Answer win bounty	+bounty amount	Receive thumbs-up	+1
Offer bounty	-bounty amount	Receive a violation	-10
Answer marked as spam	-100		

Table 3.2: Description about data.

Site	**Period**	**# of Users**	**# of Questions**	**# of Answers**
Yahoo! Answers	Jan ’08 to Dec ’09	1.07 M	2.24 M	11.71 M
Stack Oveflow	Jan ’14 to Sep ’14	3.4 M	1.3 M	3.68 M

3.3.2 Characterization of the Data

In this section, we describe some of the CQA characteristics that help us gain a better understanding of and rationale to justify our method. For example, when calculating the score between a user profile and a question, we use the log instead of the real value of activity level. This decision is based on the distribution of user activity levels.

User Activity Level

Figure 3.3 plots the distribution of several answers given by users. We see that the distribution follows the power law distribution with heavy tail. Many users in CQA only answer a few questions, while a small number of users are very active. In both CQA sites, a small fraction of users contributes the a majority of the content. Since the distribution follows the power law, our formula to calculate the score uses the log of activity level.

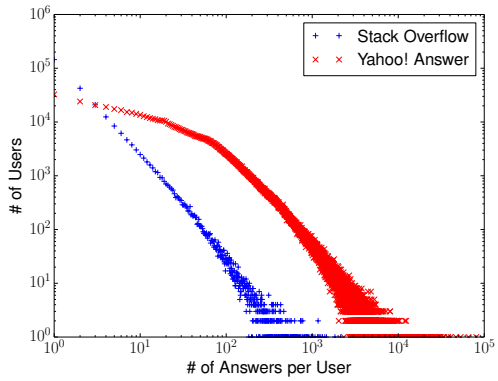


Figure 3.3: Distribution of number of answers given by users. There is a small percentage of users who are very active, while many users only give a few answers. A small fraction of users contribute the majority of a CQA site's contents.

The Length of Question

Figure 3.4 plots the distribution of the number of words per question. Questions in Stack Overflow are normally longer than questions in Yahoo! Answers. Stack Overflow is a focused CQA site and the questions are carefully managed. Unnecessary or meaningless questions are

deleted to maintain the site's high quality. But the length of questions in CQA sites is normally short. For example, half of the questions in Yahoo! Answers and Stack Overflow are less than 47 and 160 words, respectively.

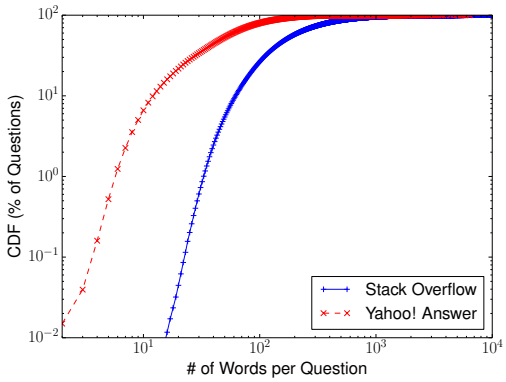
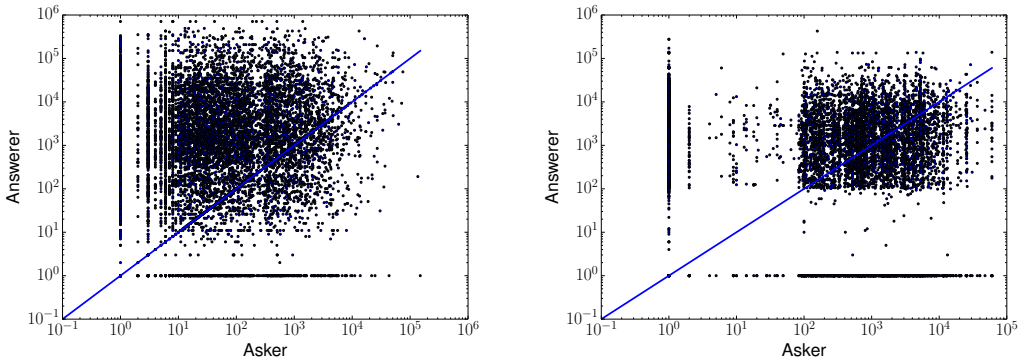


Figure 3.4: Distribution of number of words per question. Many questions in CQA are short. Questions in Stack Overflow are normally longer than questions in Yahoo! Answers.

Reputation of Asker and Answerer

We also want to compare the reputation of an asker with the reputation of an answerer. Figure 3.5 shows the reputation of an asker vs. the reputation of an answerer. The line $x = y$ is plotted by the blue line. The points above the blue line indicate that an answerer's reputation is higher than an asker's reputation. This Figure shows that the answerer often has a higher reputation than than askers in Stack Overflow. The observation makes sense because Stack Overflow is a focused CQA. Questions in Stack Overflow are generally difficult and require specialized knowledge. In contrast, the questions in Yahoo! Answers are usually general and do not require specialized knowledge. For example, many questions in Yahoo! Answers are polling or opinion-based, which anyone can answer. Furthermore, Stack Overflow penalizes users for giving poor answers. Thus, users in Stack Overflow only provide answers when they are confident about their accuracy. In contrast, users in Yahoo! Answers can earn points whenever they supply an answer, and are subsequently encouraged to provide answers whenever they can.



(a) Stack Overflow

(b) Yahoo! Answers

Figure 3.5: Asker's reputation vs. Answerer's reputation. Points above the cross line (in blue) indicate that an answerer's reputation is higher than an asker's. Answerers in Stack Overflow often have a higher reputation than askers because the questions in Stack Overflow are difficult. Since registration in Yahoo! Answers can earn 100 points, there are very few users who have a reputation score less than 100. Users with reputation scores less than 100 normally violate the rules or ask too many questions without giving answers.

Topics Distribution

Questions in Stack Overflow and Yahoo! Answers are grouped into topics. There are 31,250 topics in Stack Overflow and 945 topics in Yahoo! Answers. Figure 3.6 plots the distribution of several questions per topic. We see that many topics contain only a few questions. The majority of these questions belong to a few popular topics. Table 3.3 lists some of the most popular topics on these sites.

Questions in Yahoo! Answers belong to one topic only, while questions in Stack Overflow can belong to multiple topics. The number of topics ranges from 1 to 5. Figure 3.7 plots the number of topics per question. The majority of questions in Stack Overflow belong to multiple topics. Only 11 % questions belong to one topic only.

Next, we discuss the results of experiments on these two popular CQA sites.

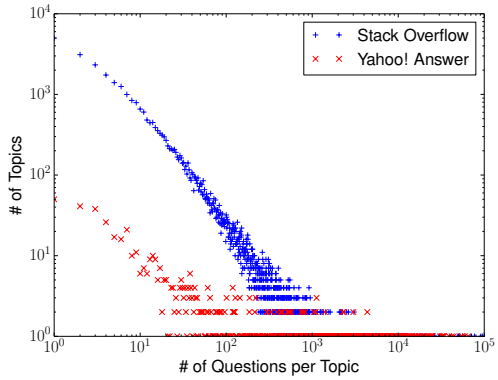


Figure 3.6: Distribution of number of questions per topic. A large number of topics contain only a few questions. There is a small number of topics that contain the majority of questions asked.

Stack Oveflow	**% question**	**# Yahoo! Answers**	**% question**
Javascript	3.75	Video & Online Games	2.93
Java	3.40	Current Events	2.67
php	2.89	Polls & Surveys	2.48
C#	2.64	Singles & Dating	2.47
Android	2.46	Psychology	1.99

Table 3.3: Most popular topics in two CQA sites.

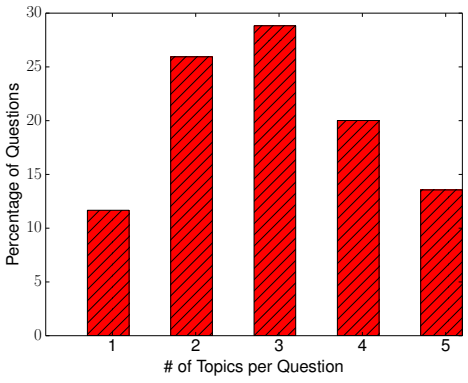


Figure 3.7: Distribution of number of topics per question. Majority of the questions in Stack Overflow belongs to multiple topics.

3.4 Experiments

In this section, we describe our experiments with Yahoo! Answers and Stack Overflow datasets using the *QRec* algorithm presented earlier. For comparison, we will use a randomized method, a probabilistic question recommendation, and White's method as the baselines. This section is divided into three parts: experimental setup, results and discussion of the results.

3.4.1 Experimental Setup

We worked with datasets collected from 9 months of Stack Overflow and 2 years of Yahoo! Answers to evaluate the proposed algorithm. For Stack Overflow, we used the postings from Jan-June 2014 to predict the July-Sept 2014 postings. For Yahoo! Answers, we used the postings in 2008 to predict the answer behavior in 2009.

Data prepossessing: In the first step, we eliminated questions that were not answered. In the Stack Overflow dataset, 29.5% of questions were not answered. In Yahoo! Answers, 32.7% of questions were unanswered, and thus eliminated. All the content was then converted to lower case. In the final step, we eliminated all stop words from the question content.

Competing methods: We compared our approach with the following methods:

- White's: We implement White's algorithm proposed in [127] and [128]
- PLSA: A probabilistic question recommendation for CQA [98]
- Rand: Potential users are selected randomly
- Active: Recommend questions to most active users

In White's method, authors computed the *tf-idf* between user profile and question. White's method also multiplied the *tf-idf* with a decay function. The decay function is defined as:

$$Decay = \begin{cases} 0 & \text{if } \Delta_t \leq \beta \\ 1 - e^{\frac{-\Delta_t}{\alpha}} & \text{if } \Delta_t > \beta \end{cases}$$

where Δ_t is the duration since the last answer given by this user, β is the minimum duration between two questions, and α is set to 120 / maximum number of questions answered per day.

The decay function has a lower value if the user answered a question recently. The purpose of the decay function is to balance the load across users.

In the PLSA method, the probability that user u will answer question $q = w_1, w_2, .., w_l$ is computed by $(\prod_{i=1}^{l} P(u, w_i))^{1/l}$.

We changed the size of potential answerers from 1 to 1000 users. The evaluation criteria is whether any users in the potential list answer the question. The larger size of potential answerers makes the correctness ratio higher. In cases where we select the whole community as potential answerers, we can make sure that at least one of them will answer the question. In practice, it is unrealistic to recommend a question to all users due to the community's large size.

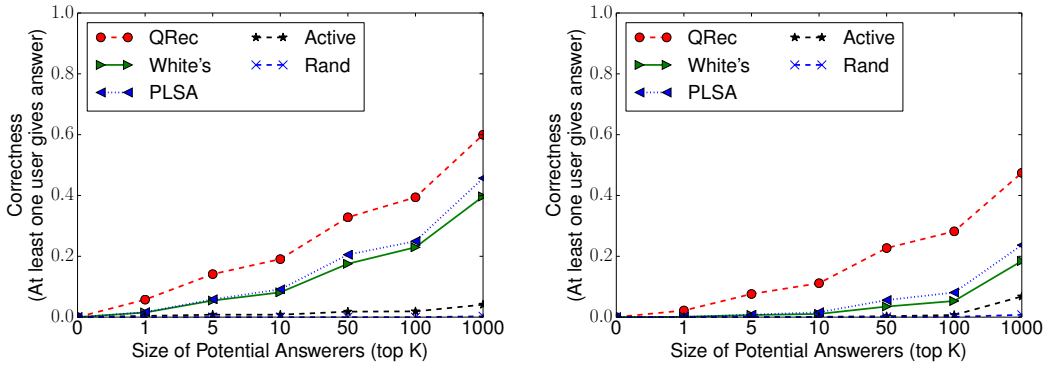
3.4.2 Results

The results and evaluation for this problem are based on the accuracy of a proposed method for finding the potential answerers. Specifically, the accuracy/effectiveness of the method is determined by the percentage of questions answered by at least one person in a set of identified potential answerers.

Accuracy

Figure 3.8 plots the correctness of different systems. The x-axis is the size of k potential answerers. It is trivial that the higher value of k, the higher the chance that at least one answerer will answer the question. The y-axis is the percentage of questions that will be answered by least one user in the set of identified potential answerers. A higher y-value is better. We see that adding topics and the user activity level improve the correctness of our algorithm. Since the questions in CQA are often short, computing the similarity between the question and the user profile does not perform well. Our *QRec* achieves the best correctness in both datasets.

Even though there are more than one million users on both sites, the probability that *QRec* can find at least one user to answer the question is higher than 0.5 if the size of potential answerers is 1000. Obviously, if we enlarge the group size, we could increase this accuracy.



(a) Stack Overflow

(b) Yahoo! Answers

Figure 3.8: Compare the correctness in selecting potential answerers. Higher is better. The *QRec* achieves the highest accuracy.

While this may not seem like a big feat, one needs to consider the enormity of the communities considered here. For example, the results showed that randomly selected answerers will perform very poorly due to the large community size. We also see that including topics when computing the similarity can improve the accuracy. This is because many questions in CQA are short. Thus, *tf-idf* of raw content does not reflect user expertise.

Mean Reciprocal Rank

We also measure the Mean reciprocal rank (MRR) of the ranking [123]. Let the set of questions be Q. For each question $q_i \in Q$, we find $rank_i$, which is the first correct answerer in the potential list. MRR is the average of the reciprocal ranks of results for all questions in Q as in Equation 3.6. Table 3.4 compares the MRR scores of *QRec* and competing methods. Our approach, *QRec*, achieves the highest MRR score in both data sets. Rand method has very low MRR score due to the large number of users in these CQA sites. *Active* method, which simply recommends questions to the most active users, also performs poorly due to the diversity of content in the community.

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\text{rank}_i} \tag{3.6}$$

Table 3.4: Compare the MRR of different algorithms. *QRec* achieves highest MRR score in both data sets.

Algorithm	**Stack Overflow**	**Yahoo! Answers**
Rand	0.000012	0.000009
Active	0.00019	0.00026
White's	0.033	0.029
PLSA	0.038	0.032
QRec	**0.053**	**0.039**

Importance of Each Similarity Feature

Furthermore, we measure the importance of each similarity feature in *QRec*. In order to evaluate the importance, we drop each similarity metric from our framework and measure the resulting drop in efficacy. Let $Acuracy$ be the accuracy when using all similarity metrics and $Acuracy^*$ be the accuracy when we drop one similarity metric. The relative drop in accuracy is measured as: $Drop = \frac{Accuracy - Accuracy^*}{Accuracy}$. Table 3.5 lists how much accuracy drops when removing the similarity features. The higher the drop, the more important the removed similarity feature. We see that the similarity in content and similarity in topics is more important in *QRec*.

Table 3.5: The importance of each similarity feature in *QRec* (k =1000). For example, dropping the similarity content in Stack Overflow causes 14.82% in accuracy. The higher value indicates the higher importance of this similarity metric. Content and topic similarity are more important than information network.

Similarity metrics	**Stack Overflow**	**Yahoo! Answers**
Content	14.82%	19.86%
Topics	16.85%	18.87%
Information network	7.28%	4.31%

The Effect on the Community

Loss of Anarchy: Since we are controlling the community–for example, by sending each question to a subset of users–we might lose answer quality if the question is only seen by the small set of potential answerers. As we described in Section 3.3.1, the community will vote on or rate each answer. Each answer's score is the aggregation of the humans' votes or rates. The best answer is defined as the answer that received the highest score. In order to measure the effect of the best answer, we define a metric called *Loss of Anarchy (LoA)*, which is calculated as:

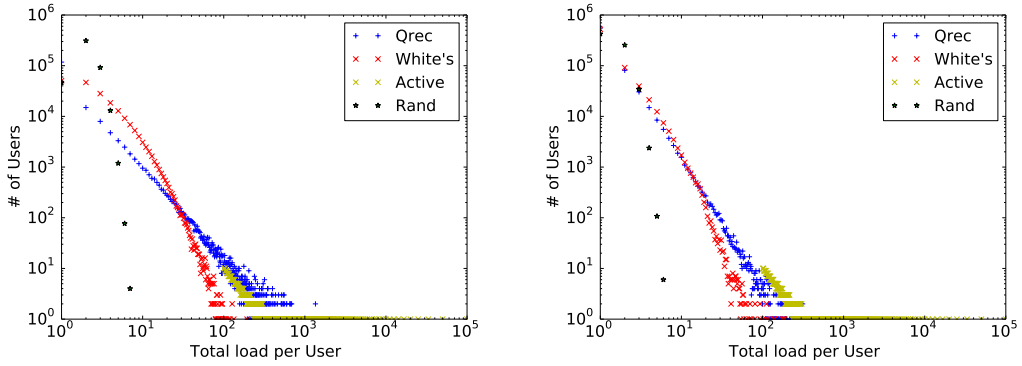
$$LoA = 1 - \frac{Best\ Answer\ Score\ (Recommended\ Set)}{Best\ Answer\ Score\ (All\ Users)} \tag{3.7}$$

Table 3.6 lists the *LoA*. We see that *QRec* preserves the quality of best answers, but the *LoA* is higher on Yahoo! Answers. The *Active* method has a high *LoA* value due to diverse CQA content and the fact that active users cannot cover many different topics. White's method and the PLSA method also have low *LoA*.

Table 3.6: Comparing the Loss of Anarchy. *QRec* preserves the quality of best answers when the question is answered by a small set of potential answerers.

Algorithm	**Stack Overflow**	**Yahoo! Answer**
Rand	0.91	0.95
Active	0.26	0.75
White's	0.14	0.27
PLSA	0.11	0.28
QRec	**0.09**	**0.21**

The load on each user: Since the question can be sent to the ordered list of users, some users might be overloaded, while others will not receive much. In this experiment setting, there is a set of questions and the list of potential answerers for each question q is $U = \{u_1, u_2, ..., u_k\}$, which is ranked in order. We assign the value of each user based on their rank. If the user is the i^{th} in the list, his reciprocal rank (RR) is $\frac{1}{i}$. Then, given the list of Q questions, we can compute the total RR of each user to see the total workload that each user could get.



(a) Stack Overflow

(b) Yahoo! Answers

Figure 3.9: The loading distribution on users. *QRec* reflects the "power law" nature of users' activities on CQA.

Figure 3.9 examines the distribution of the users' workload. In the *Rand* method, all the users have small RR values. In the case of the *Active* method, only a small number of users will be potential answerers, but the RR of each user is very high. In *QRec* and *White's* methods, the distributions of RR follow the power law, but the distribution of *QRec* is skewer. We use linear regression on the log-log scale to estimate the slope of distribution [18]. On Stack Overflow, the slopes of *QRec* and *White's* methods are -1.53 and -2.89 respectively. Similarly, the slopes of *QRec* and White's methods are respectively -2.22 and -3.01 on Yahoo! Answers. The workload distribution of *QRec* reflects the nature of users' contributions in CQA; for example, the number of answers per user follows the power law.

Overload on user

We also examine the overloaded on the users. Let *RealLoad* be the number of answers given by users during the recommendation period and *RecLoad* be the load created by different recommendation algorithms. The ratio of overload is define as $OverLoad = \frac{RecLoad - RealLoad}{RealLoad}$. Table 3.7 describes the average overload of users in the community. We see that the sending the question to most active users create a very high overload, while Rand method has low overload. The *White's* and *PLSA* method also has low workload but achieve low efficacy to get the question be answered. The *QRec* has reasonable overloading on users. In the real system, the users can see more question than the question they can answers.

Table 3.7: Comparing the Overload. *QRec* has reasonable overload.

Algorithm	**Stack Overflow**	**Yahoo! Answer**
Rand	0.62	0.46
Active	182.3	88.2
White's	2.04	1.51
PLSA	2.11	1.54
QRec	6.96	2.87

3.5 Discussion

The results showed that a user's topic interest and activity level (or social capital) are useful features when finding potential answerers. In the method we used to calculate the score between a user profile and a question, we used $log(activity\_level)$ instead of the *activity level* due to the skewed distribution of user activity. Figure 3.3 shows that some users answer a larger number of questions. These active users will always have a very high score if multiplying with *activity level* even if *tf-idf* is low. Thus, using the log value is important. Additionally, using users' topical expertise is better than using the raw content due to the brevity of questions in CQA. We believe that finding correct potential answerers can help CQA sites improve their users' experiences.

At this point, it is important to reflect on the newly proposed approach on theoretical grounds. Specifically, we want to discuss how and why a user's topical expertise and activity level, which we incorporated in the proposed method, help make this approach generalizable–theoretically and practically–for other applications of finding people through CQA that go beyond retrieving a question's potential answerers.

Many CQA sites, including those used in the work reported here, contain highly diverse and massive amounts of content. Our analysis shows that there are thousands of topics in Stack Overflow and Yahoo! Answers. Users are normally interested in a small number of topics, and they have a higher tendency to answer a question that pertains to topics with which they are familiar. In order to confirm this hypothesis, we did a test to measure a user's *answer-lift*. Given a user *u*, a question *q* belongs to topic *T*, *answer-lift* is defined as in Equation 3.8.

$$AnswerLift(u,q) = \frac{P(u\ answers\ q \mid u\ answered\ T\ before)}{P(u\ answers\ q \mid u\ not\ answered\ T\ before)} \tag{3.8}$$

In general, the *answer-lift* measures the ratio of the proportions of answering a question by users who participated in a topics opposed to users who were never participated in topic. We randomly picked 10,000 answers in the site and found that the answer lifts were 2.81 and 2.37 in Stack Overflow and Yahoo! Answers, respectively. These results support our method's effectiveness.

We expect that using topic interest can help us to solve other important problems that concern user retrieval in CQA, such as grouping users or finding special types of users. For example, we could use the *QRec* algorithm proposed here with a slight modification to find people who are likely to quit the community, as they are either losing interest in the topics covered by the community or have not found enough activities that pertain to their interests. Another example is finding a moderator in CQA. Due to the popularity of its community, a CQA site must have a set of users who will monitor others' activities. Topic interest is a strong indicator of these potential moderators.

This work is not without its limitations. In our experiments, we only evaluated our method using questions that received at least one answer. Due to the limitation of our datasets, it is not possible to test our method on unanswered questions. In this work, we do not investigate why certain questions are not answered. Understanding the answerability of questions is studied in several works [133], [34], [106]. Questions often go unanswered because they are spam, duplicates or annoying. Furthermore, our method did not consider users' temporal activity, such as changes in topic interest over time. Different forms of data and experiments are needed to address such issues.

3.6 Conclusion

In the work reported here, we presented a large scale study on the two of the most successful CQA sites: Yahoo! Answers and Stack Overflow. Our analysis highlighted the similarities and differences between these two sites, since they serve different purposes and communities. We also proposed a new method, *QRec*, to find potential answerers, addressing an important

problem in CQA. The results showed that our method can achieve high accuracy in both CQA sites. This could help CQA sites forward questions to suitable users. We expect that finding the correct answerers would allow a question to be answered more quickly and accurately. In both cases, users should be more satisfied with the site, increase their engagement and ultimately build a larger and healthier CQA community. Furthermore, the method used to compute scores in *QRec* is very efficient, which makes it applicable to large datasets.

In our current work, we only consider the list of questions answered when constructing user profiles. A profile based on answering activity can be considered a user's *expertise*. Future work will incorporate the list of questions asked into user profiles. A profile based on asking activity is considered a user's *interest*. We expect that combining both user expertise and user interest will provide a better view of CQA.

Chapter 4

Good and Bad Answerers

4.1 Motivation and Problem Definition

CQA is a user-driven community where registered users voluntarily create all content, including questions and answers. Therefore, content quality can easily vary, and it is paramount that sites offer high quality information to retain existing users and attract new users to support their online information seeking behaviors. The quality of information for educational purposes is even more critical, as educational CQA users may develop their knowledge base through contents within the platform. Students, in particular, may suffer when faced with inaccurate information. For example, it may be possible that students who use a CQA to ask questions about homework problems could be misled by incorrect answers without making proper judgements about answer quality. Thus, quality assessment is critical to ensure students' ability to learn through educational CQA via information seeking and sharing activities. At the moment, traditional CQAs depend on human judgments to evaluate content quality. Unfortunately, human assessors have many drawbacks including subjective (and possibly biased) assessments and insufficient availability to sort through the ever-increasing content in CQA sites. Additionally, such evaluators are difficult to recruit. Here, we address these concerns with a new framework for assessing content quality.

In order to reduce the workload involved in manually assessing answer quality, we developed a framework to automatically detect the quality of answers. This was a difficult task due to the complexity of CQA content. Here is the formal definition of our problem:

Formal definition:

Given:

- a set of users $U = \{u_1, u_1, ..., u_n\}$

- a set of posts $P = Q \cup A$,

  Q is the set of questions $Q = \{q_1, q_2, ..., q_{m1}\}$, and

  A is the set of answers $A = \{a_1, a_2, ..., a_{m2}\}$

- a set of interactions $I = \{i_1, i_2, ..., i_{m3}\}$ (such as giving thanks, up-votes, making friends)

Task: For arbitrary answer $a \in A$, predict whether a will be $deleted$ or $approved$? (or will a be a bad or a $good$ answer?)

4.2 Examining the Quality of an Answer

Our framework follows a classification problem. In the first step, we collect users' history and information, the interactions in the community, and the characteristics of answers. In the second step, we build the classification model based on history. In the last step, we predict the quality of new answers based on our trained models.

4.2.1 Feature Extraction

In order to classify answer quality, we build a list of features for each answer. Table 4.1 lists the features used in our study. They are divided into four groups: Personal Features, Community Features, Textual Features, and Contextual Features.

- Personal Features: These features are based on users' characteristics. Personal features include the activity of an answer's owner, including the number of answers given by the user, the number of questions asked by the user, the rank achieved by the user in the community and the user's grade level.

- Community Features: These features are based on the community's response to a user's answers, such as how many thanks were given or how many bans were made. Furthermore, we also consider the social connectivity of users in the community. In Brainly, users can make friends and exchange information. The friendships can be placed on a graph where users are nodes and the edge between two nodes represents their friendship.

We extract several features–such as the number of friends–pertaining to their connection, clustering the coefficient of a user and their ego-net (aka, the friends of friends). The clustering coefficient (CC_i) of a user measures how closely their neighbors form a clique, defined as

$$CC_i = \frac{\#\ of\ triangles\ connected\ i}{\#\ of\ connected\ triples\ centered\ on\ i} \tag{4.1}$$

Higher values mean that this user and their friends form a stronger connection. We denote $d_i = |N(i)|$ as the number of friends of user i, $|N(i)|$ denotes set of neighbors of i. Average degree of neighborhood is defined as

$$\bar{d}_N(i) = \frac{1}{d_i} \times \sum\nolimits_{j \in N_i} d_j \tag{4.2}$$

We also use egonet features of a node. A node's egonet is the subgraph created by the node and its neighbors. Egonet features include the size of egonet, the number of outgoing edges of egonet and the number of neighbors of egonet.

These features incorporate four social theories: Social Capital, Structural Hole, Balance and Social Exchange [10]. The capacity of social connection in information dissemination was conducted in [60]. Furthermore, these features are all computed locally, which is scalable and efficient. Computing the community features is an *almost* linear time algorithm, taking *almost* $O(n \log n)$, where n is number of nodes in graph.

- Textual Features: These features are based on answer content, such as length and format. We also check whether users use Latex for typing, since many answers provided in topical areas concerning mathematics and physics are easier to read if Latex is used. Furthermore, we measure the text readability based on two popular indexes: automated readability index (ARI) and Flesch reading ease score of answer (FRES) [57]. The ARI measures what grade level should understand the text, which is measured by

$$4.71 * \frac{\#\ of\ characters}{\#\ of\ words} + 0.5 * \frac{\#\ of\ words}{\#\ of\ sentences} - 21.43 \tag{4.3}$$

The FRES index measures the readability of the document. Higher FRES scores indicate the text is easier to understand. FRES index is calculated as

$$206.8 - 1.01 * \frac{\#\ of\ words}{\#\ of\ sentences} - 84.6 * \frac{\#\ of\ syllables}{\#\ of\ words} \tag{4.4}$$

- Contextual Features: These features include the question's grade level, the device types used to answer the question, the similarity between answer and question, how long it took to type the answer and the typing speed. The typing speed measures how many words the user types per second. The devices let us know whether the participant used a computer or a mobile device to answer. In order to compute the similarity between the answer and the question, we treat the answer and question as two vectors of words. The cosine similarity between these two vectors returns the similarity between them. Value 0 means that there are no common words between them. We believe that no common words between the answer and the question might indicate unrelated answers.

Similarly, we have a list of features that pertain to subject-focused CQA. Though they are slightly different, they are essentially equivalent to those that apply to general CQA. For example, the ranking of users in Brainly is similar to the reputation of users in Stack Overflow. In focused CQA such as Stack Overflow, the good content is up-voted, and the user can achieve a respected reputation by providing high quality content. In many CQAs, friendship connections are not available. Thus, we can create the virtual friendship network [38]. In particular, there is a connection between users if they interact via asking-answering activity. Furthermore, some features, such as the device type or typing speed, are not available in the public data set. Fortunately, the results suggest that these features are not very important and have low prediction power.

Building training set:

In order to build the training data, we extracted features for each answer as seen in Table 4.1. These can also be divided into two types of features. *(i): Immediate features* are the length, device type, typing speed, and similarity between answers and questions. These features are extracted immediately when the answer is posted. *(ii:) History features*, such as the number of thanks and number of answers given, can be built beforehand and updated whenever these

Table 4.1: Lists of features on educational CQA are classified into four groups of features: Personal, Community, Textual and Contextual. The abbreviations of features are in brackets.

Personal Features
Number of answers given (n_answers)
Number of questions asked (n_questions)
Ranking of users (rank_id)
Grade level of users (u_grade)
Community Features
Number of thanks that user received (thanks_count)
Number of warnings that user received (warns)
Number of spam reports that user received (spam_count)
Number of friends in community (friends_count)
Clustering Coefficient in friendship network (cc)
Average degree of neighborhood (deg_adj)
Average CC of friends (cc_adj)
Size of ego-network of friendship (ego)
Number of outgoing edges in ego-network (ego_out)
Number of neighbors in ego-network (ego_adj)
Textual features
The length of answer (length)
The readability of answer (ari)
The Flesch Reading Ease Score of answer (fres)
The format of answer (well_format)
Using advance math typing: latex (contain_tex)
Contextual features
The grade level of question (q_grade)
The grade difference between answerer & question (diff_grade)
The rank difference between answerer & asker (diff_rank)
The similarity between answer and question (sim)
Device used to type answer (client_type)
Duration to answer (time_to_answer)
Typing speed (typing_speed)

Table 4.2: Lists of features on Focused CQA (Stack Overflow) are classified into four groups of features: Personal, Community, Textual and Contextual. The abbreviations of features are in brackets.

Personal Features
Number of answers given (n_answers)
Number of questions asked (n_questions)
Reputation of users (answerer_rep)
Community Features
Number of up-votes that user received (up_votes)
Number of down-votes that user received (down_votes)
Number of friends in community (friends_count)
Clustering Coefficient in friendship network (cc)
Average degree of neighborhood (deg_adj)
Average CC of friends (cc_adj)
Size of ego-network of friendship (ego)
Number of outgoing edges in ego-network (ego_out)
Number of neighbors in ego-network (ego_adj)
Textual features
The length of answer (length)
The readability of answer (ari)
The Flesch Reading Ease Score of answer (fres)
The format of answer (well_format)
Using advance math typing: latex (contain_tex)
Contextual features
The rank difference between answerer & asker (diff_rep_users)
The similarity between answer and question (sim)
Duration to answer (time_to_answer)

features change. Thus, when a new answer is posted, we can immediately extract all proposed features, allowing our method to work in real time. Further details about these settings are described in Section 4.4. Next, we describe three classifiers used in our study.

4.2.2 Classification

Because our framework could operate with almost any classification model, we compared the performance of different models in this study. In particular, we tested the classification algorithms below [12]. Let $X = x_1, x_2, ..., x_n$ be the list of features. The list of classification algorithms are summarized as:

- Logistic regression (log-reg): Log-reg is a generalized linear model with sigmoid function

$$P(Y = 1|X = \frac{1}{1 + exp(-b)}) \tag{4.5}$$

  where $b = w_0 + \sum(w_i.x_i)$, w_i are the inferred parameters from regression.

- Decision trees: The Tree-based method is a nonlinear model that partitions features into smaller sets and fits a simple model into each subset. The decision tree includes two-stage processes: tree growing and tree pruning. These steps stop when a certain depth is reached or each partition has a fixed number of nodes.

- Random Forest (RF): RF is an average model approach [47], [15] and we use a bag of 100 decision trees. Given a sample set, the RF method randomly samples data and builds a decision tree. This step also selects a random subset of features for each tree. The final outcome is based on the average of these decisions. The pseudo-code of RF is described in Algorithm 2. There are some advantages of RF. When building each tree in Step 4, RF randomly selects a list of features and a subset of data. Thus, RF can avoid the overfitting problem of the decision tree. Furthermore, each tree can be built separately, which makes distributively computing the trees extremely easy.

Figure 4.1 summarizes the architecture of our method. In the framework, textual features

Algorithm 2 Pseudo-code of Random Forest algorithm

Input:

- A set of training input $T = \{(X_i, y_i)\}$, i = 1, ..., n.
- Number of trees N_{trees}
- A new feature vector X_{new}

Output: the prediction outcome of X_{new}

1: **for** $i = 1 : N_{trees}$ **do**
2: Randomly select a subset of training $T_{rand} \subset T$
3: Build the tree h_i based on T_{rand}
4: In each internal node of h_i, randomly select a set of features and split the trees based on these selected features
5: **end for**
6: $Pred(X_{new}) = \sum_{i=1}^{N_{trees}} h_i(X_{new})$
7: **return** $Pred(X_{new})$

and contextual features can be quickly calculated as soon as a new answer is posted. Personal and community features are extracted from the history database. After querying personal and contextual features, some features related to a user's activities (e.g., number of answers increased over time, etc.) are also updated accordingly.

Next, we will describe the data sets used in our study, as well as some characteristics of users in online learning communities.

4.3 Datasets and Characterization of the Data

We used data in Stack Overflow and Brainly. Table 4.3 describes the basic characteristics of these sites.

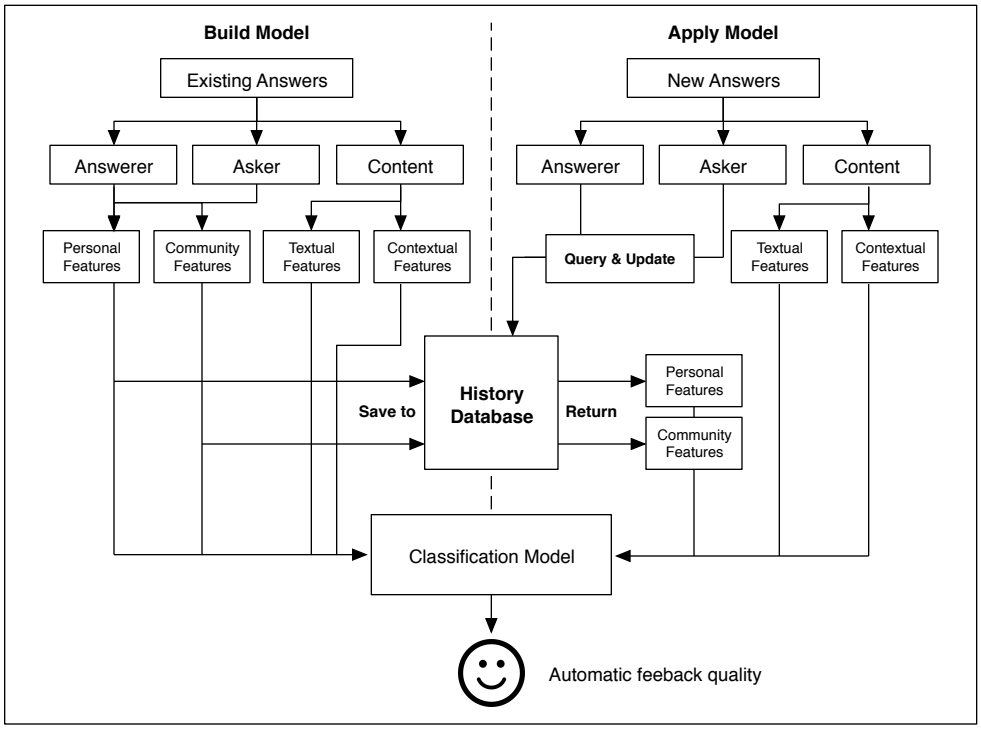


Figure 4.1: An overview of a framework proposed in the study.

Table 4.3: Description about data. US is the Brainly data in the United States market, PL is Brainly data in Poland.

Site	Period	# of Users	# of Posts	# of Answers
US	Nov '13 to Dec '15	800 K	1.5 M	700 K
PL	Mar '09 to Dec '15	2.9 M	19.9 M	10 M
Stack Overflow	Jul '08 to Sep '14	3.4 M	21.2 M	7.7 M

4.3.1 Brainly: An Educational CQA

Overview: Brainly.com is an online Q&A for students and educators with millions of active users. In our study, we use the data from two markets: the United States (US) and Poland (PL). Table 4.3 describes some characteristics of these datasets. Our study uses both deleted and approved answers. Brainly requires high quality answers, meaning that moderators delete incorrect answers, incomplete answers or spam posts. A moderator is an experienced user who

has contributed significantly to the community. Established in 2013, the United States market–currently consisting of 60,000 active users–is emerging in Brainly. In contrast, Brainly has been used since 2009 in Poland, and thus boasts an established market of 760,000 active users in that country. The posts in Brainly are divided into three levels (grades): primary, secondary and high school. There is no detail category for each level.

Manual Assessment

A total of 400 answers were extracted from Brainly in the first week of January 2016. From this, 200 deleted answers and 200 approved answered were examined manually to discover the reasons for deleting answers. We see that approved answers have high quality and provide complete and detailed information to answer the questions. Brainly is an educational CQA; the website wants to promote the correct answers since the user is in the learning stage and learning the wrong information is not desirable. Thus, the site strictly deletes mistaken, incomplete and/or uninformative answers. The main reasons for deleting answers include:

- *Lack of explanation:* The answer is uninformative and does not provide any supporting work and/or examples. This often occurs in mathematics and physics where answers require calculations.
- *Mistakes:* The answer provides wrong information or results.
- *Incomplete:* The answer missed crucial information.
- *Too Vague:* The answer is too vague to be considered valid. The answer needs to elaborate on the topic or demonstrate more effort.
- *Pointless Answer:* The answer is unhelpful and/or pointless. For example, “I want to help but I need more knowledge” or “I think the answer is ...”
- *Plagiarism:* The answer copies the content from other websites or previous answers.
- *Grammar:* The answer has grammatical, syntactical or linguistic issues.
- *Others:* The answer advertised other services or seems antisocial. For example, “Pay me $20 and I can write for you” or “Do a little math loh”.

These reasons can also be grouped into three main types includes *(I) Erroneous:* Mistakes, Pointless Answer, Grammar Error, *Not Clear:* Lack of explanation, Incomplete, Too Vague, and *Miscellaneous:* Plagiarism, Others. Figure 4.2 plots the distribution of reasons why answers are deleted. In the majority of cases, answers are deleted due to answerers' insufficient knowledge. In some cases, an answer might be considered low quality due to improper grammar, language and other details. Table 4.4 lists some examples of deleted answers and approved answers.

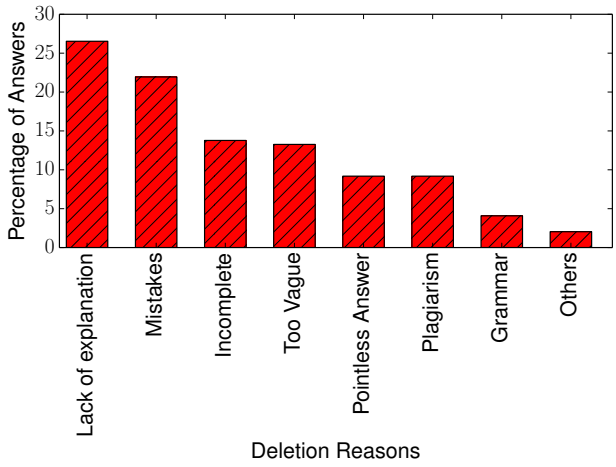


Figure 4.2: Distribution of deletion reasons. Some popular reasons are lack of explanation, containing mistakes and incomplete answers.

Other Characteristics

Ranking of users: Brainly uses a gamification-based feature that illustrates how actively users participate in answering questions. In the current Brainly system, there are seven hierarchical ranks, from Beginner to Genius. Users can advance through these categories in two ways: receiving a high number of points on their answers, and having their answers selected as "best" by askers. This ranking mechanism is similar to other CQA sites such as Yahoo! Answers and Stack Overflow, which encourage users to contribute to the site in order to earn a high reputation.

Table 4.4: Popular reasons for deleting answers. Answers are deleted for diverse reasons. High quality answers are approved on Brainly.

Reason	**Example**
Lack of explanation (or uninformative)	**Question**: Out of 32 students in a class, 5 said they ride their bikes to school. Based on these results, how many of 800 students in the school ride their bikes to school? **Approved Answer:** (5/32)=(x/800). Set up the equation, multiply both sides by 800, (4000/32)=x, simplify x=125 . **Deleted Answer:** 125 students
Mistakes	**Question:** Simplify 3(7x+2y)? **Approved Answer:** It is 21x+6y. You would just multiply 7*3 =21 then u would bring down the x, then bring down the addition sign then multiply 3*2= 6 **Deleted Answer:** 3(9x) is the simplified
Incomplete	Q: The two sociologists who referred to society as being a kind of living organism were A) Auguste Comte; Emile Durkheim B) Karl Marx; Max Weber C) Auguste Comte; Herbert Spencer D) Emile Durkheim; Max Weber **Approved Answer:** Auguste Comte; Emile Durkheim **Deleted Answer:** AUGUSTE COMTE
Too vague	**Question:** How do u add and subtract fractions with different denominator? **Approved Answer:** You first have to get both of the denominators to be the same. For example: if you're adding $\frac{1}{2} + \frac{1}{4}$, you need to multiply the 1/2 by 2/2 to get the fraction to be 2/4 so you can add it with 1/4. **Deleted Answer:** You need to make them have a common denominator.
Plagiarism	Q: Why shouldn't we bath after having lunch or dinner? **Approved Answer:** Due to the cold water flowing on the body, the blood circulation increases around the body near the skin. This reduces the blood flow in the stomach... **Deleted Answer**: Following ingestion food the splanchnic circulation or blood circulation to the viscera or the different parts and organs concerned with food ...
Grammatical linguistic errors	Q: What color is given for desert in map? **Approved Answer:** Brown **Deleted Answer**: The answer is (BEROWN) .

Deleting answers in Brainly: Brainly tries to maintain high quality answers, and moderators are recruited to participate heavily in deleting questions. Only experienced users, such as moderators, are allowed to delete answers. Answers may be deleted if they are incomplete, incorrect, irrelevant or spam. A significant portion of answers are deleted (30%) to maintain the site's high quality. But deleting this many answers is time-consuming and labor intensive. Furthermore, manual deletion might not be prompt, meaning unsuitable content remains on the site until moderators have a chance to review it. Thus, developing an automatic mechanism to assess answer quality is a critical task.

Friendship in Brainly: Users in this social CQA can make friendships and exchange ideas and solutions. After joining the community, users can request to make friends with other users if their topics of interest are related. The friendship feature in Brainly is a new mechanism that encourages students to exchange ideas and solutions. In traditional CQA such as Yahoo! Answers and Stack Overflow, there is no formalized friendship connection. Figure 4.3 depicts the distribution of number of friends per user. We see that it follows the power law with long tail. Some users have many connections in the community while others make only a few contacts. We expect that users with many connections are more active and more committed to answering questions.

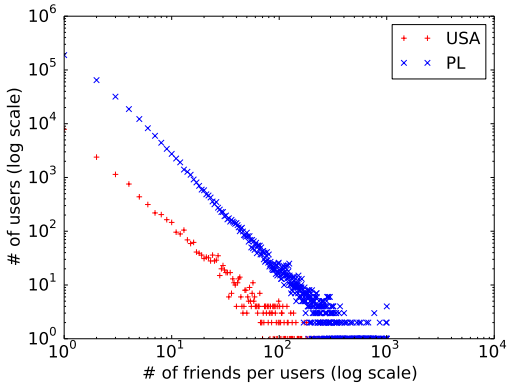


Figure 4.3: Distribution of number of friends per user in log-log scale. The number of friends follows the power law. Some users make a lot of friends in this community.

Activity in Brainly: This is a free community. Anyone can contribute by asking questions, giving answers, giving thanks and making friends. Due to the nature of the community, the

contribution of each user is different and based on their interests and availability. Figure 4.4 plots the distribution of number of answers given per user. Again, this follows the power law with some very active users. Answering questions is a popular way for users to earn higher scores and increase their ranking in the community. Active users provide many answers to demonstrate that they are willing to devote their time to helping others. Answerers can also gain knowledge and trust from the community by answering a high volume of questions. Thus, these users may provide high quality answers.

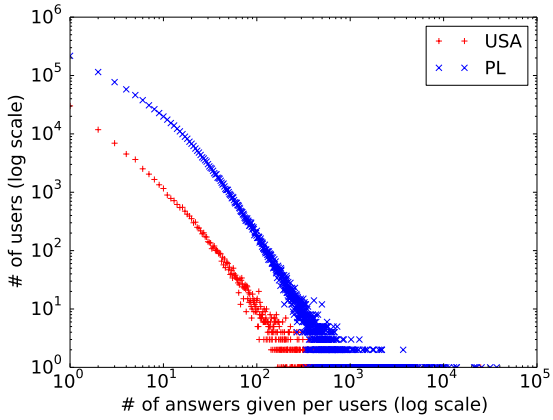


Figure 4.4: Distribution of number of answers given per user. A small fraction of users answer a lot of questions while many users answer a few questions.

Subjects of interest: The questions in Brainly are divided into different subjects/topics, such as Mathematics, Physics, etc. We examine how students participate in these topics between two countries. Figure 4.5 shows that students in both countries participate more in the topical areas of Mathematics, History and English. The percentage of posts on Mathematics in the United States is significantly higher than in Poland (42% vs. 35%). This might indicate that students in the US need more help with Mathematics.

The readability of answers: We want to see whether approved answers are more readable–or clearer–than deleted answers. We use ARI to measure answers' readability. Findings indicate that the ARI of approved answers is 6.9 ± 3.1, while the ARI of deleted answers is 5.1 ± 3.2. Similarly, the FRES indexes of deleted and approved answers are 69.9 ± 23.1 and 62.2 ± 22.5 respectively. A higher FRES value means that an answer is easier to read. We see that the

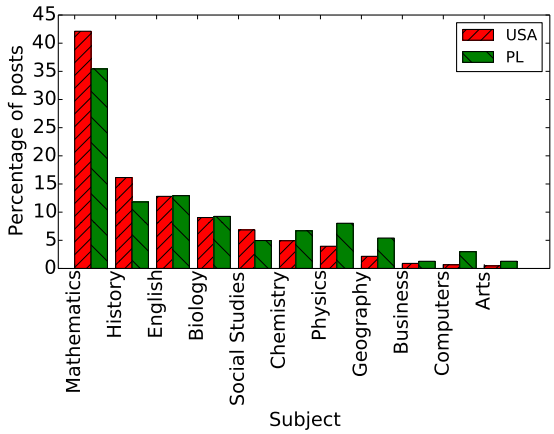


Figure 4.5: Percentage of posts in different subjects. Both countries are similar and students are most active in discussing Mathematics, History and English.

standard deviation is large for both indexes due to the diversity of content. We conducted a t-test and saw that the difference is significant with $p = 0.05$. This difference exists because many answers in primary and secondary levels are deleted. In general, the answers in primary and secondary levels are easy to read.

Quality of experienced and new users: We examine the quality of answers from both new users and experienced users. We examine the deletion rate of answers based on the ranking of users. Figure 4.6 plots the rate of answers deleted from differently ranked users. We see that low-ranked users have a very high rate of deletion. Because Brainly supports education, the community expects correct answers. Even incomplete answers are deleted. We see that many intermediate users' (such as rank 3 or rank 4 users') answers are deleted. This demonstrates that Brainly maintains a very high standard to ensure quality answers.

In the next section, we will describe the experiment set up, the main results and the discussion of the results.

4.3.2 Stack Overflow: A Focused CQA

Table 4.3 lists the characteristics of the data set used in our experiment. The data includes information on all users and all posts from Stack Overflow since its creation in 2008 until September 2014. There are more than 21 million posts in this dataset. Each question in Stack

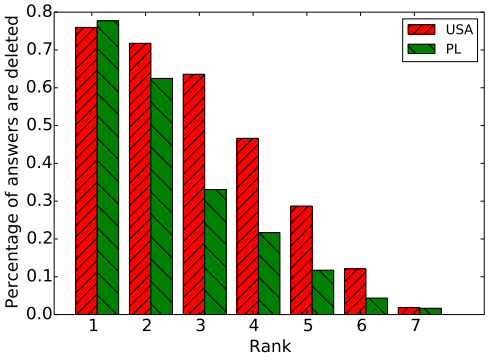


Figure 4.6: Percentage of answers deleted vs. rank level. Rank 1 is beginner while rank 7 is genius. Highly ranked users have fewer deleted answers due to their experience. A high deletion rate indicates that the site's answer requirements are very strict.

Overflow has its own tags, which are considered the post topics. Each question has one to five tags. Users in Stack Overflow can engage in different activities such as posting questions, giving answers, voting for the best answer and up-voting or down-voting a post. They can also earn a reputation by posting high quality questions and answers. The table shows how this reputation is calculated. Table 3.1 summarizes the score system in Stack Overflow. In order to maintain high quality posts and encourage high quality posting, the score system is very rewarding and but strict.

Number of Posts

Figure 4.7 plots the number of answers per user. It shows that a few users are very active and contribute a majority of the content in Stack Overflow. This reflects the general observation that many users in CQA only answer a few questions, while a small number of users are very active.

Number of Posts vs. User Reputation

Figure 4.8 shows the relationship between 1000 randomly selected users' number of posts and earned scores. The Figure demonstrates that if a user contributes more frequently, they can earn a higher score. This relationship, however, is not linear because post quality significantly

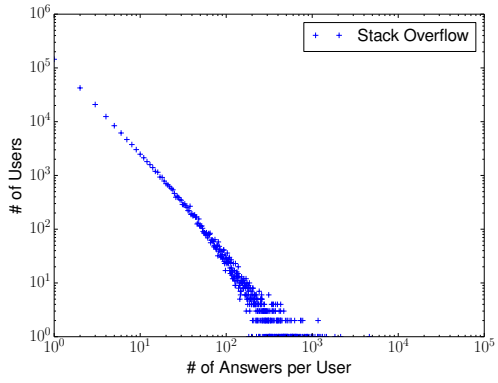


Figure 4.7: Number of answers given per user in Stack Overflow. The contribution follows the power law.

affects a user's score. For example, if a user posts 1000 mid or low quality posts, they may earn the same score as another user who contributes less than 100 high quality posts. But in general, a user can only become a top contributor if they are active in the community.

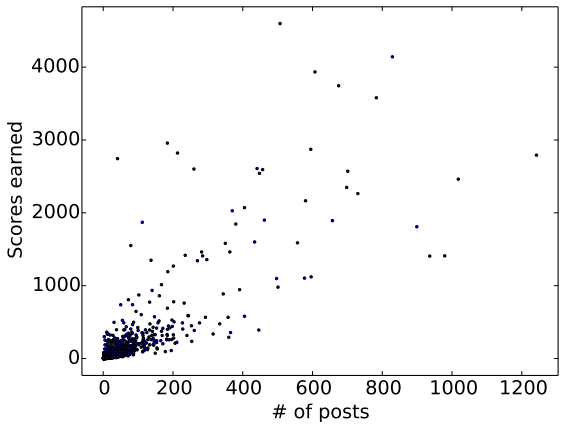


Figure 4.8: The number of posts vs. the score earned by 1000 randomly selected users. A more active user can earn a higher score.

Question-Answering Behavior vs. User Reputation

Users in CQA can contribute to the community and earn their reputation by posting useful questions and answers. We want to see whether there are differences between normal users and

top contributors. Figure 4.9 shows the distribution for different types of users. We see that top contributors' posts are often answers. For example, in the group of top contributors, 60% of these users list more than 90% of their posts as answers. In contrast, more than 30% of normal users have less than 10% of their posts as answers. This indicates that many normal users may only join a CQA community to find answers.

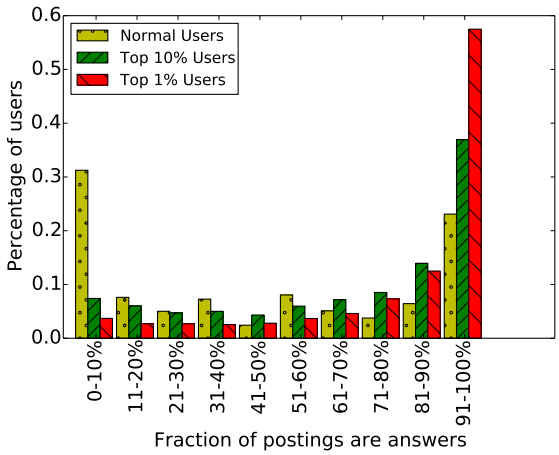


Figure 4.9: Comparing users' question-answering behavior. The x-axis is the percentage of posts that are answers. Most top contributors' posts are answers.

4.4 Experiments and Results

In this section we will describe our experimental setup, highlight the main results, and provide a discussion around these experiments and findings.

4.4.1 Experimental Setup

To compare the performance of classifications, we used different classification algorithms with different sets of features. In the default setting, we used the Random Forest of 100 decision trees. In the evaluation, we randomly selected 200,000 answers in each data set to validate our framework's the accuracy. We used 10-fold cross validation to select parameter classification with 70-30% training, testing set. In order to compare the efficacy, we examined the accuracy, F1-score, confusion matrix and Area Under Curve.

In Brainly's data sets, our framework predicts whether an answer is deleted or approved by the community moderators. In Stack Overflow, we use the community feedback as the ground truth of answer quality. The high quality answer gets the positive score and the low quality answer receives the negative score. Section 4.3 describes the score-assigning mechanism, which is based on the up-votes and down-votes.

4.4.2 Main Results

Accuracy

Accuracy is defined as the percentage of correctly classified answers. Figure 4.10 plots the accuracy of applying Random Forest (RF) to different groups of features. *PF, CmF, TF, CtF* denotes the results when our frameworks used personal features, community features, textual features and contextual features, respectively. *All* presents the accuracy when using every feature. The results demonstrate that personal features and community features are more useful in predicting answer quality. This makes sense because good users normally provide good answers. The textual features have less prediction value due to the complexity of the site's content. We will examine the details of each feature later. Furthermore, our classifier achieves a very high accuracy of more than 78.5% in all data sets. Given the complexity of answers within the community, these results are very encouraging.

F1-score

We also measure $F1$ score, which considers both precision and recall. Precision is the fraction of instances that are relevant, while recall is the fraction of relevant instances that are retrieved. The value of $F1$ is defined as

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{4.6}$$

Figure 4.11 shows that using all features achieves the highest $F1$ score, which is more than 84% in both Braily data sets and 78.5% in Stack Overflow. High $F1$ scores demonstrate that our method can achieve both high precision and recall. Similar to accuracy, the results suggest that personal features and community features are the most important in the model.

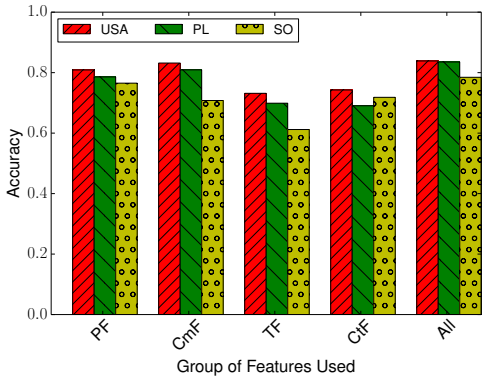


Figure 4.10: The accuracy of using different groups of features. *PF, CmF, TF, CtF* denotes the results when our frameworks used personal features, community features, textual features and contextual features respectively. *All* means using all features. *PF, CmF* are more useful in predicting answer quality. (Random Forest is the classifier used.)

The performances in the Brainly data sets are higher than those in Stack Overflow (83.5% vs. 78.5%). Several reasons explain this discrepancy. First, Stack Overflow provides a public data set that does not contain every feature. Furthermore, the content in Stack Overflow is more complicated than that in Brainly because Stack Overflow is used widely by professionals, such as programmers, with specialized knowledge.

Comparing Different Classifiers

Table 4.5 compares the accuracy when applying different classification algorithms. We see that Random Forest outperforms logistic regression and decision trees. This is due to the non-linear relationship between features and answer quality. Furthermore, Random Forest also randomly selects different sets of features to build trees, which avoids over-fitting in classification. Random Forest is also an efficient algorithm that can work well on large data sets. Our experiment was conducted on a machine with 2.2 GHz quad-core, 16 GB of RAM. It was implemented in Python code on a data set consisting of 200,000 answers. The experiment took 34 seconds to train the model and less than 1 millisecond to predict each answer. Training is one time cost. It implies that our framework can determine the quality of answers in real time. Thus, our suggestion is to use Random Forest as a classifier in a real system.

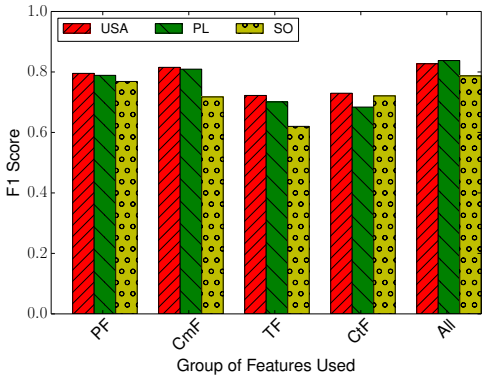


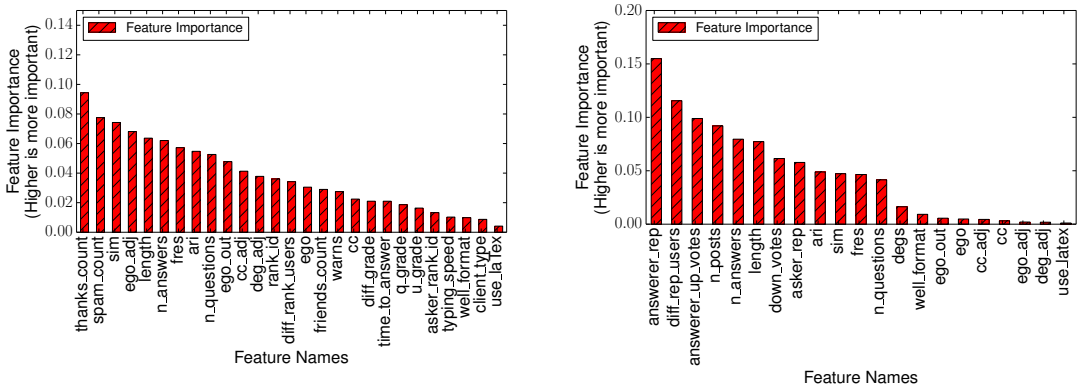
Figure 4.11: Comparing the $F1$ score (higher is better) when using different groups of features. Random Forest is the classifier used. High F1 score shows that our method achieves high value in both precision and recall. Again, personal features and community features are more important in the model.

Table 4.5: Compare the accuracy of different classifiers. Random Forest (bag of 100 trees) outperforms logistic regression and decision trees.

Classification	**USA**	**PL**	**Stack Overflow**
Logistic Regression	79.1%	76.8%	73.1%
Decision Trees	78.2%	77.1%	72.9%
Random Forest	**83.9%**	**83.5%**	**78.5%**

Feature Importance

In this section, we measure which features are more important. In order to determine this, we use a permutation test to remove the features and measure the accuracy of out-out-bag (OOB) samples. The important features will substantially degenerate the accuracy. Figure 4.12 reports the importance of different features used in our study. The three most important features are the number of thanks users receive, the amount of spam reported, and the similarity between answers and questions. We believe that some features strongly correlate with quality but ultimately appear less important, such as device type or using Latex when typing. For example, participants using mobile devices to submit their answers may make more mistakes, or a participant using Latex markup might indicate their significant experience with certain topics. Unfortunately, there were only a few answers that were posted from mobile devices or typed in Latex (less than 10%). Thus, these features lost their prediction value. In the Stack Overflow data, the social connection feature is less important because the site facilitates virtual connections. As we mentioned before, there is no formal friendship in Stack Overflow. In a smaller community such as a forum, the virtual connection is stronger and contains more information [38]. But because CQA is a large community, creating a virtual connection based on asking-answering interactions has less prediction value.



(a) Brainly (US) (b) Stack Overflow

Figure 4.12: Measure of important features (higher is more important). The most important features are *number of thanks - up votes*, *reputation*, *number of spams*, *similarity between question and answer*, and so on. Table 4.1 and Table 4.2 list the notations of used features.

Features Selection

One possible concern is whether features selection can improve the performance of our method. The general idea of features selection is to remove features that have no correlation with the outcome, or to remove two similar features. In both cases, such features cause over-fitting in the prediction. In Random Forest, we already randomly select features when building the trees. In particular, Step 4 in Algorithm 2 selects random features to build the trees. Furthermore, the number of features in our study is not large. Thus, features selection is unnecessary and does not help improve accuracy.

High Quality Answers and Low Quality Answers

We discuss which is more difficult to detect: high quality or low quality answers. Table 4.6 examines the confusion matrix that describes how answers are wrongly classified in different data sets. We see that detecting deleted questions achieves higher accuracy than detecting approved answers in the US. This is so because many answers in the US market are answered by newcomers, who do not often satisfy the high quality criteria established by this CQA community. In the PL market, there is no difference due to a well-established community and the fact that the majority of the participants are experienced users. Similarly, there is no difference in Stack Overflow data since Stack Overflow is also a well-established community.

Receiver Operating Characteristic (ROC)

We also evaluate the ROC of the approved answers for both data sets. The ROC denotes the classification's ability to find the correct high quality answers with different thresholds. The curve in Figure 4.13 plots the True Positive rate against the False Positive rate. We see that the area under ROC is higher than 0.91 in both data sets. In the real deployment, we can set different thresholds to select the approved answers based on various requirements. For example, the administrators of the site might believe that 17% is insufficient and require that the automatic assessments not make mistakes with a rate of more than 0.05. Figure 4.13 shows that if the False Positive Rate is 0.05, the True Positive Rates of the US and PL are 0.73 and 0.62, respectively. Otherwise, we can detect a majority of the approved answers with a

Table 4.6: Confusion matrix for predicting answer quality.

Prediction outcome

		Deleted	Approved	Total
Actual value	Deleted	90.1%	9.9%	100%
	Approved	22.4%	77.6%	100%

a. Brainly – United States

Prediction outcome

		Deleted	Approved	Total
Actual value	Deleted	81.5%	18.5%	100%
	Approved	14.5%	85.5%	100%

b. Brainly – Poland

Prediction outcome

		Negative	Positive	Total
Actual value	Negative	77.7%	22.3%	100%
	Positive	20.6%	79.4%	100%

c. Stack Overflow

small error rate. The rest of the answers are considered borderline entities, which are hard to differentiate between good or bad. Under these circumstances, we can still take advantage of moderators and askers to provide another evaluation of the questions or answers. It's more difficult to detect the quality of answers in Stack Overflow since this is a focused CQA where the questions and answers are much more difficult and complex than those in educational CQA. In all cases, our model significantly reduces humans' workload.

4.5 Discussion

Asking a question for learning purposes is not a new phenomenon within the area of information seeking. It is an innate and purposive human behavior to satisfy an information need via

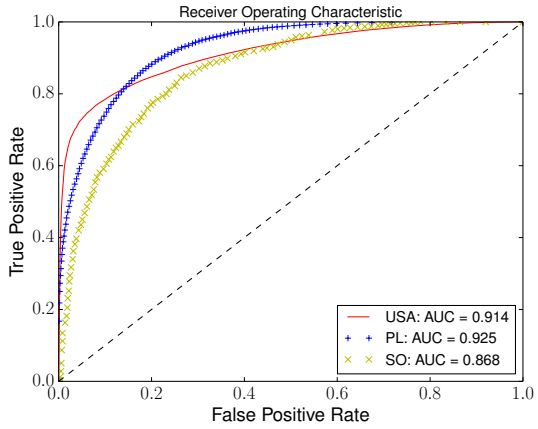


Figure 4.13: Area Under ROC curve for our frameworks are above 0.9 in both Brainly data sets and is 0.86 in the Stack Overflow data set.

searching techniques [25], and information and knowledge received through an asker's questioning behavior may become meaningful if the information acquired helps solve their problematic situations [135]. In recent years, new information and communication technologies have emerged to develop novel ways for users to interact with information systems and experts in order to seek information. These new resources include digital libraries and virtual reference tools, as well as CQA services that allow users to both consume and produce information.

According to Ross et al. [102], librarians and experts in both face-to-face and virtual reference environments engage in a process of negotiating an asker's question. This helps identify an asker's information need and allows them to construct a better question that will receive higher quality answers. However, this process of question negotiation may not occur in the context of CQA, which significantly impedes an asker's ability to receive satisfactory answers. Identifying what constitutes the content quality of information generated in CQA (or for that matter, any online repository with user-generated content) can be critical to the applicability and sustainability of such digital resources.

When it comes to CQA in educational contexts, seeking and sharing high quality answers to a question may be more critical since question-answering interactions for educational answers is likely to solicit factual and verifiable information, in contrast to general-purpose CQA services where advice and opinion-seeking questions are predominant [24]. Thus, evaluating

and assessing the quality of educational answers in CQA is important for not only improving user satisfaction, but also for supporting students' learning processes. In more complex CQA that pertains to focused and difficult topics, communities heavily rely on moderators, who often find themselves with inordinately high work loads. Thus, the accuracy of our framework is encouraging, as it suggests that an automated process can alleviate stress and improve content within CQA communities.

In this work, we attempted to investigate an educational and a focused CQA by examining a series of textual and non-textual answer features in order to identify levels of content quality among the answers. The study first attempted to identify a list of content characteristics that would constitute answer quality. In the second step, we applied these features in order to automatically assess the quality of answer. The results showed that personal features and community features are more robust in determining answer quality. Most of these features are available and feasible to compute in other CQA sites, making our approach applicable to the wider community. Furthermore, the efficacy and efficiency of our method make it possible to implement within the real system. In our experiment on a standard PC, it takes less than one millisecond to return the prediction. It also only takes less than one minute to train the model with 200,000 answers from Brainly. However, the training step is a one-time cost and can be accomplished using distributed processing. By applying this technique to the real system, we believe that we can reduce the number of deleted answers by giving a warning immediately before a user submits a response to the community. Furthermore, the approach can approve high-quality answers and thus significantly reduce an asker's wait time.

Most of the previous work applied logistic regression in order to evaluate the quality of answers. Our work showed that a "wisdom of the crowd" approach, such as Random Forest, can significantly improve the accuracy of assessing answer quality due to a non-linear relationship between the features and the quality of answers. For example, the results showed that longer answers are more likely to be approved compared to deleted answers. But very lengthy answers might signal low-quality answers, such as confusing or spam answers. Even though the current study focused on evaluating answer quality in terms of educational information on CQAs for online learning in particular, the study also suggests an alternative way to investigate general

CQAs' answer quality via our method–i.e., a "wisdom of the crowd" approach–in order to improve the accuracy of quality answer assessment. Moreover, in terms of practical implications of users' interactions for content moderation on CQA, the findings may propose a variety of features or tools (e.g., detecting spams, trolling, plagiarism, etc.) that support content moderators in order to develop a healthy online community in which users may be able to seek and share high quality information and knowledge via question-answering interactions.

There are also limitations to our work. For example, it can only detect high and low quality answers. It would be helpful if we could provide suggestions to improve the overall quality of these answers. We believe this is challenging but highly rewarding work that might require a significant effort to examine answer meaning. Furthermore, our approach was heavily based on the communities' past interactions, and thus has limited applicability to a newly-formed CQA community.

4.6 Conclusion

The work described here focused on answer quality in both educational and subject-focused CQAs. An educational CQA supports young students in the learning stage while a focused CQA targets professional users. We strove to improve the efficiency of community management in terms of facilitating users' ability to seek and share high quality answers to a question. Traditional methods involving human assessments may not be sufficient due to the large amount of content available, as well as subjective assessments of answer quality; therefore, we propose a framework to automatically assess the quality of answers for these communities. In general, our framework integrated four different aspects of answers, such as personal features, community features, textual features and contextual features. We presented here the first large-scale study on CQA for education and focused programming topics. Our method achieves high performance in all important metrics such as accuracy, F1 score and Area under ROC curve. Furthermore, the experiment demonstrates that our method is highly efficient and can work well in a real time system.

We conducted a large scale study of two popular CQAs: the educational Brainly, which spanned two major markets, and the programming-focused Stack Overflow. Our framework

performs well in all data sets. In this study, we had access to all questions generated in Brainly, while the Stack Overflow data contained that which was publicly released. The content in Stack Overflow is more complicated than that in Brainly because Stack Overflow is used widely by professional programmers with specialized knowledge. Because Stack Overflow contains more complex content, our model demonstrated slightly lower accuracy when applied to its data set. However, we find that personal features and community features are more robust in assessing the quality of answers in an online community. The textual features and contextual features are less robust due to the diversity of users and content in these communities. For example, users with an esteemed reputation are more likely to give a good answer than new users who logically have a lower reputation.

For both communities (i.e., Brainly and Stack Overflow), we found that the higher the user's reputation, the better the quality of the answers they provide. Furthermore, all features used in this study can be computed easily, which makes the framework's implementation feasible.

Chapter 5

Struggling Users in Educational CQA

Many initial CQA venues, such as Yahoo! Answers and AnswerBag, were developed to support general-purpose questions. Other CQA platforms focus on more specific topics; for example, Stack Overflow supports issues related to computer programming. Recently, CQA has evolved to support online learning. Some small-scale CQAs were introduced in order to support small groups of university students [8], [110]. These educational CQA sites include Chegg[1], Piazza[2] and Brainly[3]. Brainly, for instance, specializes in online learning for primary and secondary students, helping them interact with each other by asking and answering questions related to school subjects (e.g., English, Mathematics, Biology, Physics, Chemistry, etc.) [23]. Although most CQAs are publicly available and any student is welcome to join for educational purposes, a majority of new CQA users may be fully or partially unaware of these sites' community norms. These norms constitute user behaviors that govern how to appropriately ask and answer a question in order to satisfy an asker's information need. Misunderstanding these norms may affect a user's ability to post appropriate and mutually beneficial answers to a CQA community. Though users who struggle to understand community norms but continue to answer questions differ from online lurkers who most likely consume content without creating it [113], they both need guidance in order to create appropriate answers and participate in healthy question-answering activities. Thus, the main research objective of this Chapter is to understand characteristics of those struggling CQA users and investigate a series of features that indicate their behaviors. To do so, we propose a framework to automatically identify struggling users based on their question-answering activities. We also attempt to investigate the feasibility of detecting

[1]https://chegg.com/study/qa

[2]https://piazza.com/

[3]http://brainly.com

struggling users in the early stages of their problems by using community feedback. Community feedback is the feedback other users provide on answer quality; for example, users vote for best answers or moderators delete bad quality answers. Understanding struggling users and identifying them in the early stages of their CQA activities may help create appropriate user guidelines that demonstrate how students can properly seek and share information within an online education community in order to increase or improve their knowledge.

In the current study for identifying struggling users, we attempt to examine Brainly, one of the largest CQA services specifically targeted towards education. Brainly is an online social learning network for students and educators with millions of active users. It has approximately 60 million monthly unique visitors as of January 2016 and is available in 35 countries, including the United States, Poland, Russia, Turkey, Brazil, France, Indonesia, and more.

5.1 Datasets and Characterization of the Data

Overview: In this study, we use data provided by Brainly.com. This is an online Q&A for students and educators with millions of active users. Here, we use data from two markets: the United States (US) and Poland (PL). US is an emerging market that started in 2013 while PL is a well-established market that began in 2009. Table 3.2 describes some characteristics of these data sets. Brainly requires high quality answers. Thus, moderators delete incorrect answers, incomplete answers or spam posts. A small fraction of highly experienced users are promoted to be the moderators. The moderators not only have experience, but also have a history of significantly contributing to the community. In terms of assuring answer quality, moderators are able to delete wrong, poor and spam answers with additional explanations to answerers; approve appropriate answers; warn and ban users who behave against policies; and participate in the forum to share their experiences with content moderation. There are 207 active moderators in US and 377 active moderators in PL who voluntarily curate content on a daily basis. Moderators also evaluate the majority of new material. The posts in Brainly are divided into three levels (grades): primary, secondary and high school. Brainly also integrates social networking into the platform, as it allows users to make “friends” and exchange ideas.

5.2 Examining Struggling Users

To understand struggling users based on their online activities, we first show that these users exist in the community and display behaviors that deviate from established community norms.

5.2.1 Definition of Struggling Users

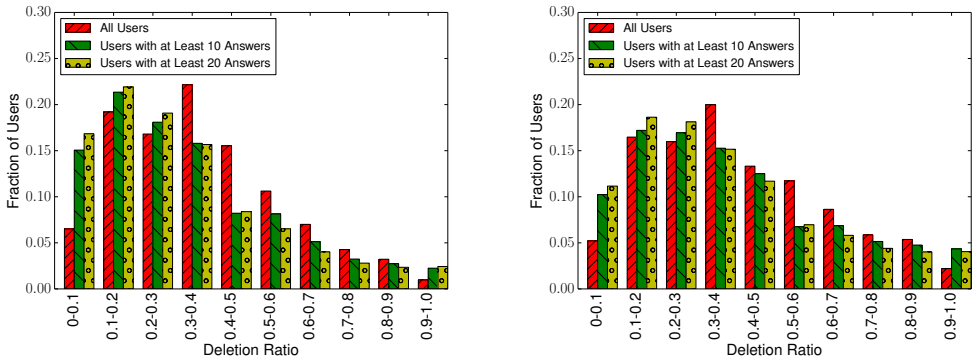
In our work, we are focused on active users who have a limited ability to make meaningful contributions. We define struggling users as those who actively provide low-quality answers to other users' questions. In particular, struggling users generate at least X posts, with a ratio of deletion of at least Y percent. The value depends on the site's requirement. In this study, we work directly with Brainly's data analysis and business intelligence team to determine the threshold. The Brainly data analysis team suggests $X = 10$ and $Y = 0.7$. We also extensively evaluate our method with a wide threshold range to show its efficacy. Our method works well with different thresholds.

5.2.2 Existence of Struggling Users

We examine the proportion of active users who are struggling within the community. We consider struggling users to be those who have the majority of their answering contributions deleted. Figure 5.1 plots the histogram of the deletion rate of different types of users. We plot the histogram of the deletion rates for the general population, as well as users with at least 10 and at least 20 answers. We see that some users have a higher deletion rate.

Figure 5.1 describes the percentage of struggling users in our data sets. We see that there are quite a few users who want to learn and contribute to the community, but cannot provide sufficient answers. These users might need significant help, making it important to detect and assist them. For example, 7.7% and 13.9% of active users in US and PL markets have a deletion rate higher than 70%. This is a large number of users equivalent to 28,364 individuals in PL's market alone. The number of struggling users will increase when the site becomes more popular. As Brainly focuses on education, it attracts a student user population whose learning process is of particular importance.

There are other reasons for deleting the posts such as spam or antisocial elements (e.g.,



(a) USA

(b) PL

Figure 5.1: **Deletion Rate.** Histogram of deletion rate in two markets. Percentage of users with deletion rate higher than 0.7 is the sum of the last three bars (0.7-0.8, 0-8-0.9 and 0.9-1).

abusing, cyber bullying, trolling). These actions are prevalent in news sites and discussion forums. For example, around 5% of Yahoo! Finance posts are antisocial [81]. Since Brainly is an educational CQA, we expect the rate of antisocial posts to be much lower. In order to evaluate our hypothesis, we collect the responses from moderators. Moderators are provided with a tool through which they can report spam and issue warnings. When they delete answers, moderators may also provide their reasoning so that the users can improve their future answers. We also analyzed the feedback from moderators.

Table 5.1 shows that a small percentage of educational CQA users receive a warning for posting inappropriate content. The users receive the warning when they commit certain offenses such as bullying, rudeness, profanity or posting sexual content. For example: *"You should know how do this already, you are bad at this"*. Spam answers, on the other hand, try to divert users to other sites or advertise services such as *"Experienced editor, will write your essay $20/hr, PM me"* or *"Ask on ...com you will get the answer very fast"*. The number of users who receive spam reports is higher than the number of answerers who receive warnings, but both represent only a small fraction of site users.

We also examine the content of the moderators' feedback. A fraction of the deleted posts include moderators' comments; each moderator can explain why they chose to delete a post in their own words. A total of 341,048 responses were collected. We track some keywords

Table 5.1: Percentage of users who received warnings or spam flags. The majority of users do not receive any warnings or spam reports. For example, 99.89% of users do not receive any warnings at all. Furthermore, 97.34% users do not receive any spam reports.

Count	Warning	Spam
0	99.89%	97.35%
1	0.036%	1.76%
2	0.017%	0.38%
3	0.008%	0.14%
4	0.0005%	0.08%
$\geq$5	0.036%	0.24%

that are likely used to describe the spam or antisocial behavior. The list of words describing spam includes *spam, advertisement, unrelated content* and *forbidden content*. The list of words describing the antisocial behavior includes *inappropriate, offensive, abuse, don't be mean* and *bullying*. The lists are not exhaustive, but we expect they can capture the majority of spam and antisocial behavior. Using the lists, we show that 3.2% and 0.8% of answers are spam or antisocial content, respectively. The results correspond with Table 5.1, which stipulates that only a small number of users received warnings or spam flags. Table 4.2 describes the reasons answers may be deleted in educational CQA in more detail.

5.2.3 Social Connections of Struggling Users

We examine whether there is any difference between the social connections of struggling users and those of normal users. To encourage users to exchange information, Brainly includes a social network-like structure in its architecture. We graphically represent these social connections. Each user is a node, and each friendship is an edge in the graph. The number of edges, a user's clustering coefficient and a user's egonet (or friends of friends) represent some features that demonstrate a user's social connections. For example, the clustering coefficient (CC_i) of a node measures how closely a user's neighbors form a clique–or cluster together–and is defined as:

$$CC_i = \frac{\#\ of\ triangles\ connected\ i}{\#\ of\ connected\ triples\ centered\ on\ i} \tag{5.1}$$

The higher clustering coefficient means that this user and their friends form a stronger connection. For example, a fully connected graph has (CC_i) equals to 1, and a star graph has (CC_i) equals to 0. We denote d_i = $|N(i)|$ as the number of friends of users i, while $|N(i)|$ denotes i's set of neighbors. The average degree of neighborhood is defined as

$$\bar{d}_N(i) = \frac{1}{d_i} \times \sum\nolimits_{j \in N_i} d_j \tag{5.2}$$

We also use a node's egonet features. A node's egonet is the subgraph created by the node and its neighbors. Egonet features include its size, number of outgoing edges and number of neighbors.

We use these features because they incorporate four social theories: Social Capital, Structural Hole, Balance and Social Exchange [10]. The capacity of social connection plays an important role in information propagation [42, 60]. Furthermore, these features can be calculated locally which is scalable for large networks.

Table 5.2 summarizes the social connection features of normal users and struggling users. In general, struggling users have fewer social connections compared to normal users. Several reasons could explain this deficiency. For example, it is possible that users who provide low quality answers are less attractive to others. We perform t-tests on these two user groups. The degree and the clustering coefficient are significantly different with $p = 0.01$. A student might want to connect with someone who gives high quality answers. Thus, they are less likely to connect with struggling users. Alternatively, struggling users may not know how to enrich their social connections. In the case that struggling users do not know how to connect with good peer users, a recommendation from a new friend could help.

5.2.4 Time to Answer Question

In the majority of tasks, greater enthusiasm and effort lead to better results. We want to see this happen in educational CQA. We measure the average time that users take to answer a question. We also measure answer length. Table 5.3 shows that, compared to struggling users, normal

Table 5.2: Comparing the social connections between normal and struggling users. The table presents the mean values and standard error mean in the parentheses. Struggling users have fewer social connections than normal users.

Features	USA		PL	
	Normal	**Struggling**	**Normal**	**Struggling**
Degree	9.09 (0.41)	7.5 (1.07)	6.5 (0.08)	4.3 (0.10)
Degree_Adj	80.3 (1.60)	78.6 (1.41)	65.7 (0.3)	60.9 (0.6)
CC	0.13 (.003)	0.14 (.012)	0.04 (.002)	0.03 (.001)
CC_Adj	0.03 (.001)	0.02 (.0005)	0.09 (.002)	0.12 (.004)
Ego	25.8 (2.59)	14.2 (2.37)	14.6 (0.74)	4.5 (0.58)
Ego_Out	618.2 (19.7)	517.5 (29.6)	573.3 (7.9)	311.7 (11.2)

users spend much more time and effort on their answers. Additionally, answers provided by struggling users display a shorter length than those provided by the general population. We perform t-tests on normal and struggling users. The average length of answer and average time spent per answer are significantly different with $p = 0.01$. The statistics show that struggling users should put more effort into crafting high quality answers.

5.2.5 Difference Between Level of Education

Users with varying levels of education may not be equally capable of expressing answers. We examine whether there are differences among users from primary, secondary and high school levels. Figure 5.2 describes the percentage of struggling students based on their respective levels of education. The percentage of struggling primary students is higher than secondary and high school students, likely because primary students lack the ability to clearly express their ideas. Fortunately, when detected and assisted at their primary (young) age, these users can significantly improve their capability.

5.2.6 Activeness of Struggling Users

As we mentioned, we only studied users who generated a certain amount of answers. In particular, we examined users who had at least ten posts. Among these users, we found that struggling users answered 33 questions in the US data set, and 25 questions in the PL data set. These values are lower compared to normal active users, but still high. They suggest that some struggling users want to participate in the community, but their limitations prevent them from making quality contributions.

Table 5.3: Comparing the effort put into creating an answer. The table presents the mean values and standard error mean in the parentheses. Struggling users spent less time and effort when providing an answer compared to normal users.

Features	USA		PL	
	Normal	**Struggling**	**Normal**	**Struggling**
Avg Time (sec)	80.4 (0.8)	42.2 (1.3)	171.4 (0.3)	79.5 (0.4)
Avg Length	119.9 (0.9)	94.8 (1.2)	367.7 (0.6)	321.1 (1.2)

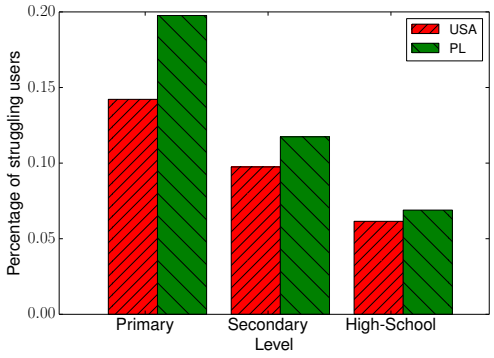


Figure 5.2: Percentage of struggling users with different education levels. Many primary school students struggled to provide good quality answers.

5.2.7 Readability of Answers

To determine the quality of struggling users' written text, we measure its readability based on two popular indexes: automated readability index (ARI), and Flesch reading ease score of answer (FRES) [57]. The ARI measures what grade level should understand the text while the FRES index measures the readability of the document. ARI and FRES indexes are calculated as in Chapter 4

The average FRES value for normal users is 64.1(0.4) while the FRES value of struggling users is 39.1(1.5). This reveals that it is more difficult to read the answers given by struggling users, which explains why some of their posts are deleted. But the distribution of the FRES value varies widely between struggling users. Many struggling users compose easy-to-read posts that provide incorrect information. This observation bolsters the claim that posts are deleted for a diverse array of reasons.

5.3 Detecting Struggling Users without Human Judgment

In the CQA community, a small number of high profile users are authorized to examine answer quality. As the community develops, users generate more content, and this places a growing burden on moderators. In this section, we examine whether it is possible to automatically detect struggling users without any human judgment. Human judgment defines the actions taken by moderators or other users, such as voting for the best answers or bad answers. The formal definition is: *"Given a user and all of their posts, can we predict, without any human judgment, whether this user has struggled with the site?"*.

Finding these users is a classification problem, which includes extracting features, building the classifier and applying the model automatically.

5.3.1 Features Extraction

We identified a list of features based on our observations to extract struggling users. Table 5.4 lists the features used in our study. They are divided into four groups: *Personal Features*, *Community Features*, *Textual Features* and *Contextual Features*. The *Personal Features* include the number of answers given by the users, users' grade level, users' lifetime site presence

and how much time users spend on the site. The *Community Features* include users' social connections, as described in the previous section. The *Textual Features* include the ARI and FRES index, answers' average length, and features to represent the format of users' writing, such as whether their text is well-formatted or whether it contains latex typing. The *Contextual Features* include the time to answer or typing speed.

These features are based on users' activities. We excluded features generated by moderators or other human judgment. The purpose is to examine whether we can automatically evaluate the users.

5.3.2 Classification

Since our problem is a typical binary classification problem, any classification method can work with our framework. In this work, we applied different popular classification algorithms, including logistic regression, support vector machine, decision trees and Random Forest [12]. The explanation of these methods is described in Section 4.2.2.

5.3.3 Experiment Setup

We perform the prediction on a balanced data set of normal and struggling users by under sampling method to create a balanced data set [20]. We perform 10-fold cross validation in all experiments. We measure the efficacy of our method with importance metrics including accuracy, F1-score and Receiver Operating Characteristic (ROC).

Setting different thresholds: According to our definition, the struggling user makes at least X posts, and the ratio of deletion is at least Y percent. The value depends on the site's requirement. Even Brainly's analytic teams suggest $X = 10$ and $Y = 0.7$; we evaluate the efficacy of our method with different thresholds. Below is the set of thresholds tested in our method X and Y to show the efficacy of our method.

- $X = \{5,\ 10,\ 15,\ 20\}$
- $Y = \{0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9\}$

Table 5.4: Lists of features are classified into four groups: Personal, Community, Textual and Contextual. Features' abbreviations are in brackets.

Personal Features
Number of answers given (n_answers)
Grade level of users (u_grade)
Lifetime of users (life_time)
Time spent with the site (spent_time)
Community Features
Number of friends in community (friends_count)
Clustering Coefficient in friendship network (cc)
Average degree of neighborhood (deg_adj)
Average CC of friends (cc_adj)
Size of ego-network of friendship (ego)
Number of outgoing edges in ego-network (ego_out)
Number of neighbors in ego-network (ego_adj)
Textual features
The avg. length of answer (length)
The avg. readability of answer (ari)
The avg. Flesch Reading Ease Score of answer (fres)
The format of answer (well_format)
Using advance math typing: latex (contain_tex)
Contextual features
Duration of time taken to answer (time_to_answer)
Typing speed (typing_speed)

5.3.4 Results

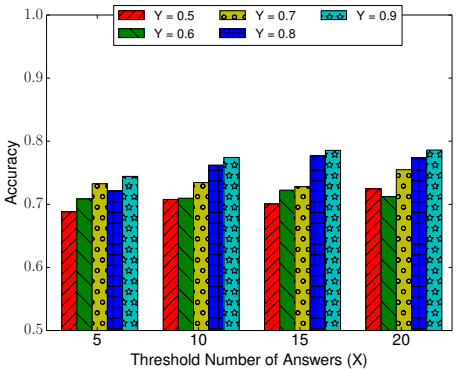
Figure 5.3 plots the accuracy and F1-score when we use different number of posts (X) and deletion rate (Y) thresholds by applying the Random Forest algorithm. These values show that we do not achieve high performance if we only rely on user activity and do not include community feedback. Our method performs better with higher thresholds X and Y. Higher value X indicates that we can observe more activity and gather more information about the users. Higher threshold Y means that the users are very different from the community norm. For example, users with a deletion rate of 0.9 are very extreme users. In such cases, it is easier to detect these extreme users.

Table 5.5 describes the accuracy from different classifiers applied in the current study. In both data sets, Random Forest achieves the highest accuracy. Random Forest achieves the highest performance and is a scalable algorithm. In our experiment with a single machine with 2.2 GHz quad-core, 16 GB of RAM, implemented in Python code, it took less than one millisecond to classify each user. Furthermore, the tree can be built separately and is easily computed to be distributed for a larger data set.

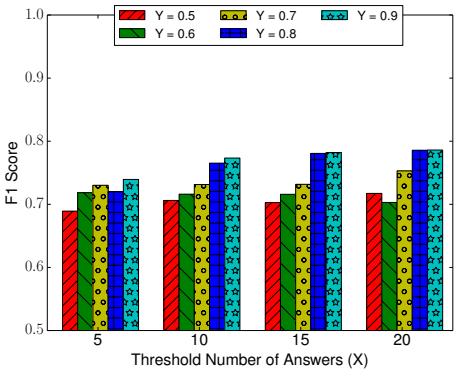
Table 5.5: Comparing different classifiers. We change the thresholds of number of answers and deletion rate as in experiment setting and take the average. The table presents the average accuracy with standard deviation in the parentheses. Random Forest achieves the best accuracy.

Data sets	**LogReg**	**SVM**	**Decision Trees**	**RF**
USA	70.8 (.042)	74.2 (.028)	68.1 (.039)	**73.7 (.031)**
PL	75.1 (.045)	76.3 (.029)	70.7 (.037)	**78.6 (.035)**

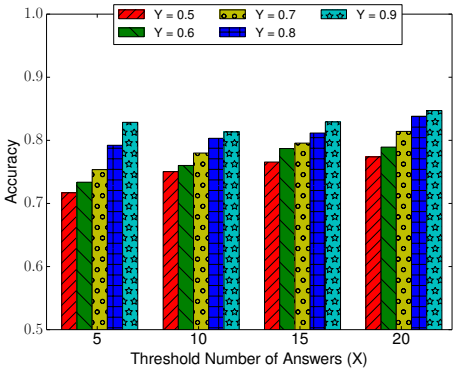
We also measure the Area Under ROC curve to see the trade-off between true positive and false positive rates. The Area Under ROC for US and PL are 0.83 and 0.84, respectively. Again, these values reflect moderate performance value. We need to improve for a real application.



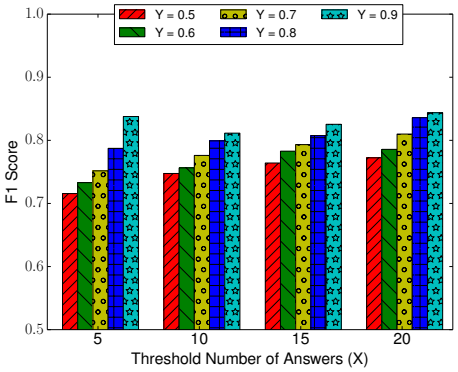
(a1) USA-Accuracy



(a1) USA-F1 Score



(b1) PL- Accuracy



(b2) PL- F1 Score

Figure 5.3: **Accuracy and F1 Score.** The accuracy and F1 Score of detecting struggling users without human judgment. The Figures present the results when applying Random Forest with all Features. The performance is moderate without human judgment.

5.4 Detecting Struggling Users at Their Early Stage with Human Judgment

In the previous Section, we demonstrate the low accuracy involved in predicting whether a user will struggle. In this Section, we will examine if it is possible to predict whether users will struggle at an early stage with community feedback. When a new answer is generated in CQA, other users can judge the content. Some typical human judgment includes moderators deleting bad content or other users voting for good content. The formal definition is *"Given a user with his T posts and human feedback, can we predict whether this user will struggle with the site in the future?"*. Since the moderator judges users' answers, we can get some early

feedback on answer quality. We use the same features as Table 5.4 in Section 5.3. We add new features, including: *del_ratio* represents the percentage of the first T answers that are deleted, and *n_best_answers* represents the number of answers that are selected as the best answer. Other features, such as readability or answer length, are similar to those in the previous section, but we only calculate the features based on the first T answers.

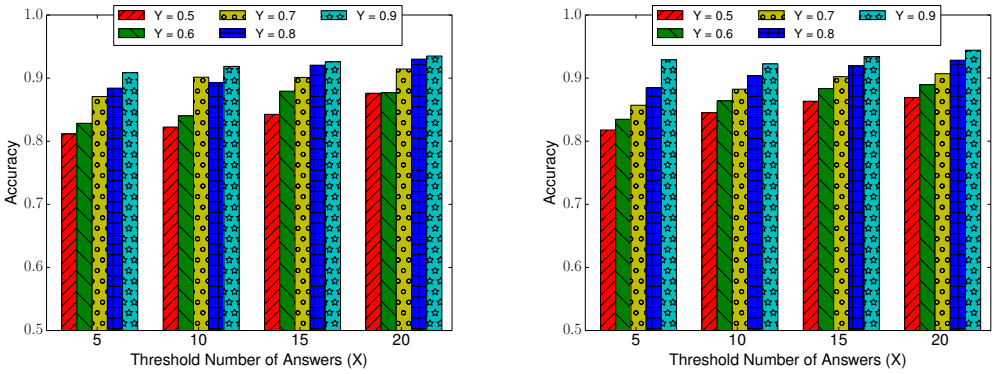
5.4.1 Experiment Set Up

We observe the community behavior and feedback on the first $X/2$ posts of each user and predict whether these users will struggle in the long term. Higher value X means we observe more activities. We also add two new features–*del_ratio* and *n_best_answers*–in the *Community Features* group. Other settings are similar to the previous task. We also try different thresholds including the number of post X and the deletion rate Y to show the efficacy of our method.

5.4.2 Results

Figure 5.4 plots the accuracy of our method. The new features, *n_deletion* and *n_best*, significantly increase the accuracy. For brevity, we do not report the F1-score, which is similar to the accuracy. Results demonstrate that our method achieves high accuracy and F1-score in both US and PL data sets, which are both at 90% for high deletion rate threshold. These key performance metrics predict struggling users more accurately than those that measure all posts without human judgment. Furthermore, two new features significantly increase the accuracy. We will discuss the importance of each feature separately in a later part of this paper.

When using Random Forest, our method's Area Under ROC is 0.93 and 0.94, respectively. These are very high performance metrics reiterated by Accuracy and F1-score. We see that some community feedback helps us significantly increase the system's performance. Furthermore, we only need feedback on a user's few initial posts to make the framework perform well, which could significantly reduce the moderator's workload.



(a) USA (b) PL

Figure 5.4: **Accuracy.** The accuracy of detecting struggling users in the early stage. The Figures present the results when applying Random Forest with all Features. Our method is robust to a different threshold of X and Y. Our method performs better with higher value X and Y. Users with higher deletion rate Y are different from the community norm.

5.4.3 Experiment Results Discussion

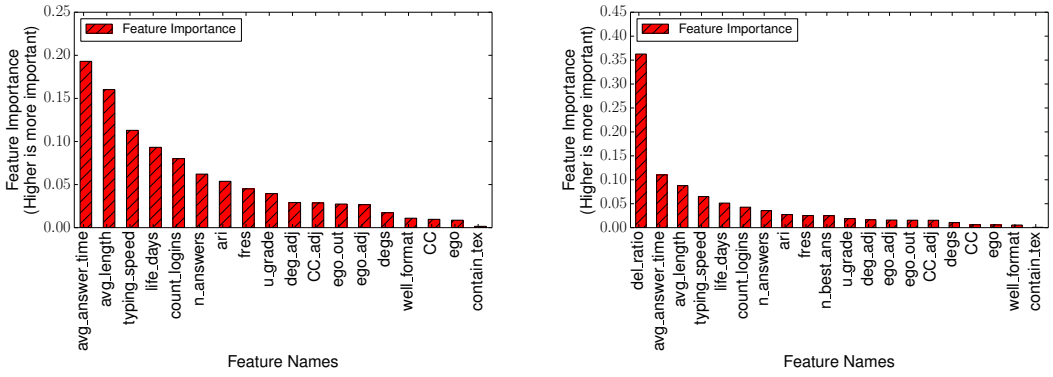
Feature Importance

We measure the importance of each feature in our prediction framework. There are multiple ways to measure the importance of each feature. The general idea is to measure how a particular feature's removal affects the framework's overall accuracy. In the bagging method, we use a permutation test to remove the features and measure the accuracy of out-out-bag (OOB) samples. The important features will decrease the accuracy more than others.

Figure 5.5 measures the importance of each feature in our study. In both scenarios, the average time users spend on an answer and the average length of an answer are important features. When detecting early-stage struggling users, human judgment is vital. In Figure 5.5b, the deletion ratio has a higher value than other features. Thus, we can achieve better performance with some human judgment in our framework.

Feature Selection

We use a small number of features in our study. Thus, feature selection does not improve performance; even some features, such as using latex, are not powerful. Furthermore, the



(a) Without Human Judgment (b) With Human Judgment in the Early Stage

Figure 5.5: **Measure the importance of each feature.** Average answer time and the average length of answers are important features. Furthermore, the deletion ratio in the case of human judgment is a vital feature.

main result presented in this work is achieved via Random Forest. When building each tree of the random forest, the algorithm already randomly selects some features. This suggests that performing feature selection is unnecessary.

Setting the Threshold

In the real system, we might choose a different threshold to identify the users who may need guidance to create appropriate answers. If we want to detect the majority of struggling users, we might need to accept a higher error rate. If we want a very low error rate, we would detect fewer struggling users. The curve in Figure 5.6 plots the True Positive rate against the False Positive rate. We first observe the high Area ROC in both data sets. Secondly, we can detect the majority of struggling users with the small False Positive Rate. For example, if we want to identify 80% of struggling users (True Positive Rate = 0.8), the error rate is very small. In this case, the False Positive Rate is 0.03 and 0.05, respectively. Thus, the framework is quite promising in a real system.

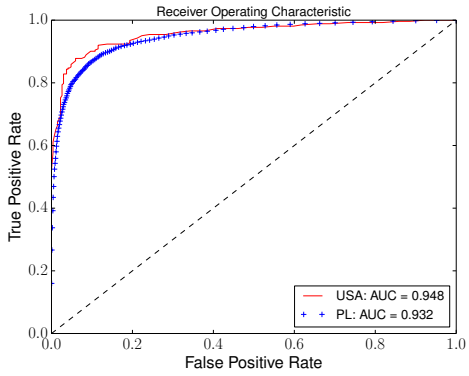


Figure 5.6: Plotting of ROC to select the suitable threshold. We can detect the majority of struggling users with minuscule error rate.

5.5 Discussion

CQA has become an important knowledge sharing avenue. Education is also an important field in which society has traditionally held a vested interest. Our findings show that many users are struggling within online CQA communities that are geared towards students. These struggling users should not be neglected, especially because they hail from educational institutions. With some help from the CQA community, these users could be detected with high accuracy. In the current system, all feedback comes from moderators, who must exert an undue amount of human effort to properly assist struggling users. Furthermore, detecting late-stage struggling users comes with a high cost in terms of those users and their wider online communities; e.g., they could become frustrated and leave the community. Furthermore, it might be too difficult to fix their knowledge gaps if discovered too late. Our work shows that it is possible to find struggling users in their early stage with some community feedback.

Our study has some limitations, which can be addressed in future research. First, we do not discern the reason that users are struggling in the community. Providing users with automatic, concrete feedback could improve their performance and allow them to quickly transcend their shortcomings. Second, it would be highly beneficial to directly interact with struggling users to determine how to best assist them. Observing the effect of such an early interaction would be valuable. In our future work, we hope to resolve these limitations.

5.6 Conclusion

The current work focuses on struggling users in community question answering (CQA). Some students are found to be struggling within education-oriented CQA sites, such as Brainly. These struggling users want to participate in the community, but are unable to produce acceptable answers due to their inexperience and lack of knowledge. While these characteristics may be undesirable outside of an educational context, here they are to be expected and should be addressed to enhance students' learning experience. It follows that flagging these users simply because they are producing *bad* content may not be the best course of action. Instead, if we could use this information to identify such users and recognize that their behavior could indicate a learning-related problem and opportunity, we could help them become better users and better students. In order to identify and eventually assist these individuals, the current study compares their behavior to that of regular active CQA users. Our study reveals that struggling users want to participate in their community, but often do not possess the proper skills or attention to detail that are necessary to make valuable contributions.

Currently, in Brainly and other CQA sites, only moderators judge user performance, and this gives them an unfairly heavy workload. We show that it is possible to detect struggling users without any human judgment with moderate accuracy. However, detecting struggling users without human judgment may come too late, as many users leave the community if they are dissatisfied with their respective performances. Our framework shows that it is more accurate and effective to detect struggling users in their early stage with initial community feedback. Furthermore, using early community feedback can significantly reduce a moderator's workload.

Understanding struggling users' behavior can help us design CQA sites that better help these users. This applies to general CQA as well as educational CQA. However, detecting early-stage struggling users is particularly important in the realm of education since educators can effectively intervene and assist if they know which students are struggling. Detecting struggling users in their early stage ensures that we have enough resources and time to intervene [59]. Although the work in this paper is limited to educational CQA, the framework–including feature extraction, classifier and evaluation–could apply to other general and topical CQA sites.

In future work, we hope to not only detect struggling users, but also to understand the more

specific reasons behind their difficulties. At the moment, the CQA system still needs some input from its moderators, and automatic feedback remains the ultimate goal. To achieve automatic feedback, we might need a more complex taxonomy to analyze user-generated content. Furthermore, finding struggling users is also an important task in other domains such as search or information seeking. If we can detect struggling users in their early stage, suitable action could be taken to help them, including early intervention or content recommendation. The findings from this study give us an approach to detect such users.

Chapter 6

Retrieving Rising Stars in CQA

In this section, we will discuss rising stars in focused CQA. Because of their popularity, CQA sites have become important sources of information for Internet users. However, previous studies have shown that a small group of users contributes the majority of the content on CQA sites. [1], [78]. Furthermore, high quality content can attract traffic in CQA sites. In this work, we propose a new framework to identify the *rising star* in CQA. The rising star is the low profile user who has strong potential to contribute to the community in the future. This is a difficult task because unlike publication networks or social networks, CQA sites do not provide information about participants' peers or collaborators. Discovering the rising star early can help CQA administrators give support or incentives to these potential users. For example, the earlier administrators recognize a rising star's contribution, the sooner they can encourage and cultivate their activity in CQA [6], [7].

6.1 Problem Definition

As prior research indicates, predicting rising stars was previously studied in publication or social network contexts where information about peers and collaborators was readily available. However, in CQA sites, information about peers, collaborators or other relationships is not apparent, making it difficult to predict a user's future performance. In this work, we want to address several questions related to identifying rising stars in a CQA community. The main questions are:

- Given a user with all of their posts in the first T weeks, will they become a top contributor to the site?
- Given a user with his first T posts, will he become a top contributor?

In order to address these questions, we need to understand the list of factors used to identify the users who have the potential to become top contributors. It might be impossible to reveal all factors affecting a user's performance. For example, we might not know that a user changed careers and left a site. However, we try to identify as many features as possible and measure the importance of these features. We also want to see the difference between average users and high profile users. To make predictions regarding rising star behavior, we observe their contributions for a short period or within a few posts to predict whether they will become top contributors in the long term.

6.2 Our Method

Our method includes three main steps. In the first step, we extract features that provide intuitive ideas about users' future performance. Then, we build training and testing sets. The last step is executing an applied classification algorithm to find the rising stars.

6.2.1 Feature Extraction

In order to find the rising stars, we built a user's profile using several features. Table 6.2.1 describes the features used in our experiment. The features are divided into four groups:

- Personal Features: These features are based on users' characteristics. Personal features include number of posts, the topics that the user participated in during the observation period, and the ratio between questions and answers. Users' personal features definitively affect future performance.

- Community Features: The features are based on the community's response to users' posts. These features include the average score that the community gives to the posts and the average comments the posts receive. In general, a higher value indicates better post quality.

- Textual Features: These features represent the content generated by users. Textual features include the average length of the posts and the posts' respective topics.

- Contextual Features: These features measure the consistency of user's posts such as standard deviation (std) gap between the posts, standard deviation scores of the post, standard time gap between recent posts, average duration between the posts, the time gap between two recent posts, and the increasing/decreasing activity of users.

Table 6.1: List of features are classified into four groups of features.

Personal Features
Number of posts
Participated topics
Ratio between # questions and # answers
Community Features
Avg. score
Avg. comments the post received
Avg. Favorite marked
Textual Features
Avg. length of post
Trending of the posts
Contextual Features
Avg. duration between posts
Duration between the two recent posts
Std. time gap between posts
Std. scores between the posts
Std. time gap between posts in 2nd half
More or less active

6.2.2 Building Training Set

In order to build the training data, we extracted features for each user as seen in Table 6.2.1 for different observation durations or the first few posts. A rising star user is defined as a user who will be among the top-K users who have the highest score after 1 year (K = 1% or 10%). Normal users are the rest of the community. Since the majority of CQA users are very inactive,

we applied a sampling technique to make class balanced [20]. The details of the settings are described in Section 6.4.

6.2.3 Classification

Since our framework could use almost any classification model we compared the performance of different models in this study. In particular, we tested with the below classification algorithms [12]. Let $X = x_1, x_2, ..., x_n$ be the list of features. The list of classification algorithms is summarized as:

- Logistic regression (log-reg): Log-reg is a generalized linear model with sigmoid function: $P(Y = 1|X = \frac{1}{1+exp(-b)})$, where $b = w_0 + \sum(w_i.x_i)$, w_i are the inferred parameters from regression.

- Support vector machine (SVM) with Radial basis (RBF) kernel: The RBF kernel is defined as $K(x, x') = exp(-\frac{1}{2}||x - x'||)^2$

- Decision trees: The Tree-based method is a nonlinear model that partitions features into smaller sets and fits a simple model into each subset. The decision tree includes two-stage processes: tree growing and tree pruning. These steps stop when a certain depth is reached or each partition has a fixed number of nodes.

- Random Forest (RF): RF is an average model approach. We use a bag of 100 decision trees. Given a sample set, the RF method randomly samples data and builds a decision tree. This step also selects a random subset of features for each tree. The final outcome is based on the average of these decisions.

- Adaptive Boosting (AdaBoost). This approach uses a list of weak learners to obtain a stronger learner by performing adaptive sampling. The general idea is to put a heavier weight on difficult examples.

This section is organized as follows: data description, characterization of data and features, experimental setup, results and discussion.

6.3 Data Description

Stack Overflow is a focused CQA site that hosts programming-related questions. All questions and answers must be related to programming, which is different from other general CQA sites. Stack Overflow's data-dump was released to the public[1]. Table 6.2 lists the characteristics of the data set used in our experiment. Users in Stack Overflow can engage in different activities such as posting questions, giving answers, voting for the best answer and up-voting or down-voting a post. They can also earn a favorable reputation by posting high quality questions and answers.

Table 6.2: Description about data.

Site	Period	# of Users	# of Posts	Reputations
Stack Overflow	July '08 to Sep '14	3.4 M	21.2 M	Score

6.3.1 Data Pre-processing

A majority of users in CQA are inactive. We consider the life span of a user as the period between their last and their first post in our dataset. Since users join the site at different points in time, the first post is aligned at time $t = 0$. We only select sets of users who remain in the community for at least one year. After pre-processing, we identified a set of of 376K users.

6.3.2 Defining the Rising-Star

Previous research on finding rising stars in academic publishing used successful publications or positions that an author might hold to measure present and future contributions. Sufficient measures of successful authors include indicators such as tenure/academic positions, ACM/IEEE fellows and editors of top journals [31], [66]. These works also used the number of citations to evaluate potential rising stars. The definition of the rising star in our work is explained next.

In our definition, a rising star is a new user who will attain the top earned score in the future. It is not fair to compare a user who joined and stayed in the community for 3 years

[1] https://archive.org/details/stackexchange

and a user who has only been involved in the community for 1 year. Thus, we compare their scores after a fixed period (i.e, 1 year) of their association with the community. Rising stars are classified as the top-K percent of users. Here, the value K is selected depending on our purpose. For example, the top-1% of users are very exceptional contributors in the community and the top-10% of users are very good contributors. In our experiment, we evaluate for top-1% and top-10% users.

6.4 Experimental Setup

Competing methods We compare our method with the following baseline methods:

- RAND: Users are selected randomly.
- ARIMA (Autoregressive integrated moving average): This is the most general time series analysis technique [108].

In ARIMA, we build the time series for each user with the score they earn at time t_i. ARIMA includes three factors in its prediction. These are the (i) Auto-regression: the output depends linearly on the previous value, (ii) Integration and (iii) Moving average: a regression of current value of the series including current value and noise. **Metrics** We pick up a set of rising star users and "normal users". We then predict whether the user will become a star. The rising star user is defined as part of the top-$K(\%)$ users having the highest score after belonging to the community for one year. In our experiment, we test the top-$1(\%)$ and the top-$10(\%)$. Since the number of users in top-$K(\%)$ is much smaller than the community's general population, we use a sampling method to create a balanced data set [20]. Since the class is balanced, the probability that a user becomes a rising star is 50% in the RAND method.

Our method is classification problem and we use cross validation to evaluate the accuracy of our approach. In the default setting, 10-fold cross validation is used.

6.5 Results

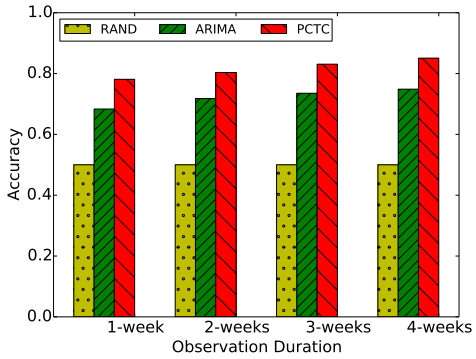
6.5.1 Overview Results

Figure 6.1 shows the correctness when predicting whether a user becomes a rising star. Our method is denoted as *PCTC* which stands for **P**ersonal- **C**ommunity- **T**extual- **C**ontextual. The x-axis is the duration of our observation while the y-axis is the precision of prediction. The observation can be the first T weeks or the first T posts. It shows that our method can predict the long term performance after a short observation period. Further, the longer a user is observed, the better the prediction. Figure 6.1 also implies that it is more difficult to predict whether a user will become a top-10% contributor than which users will comprise the top-1% contributors after observing for T weeks. The reason is the distinction between top-1% and normal users is clearer than the top-10%. The result in Figure 6.1 is achieved by applying *Log-reg* algorithm. Next, we will examine the performance fluctuation when we apply different classification algorithms.

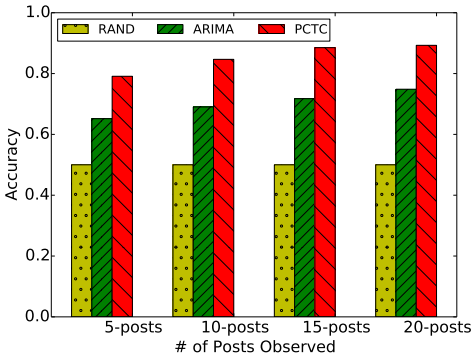
6.5.2 Applying Different Classification Algorithms

We apply different classifications and compare the efficacy. Table 6.3 shows the efficacy of *LogReg, SVM, Decision Trees, RF* and *AdaBoost*. The result in Table 6.3 summarizes the correctness when detecting whether a user will become a top-1% contributor. The decision trees have the lowest accuracy while RF and AdaBoost have the highest accuracy. In general, the accuracy is not affected greatly when applying different classification algorithms. The results show that our method is robust when tested against different classification algorithms.

We also measure $F1$ score, which considers both precision and recall. Precision is the fraction of instances that are relevant, while recall is the fraction of relevant instances that are retrieved. The value of $F1$ is defined as $F1 = 2 * \frac{precision * recall}{precision + recall}$. Figure 6.2 shows that our approaches achieve higher $F1$ scores.

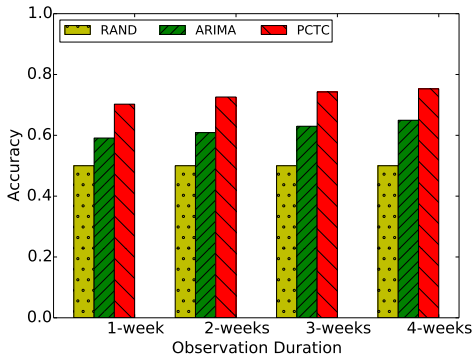


(a) Observe T weeks

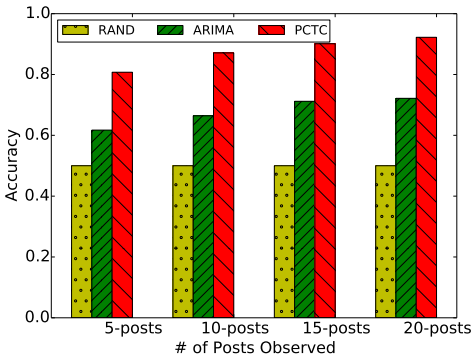


(b) Observe T posts

Prediction whether a user becomes top-1(%) contributors



(c) Observe T weeks



(d) Observe T posts

Prediction whether a user becomes top-10(%) contributors

Figure 6.1: Compare the correctness in predicting whether a user will become a top contributor after one year by observing their performance in the first T weeks (T = 1,2,3,4) or the first T posts (T = 5, 10, 15, 20). The result of *PCTC* is presented when applying *log-reg* classification algorithm.

Table 6.3: Comparing different classifiers. The performance is affected slightly by choosing different classification algorithms.

Observe Duration	LogReg	SVM	Decision Trees	RF	AdaBoost
Observe the first T weeks					
1 week	80.6	80.2	78.5	79.1	79.2
2 weeks	83.4	83.3	80.2	83.3	83.4
3 weeks	84.8	84.9	81.1	84.9	85.9
4 weeks	85.6	85.7	82.3	86.6	86.7
Observe the first T posts					
5 posts	79.1	79.8	78.6	79.9	80.1
10 posts	84.6	84.5	82.3	84.6	84.2
15 posts	88.5	85.5	84.5	88.8	88.7
20 posts	89.6	89.9	87.3	91.1	91.2

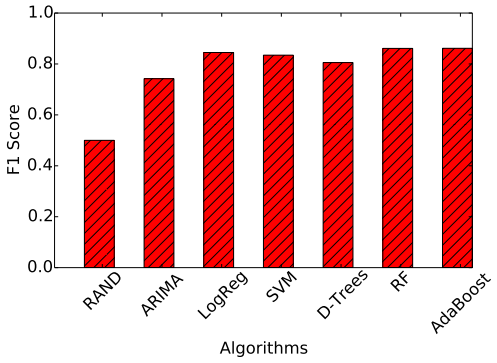
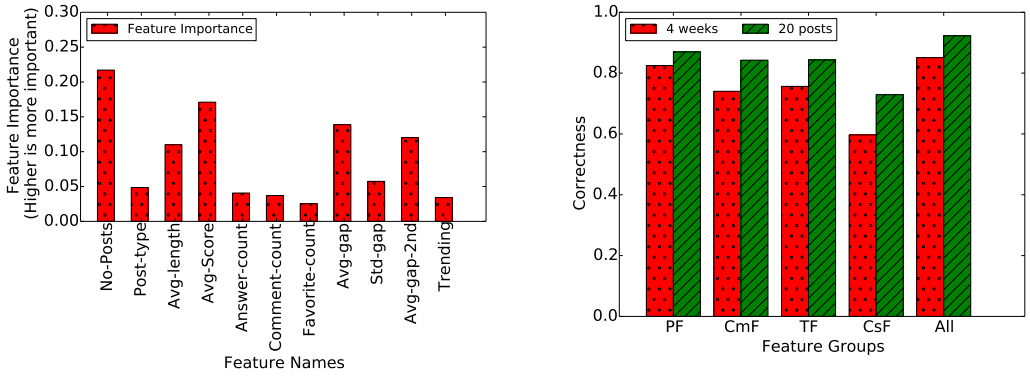


Figure 6.2: Compare the $F1$ score (higher is better).

6.6 Discussion

6.6.1 Feature Importance

We want to quantify the feature importance in classification. A popular method to evaluate the feature importance in Random Forests is measuring the mean square error of prediction [47]. When constructing the decision trees in RF, we use out-of-bag to compute the prediction error. In order to quantify a feature f, we randomly permute f and recompute prediction error. The change is used to evaluate the importance of features in prediction. Figure 6.3a plots the importance of each feature in our prediction framework. The feature importance is normalized so that the sum is equal to 1. The higher value indicates that the feature is more important in prediction. We see that the number of posts, the average gap between the posts, and the average score of the posts are the most important features.



(a) Feature importance (b)Using the different groups of features

Figure 6.3: Evaluate the importance of different features.

6.6.2 Using Different Groups of Features

Another way to quantify the feature importance is to isolate the features and perform classifications separately. There are four different groups of features in our approach. We also measure the efficacy when using each group of features separately. Figures 6.3a depicts the correctness when using different features separately. The higher efficacy indicates that the group of features is more important. We see that the Personal Features group achieves highest accuracy, which is close to the accuracy when using all features. The Consistency Features group performs

slightly higher than RAND in the case where we observe for 4 weeks. The results match with measuring feature importance in Figure 6.3a.

6.6.3 Effect of the Community Size

Stack Overflow is a large focused community. There are other sites that might not achieve the same size as Stack Overflow. It would be useful if our method was scalable to other sites, which motivates us to test whether our method can be applied to smaller communities.

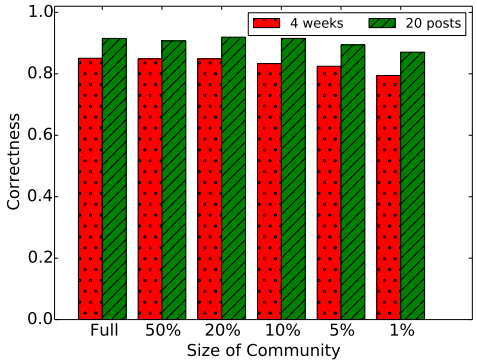


Figure 6.4: Effect of the community size in our prediction. The x-axis is the sample size of the community, ranging from the total Stack Overflow community to 1% of this original community (or 3,761 users). Our approach can maintain the efficacy when the size of the community is much smaller than Stack Overflow.

In order to test our method in smaller communities, we sample users in Stack Overflow. The sample ratios determine the size of the community. We sample with different values from a half to 1% of the community and perform our method. Figure 6.4 shows the correctness when using a smaller community. The x-axis is the size of the community and the y-axis is the correctness. We predict the rising stars after observing for 4 weeks and 20 posts. We show that our approach can maintain its effectiveness even when the community size is smaller. This effectiveness starts to be affected when the sample size is less than 5%. For example, when the community size is only 1%, the precision drops from 85% to 79% when observing for 4 weeks. One percent of this community is quite small; only 3761 users. This result shows that our approach might be applicable for a newly-established or smaller community.

6.7 Conclusion

In this work, we investigated the problem of identifying rising stars in CQA sites. The results show that it is possible to identify rising stars in their early stages. Those potential users contribute the majority of site content to the community and will be active participants in the community's development. A large collection of knowledge and high quality posts could help the community to retain current users and attract new users. Early detection of these users could help administrators provide incentive and support to these rising stars.

We conducted a large scale experiment with Stack Overflow. By extracting different groups of features, we can apply several classification algorithms to predict future performance. The results show that our method can achieve high accuracy and is robust enough to perform well with different classification algorithms. We expect our work would help CQA administrators evaluate the current state of their sites. Then, the site's administrators can make suitable adjustments to keep the community growing.

There are limitations with our work primarily due to the nature of the data. For example, it would be helpful if we could see the effects of incentives on site users. Future work will also investigate in greater detail top performers in CQA. For example, we can examine whether rising stars form a "small community" or if they are separate users within a large community.

Chapter 7

Conclusions and Future Work

In this Chapter, we will discuss our findings and propose some future works.

7.1 Conclusions

CQA is a community where registered users voluntarily create all contents, including questions and answers. Therefore, the contribution of users can easily vary, and it is paramount that sites offer high quality information to retain existing users and attract new users to support their online information seeking behaviors. Thus, identifying different user types is critical for the community to seek and share the right information. At the moment, traditional CQAs depend on human judgments to rate their users' contributions. Unfortunately, using human assessors can have many drawbacks, including subjective (and possibly biased) assessments, seeming difficulty in recruiting such evaluators, and the time it could take for human assessors to go through the ever-increasing amount of users and content in CQA sites. Here, we address these concerns with a new framework for identifying different types of users in the community.

In general, the work in this dissertation extracts different types of CQA users. Finding these specific users provides a framework to extract other types of users in the community. We expect that our findings will help the CQA sites to improve users' experiences and, ultimately, to serve users better. Furthermore, these findings help CQA administrators design their sites, as results demonstrate the need to consider users' strengths and to encourage users to contribute content to a CQA community. We have applied different data mining approaches and the framework leverages the Personal, Community, Textual and Contextual Features to extract particular user types.

- Personal Features: These features are based on users' characteristics. Personal features

include the activity of an answer's owner, such as the number of answers given by the user, the number of questions asked by the user and the rank that user achieved in the community.

- Community Features: These features are based on the response of the community to a user's answers, such as how many thanks they received. Furthermore, we also consider the social connectivity of users in the community.

- Textual Features: These features are based on the text generated by users, such as the content, the topics of answers, the length of answers and the format of answers.

- Contextual Features: These features contain some contextual features of the post such as the similarity between answer and question.

In this dissertation, we study different special types of CQA users. These types of users are defined in different granularity levels or in the context of their respective communities. For example, finding the good or bad answerer should be defined in which context. The context might be a particular question or the entire community atmosphere.

The first component of this work involved finding potential answerers when a new question is posted in CQA. This is a challenging problem due to the diversity of CQA contents and users. But this is an important task because it is applicable to question routing and reducing askers' wait time. Furthermore, finding potential answerers can also increase the chance that the question can be answered. In this study, we compared the similarities between general and focused CQA, including users' activity level and short content. Then, we proposed *QRec*, which integrates different similarity metrics and also users' activity level. The empirical study shows the robustness of that approach. The work also showed that users' activity level and commonalities between content and topics of interest are robust in finding the potential answerer.

Finding potential answerers will ensure that queries receive responses. But answer quality is also important, which motivates us to evaluate the quality of answers in CQA. Receiving high quality answer(s) to a posed CQA query is a critical factor to both user satisfaction and supported learning in these services. This process can be impeded when experts do not answer questions and/or askers do not have the knowledge and skills needed to evaluate the quality of

the answers they receive. Such circumstances may cause learners to construct a faulty knowledge base by applying inaccurate information acquired from online sources. Though site moderators could alleviate this problem by surveying answer quality, their subjective assessments may cause evaluations to be inconsistent. Another potential solution lies in human assessors, though they may also be insufficient due to the large amount of content available on a CQA site. Our study addresses these issues by proposing a framework for automatically assessing answer quality. The frameworks achieve high efficiency in different sites.

Answer quality is very specific in a fine granularity (i.e, in a particular question). A user can be wrong for a certain question, but they might be an expert in another topic. This dissertation also takes an extended look at each user's long term contribution. In the long term, the user can be a high quality contributor or a struggling user. For example, due to insufficient knowledge, lack of experience, and other reasons, users often struggle in producing quality or even appropriate content. This low quality production causes their content to be flagged or deleted, further discouraging them from participating in the CQA process and instigating a vicious cycle of bad users and bad content. In an effort to break this cycle, the work reported here focuses on identifying users whose postings demonstrate a high deletion rate with a presumption that the *bad* content is an indication of a struggling student rather than a malicious user. In this dissertation, experiments are conducted on a large student-oriented online CQA community to understand struggling users' behaviors. A framework is proposed to find these users based solely on their activities. Finally, community feedback (i.e., human judgment) such as moderator evaluation or community votes for good content is used to detect these users in the early stages of their respective struggles. The results show that the human judgment feature identifies early stage struggling users with high accuracy.

Top contributors contrast struggling users. In CQA, there is typically a small fraction of users who provide high-quality posts and earn a very high reputation status from the community. These top contributors are critical to the community since they drive the development of the site and attract traffic from Internet users. Identifying these individuals could be highly valuable, but this is not an easy task. The results show that it is possible to identify rising stars in their early stages. The results also show that those potential users contribute the majority of site content to the community and will be active participants in the community's development.

A large collection of knowledge and high-quality posts could help the community to retain current users and attract new users. Early detection of these users could help administrators provide incentive and support to these rising stars.

In conclusion, the work demonstrates the possibility of finding different special types of users. We studied potential answerers, good answerers, struggling users and rising stars. These users represent different segments of CQA participants with different granularity levels. Understanding users and detecting them in the early stage provides us an overview of the community's health. In popular CQA services including Stack Overflow, Yahoo! Answers and Brainly, a majority the feedback comes from human moderators who must exert an undue amount of human effort to properly maintain the content and assist users. In these services, early detection has many benefits. For example, detecting late-stage struggling users comes with a high cost regarding those users and their wider online community; the users could be frustrated and leave the community. Furthermore, it might be too difficult to fix their knowledge gaps if they are discovered too late. Our work shows that it is possible to find different types of users in their early stage. We expect the framework can be applied to find other types of users in CQA.

7.2 Future Work

The work presented in this dissertation shows the high efficacy in identifying users in CQA, but there are some limitations to this study, primarily due to the nature of the data. For example, it would be helpful if we could see the effects of incentives on site users. Future work will also investigate in greater detail top performers in CQA. For example, we can examine whether rising stars form a "small community" or if they are separate users within a large community. Furthermore, understanding the latent features of different types of users, such as why they are more active or more effective in generating content than others, is very important.

Additionally, even though this work concentrates on CQA, finding various types of users is not only useful in CQA, but also crucial to other online communities. Another potential direction involves extending our approach to other types of communities such as online forums or social networks. In most communities, we can observe user actions. Different online communities contain varying levels of rich user information. For example, a user in a social network

typically interacts with peers they know, while users in forums and CQAs interact with many strangers; it follows that social networks demonstrate stronger social influence between users. We hope that that our findings in this dissertation will help with understanding users in other types of communities.

In future work, we hope to find different types of users in finer granularity levels. For example, we would not only detect struggling users, but also understand the more specific reasons behind their difficulties. At the moment, the CQA system still needs some input from its moderators, and automatic feedback remains the ultimate goal. To achieve automatic feedback, we might need a more complex taxonomy to analyze user-generated content.

References

[1] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman. Knowledge sharing and yahoo answers: Everyone knows something. In *International Conference on World Wide Web (WWW)*, pages 665–674, 2008.

[2] M. Ageev, D. Lagun, and E. Agichtein. The answer is at your fingertips: Improving passage retrieval for web question answering with search behavior data. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1011–1021, 2013.

[3] E. Aktolga, J. Allan, and D. A. Smith. Passage reranking for question answering using syntactic structures and answer types. In *European Conference on Information Retrieval (ECIR)*, pages 617–628, 2011.

[4] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: A case study of stack overflow. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 850–858, 2012.

[5] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Effects of user similarity in social media. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 703–712, 2012.

[6] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Steering user behavior with badges. In *International Conference on World Wide Web (WWW)*, pages 95–106, 2013.

[7] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Engaging with massive online courses. In *International Conference on World Wide Web (WWW)*, pages 687–698, 2014.

[8] C. Aritajati and N. H. Narayanan. Facilitating students' collaboration and learning in a question and answer system. In *ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW) Companion*, pages 101–106, 2013.

[9] P. Arora, D. Ganguly, and G. J. F. Jones. The good, the bad and their kins: Identifying questions with negative scores in stackoverflow. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1232–1239, 2015.

[10] M. Berlingerio, D. Koutra, T. Eliassi-Rad, and C. Faloutsos. Network similarity via multiple social theories. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1439–1440, 2013.

[11] J. Bian, Y. Liu, E. Agichtein, and H. Zha. Finding the right facts in the crowd: Factoid question answering over social media. In *International Conference on World Wide Web (WWW)*, pages 467–476, 2008.

[12] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[14] Brainly. `https://brainly.com`.

[15] L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, 2001.

[16] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *International Conference on World Wide Web (WWW)*, pages 107–117, 1998.

[17] D. Carmel, A. Mejer, Y. Pinter, and I. Szpektor. Improving term weighting for community question answering search using syntactic analysis. In *ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 351–360, 2014.

[18] D. Chakrabarti and C. Faloutsos. *Graph Mining: Laws, Tools, and Case Studies*. Synthesis Lectures on Data Mining and Knowledge Discovery. Morgan & Claypool Publishers, 2012.

[19] S. Chaturvedi, D. Goldwasser, and H. Daumé III. Predicting instructor's intervention in mooc forums. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1501–1511, 2014.

[20] N. V. Chawla. Data mining for imbalanced datasets: An overview. In *The Data Mining and Knowledge Discovery Handbook*, pages 853–867. Springer, 2005.

[21] Chegg. `https://www.chegg.com`.

[22] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec. Antisocial behavior in online discussion communities. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 140–149, 2015.

[23] E. Choi, M. Borkowski, J. Zakoian, K. Sagan, K. Scholla, C. Ponti, M. Labedz, and M. Bielski. Utilizing content moderators to investigate critical factors for assessing the quality of answers on brainly, social learning Q&A platform for students: a pilot study. In *Annual Meeting of the Association for Information Science and Technology (ASIST)*, pages 69:1–69:4, 2015.

[24] E. Choi, V. Kitzie, and C. Shah. Developing a typology of online Q&A models and recommending the right model for each question type. In *Annual Meeting of the Association for Information Science and Technology (ASIST)*, pages 1–4, 2012.

[25] E. Choi and C. Shah. User motivation for asking a question in online Q&A services. *Journal of American Society for Information Science and Technology (JASIST)*, pages 1182–1197, 2016.

[26] R. A. Cole. *Issues in Web-based pedagogy: A critical primer*. Greenwood Press, 2000.

[27] M. Coletto, C. Lucchese, S. Orlando, and R. Perego. Polarized user and topic tracking in twitter. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 945–948, 2016.

[28] M. D. Conover, J. Ratkiewicz, M. Francisco, A. Flammini, and F. Menczer. Political polarization on twitter. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 89–96, 2011.

[29] D. H. Dalip, H. Lima, M. A. Gonçalves, M. Cristo, and P. Calado. Quality assessment of collaborative content with minimal information. In *ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 201–210, 2014.

[30] C. Danescu-Niculescu-Mizil, R. West, D. Jurafsky, J. Leskovec, and C. Potts. No country for old members: User lifecycle and linguistic change in online communities. In *International Conference on World Wide Web (WWW)*, pages 307–318, 2013.

[31] A. Daud, R. Abbasi, and F. Muhammad. Finding rising stars in social networks. In *DASFAA (1)*, Lecture Notes in Computer Science, pages 13–24. Springer, 2013.

[32] G. Dror, Y. Koren, Y. Maarek, and I. Szpektor. I want to answer; who has a question?: Yahoo! answers recommender system. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1109–1117, 2011.

[33] G. Dror, Y. Maarek, A. Mejer, and I. Szpektor. From query to question in one click: Suggesting synthetic questions to searchers. In *International Conference on World Wide Web (WWW)*, pages 391–402, 2013.

[34] G. Dror, Y. Maarek, and I. Szpektor. Will my question be answered? predicting "question answerability" in community question-answering sites. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, volume 8190, pages 499–514, 2013.

[35] G. Dror, D. Pelleg, O. Rokhlenko, and I. Szpektor. Churn prediction in new users of yahoo! answers. In *WWW '12 Companion*, pages 829–834, 2012.

[36] S. Dumais, R. Jeffries, D. M. Russell, D. Tang, and J. Teevan. Understanding user behavior through log data and analysis. In *Ways of Knowing in HCI*, pages 349–372. Springer New York, 2014.

[37] H. Fang, F. Wu, Z. Zhao, X. Duan, Y. Zhuang, and M. Ester. Community-based question answering via heterogeneous social network learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 122–128, 2016.

[38] G. Ganu and A. Marian. Personalizing forum search using multidimensional random walks. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 140–149, 2014.

[39] K. Garimella, G. De Francisci Morales, A. Gionis, and M. Mathioudakis. Quantifying controversy in social media. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 33–42, 2016.

[40] S. D. Gollapalli, P. Mitra, and C. L. Giles. Ranking experts using author-document-topic graphs. In *ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 87–96, 2013.

[41] W. Gong, E.-P. Lim, and F. Zhu. Characterizing silent users in social media communities. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 140–149, 2015.

[42] A. Goyal, F. Bonchi, L. V. S. Lakshmanan, and S. Venkatasubramanian. On minimizing budget and time in influence propagation over social networks. *Social Netw. Analys. Mining*, 3(2):179–192, 2013.

[43] I. Guy and D. Pelleg. The factoid queries collection. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 717–720, 2016.

[44] A. Halavais, K. H. Kwon, S. Havener, and J. Striker. Badges of friendship: Social influence and badge acquisition on Stack Overflow. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1607–1615, 2014.

[45] X. Han, L. Wang, N. Crespi, S. Park, and A. Cuevas. Alike people, alike interests? inferring interest similarity in online social networks. *Decis. Support Syst.*, 69:92–106, 2015.

[46] F. M. Harper, D. Raban, S. Rafaeli, and J. A. Konstan. Predictors of answer quality in online Q&A sites. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 865–874, 2008.

[47] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics, 2009.

[48] J. He, J. Bailey, B. I. P. Rubinstein, and R. Zhang. Identifying at-risk students in massive open online courses. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1749–1755, 2015.

[49] L. Hirschman and R. Gaizauskas. Natural language question answering: The view from here. *Nat. Lang. Eng.*, 7(4):275–300, 2001.

[50] M. D. Hoffman, D. M. Blei, and F. R. Bach. Online learning for latent dirichlet allocation. In *Conference on Neural Information Processing Systems (NIPS)*, pages 856–864, 2010.

[51] M. Iyyer, J. L. Boyd-Graber, L. M. B. Claudino, R. Socher, and H. D. III. A neural network for factoid question answering over paragraphs. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 633–644, 2014.

[52] J. Jin, Y. Li, X. Zhong, and L. Zhai. Why users contribute knowledge to online communities. *Inf. Manage.*, 52(7):840–849, 2015.

[53] P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 919–922, 2007.

[54] S. R. Kairam, D. J. Wang, and J. Leskovec. The life and death of online groups: Predicting group growth and longevity. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 673–682, 2012.

[55] I. Kayes, N. Kourtellis, D. Quercia, A. Iamnitchi, and F. Bonchi. Cultures in community question answering. In *ACM Conference on Hypertext and Social Media (HT)*, pages 175–184, 2015.

[56] I. Kayes, N. Kourtellis, D. Quercia, A. Iamnitchi, and F. Bonchi. The social world of content abusers in community question answering. In *International Conference on World Wide Web (WWW)*, pages 570–580, 2015.

[57] J. P. Kincaid, R. P. Fishburne, R. L. Rogers, and B. S. Chissom. Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. Technical report, Naval Air Station Memphis, 1975.

[58] J. Kiseleva, K. Williams, J. Jiang, A. Hassan Awadallah, A. C. Crook, I. Zitouni, and T. Anastasakos. Understanding user satisfaction with intelligent assistants. In *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*, pages 121–130, 2016.

[59] H. Lakkaraju, E. Aguiar, C. Shan, D. Miller, N. Bhanpuri, R. Ghani, and K. L. Addison. A machine learning framework to identify students at risk of adverse academic outcomes. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1909–1918, 2015.

[60] L. T. Le, T. Eliassi-Rad, and H. Tong. MET: A fast algorithm for minimizing propagation in large graphs with small eigen-gaps. In *SIAM International Conference on Data Mining (SDM)*, pages 694–702, 2015.

[61] L. T. Le and C. Shah. Retrieving rising stars in focused community question-answering. In *Asian Conference on Intelligent Information and Database Systems (ACIIDS)*, pages 25–36, 2016.

[62] L. T. Le, C. Shah, and E. Choi. Evaluating the quality of educational answers in community question-answering. In *ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 25–36, 2016.

[63] L. T. Le, C. Shah, and E. Choi. Bad users or bad content? breaking the vicious cycle by finding struggling students in community question-answering. In *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*, 2017.

[64] A. Y. Levy, A. Rajaraman, and J. J. Ordille. Querying heterogeneous information sources using source descriptions. In *VLDB*, pages 251–262, 1996.

[65] B. Li, T. Jin, M. R. Lyu, I. King, and B. Mak. Analyzing and predicting question quality in community question answering services. In *International Conference on World Wide Web (WWW)*, pages 775–782, 2012.

[66] X.-L. Li, C. S. Foo, K. L. Tew, and S.-K. Ng. Searching for rising stars in bibliography networks. In *International Conference on Database Systems for Advanced Applications (DASFAA)*, pages 288–292, 2009.

[67] J. Lin. The web as a resource for question answering: Perspectives and challenges. In *Proceedings of the 3rd International Conference on Language Resource and Evaluation*, 2002.

[68] Q. Liu, E. Agichtein, G. Dror, E. Gabrilovich, Y. Maarek, D. Pelleg, and I. Szpektor. Predicting web searcher satisfaction with existing community-based answers. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 415–424, 2011.

[69] Q. Liu, E. Agichtein, G. Dror, Y. Maarek, and I. Szpektor. When web search fails, searchers become askers: Understanding the transition. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 801–810, 2012.

[70] Y. Liu, S. Li, Y. Cao, C.-Y. Lin, D. Han, and Y. Yu. Understanding and summarizing answers in community-based question answering services. In *International Conference on Computational Linguistics (COLING)*, pages 497–504, 2008.

[71] Y. Lu, J. Warren, C. Jermaine, S. Chaudhuri, and S. Rixner. Grading the graders: Motivating peer graders in a mooc. In *International Conference on World Wide Web (WWW)*, pages 680–690, 2015.

[72] L. Luo, F. Wang, M. X. Zhou, Y. Pan, and H. Chen. Who have got answers?: Growing the pool of answerers in a smart enterprise social qa system. In *International Conference on Intelligent User Interfaces (IUI)*, pages 7–16, 2014.

[73] C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[74] A. Mao, E. Kamar, and E. Horvitz. Why stop now? predicting worker engagement in online crowdsourcing. In *AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*, pages 103–111, 2013.

[75] Y. Mehdad and J. R. Tetreault. Do characters abuse more than words? In *Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pages 299–303, 2016.

[76] E. Momeni, K. Tao, B. Haslhofer, and G.-J. Houben. Identification of useful user comments in social media: A case study on flickr commons. In *ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 1–10, 2013.

[77] M. R. Morris, J. Teevan, and K. Panovich. What do people ask their social networks, and why?: A survey study of status message Q&A behavior. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1739–1748, 2010.

[78] D. Movshovitz-Attias, Y. Movshovitz-Attias, P. Steenkiste, and C. Faloutsos. Analysis of the reputation system and user contributions on a question answering website: Stackoverflow. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 886–893, 2013.

[79] K. K. Nam, M. S. Ackerman, and L. A. Adamic. Questions in, knowledge in?: A study of naver's question answering community. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 779–788, 2009.

[80] B. Ngonmang, E. Viennet, and M. Tchuente. Churn prediction in a real online social network using local community analysis. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 282–288, 2012.

[81] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang. Abusive language detection in online user content. In *International Conference on World Wide Web (WWW)*, pages 145–153, 2016.

[82] M. Noer. *One Man, One Computer, 10 Million Students: How Khan Academy Is Reinventing Education*. Forbes, 2013.

[83] H. L. O'Brien, L. Freund, and R. Kopak. Investigating the role of user engagement in digital reading environments. In *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*, pages 71–80, 2016.

[84] R. J. Oentaryo, E.-P. Lim, D. Lo, F. Zhu, and P. K. Prasetyo. Collective churn prediction in social network. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 210–214, 2012.

[85] S. Overflow. `http://stackoverflow.com`.

[86] A. Pal, S. Chang, and J. A. Konstan. Evolution of experts in question answering communities. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 274–281, 2012.

[87] A. Pal, R. Farzan, J. A. Konstan, and R. E. Kraut. Early detection of potential experts in question answering communities. In *ACM Conference on User Modeling, Adaptation and Personalization (UMAP)*, pages 231–242, 2011.

[88] A. Pal, F. Wang, M. X. Zhou, J. Nichols, and B. A. Smith. Question routing to user communities. In *ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 2357–2362, 2013.

[89] K. Panovich, R. Miller, and D. Karger. Tie strength in question & answer on social network sites. In *ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW)*, pages 1057–1066, 2012.

[90] D. Pelleg. When the crowd is not enough: Improving user experience with social media through automatic quality analysis. In *ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW)*, pages tba–tba, 2016.

[91] D. Pelleg, E. Yom-Tov, and Y. Maarek. Can you believe an anonymous contributor? on truthfulness in yahoo! answers. In *ASE/IEEE International Conference on Social Computing and International Conference on Privacy, Security, Risk and Trust (SOCIALCOM-PASSAT)*, pages 411–420, 2012.

[92] Piazza. `https://piazza.com/`.

[93] J. Preece, B. Nonnecke, and D. Andrews. The top five reasons for lurking: improving community experiences for everyone. *Computers in Human Behavior*, 20(2):201 – 223, 2004.

[94] PRIUSchat. `http://priuschat.com/`.

[95] J. S. Pudipeddi, L. Akoglu, and H. Tong. User churn in focused question answering sites: Characterizations and prediction. In *WWW Companion '14*, pages 469–474, 2014.

[96] J. Qiu, J. Tang, T. X. Liu, J. Gong, C. Zhang, Q. Zhang, and Y. Xue. Modeling and predicting learning behavior in moocs. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 93–102, 2016.

[97] X. Qiu and X. Huang. Convolutional neural tensor network architecture for community-based question answering. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1305–1311, 2015.

[98] M. Qu, G. Qiu, X. He, C. Zhang, H. Wu, J. Bu, and C. Chen. Probabilistic question recommendation for question answering communities. In *International Conference on World Wide Web (WWW)*, pages 1229–1230, 2009.

[99] M. Rath, L. T. Le, and C. Shah. Discerning the quality of questions in educational q&a using textual features. In *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*, 2017.

[100] S. Ravi, B. Pang, V. Rastogi, and R. Kumar. Great question! question quality in community Q&A. In *International AAAI Conference on Web and Social Media (ICWSM)*, pages 426–435. The AAAI Press, 2014.

[101] Y. Richter, E. Yom-Tov, and N. Slonim. Predicting customer churn in mobile networks through analysis of social groups. In *SIAM International Conference on Data Mining (SDM)*, pages 732–741, 2010.

[102] C. Ross, K. Nilsen, and P. Dewdney. *Conducting the reference interview: A how-to-do-it manual for librarians.* New York: NealSchuman, 2002.

[103] C. Shah and V. Kitzie. Social q&a and virtual reference - comparing apples and oranges with the help of experts and users. *Journal of American Society for Information Science and Technology (JASIST)*, 63:2020–2036, 2012.

[104] C. Shah, S. Oh, and J. S. Oh. Research agenda for social Q&A. *Library & Information Science Research*, 31(4):205 – 209, 2009.

[105] C. Shah and J. Pomerantz. Evaluating and predicting answer quality in community qa. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 411–418, 2010.

[106] C. Shah, M. Radford, L. Connaway, E. Choi, and V. Kitzie. How much change do you get from 40$? analyzing and addressing failed questions on social Q&A. In *Annual Meeting of the Association for Information Science and Technology (ASIST)*, pages 1–10, 2012.

[107] Y. Shoji, S. Fujita, A. Tajima, and K. Tanaka. Who stays longer in community qa media? - user behavior analysis in cqa. In *SocInfo*, volume 9471 of *Lecture Notes in Computer Science*. Springer, 2015.

[108] R. H. Shumway and D. S. Stoffer. *Time Series Analysis and Its Applications: With R Examples*. Springer Texts in Statistics, 2011.

[109] A. Singla and A. Krause. Incentives for privacy tradeoff in community sensing. In *AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*, pages 165–173, 2013.

[110] I. Srba and M. Bielikova. Askalot: Community question answering as a means for knowledge sharing in an educational organization. In *ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW) Companion*, pages 179–182, 2015.

[111] I. Srba, M. Grznar, and M. Bielikova. Utilizing non-qa data to improve questions routing for users with low qa activity in cqa. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 129–136, 2015.

[112] N. Sun, P. P.-L. Rau, and L. Ma. Understanding lurkers in online communities: A literature review. *Comput. Hum. Behav.*, 38:110–117, Sept. 2014.

[113] R.-P. M. L. Sun, L. Understanding lurkers in online communities: a literature review. *Computers in Human Behavior*, 38:110–117, 2014.

[114] M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to rank answers on large online qa collections. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 719–727, 2008.

[115] J. Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.

[116] M. A. Suryanto, E. P. Lim, A. Sun, and R. H. L. Chiang. Quality-aware collaborative question answering: Methods and evaluation. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 142–151, 2009.

[117] V. Szymczak. Github and stackoverflow in technical recruitment. `http://sourcingrecruitment.info/2015/05/github-and-stackoverflow-in-technical-recruitment/`. Accessed: 2016-12-30.

[118] C. H. Tan, E. Agichtein, P. Ipeirotis, and E. Gabrilovich. Trust, but verify: Predicting contribution quality for knowledge base construction and curation. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 553–562, 2014.

[119] P. A. Tess. The role of social media in higher education classes (real and virtual) - a literature review. *Computers in Human Behavior*, 29:A60–A68, 2013.

[120] P. Thomas, H. O'Brien, and T. Rowlands. Measuring engagement with online forms. In *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*, pages 325–328, 2016.

[121] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *IEEE International Conference on Data Mining (ICDM)*, pages 613–622, 2006.

[122] G. Tsur, Y. Pinter, I. Szpektor, and D. Carmel. Identifying web queries with question intent. In *International Conference on World Wide Web (WWW)*, pages 783–793, 2016.

[123] E. M. Voorhees. The trec-8 question answering track report. In *In Proceedings of TREC-8*, pages 77–82, 1999.

[124] G. Wang, K. Gill, M. Mohanlal, H. Zheng, and B. Y. Zhao. Wisdom in the social crowd: An analysis of quora. In *International Conference on World Wide Web (WWW)*, pages 1341–1352, 2013.

[125] G. Wang, K. Gill, M. Mohanlal, H. Zheng, and B. Y. Zhao. Wisdom in the social crowd: An analysis of quora. In *International Conference on World Wide Web (WWW)*, pages 1341–1352, 2013.

[126] X.-J. Wang, X. Tu, D. Feng, and L. Zhang. Ranking community answers by modeling question-answer relationships via analogical reasoning. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 179–186, 2009.

[127] R. W. White and M. Richardson. Effects of expertise differences in synchronous social Q&A. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1055–1056, 2012.

[128] R. W. White, M. Richardson, and Y. Liu. Effects of community size and contact rate in synchronous social Q&A. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 2837–2846, 2011.

[129] H. Wu, W. Wu, M. Zhou, E. Chen, L. Duan, and H.-Y. Shum. Improving search relevance for short queries in community question answering. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 43–52, 2014.

[130] X. Xue, J. Jeon, and W. B. Croft. Retrieval models for question and answer archives. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 475–482, 2008.

[131] Yahoo. Yahoo! answers. `https://answers.yahoo.com`.

[132] Z. Yan and J. Zhou. Optimal answerer ranking for new questions in community question answering. *Information Processing & Management*, 51(1):163–178, 2015.

[133] L. Yang, S. Bao, Q. Lin, X. Wu, D. Han, Z. Su, and Y. Yu. Analyzing and predicting not-answered questions in community-based question answering services. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1273–1278, 2011.

[134] P. Yang and H. Fang. Opinion-based user profile modeling for contextual suggestions. In *ACM International Conference on the Theory of Information Retrieval (ICTIR)*, pages 80–83, 2013.

[135] S. Yang. Information seeking as problem-solving using a qualitative approach to uncover the novice learners' information-seeking process in a perseus hypertext system. *Library and Information Science Research*, 19(1):71–92, 1997.

[136] Y. Yao, H. Tong, F. Xu, and J. Lu. Predicting long-term impact of cqa posts: A comprehensive viewpoint. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1496–1505, 2014.

[137] S. Yarosh, T. Matthews, and M. Zhou. Asking the right person: Supporting expertise selection in the enterprise. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 2247–2256, 2012.

[138] X. Yin, J. . Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, pages 796–808, 2008.

[139] J. Zhang, X. Kong, R. J. Luo, Y. Chang, and P. S. Yu. Ncr: A scalable network-based approach to co-ranking in question-and-answer sites. In *ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 709–718, 2014.

[140] J. Zhang, J. Tang, and J.-Z. Li. Expert finding in a social network. In *International Conference on Database Systems for Advanced Applications (DASFAA)*, pages 1066–1069. Springer, 2007.

[141] Z.-M. Zhou, M. Lan, Z.-Y. Niu, and Y. Lu. Exploiting user profile information for answer ranking in cqa. In *International Conference on World Wide Web (WWW) Companion*, pages 767–774, 2012.

[142] Y. Zhu, E. Zhong, S. J. Pan, X. Wang, M. Zhou, and Q. Yang. Predicting user activity level in social networks. In *ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 159–168, 2013.