Good or Bad Question?
A Study of Programming CQA in Stack Overflow

Gary Kuen Wen Hao, Zhou Shu & Jemmy Irawan
Nanyang Technological University
Wee Kim Wee School of Communication & Information
31 Nanyang Link, Singapore 637718
Email : {gkuen001; zhou0226; jemmy001}@e.ntu.edu.sg

ABSTRACT
Community question answering (CQA) services accumulate large amount of knowledge through the voluntary services of the community across the globe. In fact, CQA services gained much popularity recently compared to other internet services in obtaining and exchanging information. Stack Overflow is an example of such a service that targets programmers and software developers. In general, most questions in Stack Overflow are usually ended up with an answer accepted by the askers. However, we found that the number of unanswered or ignored questions has increased significantly in the past few years. Understanding the factors that contribute to questions being answered as well as questions remain ignored can help information seekers to improve the quality of their questions and increase their chances of getting answers from the community in Stack Overflow. In our study, we have identified the relevant features that will help in the predicting of the quality of questions and validate the reliability of the features using some of the state-of-the-art classification algorithms. The features revealed in this study is significant in the sense that they can help Stack Overflow to improve their existing CQA service in terms of user satisfaction in obtaining quality answers from their questions.

Author Keywords
Community question answering (CQA), Stack Overflow, classification.

INTRODUCTION

Background
The rapid growth of the Internet has changed the way people communicate. More people are increasingly relying on their distributed peer communities for information, advice, and expertise. In fact, internet services like public discussion forums, community-built encyclopedias (e.g., Wikipedia) and community question answering (CQA) sites (e.g., Yahoo! Answers, Answerbag[1], Quora, etc) are used globally by people for exchanging information and learning from each other. Of all these Internet services, CQA has recently gained much popularity among the general public for information seeking and knowledge sharing. In reality, it is estimated that the number of questions answered on CQA sites surpass the number of questions answered by library reference services (Shah & Pomerantz, 2010).

CQA is defined as community services which allow users to post questions for other users to answer or respond (Li et. al. 2008). It aims to provide community-based (Chen et. al, 2012) knowledge creation services (Anderson et. al, 2012). Compared to the previous keywords-based querying, CQA websites has been recognized as more precise (Shah & Pomerantz, 2010) and trustworthy (Chen et. al., 2013). The reason is as opposed to traditional search engine such as Google, CQA provide a valuable alternative solution to information seeking for a number of reasons (Cai & Chakravarthy, 2011). Firstly, the answers given by users with actual knowledge or experience are in no doubt more fruitful and foolproof for the questioner. CQA tends to be more efficient and useful to the answerers to get the information regarding particular questions asked by them rather than to go through a list of

[1] http://www.answerbag.com/

related documents. Secondly, CQA presents a centralized communication environment where it is possible to facilitate multiples answers from different perspectives and also allow the questioners to interact with the answerers for further clarifications. Thirdly, CQA provides an enticement for people to demonstrate their expertise and get recognized globally. This surely attracts the users to answer in order to be recognized as experienced ones.

Lately, CQA websites specifically in the programming context are gaining momentum among programmers and software developers (Barua et al., 2012). This is because today's software engineering field involves a wide range of technologies, tools, programming languages, and platforms. Even for experienced developers, it can be difficult for them to be proficient and to keep up with the rapid growth of all the different technologies. CQA can therefore provide them with the environment for seeking help and advice from their distributed peers about technical difficulties that they face. In addition, software development has been described as knowledge intensive because it requires different areas of expertise and capabilities. For this reason, CQA can also play a role as knowledge management in this field (Treude et al., 2011).

One of the most popular programming CQA website currently, Stack Overflow[2], managed to capture compelling technical knowledge sharing among software developers globally (Mamykina et. al, 2011). From the perspective of practices, Stack Overflow is a place for people to post their programming or IT related questions for other people to provide answers (Nasehi et. al., 2012). After that, the questioners can select the most helpful answer and the question is considered solved. Registered members in Stack Overflow can vote on questions and also answers. The positive and negative votes show the helpfulness and quality of a question and answer. There is a reputation system in Stack Overflow, the members can increase their reputation in the website by participating in various activities like posting questions, answering, voting, posting comments, etc. With better reputations, they will obtain extra capabilities such as editing question/answers and closing a topic. From the perspective of theories, recently, many researches have been conducted to investigate this popular CQA website. For instance, the different online behaviors on Stack Overflow caused by user gender (Vasilescu et. al., 2012), the approaches of assigning new questions to Stack Overflow experts so as to facilitate the management process (Riahi et. al., 2012), the influence factors that could lead to the question deleted issues on Stack Overflow (Correa and Sureka, 2014), and the trends of Stack Overflow (Barua et al., 2012). All these practices and researches provided the authors with more comprehensive thinking on CQA websites especially Stack Overflow.

Objectives

This study aims to examine the predictors of ignored questions in a CQA service specifically those posted in Stack Overflow. Thus, there are two main objectives in this study. The first objective is the identification of the crucial factors or features that affect the quality of the questions. The quality of the questions is divided into two classes: good and bad questions. In our context, good questions are defined as the questions that are solved by the community members. Contrarily, bad questions are defined as the ignored questions, which specifically mean the questions without any answers or comments from the community for at least three months. The second objective is the establishment of classification models to classify between good and bad questions, in order to verify the validity of the identified features in predicting the quality of questions in CQA.

Justification

The high number of unanswered or ignored questions is a concern for CQA websites. Yang et. al. (2011) found out that of 76251 questions in Yahoo! Answers, among which 10424 (about 13.67%) questions get no answers. Similarly, in Quora, another rapidly growing CQA website, 20% of questions remain unanswered even though almost all questions were viewed at least by 1 user (Wang et. al, 2013). On the other hand, we crawled data from Stack Overflow to investigate the number of ignored questions for the last few years to generate Figure 1 and Figure 2. In our context, an ignored question is defined as question that is without any answers or comments from the community for at least three months. From Figure 1, there was a rapid growth in the number of ignored questions in Stack Overflow from the beginning of Stack Overflow in 2008 to 2012. If the ignored questions are not managed properly, the number of ignored questions would keep increasing exponentially resulted from the

[2] http://stackoverflow.com/

increasing users registering to Stack Overflow in the future. Similarly, Figure 2 shows that the percentage of ignored question also increase over the years. To control the growth, it would be important for Stack Overflow to understand the factors that contribute to low quality questions and introduce a mechanism that helps reduce the number of ignored questions. Thereby, the mechanism can lead to a better management of the CQA site and to help users increase their chances of getting answers from their questions.

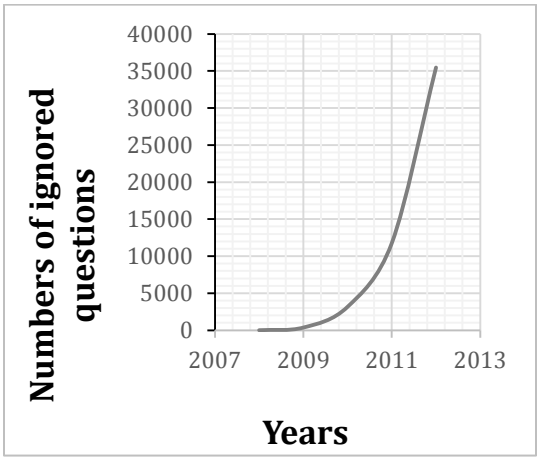


Figure 1. Number of ignored questions each year.

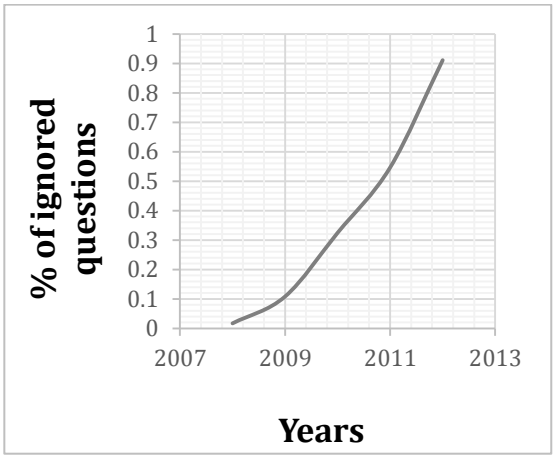


Figure 2. Percentage of ignored questions each year

(Source of Figure 1 and Figure 2: crawled data from Stack Overflow)

Most of the previous study focused on investigation of high quality answers in CQA, with the anticipation to improve the quality of user generated contents of CQA websites in the future. However, not many researchers focused on the quality of the questions. Low quality questions often lead to bad answers whereas high quality questions usually receive good answers (Agichtein et. al., 2008). Thus, we would argue that the quality of the questions is one of the main influences to the low quality answers in CQA websites. *"Questions are never indiscreet, answers sometimes are."* a quote by Oscar Wilde, suggested that a question might be asked poorly or incorrectly if there are no desirable answers to the question. Therefore in our study, we will attempt to identify the factors that influence the selection of high and low quality questions in a CQA service specifically questions related to programming in Stack Overflow.

For further justification, identifying the quality of question in CQA service is significant for a number of other reasons. Firstly, high quality questions can actually promote the development of the CQA community. This is because high quality questions will attract more authentic expert users to share their knowledge and experience, eventually further enhancing the knowledge base of the CQA community. The users can therefore gain more value from the CQA community. Secondly, low quality questions hinder the service quality of CQA websites. For example, low quality questions, such as posting advertisements as questions, are eyesores for the users and will highly reduce the overall user experience. Lastly, question quality can aid question searching in CQA. The accuracy of question retrieval and question recommendation in CQA services can be improved if the system can take advantage of the high quality questions to understand their meaning from the natural language.

RELATED WORKS

CQA websites provides community-based (Chen et. al, 2012) knowledge creation services (Anderson et. al, 2012), which has been recognized as more precise (Shah & Pomerantz, 2010) and trustworthy (Chen et. al., 2013) than previous keywords-based querying. In recent years, researchers have started to analyze CQA websites on different perspectives, mainly on the users, user generated contents and the features of CQA.

Research on Users

Firstly, studies have been done on the users in CQA websites. These researches have mainly focused on user intentions, gender issue, user attention, user activity, and user typology. Based on different user intentions, Chen et. al. (2012) have proposed the taxonomy that categorizes the questions on CQA websites into three types:

objective questions (posed by users so as to get factual knowledge), subjective questions (posed by users so as to get opinions or experiences), and social questions (posed by users so as to establish social interactions). Vasilescu et. al. (2012) conducted a quantitative study of online communities and found out that users' gender differences leading to different online behavior. This study have claimed that men contributes vast majority contents, and earned more reputation by participating than women on Stack Overflow. Wang et. al. (2013) have suggested that users are connected by three networks, namely topics, social networking and questions on CQA websites, and have pointed out that heterogeneity is the key success factor in directing users' attention and activities into small popular questions. Dror et. al. (2012) have studied the churn prediction in new users of CQA websites by crawling users' personal information, activity data and the data on social interaction. Similarly, Lai and Kao (2012) have claimed that both users' expertise and also users' activities are crucial criteria for question routine. Anderson et. al. (2012) have found out that users can be divided into answers and voters, and have studied how users' community activities that affect the answers in CQA websites.

Research on User-generated Contents

Secondly, studies have been done on the user-generated contents in CQA websites, researches had mainly focused on questions, answers and the pairs of question-answer. Many research were conducted on answers. Blooma et. al. (2012), Nasehi et. al. (2012), Cai and Chakravarthy (2011), Shah and Pomerantz (2010) had analyzed the quality of answers in CQA and the factors that will affect their quality. Besides quality issue, the categorizing issue of answers had also emerging. Miao et al. (2012) had studied on new answer categories that had not been concluded in the existing category hierarchy, and proposed two modeling methods to solve this problem.

On the other hand, research on questions were emerging. Suzuki et. al. (2011) and Li et. al. (2008) had analyzed the questions in CQA, Singh and Visweswariah (2011) had studied the question classifications of CQA website, Quan and Wenyin (2012) analyzed the attractiveness of questions. Riahi et. al. (2012) had focused on assigning new questions to experts who got the most suitable expertise in Stack Overflow. Similarly, Xuan et. al. (2013) had also proposed topic cluster for experts narrowing down the domain expertise, and combined with latent links for experts ranking in specific topic, so as to solve the expert find problem. Furthermore, Barua et al. (2012) had studied the trends and topics on CQA website specifically in Stack Overflow, Correa and Sureka (2014) had focused on the deleted questions on Stack Overflow, and they proposed that all four categories (users' profile, community generated, question content and syntactical style) would lead to the question-deleted issue. Although this model was one of the earliest prediction model that focused on poor quality question of Stack Overflow, but the accuracy was only reported as 66% and it seemed a lot of further work to be conducted in this field.

Apart from that, the research on similarity analysis of question-answer pairs in CQA had also been popular. Wang et. al. (2009) had proposed the ranking algorism to find the best answer by utilizing the relationships between questions and answers. Pera and Ng (2011) had introduced a CQA refinement system based on analyzing 4,000,000 pair of questions-answers crawdad from Yahoo! However, Chen et. al. (2013) figured out that focus solely on textual similarities would fail in detecting commercial answers in CQA websites, hence, they had proposed an effective detecting algorism that involves more context indexes (answering pattern and answerer's reputation track).

Research on Functionality of CQA Websites

Thirdly, studies had been done on the features in CQA websites. Both functionalities and non-functionalities of CQA websites had been explored. In terms of functionalities of CQA websites, Mamykina et. al. (2011) had studied the design features of successful CQA websites. Whereas Danescu et. al. (2009) and Hong et. al. (2009) studied the voting and reputation system of CQA, Souza et. al. (2013) and Li et. al. (2011) studied the routing of questions to appropriate answerers in CQA. In order to improve users' searching accuracy and efficiency, Tang et. al. (2010) developed an answer summary system to replace existing lists of similar queries. Zhang et. al. (2012) had suggested a mobile multimedia functionality that could be used in CQA websites, and the authors had claimed that supported by identifying mobile screenshots, matching these instances and retrieving candidate answers, the question asking process could be effective facilitated.

In terms of the non-functionalities of CQA websites, the characteristic of communities of CQA websites had attracted intense research attentions. Zhang et. al. (2010) had proposed a unified framework to examine the typology of community structures in CQA websites. Shachaf (2010) had regarded the collaborations in CQA

websites as an enabler of better services. However, Fichman (2011) had defined the popularity as higher user participation, and had reported the unclear correlations between website popularity and answer quality. Chua and Balkunje (2012) had measured the system usability issues and provided general recommendations of improving usability features for CQA websites.

Based on the extensive literature review, it is suggested that the past works on CQA had been quite fruitful, and many research had already contributed a lot of insights to this topic. On the one hand, all achievements had provided the authors with adequate inspirations. On the other hand, since the management of CQA website is a dynamic problem that contains multiple-roles, it is found that previous research mainly focused on one side of the problem. As we argued previously, the importance of questions have attracted emerging attentions. However, past works had mainly concentrated on one side of questions, for instance, the classification or attractiveness of questions, or the approaches to assigning questions to experts, or some specific type of questions like deleted ones and etc. But the distinctions between different types of questions had not been clearly articulated. In another word, the possible influence factors of questions had been identified, but the comparisons between how these factors perform in different types of questions and the causes of different performances have not been taken care of. Therefore, we would like to investigate the comparisons between good quality questions and bad quality questions so as to contribute more from a new perspective.

METHODOLOGY

Stack Overflow's data is used for our study due to the popularity of Stack Overflow among programmers globally. Stack Overflow is one of the largest community question answering site for questions related to programming. In addition, they are rich in metadata such as user's reputation which are suitable to be used for our study. Stack Overflow had been used in many existing studies (e.g.: Nasehi et. al., 2012; Asaduzzaman et. al., 2013).

Dataset

This study will focus on Java solely in Stack Overflow, to keep balance of both macro and micro research perspectives. From the macro perspective, Java is the most popular programming language on average. According to Tiobe[3] Programming Community Index (Figure 3), a measure of popularity of programming languages, Java has been the lingua franca and long-term favored in the past decade and, without no doubt, gets the highest average ratings in popularity from 2002 to 2014. From the micro perspective, based on data collected from Stack Overflow (Table 1), Java-related questions are the most tagged and the most ignored, with 456748 and 8463 respectively, which are extremely superior compared to other top two popular languages, C (103461 and 557) and C++ (217951 and 1896).

[3] http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html

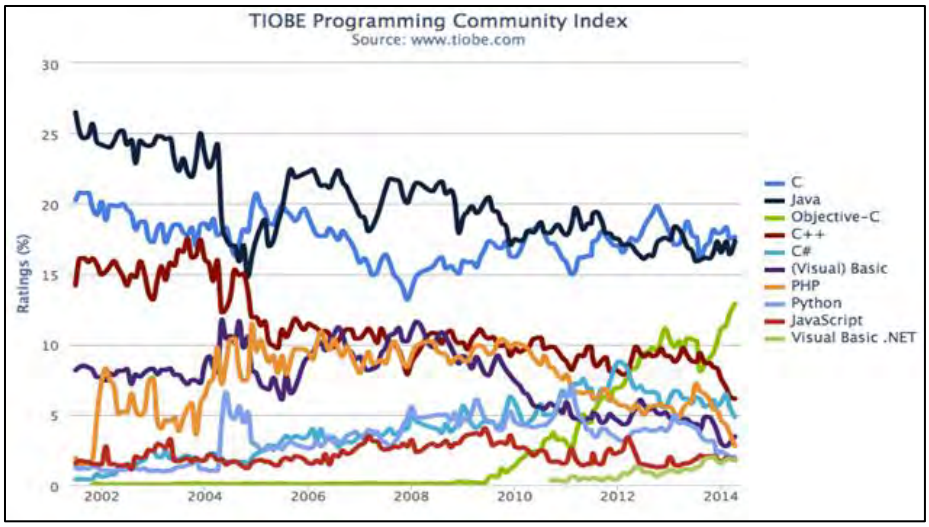


Figure 3. Tiobe Programming Community Index (Source: Tiobe, April 2014).

Popular language	Number of tagged questions	Number of ignored questions
Java	456748	8463
C	103461	557
C++	217951	1896

Table 1. Number of questions on SO (Source: crawled data from Stack Overflow).

As well as that, this study will focus on questions on Stack Overflow. Previous researches pay more attention on either high-quality answers (Blooma et al., 2012) or the pairs of question-answer (Nasehi et. al., 2012). It is estimated that investigation on the questions in CQA service are seriously underestimated or even neglected. However, questions actually generate great value for CQA. In order to compensate for this research gap, our study aims to focus on questions in CQA service. Data from Stack Overflow are considered suitable for examining the predictors of questions for three causes. Firstly, practically, the numbers of both solved and ignored questions of Stack Overflow are soaring continually from 2008 to 2012. In order to deal with this serious challenge in Stack Overflow, a practical solution is highly expected. Secondly is the richness data available on Stack Overflow. Large number of data from both kinds of questions could be served to make comparisons. Hence, the causes behind could be mined out. Thirdly is the accessibility. Free services of Stack Overflow (unlike Expert Exchange[4]) enable data retrieval at a low cost, so valuable previous work could also be referred (e.g. Nasehi et. al., 2012).

Data from Stack Overflow are collected during February 2014. We utilized the Data Explorer[5] service provided by Stack Exchange to obtain the data. SQL queries are be written and executed in Data Explorer to crawl the required data from the database of Stack Overflow. For our study, we crawled the data with the tag 'Java' of solved and ignored questions starting from year 2008. After that, disproportionate stratified sampling are be used to sample the data from the two categories. A total of 3000 data are be sampled, of which 50% are the good questions and another 50% are the bad questions. Good questions are defined as the questions that are solved by

[4] http://www.experts-exchange.com/

[5] https://data.stackexchange.com/

the community members. On the other hand, bad questions are defined as the ignored questions, which specifically mean the questions without any answers or comments from the community for at least three months. The terms, good questions and bad questions, are exempted to any subjective judgments and only confined in the scope of this research.

FEATURE IDENTIFICATION

This section represents a literature review to identify the features used for our study. In summary, we have identified 24 features that are used for classifying the quality of the questions in CQA from the literatures. The features are divided into two main categories: metadata features and content features; and four sub-categories. Table 1 shows a summary of all the features identified from the literatures. The detailed explanations of the features are given in the sub-sections below.

Metadata Features

Metadata is data that describe other data, which summarizes basic information about data. The main purpose of metadata is to facilitate in the identification and retrieval of relevant information, in which also known as resource discovery. In the context of question in CQA, metadata can refer to the category of the question, the time and date of creation, the creator of the question and other available information that describe the question. Additionally, there are also metadata that describe about the users such as their personal information and reputation in the CQA site.

In classification, metadata features are defined as features that can be obtained directly from the metadata to be used as predictors, without the need to perform further complicated extractions on the features. In classifying the quality of questions in CQA, the metadata that describe the questions and the askers of the questions are commonly used (Yang et. al., 2011; Bian et. al., 2009). Only the information on the metadata available at question time are used because the objective of the study is to predict the quality of the question, when the questions are first posted by the users. The metadata features in our study were divided into two main two sub-categories: the asker's user profile and questions. Below are the descriptions of the identified features.

Asker's User Profile

This metadata information is related to the users that were asking the questions, which may include their personal information as well as the information obtained from the activities performed by the users. The metadata features about the askers used are reputation of the askers, days since the first day of joining data, upvotes, downvotes, the ratio of upvotes to downvotes, number of questions asked, number of answers posted and lastly the ratio of answer posted to questions asked. One of the reason we choose to include the metadata features from the askers' user profile is that they give information about the background and involvement of the askers in a particular CQA site. Experienced users are more familiar with the CQA services as well as the community while new users probably wonder what to ask and how to ask (Yang et. al., 2011). In addition, users are also able to assess the quality of the questions and answers posted by a particular user by giving positive or negative votes, the information can be utilized to find out the overall quality of the information provided by the askers in the past. Furthermore, these set of features about the askers are used as predictors of high quality questions because they are readily available at the time when the questions were posted. A description of these metadata features from the askers' user profile is given below:

1) **Reputation**. The stature of the askers in the CQA site (Agichtein et. al, 2009; Asaduzzaman et. al., 2013; Liu et. al., 2008; Li et. al, 2012; Bian et. al., 2009).
2) **Days since join**. The total number of days since the askers joined the CQA site (Agichtein et. al., 2008; Liu et. al., 2008).
3) **Upvotes**. The total number of positive votes of the questions and answers from the askers by other users in the past (Asaduzzaman et. al., 2013; Agichtein et. al., 2008).
4) **Downvotes**. The total number negative votes of the questions and answers from the askers by other users in the past (Bian et. al., 2009).

5) **Upvotes/Downvotes**. The ratio between the total number positive votes and the negative votes of the questions and answers from the askers by other users in the past (Bian et. al., 2009).

6) **Questions asked**. The total number of questions the askers asked in the past (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008).

7) **Answers posted**. The total number of answers the askers posted in the past (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008).

8) **Answers posted/questions asked**. The ratio between the total number of question asked and the total number of answers posted (Liu et. al., 2008)

Questions

In CQA services, metadata that describe about the questions posted can also be found. Two important metadata features on the questions that determine whether the questions will be answered are the time and day the questions were being posted. The chances of the questions quickly being answered are largely affected by the number of users that are active at the moment when the questions are posted. Therefore, we use the features time and the day in the week the questions were posted as predictors in our study. However, due to the periodic nature of time and day, we make a modification to the features during the feature extraction process so that they are more appropriate to be used in classification, each feature is converted to two features to represent the feature; the novel approach is explained in the feature extraction section of this paper.

1) **Time**. The time of the day the questions were posted by the askers (Yang et. al., 2011; Bian et. al., 2009; Liu et. al., 2008).

2) **Day**. The day in the week the questions were posted by the askers (Yang et. al., 2011; Bian et. al., 2009).

Content Features

In the context of questions in CQA, content features are defined as the metrics that track intrinsic and extrinsic content quality of the questions. Content features are divided into two sub-categories, textual features and content appraisal. Specifically, textual features are defined as the intrinsic quality metrics related to tangible features of the text in the questions. Textual features can be automatically generated using a sequence of procedures and are usually done using computer programs. On the other hand, content appraisal features are defined as extrinsic quality metrics related to intangible features of the content in the questions. Content appraisal features require human expert judgment in ranking the content of the questions Intuitively, these types of features are used in the classification of the quality of questions in CQA given that the content of the questions are primarily textual in nature.

Textual Features

From the literatures, we have identified five textual features that are useful in classification of the quality of questions in CQA. The features are: tags, title length, question length, code snippet and whether the titles starts with a question word. Tags is the number of tags (or category) of the questions, a question may be assigned to one or more tags, however in our study, the questions contain the 'Java' tag alone or along with other tag(s) Title and question length is the number of word count computed from the textual content of the title and questions. A description of these textual features from the content of the questions is given below:

1) **Tags**. The number of tags or categories the questions are associated to (Agichtein et. al., 2008).

2) **Title length**. The number of words in the title of the questions (Yang et. al., 2011; Bian et. al., 2009; Liu et. al., 2008).

3) **Question length**. The number of words in the content of the questions (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008).

4) **Code snippet**. Whether the content of the questions contain code snippet(s) (Asaduzzaman et. al., 2013).

5) **Wh word**. Whether the title of the questions starts with wh question words: "what", "when", "where", which", "who", "whom", "whose", "why", and "how" (Li et. al, 2012; Liu et. al., 2008).

Content appraisal

The inclusion of content appraisal features in our study motivated by Asaduzzaman et. al. (2013), where they incorporate a series of extrinsic features to analyze the unanswered questions in CQA. In addition, there are research on the way people make judgment on the quality of information in CQA (e.g.: Bovee et. al., 2003) and the theoretical quality measures created are utilized in our study. The content appraisal features used in our study to classify the quality of questions are: completeness, complexity, language error, presentation, politeness, and question subjectivity. A description of the six content appraisal features identified from the literatures is given below:

1) **Completeness**. (Asaduzzaman et. al., 2013)
2) **Complexity**. The degree of difficulty of the questions (Asaduzzaman et. al., 2013; Agichtein et. al., 2008).
3) **Language error**. The degree of grammar and typing errors made by the askers in the content of the questions (Agichtein et. al., 2008).
4) **Presentation**. The format, clarity and writing style used to present the questions to ensure readability (Asaduzzaman et. al., 2013)
5) **Politeness**. The courtesy and sincerity shown in content of the questions by the askers (Yang et. al., 2011; Asaduzzaman et. al., 2013).
6) **Subjectivity**. Subjective questions are those asking for answers with private states, e.g. personal interests, opinions, judgments; on the other hand, objective questions require authoritative information (Yang et. al., 2011; Li et. al., 2008).

Category	Sub-category	Features
Metadata features	Asker's user profile	Reputation (Agichtein et. al, 2009; Asaduzzaman et. al., 2013; Liu et. al., 2008; Li et. al, 2012; Bian et. al., 2009)
		Days since join (Agichtein et. al., 2008; Liu et. al., 2008)
		Upvotes (Asaduzzaman et. al., 2013; Agichtein et. al., 2008)
		Downvotes (Bian et. al., 2009)
		Upvotes/Downvotes (Bian et. al., 2009)
		Questions asked (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008)
		Answers posted (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008)
		Answers posted /questions asked (Liu et. al., 2008)
	Question	Time (Yang et. al., 2011; Bian et. al., 2009; Liu et. al., 2008)
		Day (Yang et. al., 2011; Bian et. al., 2009)
Content features	Textual features	Tags (Agichtein et. al., 2008)
		Title length (Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008)
		Question length (Yang et. al., 2011; Li et. al, 2012; Asaduzzaman et. al., 2013; Bian et. al., 2009; Liu et. al., 2008)
		Code snippet (Asaduzzaman et. al., 2013)
		Wh word (Li et. al, 2012; Liu et. al., 2008)
	Content appraisal	Completeness (Asaduzzaman et. al., 2013)
		Complexity (Asaduzzaman et. al., 2013; Agichtein et. al., 2008)
		Language error (Agichtein et. al., 2008)
		Presentation (Asaduzzaman et. al., 2013)
		Politeness (Yang et. al., 2011; Asaduzzaman et. al., 2013)
		Subjectivity (Yang et. al., 2011; Li et. al., 2008)

Table 2. Features for identifying high-quality questions.

DATA PREPROCESSING

Data preprocessing is a series of steps to be performed on the raw dataset before constructing a classification model. Data preprocessing is important for three reasons. First, many raw data from the real world cannot be applied straight into the classification algorithms without any proper transforming done on the data, for example: the raw textual content found in our dataset. Second, raw data may contain many irrelevant information and those information need to be filtered out during preprocessing. Third, the value range of the features need to be normalized and standardize. This is because the wide variation of value range in the raw data is likely to affect the significance of the representations of each features during the classification process. In our study, the data preprocessing steps include: feature extraction, feature normalization and feature selection. Each of the data preprocessing steps are explained in the sub-sections below.

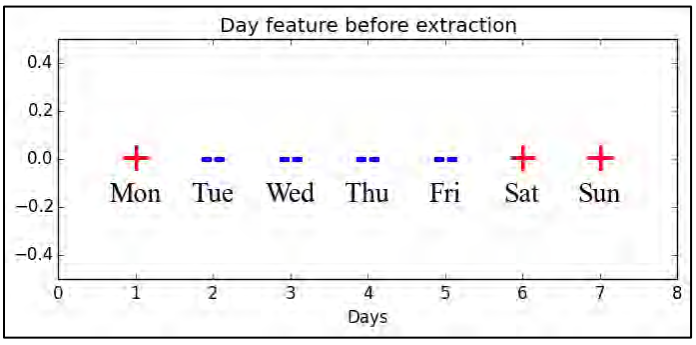
Feature Extraction

Feature extraction is an essential pre-processing step in data mining of transforming the raw dataset into a representation set of meaningful features, which involves simplifying the large and redundant raw dataset in order to generate a significantly smaller set of features to accurate describe the raw dataset. The feature extraction process for our dataset is automated by executing custom-developed Python scripts. In our study, the three sub-categories of features that requires automated extraction are: the asker's user profile, metadata features of the questions as well as the textual features from the content of the questions. In addition to that, content appraisal features from the content of the questions require manual extraction, which is the expert judgment from human.

The features from the users' asker profile (reputation, upvotes, downvotes, questions asked, answers posted) that we crawl is incorrect as most of them are crawled a long period after the questions were asked. The features do not represent the experience of the askers in Stack Overflow at the particular time when the questions were asked. Therefore, for the purpose of out study, we estimate and convert the values of these features with the assumption that the values increase linearly with time. The formula used are shown in equation (1) where x is the value of the features and x' is the estimation of the value at the time when the questions were asked.

$$x' = \frac{(x)}{(current\_date\ - join\_date)} * (question\_date\ - join\_date) \quad (1)$$

The metadata features of the questions are the time of the day and day of the week the questions were posted by the askers. However, due to the periodic nature of the time and day, it is impossible to use them directly for classification without further extraction. This is because there are no meaning in the ranking of time and day value. Most classification algorithms define hypotheses of distinguishing different classes with a linear separator in a two dimensional data features or a hyper plane in a higher dimension data features. For example, Figure 4 shows a simple example of the day feature (from 1: Monday to 7: Sunday) used directly without any extraction, demonstrated using two classes: positive and negative. It is impossible to perform a linear classification in the given dataset. In order to solve this problem of the periodic features like time, Yang et. al. (2011), in their study of predicting unanswered questions in CQA, had attempted to split the time into 24 different features, each to represent an hour. However, this approach will generate a high number of features and causing the dataset to be unnecessary sparse with many zeroes. Therefore in our study, we apply a novel approach which is similar to an approach introduced by Bishop (2006) to solve this problem by expanding the each periodic feature into two features using *sine* and *cosine* functions, so that each periodic feature is now represented in a 2-dimensional space. This way, classification algorithms can perform linear separation correctly in distinguishing the classes: positive and negative. The equations for the *sine* and *cosine* functions to extract the time and day features are shown in Table 3. Figure 5 shows the example of the day feature after extraction and a linear line is able to separate the two different classes.



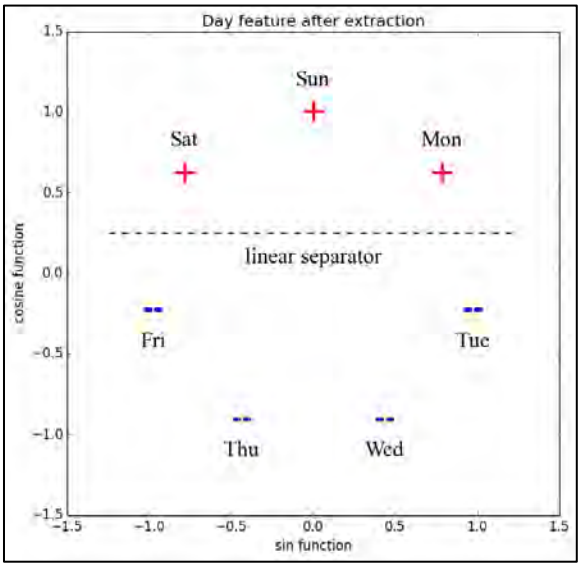


Figure 4. Periodic feature before extraction. **Figure 5. Periodic feature after extraction.**

Periodic features	*sine* function	*cosine* function
Time 1 to 24 hours	$\sin(2\pi * time/24)$	$\cos(2\pi * time/24)$
Day 1 (Monday) to 7 (Sunday)	$\sin(2\pi * day/7)$	$\cos(2\pi * day/7)$

Table 3. *Sine* and *cosine* functions for periodic feature extraction.

Some of the textual features from the content of the questions are extracted with the help of NLTK[6] (Natural Language Toolkit) module. NLTK is a free open source Python module, with a broad-coverage natural language toolkit that provides a simple, extensible, uniform framework (Bird, 2006). NTLK is widely used in academic research and teaching. We mainly use the tokenization function implemented in NLTK for the feature extraction from textual content. Tokenization is a process of separating a stream of text into individual words, which are also known as tokens. For our dataset, we tokenize the textual content of the title and questions into separated words. This is done so that the number of words or length of the title and questions can be calculated more efficiently to extract the required features.

As for the evaluation of the content appraisal features from the content of the questions, expert judgments from human are used as proxies for the users' judgments of the questions. Human experts are used for evaluation of the content appraisal because automatic analysis of appraisal approach is still under extensive research and there are still rooms for improvement (Taboada and Grieve, 2004). From previous research, Rieh and Danielson (2007) and Suryanto et. al. (2009) used similar techniques in their CQA studies. More specifically in the study by Suryanto et. al. (2009), content quality based on usefulness, readability, sincerity, readability and objectivity are rated by independent evaluators.

In our study, three independent evaluators (the three authors of this paper) evaluate the questions in terms of the six content appraisal features identified, they are: completeness, complexity, language error, presentation, politeness, and question subjectivity. Because of the nature of question in Stack Overflow which are programming related questions and we have crawled only questions with Java tag, the evaluators invited have bachelors or master degrees related to information technology and they have some experience in Java programming. Given their relevant educational and professional background, they are able to offer reliable judgment on the content appraisal of the question in Stack Overflow.

As the evaluations of the content appraisal are done by three people, it is crucial to ensure their degree of agreement in order to confirm the consistency of their judgment. Cohen's kappa coefficient is utilized to measure the inter-rater agreement among the three evaluators. Initially, each evaluators are given 180 questions for a start, which consist of 90 bad questions (without any answers) and 90 good questions (with at least one accepted answer by the askers). The average kappa coefficient was found to be 0.765 before continuing the evaluations of the content appraisal on the complete dataset (APPENDIX A shows the complete calculation of the average kappa coefficient). This suggests that there is a high degree of agreements among all the three evaluators and the content appraisal features rated by them are of high quality and consistency.

Feature Normalization

Feature normalization is a method used to standardize the value range of the features of the raw dataset. In classification, feature normalization is an important step because the range of values of raw data varies widely. If the scales for different features are wildly different, some classification algorithms may not work correctly. Ensuring standardized values among all the features implicitly weights all features equally in their representation of the dataset. In addition, normalization may improve the accuracy and efficiency of classifications algorithms like support vector machine (SVM), logistic regression, neural networks and nearest neighbor (Han et. al., 2006).

[6] http://www.nltk.org/

Among all the normalization methods available, min-max normalization is one of the best method in rescaling the values of raw data to be used in classification (Shalabi et. al., 2006). In our study, we normalized the value of all the features by adopting the **min-max normalization** method to rescale the value range of every features. Min-max normalization performs a linear transformation on the original raw data to a value range of 0 to 1. The formula of the min-max normalization is shown in equation (2), where x is the original value and x' is the normalized value, whereas $\max(x)$ and $\min(x)$ are the maximum and minimum values in the feature. After the min-max normalization process, we obtain a value range of 0 to 1 for all the features, which mean that every normalized features are comparable to each other in terms of their representation to the dataset.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{2}$$

Feature Selection

Feature selection is the process of selecting a subset of relevant features from all the features available for use in the construction of classification models. The objective of feature selection is to determine the best or good enough combination of features that improves or maintains the accuracy of classification over selecting every single features available. Feature selection technique are often used with dataset of many features and comparatively few samples, like the dataset in our study. In building a classification model, feature selection techniques provide three main advantages. First, the interpretability of the classification model is greatly improved. This is because a simpler model with few predictive features can be easily translated to human's understanding compared to a complex model which consist of a large number of features, and to understand their influence in the classification task (Guyon & Elisseeff, 2003).

Second, the time used for training the classification model are significantly reduced. This is especially true for a dataset with a large number of samples, because most classification algorithms train faster with less amount of data. Third, feature selection may reduce overfitting of the classification models. Overfitting occurs when a classification model is complex with a high number of features, which may contain redundant data or noise. The overfitting model may make prediction with high accuracy on the training dataset (overfitting the dataset) but generally have poor accuracy on new dataset not known beforehand. Reduction of features using feature selections produces less redundant data and therefore less opportunity for the classification model to make prediction based on noise and irrelevant data.

In order to select the most relevant and significant features in our dataset, we adopted two different approaches, namely the machine learning approach and statistical approach. Blooma et. al. (2012) also combine the two approaches with the help of SPSS software in their study to identify the most significant features in the prediction high quality answers in CQA. First, for the machine learning approach, we utilized the **logistic regression** classification algorithm with its L1 regularization parameter enabled to filter out the irrelevant features. Using L1 regularization, the logistic regression algorithm becomes extremely insensitive to the presence of irrelevant features, causing many logistic regression coefficient of the features to become zero (Ng, 2004). Therefore, those features with zero coefficient are natural candidates as irrelevant features and should be ignored during classification.

Second, in order to get a more solid set of most relevant features, we also consider the **Analysis of variance (ANOVA) F-test**, a statistical hypothesis testing. The F-test is done on every features of our dataset was done to find out the set of features that are statistically significant. Particularly, the output of F-test: the *p*-value is the probability of observing a result at least as extreme as the test statistic, with the assumption that the null hypothesis is true. In reality, a feature is considered statistically significant if its *p*-value is lower than a threshold significant level, usually in practice, the value of the threshold is 0.05. In our study, we combine both approaches of feature selection: machine learning and statistical approach to single out a set of features that is relevant, statistically significant and reliable, before we apply them into some of the current state-of-the-art classification algorithms (which are introduced in the next section) for the prediction of good and bad questions.

CLASSIFICATION

This section explains the about the classification algorithms, the validation approach and the evaluation metrics used in our study. The validation and evaluation process are done to ensure that the features which are identified and selected are useful and reliable in classifying the good and bad questions in our study.

Classification Algorithms

In our study, classification algorithms are used to investigate the usefulness of the identified features to predict good and bad question in Stack Overflow. We explored classification algorithms: logistic regression, support vector machine (SVM), decision tree, naïve Bayes and *k*-Nearest Neighbors. We used those algorithms implemented in the Scikit-learn[7]. Scikit-learn is a free open source Python module integrating a wide range of state-of-the-art machine learning algorithms, with the emphasis on ease of use, performance, documentation completeness and API consistency (Pedregosa et. al., 2011). In addition, Scikit-learn is very popular and widely used among academic as well as commercial researchers. The classification algorithms used are explained below.

- **Logistic regression**: Logistic regression is a type of probabilistic statistical classification algorithm, used for binary class prediction based on one or more features. In other words, it is a form of regression used when the class is Boolean value (e.g.: 0 or 1) while the features or predictors could be of any data type. Similar to other types of regression analysis, the types of features used by logistic regression to make prediction can be either continuous or categorical data. Contrarily, when compared to linear regression, logistic regression is used for predicting binary outcomes of the class rather than continuous outcomes. Logistic regression applies the maximum likelihood estimation after transforming the features into a logistic regression coefficient, the natural log of the odds on whether the feature occurs. In our study, we apply the logistic regression classification algorithm with its L1 regularization parameter enabled in the feature selection process as it was found to be insensitive to the presence of irrelevant features (Ng, 2004).
- **Support vector machine (SVM)**: SVM is a type of supervised learning algorithm which analyze and recognize patterns in a training dataset. Unlike logistic regression, SVM is a non-probabilistic binary classification algorithm. SVM trains a training data with binary class, and builds a classification model that assigns new data into either one class. Figure 6 shows that SVM works by maximizing the margin between the two different classes, the data points that are closer to the opposite class are called support vectors and they help to determine the optimal classification separator. SVM is considered as one of the best classification algorithm for many real world tasks, mainly due to its robustness in the presence of noise in the dataset used for training, and also high reported accuracy for many cases. In addition, one of the advantages of SVM is the ability to perform non-linear classification using kernel trick. However, in our study, we use the linear version of SVM so that comparison can be made with other linear classification algorithms like logistic regression and naïve Bayes, in addition to preventing overfitting of the classification on the dataset.
- **Decision tree**: Decision tree is a rules based classification algorithm. The goal of decision tree is to create a model that predicts the value of a class by learning a set of simple decision rules speculated from the features of training data. The set of decision rules are then used for predicting the class of a new data. We use the classification and regression tree CART algorithm of the decision tree implemented in the Scikit-learn module. One of the benefit of using decision tree for classification is the ease of interpretability of the models and results. However, if there are irrelevant features in the training data, decision tree algorithm can create overly complex trees used for classification and causing overfitting (do not generalize the training data to unknown new data). Because we are using cross-validation technique (refer to the next sub-section), which is able to surface the overfitting problem in classfications models, the comparison can be done between the performance of decision treeof using complete set of features along with the smaller set of features obtained from feature selection.
- **Naïve Bayes**: Naïve Bayes is a probabilistic classification algorithm based on Bayes' theorem in the area of probability theory and statistics. Naïve Bayes algorithm is naïve in the sense that it assumes that the value of a particular feature is completely unrelated to the presence or absence of all other features in the dataset, given

[7] http://scikit-learn.org/

the class variable. Despite its naïve and overly simplified assumptions, naïve Bayes classification algorithm performs fast and is one of the most efficient and effective learning algorithms for machine learning and data mining (Zhang, 2004). One of the advantages of naïve Bayes is that it requires less training data to perform a classification compared to other algorithms. Its main disadvantage is that it cannot learn the interactions between the features, because a particular feature is assumed to be unrelated to other features, as mentioned earlier. In addition, naïve Bayes can suffer from oversensitivity to redundant or irrelevant features (Ratanamahatana & Gunopulos, 2002). However we can utilize this shortcoming of naïve Bayes to verify that our feature selection step is actually successful in getting rid of the irrelevant features, by comparing the performance of naïve Bayes using complete set of features along with the smaller set of features obtained from feature selection.

- **K-Nearest Neighbors (*k*-NN)**: *K*-NN is a non-parametric classification algorithm, non-parametric in the sense that it does not make any assumptions on the underlying distribution of the dataset. *K*-NN is also a type of instant based or lazy learning algorithm, meaning that there is no any explicit training process before testing, all the computations for classification is done on-the-go with every training data during the testing process. In the classification or testing phase, *k* is a user-defined constant, and a data point is classified by a majority vote of its neighbors, the class assigned is the class most common among its k nearest neighbors. The classification is highly depending on the number of nearest neighbor, *k*. Figure 7 below illustrate the *k*-NN with different setting of *k*, if *k*=3, the new data point (green question mark in the figure) will be classified as positive, whereas if *k*=5, the class will be negative. Euclidean distance is a commonly used distance metric to find out the *k* nearest neighbors of a data point in the data space (Weinberger et. al., 2006). For our dataset, we found that *k*=55 gives us the optimal classification results.

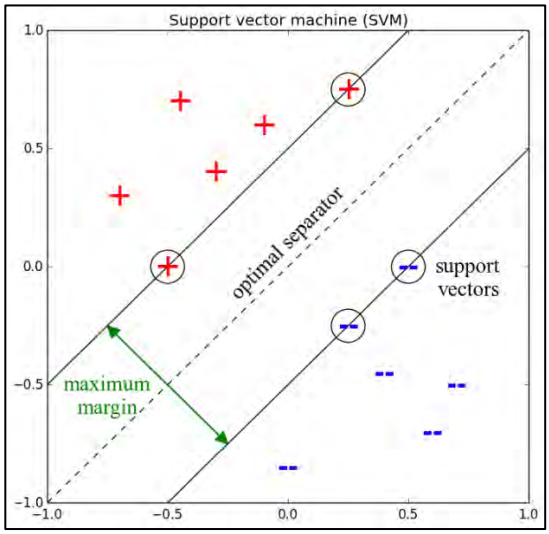


Figure 6. Support vector machine (SVM).

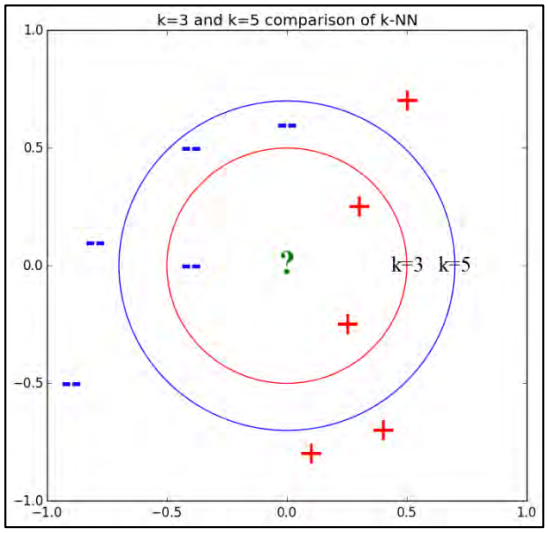


Figure 7. *k*=3 and *k*=5 comparison of *k*-NN.

Cross-validation

Cross-validation is a statistical approach used to evaluate classification algorithms by dividing the dataset into two main segments: one larger segment that is used for learning or training a classification model and the other usually smaller segment is used to validate the model. In cross validation, every segments of the dataset must cross over in in every iteration of testing so that each segment has the chance of being tested. Cross-validation is usually used to allow the training dataset to be also utilized as testing dataset in the training process, in order to prevent the overfitting problem and obtain an overall accuracy on how the classification model may perform on previously unknown new datasets.

The commonly used cross-validation approach is *k*-fold cross-validation, in which the dataset is partitioned into *k* roughly equal sized segments or folds as the name implies. After that, *k* iterations of validation are performed in a way that for each iteration, a different fold of the data is taken out and used for validation while the remaining folds are used for learning or training the classification model. Kohavi (1995) recommended using a stratified 10-

fold cross-validation where k=10 and he indicated that stratification in cross validation is better in terms of bias and variance when compared to the normal cross-validation. In cross-validation with stratified sampling, each fold contains roughly the same percentage of samples of each class as the complete dataset, this is done to ensure that each fold is a reasonably satisfactory representation of the whole dataset. In our study, we choose to use the **stratified 10-fold cross-validation** approach with the help of evaluation metrics which are discuss in the next sub-section.

Evaluation Metrics

For our study, we adopted two evaluation metrics in validating the performance of the classification, the two evaluation metrics are: **accuracy** and **area under the receiver operating characteristic (ROC) curve**. In classification, accuracy is used to measure of the performance of binary classification test, the correctness of the algorithm in identifying or excluding a condition, the accuracy is the fraction of correctly classified results (both true positives and true negatives) of the overall results. The formula for accuracy is shown in equation (3), where TP is true positives: the number of positive classes that are classified as positives and TN is the true negative: the number of negatives classes that are classified as negatives. In our study, both positive class (questions with an accepted answer by the asker) and negative class (questions that are completely ignored) are equally important classes to be considered in the evaluation of the classification performance. Therefore, the evaluation metric: accuracy is used because it considers both true positives and true negatives in the measurement, unlike precision measure which only take the true positives into account.

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

In addition to that, we also use the area under the ROC curve (AUC) as an evaluation metric in our study. ROC is a curve that represent the performance of a classification model with the true positive rate (TPR) and false positive rate (FPR) when the discrimination threshold is varied. The TPR is the fraction of correct positive results to all the positive samples, whereas FPR is the fraction of incorrect positive results to all the negative samples. An ROC curve is a two-dimensional representation of the performance of a classification model. Similarly, AUC, the calculated area under the ROC curve, can be used to measure the performance of classification models. The AUC is statistically useful in the sense that the AUC is equivalent to the probability that the classification model will rank a positive class higher than a negative class that are randomly chosen. From previous research, AUC is found to be a better evaluation metric than accuracy in terms of statistical consistency and discrimination (Bradley, 1997; Huang & Ling, 2005). Therefore, we also incorporate AUC measure apart from accuracy in order to ensure that the features used for the classification of good and bad questions are useful and reliable.

RESULTS AND DISCUSSION

The results and discussion of this study is divided into two sub-sections: feature selection and classification. The final outcome of the results of the feature selection is a set of features that are relevant and significance to the classification of good and bad questions. On the other hand, the results of the classification include the performance (accuracy and AUC) of the classification algorithms in predicting good and bad questions using the complete set of features as well as the features obtained from the feature selection step. The details of the results and discussion of the feature selection and classification are explained thoroughly in the following sub-sections.

Feature Selection

Table 4 shows the results of feature selection, obtained from logistic regression with L1 regularization and the ANOVA F-test for all the features identified from the literatures. The features are divided into two main categories: metadata features and content features, and each category is further divided into their respective sub-categories (as discussed in the feature identification section of this paper). In the table, the coefficient from logistic regression as well as the p-value from F-test are listed for all the individual features. Features with zero coefficient from the L1 regularized logistic regression are usually irrelevant features and should be ignored during classification (Ng, 2004). For a feature to be relevant and significant in the classification of good and bad questions, the absolute value of the coefficient must be more than zero and the p-value must be lower than a threshold significant level, 0.05. In addition, from the results in Table 4, the features can be broken down into

three major groups of features according to their association to the classification task in our study, namely: **positively associated features**, **negatively associated features** and lastly **irrelevant features**. Each group of the features are discussed below.

Features that are positively associated have coefficient values of more than zero and *p*-values of less than 0.05: this means that with a higher value in the coefficient, the higher the association of the features to good questions (questions with an accepted answer by the asker) and the less association to bad questions (questions that are completely ignored). From the asker's user profile features, only one feature is found to be positively associated to good questions: upvotes/downvotes. This probably means that the quality of the questions asked in Stack Overflow depends on the quality of the contents (both questions and answers) posted by the askers previously, with high number of upvotes and low number of downvotes. In addition, it is found that two content appraisal features are found to be positively associated to good questions, they are: completeness and subjectivity. This basically means that users in Stack Overflow prefer to answer questions that are that clear and understandable with reasonable amount of information provided by the askers, as well as questions which requires opinions from different perspectives. Furthermore, the time feature from the metadata of the questions is found to be positively associated for the $cosine$ function but negatively associated for the $sine$ function, this interesting finding is discussed in the discussion sub-section. Lastly, from the textual features, wh word feature is found to be positively associated. This means that the title of the question that starts with a question word are more likely to get answers from the community. The positively associated features from the most relevance and significance to the least relevance (listed in decreasing value of the coefficient) are shown in the first column of Table 5.

On the other hand, features that are negatively associated have coefficient values of less than zero and *p*-values of less than 0.05: this means that with a lower value in the coefficient, the more the features' association to good questions and the less association to bad questions. It is found that some features from the sub-categories of textual features, content appraisal features are negatively associated to good questions. The textual features include: tags, title length, question length and whether the questions contain code snippet(s). The content appraisal features that are negatively associated to good questions are: complexity and politeness. The negatively association of complexity feature basically means that the users in Stack Overflow prefer to answers questions which are simple. Politeness is found to be least relevance among the negatively associated features to good question, this probably means that the users do not really care whether the askers are polite or sincere when asking questions. It is also worth noting that the day and the time feature with the $sine$ function is also negatively associated to good questions. The negatively associated feature from the most relevance and significance to the least relevance (listed in increasing value of the coefficient) are shown in the second column of Table 5.

Last but not least, the irrelevant features are those features that are not useful in the classification of good and bad questions. All the irrelevant features have coefficient values of zero or the *p*-value is not less than the value of 0.05, suggesting a lack of associated to the classification of good and bad questions. Some of them are found in asker's user profile and the metadata of the questions, as well as the content appraisal features obtained from the questions. From the features found in the asker's user profile sub-category, it is found that the most of the features do not have any impact on whether the subsequent questions asked will get a good answers, the irrelevant features are: reputation, upvotes, downvotes, questions asked, answers posted and answers posted/questions asked. In addition, the $cosine$ function of the day feature do not affect whether the questions will get any answers. Lastly, the content appraisal features that are irrelevant include: language error and presentation. This essentially means that the overall look and feel of the questions does not affect a user's decision to answer the questions. The list of irrelevant features are populated in the last column of Table 5.

Among all the identified relevant features from feature selection step, it is also interesting to report that the textual features are found to be the most relevance sub-category of features, the specific features that are the most relevant to the classification of good and bad questions include (from the most relevant to the least): question length, tags, title length, code snippet and wh word. This is followed by the sub-category of content appraisal features, which include (from the most relevant to the least): complexity, completeness politeness and subjectivity. The third most relevance sub-category of features is from the asker's user profile. Among all the eight features from the asker's user profile, only days since join and upvote/downvotes are found to be useful.

Finally, the sub-category of features that is found to be the least relevance is the features from the questions' metadata, which are the time and day the questions are asked.

Classification

The validation of the performance of classification are used to verify the usefulness and reliability of the identified features to predict good and bad questions in Stack Overflow. In addition, it can also be utilized to investigate whether the feature extraction step done earlier produces a smaller set of highly relevance and significance features, which can be used more effectively for classification compared to using the complete set of features.

Table 6 shows the average accuracy and AUC from the stratified 10-fold cross-validation for all the classification algorithms used, containing both the average and AUC without feature selection (using the complete set of features) and with feature selection. The complete set of results of every folds of the cross-validation as well as the ROC curves can be found in APPENDIX B and C located at the end of this paper. The value of the accuracy and AUC has a maximum value of 1 if the predictions of good and bad questions from the classification model is 100% correct, and a minimum value of zero if all the predictions are wrong. For a binary classification task in our study, the value of accuracy or AUC for a random guess is 0.5, which means that the classification model is useless (performs worse than a random guess) if either the value accuracy or AUC falls under 0.5.

There are three important information that can be depicted from the results of the classification performance found in Table 6. First, it is found that all the classification algorithm perform reasonably well in the prediction of good and bad questions, better than a random guess. Without the feature selection, the overall accuracy is found to be ranging from 0.574 (naïve Bayes) to 0.735 (logistic regression), whereas for the AUC, it is found to be ranging from 0.759 (naïve Bayes) to 0.816 (logistic regression and SVM). This basically means that it is a feasible option to make use of the identified features to classify good or bad questions in Stack Overflow.

Secondly, the overall performance of the classification improves by replacing the features with a smaller set of features obtained from the feature selection step. With the inclusion of feature selection, the lowest accuracy actually improves to a 0.698 (k-NN) and the highest accuracy stays the same at 0.735, whereas the lowest AUC increases to 0.763 (k-NN). Among all the classification algorithms, the performance of naïve Bayes improves significantly with feature selection. This is because as mentioned earlier, naïve Bayes is sensitive to redundant or irrelevant features (Ratanamahatana & Gunopulos, 2002). Therefore, this essentially means that the feature selections steps successfully determine a smaller set of highly relevance and significance features, which are a better representation of the dataset compared to using the complete set of features.

Thirdly, two classification algorithms, namely: logistic regression and SVM are found to have the best overall performance both in terms of accuracy and AUC, when compared to other algorithms. This is expected because these classification algorithms represent some of the best performing supervised learning methods in the current age (Caruana & Niculescu-Mizil, 2006).

Discussion

From the results in the previous sub-section, three characteristic of the community in CQA, specifically in Stack Overflow can be observed. Firstly, the community in Stack Overflow prefers to answer simple and understandable questions, consistent with the findings from Asaduzzaman et. al. (2013). From our results of feature selection, lower number of word count in the questions, lower complexity, higher completeness and lower number of tags (categories) lead to higher chance of the questions with an accepted answer. Questions are easily understandable due to the low number of word count in the question and reasonable amount of information provided by the askers. The lower number of tags most means that the question is very specific and probably only require the answerers to have limited knowledge.

Secondly, questions which are asked after evening and before night mid, or towards the end of the week or during the weekends have a higher chance of being answered, with the assumption that most users are located in the same time zone as Stack Overflow. From the result, the time feature in the $sine$ function is negatively associated to good questions and the $cosine$ function is positively associated. Therefore, the rough estimation of the time is from 6 p.m. in the evening to 12 mid night. Figure 8 shows questions are more likely to end up with an accepted answer if they are asked within that time frame in the green region. For the day feature, it was found that $sine$

function of the feature is negatively associated and the $cosine$ function is statistically insignificant. Figure 9 shows the rough estimation of the days in a week, if questions are asked on those days, they are more likely to get an accepted answer, compared to other days. This suggests that users in Stack Overflow consist of working professionals and they are more active in answering questions off working hours (after evening and weekends). However, the time and day features are found to be not highly relevant among the relevant features partly due to the influence of different time zone of the users in Stack Overflow.

Thirdly, the experience level of the askers in Stack Overflow is found to be less important in determining the quality of the questions. Most of the features of the users' asker profile are found to be irrelevant features in identifying the quality of the questions. Interestingly, the number of days since the users joined Stack Overflow is found to be negatively associated to good questions. A user can be asking good questions and get some acceptable answers without requiring much prior experience in Stack Overflow. In addition, the only feature from users' asker profile that is found to be positively associated is upvote/downvotes. Users that consistently posted high quality content (high agreements and low disagreements from other users in terms of questions and answers) will, in the future, are likely to ask high quality question that ended up with acceptable answers.

Category	Sub-category	Feature	Coefficient from logistic regression (not equal to zero)	*p*-value from F-test (less than 0.05)
Metadata features	Asker's user profile	Reputation	0.000	0.559
		Days since join	-2.197	0.000
		Upvotes	0.830	0.333
		Downvotes	0.618	0.140
		Upvotes/Downvotes	2.307	0.042
		Questions asked	2.233	0.173
		Answers posted	0.000	0.910
		Answers posted/questions asked	0.000	0.183
	Question	Time (*sine*)	-0.253	0.016
		Time (*cosine*)	0.305	0.000
		Day (*sine*)	-0.318	0.023
		Day (*cosine*)	0.266	0.050
Content features	Textual features	Tags	-1.522	0.000
		Title length	-1.419	0.000
		Question length	-4.359	0.000
		Code snippet	-0.736	0.000
		Wh word	0.513	0.000
	Content appraisal	Completeness	2.658	0.000
		Complexity	-4.099	0.000
		Language error	-0.008	0.361
		Presentation	0.000	0.968
		Politeness	-0.547	0.004
		Subjectivity	0.197	0.000

Table 4. Coefficient and *p*-value in feature selection.

Positively associated features (from most relevant to least)	**Negatively associated features** (from most relevant to least)	**Irrelevant or not significant features**
1) Completeness	1) Question length	1) Reputation
2) Upvotes/Downvotes	2) Complexity	2) Upvotes
3) Wh word	3) Days since join	3) Downvotes
4) Time (*cosine*)	4) Tags	4) Questions asked
5) Subjectivity	5) Title length	5) Answers posted
	6) Code snippet	6) Answers posted/questions asked
	7) Politeness	7) Day (*cosine*)
	8) Day (*sine*)	8) Language error
	9) Time (*sine*)	9) Presentation

Table 5. Three groups of features from feature selection.

Algorithms	**Without feature selection**		**With feature selection**	
	Accuracy	**AUC**	**Accuracy**	**AUC**
Logistic regression	0.735	0.816	0.735	0.813
SVM	0.734	0.816	0.735	0.813
Decision tree	0.728	0.780	0.732	0.784
Naïve Bayes	0.574	0.759	0.712	0.777
***k*-NN**	0.694	0.763	0.698	0.763

Table 6. Average accuracy and AUC from 10-fold cross-validation.

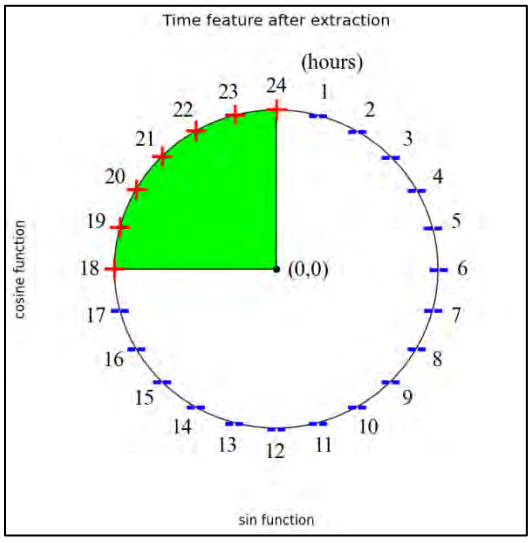


Figure 8. Time frame associate to good questions.

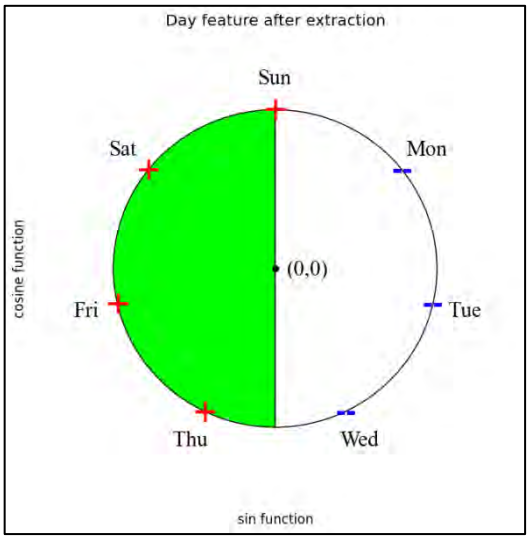


Figure 9. Days associated to good questions.

CONCLUSION

This paper describes our work and effort on classifying between the good questions (questions that contain at least an accept answer by the askers) and bad questions (questions that are completely ignored by the community) in CQA, more specifically in the context of Stack Overflow. We introduce and formalize the crucial steps of data mining and knowledge discovery from data in the problem of questions' quality classification, which include specialized feature extraction process, feature selection process and the exploration of some of the state-of-the-art classification algorithms.

Our study has revealed the important features that are useful in the classification of good and bad questions, the features that are revealed from this study come from the metadata of the askers' user profile and questions, as well as from the contents of the questions, which include textual features and content appraisal features. In addition to that, the identified features are relevant, significant and reliable because they have gone through the process of feature selection to get rid of irrelevant and statistically insignificant features, along with the performance evaluation of classification process to ensure that the selected set of features are reliable when it comes to the real task of classifying between good and bad questions. The limitation, future research direction and the significance of our study are discussed in the sub-sections below.

Limitation and future research

There are three limitation in our study. Firstly, the study is only done on one category (also known as tag in Stack Overflow) of questions, we specifically focus on questions about Java programming language due to pragmatic reasons and the limited time frame of the study. However, the problem with our approach is that we fail to take advantage of the wide range of users' metadata (details in their user profile) in Stack Overflow. Therefore, in our study, the features from the asker's user profile are reported not as highly relevant in the classification of good and bad questions. Therefore for similar future research, more categories of questions should be included and also identify the features that can be found in a certain category to help in the classification tasks (e.g.: Liu et. al., 2008).

Secondly, when predicting the quality of a question, we do not consider the previously asked similar or exactly same questions that are solved or unsolved. Information about the previously asked similar questions can be useful in classifying whether the newly asked question is good or bad, this is because questions that are similar will have identical quality (Li et. al., 2012). Therefore, future research should investigate ways to identify similar questions and the relevant features that can be extracted them.

Thirdly, the extraction of content appraisal features requires judgment from human expert, which is highly labor intensive and inefficient due to the manual extraction. Because content appraisal features are found to be quite relevant in the classification task, future research should look into this problem and explore techniques of automatic content appraisal extraction from the content of questions in CQA. For example, Taboada and Grieve (2004) had presented their method for analyzing appraisal from text automatically. This will highly enhance and speed up the future research in the area of CQA because of less reliance on the slow manual extraction of content appraisal features.

Significance of the study

We envisage that the outcome of this study would have significant benefits to the stakeholders as explained below. Firstly, more comprehensive research for researchers in the area of CQA. For a long time, the importance of the questions posed on CQA websites are underestimated or even neglected. So the management of questions in CQA websites has always been a missing part in solution. Fortunately, this study aims to fix this problem by comparing the features of good and bad questions, and revealing the important factors that influence the questions. As a result, more reasonable research within CQA service domain are expected to provide a more comprehensive solution.

Secondly, better experiences for the users of CQA. This study would help users to formulate questions in a more intelligent approach that will not be ignored easily in CQA. To increase their chances of getting answers, users with questions would be educated to organize thoughts and articulate appropriately based on the identified factors. Hence, the posed question would be attractive enough to encourage decent answers and comments in CQA

timely. As a result, the questions are more likely to be addressed in a short time, and user experiences in the community would be enhanced greatly.

Thirdly, value added for CQA owners especially Stack Overflow. With the knowledge from our study, CQA owners are able to provide better CQA services for the public by introducing an automatic system to predict the quality of a newly posted question. This way, if the question is predicted that it will probably not getting any answers from the public, the asker of the question will be prompted to revise the questions and not allowed to post the question until a quality question is formulated. Therefore, storage space will not be wasted on questions that are likely to be ignored by the public. In addition, CQA owners can also take advantage on active time and day period of the users. For example, the users are only allowed to post questions during the period when most users are active so that the questions are more likely to end up with an accepted answer. On the other hand, long-term value is expected for the CQA service. It is known that the value of CQA websites is generated by both content consumption and content creation (Nasehi et. al., 2012). High quality questions will lead to high quality answers. Without no doubts, the more high-quality contents created, the more high-quality contents consumed. Apart from that, high-quality contents would be stored and distributed for future reference. Thus, CQA websites got distributed and reputed, and the value is continuously added to the websites in the long run.

ACKNOWLEDGMENTS

The development of this critical inquiry project would not be possible without the aggregate support from a few individuals. First of all, we would like to express our deep gratitude to our supervisor, Prof. Alton Chua Yeow Kuan. His invaluable advice and great guidance have supported and inspired us over the past few months. Without his supervision, we could not come this far in accomplishing our project. On top of that, we would also like to thank Stack Exchange[8] for its Data Explorer web service, it provides us with an easy and efficient way to crawl the data from Stack Overflow. Next, we would also like to thank the school, Wee Kim Wee School of Communication and Information, Nanyang Technological University, for giving us the opportunity to carry out this experience-fulfilling critical inquiry project. Last but not least, we sincerely thank to our dearest parents and family members who had given us encouragement and moral support throughout the journey of working on this project and also our postgraduate education.

REFERENCES

Agichtein, E., Castillo, C., Donato, D., Gionis, A., & Mishne, G. (2008). Finding high-quality content in social media. In Proceedings of the international conference on Web search and web data mining, pp. 183-194.

Agichtein, E., Liu, Y., & Bian, J. (2009). Modeling information-seeker satisfaction in community question answering. ACM Transactions on Knowledge Discovery from Data (TKDD), Vol. 3 No. 2, pp. 10.

Anderson, A., Huttenlocher, D., Kleinberg, J., & Leskovec, J. (2012). Discovering value from community activity on focused question answering sites: a case study of stack overflow. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 850-858.

Asaduzzaman, M., Mashiyat, A. S., Roy, C. K., & Schneider, K. A. (2013). Answering questions about unanswered questions of stack overflow. In Proceedings of the Tenth International Workshop on Mining Software Repositories, IEEE Press, pp. 97-100.

Barua, A., Thomas, S. W., & Hassan, A. E. (2012). What are developers talking about? An analysis of topics and trends in Stack Overflow. Empirical Software Engineering, pp. 1-36.

Bian, J., Liu, Y., Zhou, D., Agichtein, E., & Zha, H. (2009). Learning to recognize reliable users and content in social media with coupled mutual reinforcement. In Proceedings of the 18th international conference on World wide web, ACM, pp. 51-60.

[8] https://stackexchange.com/

Bird, S. (2006). NLTK: the natural language toolkit. In Proceedings of the COLING/ACL on Interactive presentation sessions, Association for Computational Linguistics, pp. 69-72.

Bishop, C. M. (2006). Pattern recognition and machine learning, New York: springer, Vol. 1, p. 740

Blooma, M. J., Goh, D. H. L., & Chua, A. Y. K. (2012). Predictors of high-quality answers. Online Information Review, Vol. 36 No. 3, pp. 383-400.

Bovee, M., Srivastava, R. P., & Mak, B. (2003). A conceptual framework and belief-function approach to assessing overall information quality. International journal of intelligent systems, Vol. 18 No. 1, pp. 51-74.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern recognition, Vol. 30 No, 7, pp. 1145-1159.

Cai, Y., & Chakravarthy, S. (2011). Predicting Answer Quality in Q/A Social Networks: Using Temporal Features.

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In Proceedings of the 23rd international conference on Machine learning, ACM, pp. 161-168.

Chen, C., Wu, K., Srinivasan, V., & Bharadwaj, R. K. (2013). The best answers? think twice: online detection of commercial campaigns in the CQA forums. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ACM, pp. 458-465.

Chen, L., Zhang, D., & Mark, L. (2012). Understanding user intent in community question answering. In Proceedings of the 21st international conference companion on World Wide Web, 823-828.

Chen, Y., Dios, R., Mili, A., Wu, L., & Wang, K. (2005). An empirical study of programming language trends. Software, IEEE, Vol. 22 No. 3, pp. 72-79.

Chua, A. Y., & Balkunje, R. S. (2012). Comparative evaluation of community question answering websites. In The Outreach of Digital Libraries: A Globalized Resource Network, Springer Berlin Heidelberg, pp. 209-218.

Correa, D., & Sureka, A. (2014). Chaff from the Wheat: Characterization and Modeling of Deleted Questions on Stack Overflow, arXiv preprint arXiv, pp. 1401.0480.

Danescu, C., Kossinets, G., Kleinberg, J., & Lee, L. (2009). How opinions are received by online communities: a case study on amazon.com helpfulness votes. In Proceedings of the 18th international conference on World wide web, pp. 141-150.

Dror, G., Pelleg, D., Rokhlenko, O., & Szpektor, I. (2012). Churn prediction in new users of Yahoo! answers. In Proceedings of the 21st international conference companion on World Wide Web, ACM, pp. 829-834.

Fichman, P. (2011). A comparative assessment of answer quality on four question answering sites. Journal Of Information Science, Vol. 37 No. 5, pp. 476-486.

Frické, M., & Fallis, D. (2004). Indicators of accuracy for answers to ready reference questions on the internet. Journal of the American Society for Information Science and Technology, Vol. 55 No. 3, pp. 238-245.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. The Journal of Machine Learning Research, Vol. 3, pp. 1157-1182.

Han, J., Kamber, M., & Pei, J. (2006). Data mining: concepts and techniques. Morgan kaufmann.

Hong, L., Yang, Z., & Davison, B. D. (2009). Incorporating participant reputation in community-driven question answering systems. In Computational Science and Engineering, 2009. CSE'09. International Conference, Vol. 4, pp. 475-480.

Hsu, C. W., Chang, C. C., & Lin, C. J. (2003). A practical guide to support vector classification.

Huang, J., & Ling, C. X. (2005). Using AUC and accuracy in evaluating learning algorithms. Knowledge and Data Engineering, IEEE Transactions on, Vol. 17 No. 3, pp. 299-310.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection, in IJCAI, Vol. 14 No. 2, pp. 1137-1145.

Lai, L. C., & Kao, H. Y. (2012). Question Routing by Modeling User Expertise and Activity in cQA services. In The 26th Annual Conference of the Japanese Society for Artificial Intelligence.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. biometrics, Vol. 33 No. 1, pp. 159-174.

Li, B., Jin, T., Lyu, M. R., King, I., & Mak, B. (2012). Analyzing and predicting question quality in community question answering services. In Proceedings of the 21st international conference companion on World Wide Web, ACM, pp. 775-782.

Li, B., King, I., & Lyu, M. R. (2011). Question routing in community question answering: putting category in its place. In Proceedings of the 20th ACM international conference on Information and knowledge management, pp. 2041-2044.

Li, B., Liu, Y., Ram, A., Garcia, E. V., & Agichtein, E. (2008). Exploring question subjectivity prediction in community QA. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 735-736.

Liu, Y., Bian, J., & Agichtein, E. (2008). Predicting information seeker satisfaction in community question answering. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 483-490.

Mamykina, L., Manoim, B., Mittal, M., Hripcsak, G., & Hartmann, B. (2011). Design lessons from the fastest q&a site in the west. In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 2857-2866.

Miao, Y., Li, C., Tang, J., & Zhao, L. (2010). Identifying new categories in community question answering archives: a topic modeling approach. In Proceedings of the 19th ACM international conference on Information and knowledge management, ACM, pp. 1673-1676.

Nasehi, S. M., Sillito, J., Maurer, F., & Burns, C. (2012). What makes a good code example?: A study of programming Q&A in StackOverflow. In Software Maintenance (ICSM), 2012 28th IEEE International Conference, pp. 25-34.

Neuendorf, K. A. (2002). The content analysis guidebook. Sage Publications.

Ng, A. Y. (2004). Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In Proceedings of the twenty-first international conference on Machine learning, ACM, p. 78.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O. & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research, Vol. 12, pp. 2825-2830.

Pera, M. S., & Ng, Y. K. (2011). A community question-answering refinement system. In Proceedings of the 22nd ACM conference on Hypertext and hypermedia, ACM, pp. 251-260.

Ratanamahatana, C. A., & Gunopulos, D. (2002). Scaling up the naive Bayesian classifier: Using decision trees for feature selection.

Riahi, F., Zolaktaf, Z., Shafiei, M., & Milios, E. (2012). Finding expert users in community question answering. In Proceedings of the 21st international conference companion on World Wide Web, ACM, pp. 791-798.

Rieh, S.Y. and Danielson, D.R. (2007), “Credibility: a multidisciplinary framework”, in Cronin, B. (Ed.),Annual Review of Information Science and Technology, Information Today, Medford, NJ, pp. 307-64

Shachaf, P. (2010). Social reference: Toward a unifying theory. Library & Information Science Research, Vol. 32 No. 1, pp. 66-76.

Shah, C., & Pomerantz, J. (2010). Evaluating and predicting answer quality in community QA. In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, pp. 411-418.

Shalabi, L. A., Shaaban, Z., & Kasasbeh, B. (2006). Data mining: A preprocessing engine. Journal of Computer Science, Vol. 2 No. 9, pp. 735.

Singh, A., & Visweswariah, K. (2011). CQC: classifying questions in CQA websites. In Proceedings of the 20th ACM international conference on Information and knowledge management, ACM, pp. 2033-2036.

Souza, C., Magalhães, J., Costa, E., & Fechine, J. (2013). Routing Questions in Twitter: An Effective Way to Qualify Peer Helpers. In Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences, Vol. 1, pp. 109-114.

Suryanto, M.A., Lim, E.P., Sun, A. and Chiang, R.H.L. (2009), "Quality-aware collaborative question answering: methods and evaluation", inProceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval, ACM Press, New York, NY, pp. 142-51.

Suzuki, S., Nakayama, S. I., & Joho, H. (2011). Formulating effective questions for community-based question answering. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, pp. 1261-1262.

Taboada, M., & Grieve, J. (2004). Analyzing appraisal automatically. In Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text (AAAI Technical Re# port SS# 04# 07), Stanford University, CA, AAAI Press, pp. 158-161.

Tang, Y., Li, F., Huang, M., & Zhu, X. (2010). Summarizing similar questions for chinese community question answering portals. In Information Technology and Computer Science (ITCS), 2010 Second International Conference on, IEEE, pp. 36-39.

Treude, C., Barzilay, O., & Storey, M. A. (2011). How do programmers ask and answer questions on the web?: Nier track. In Software Engineering (ICSE), 2011 33rd International Conference, pp. 804-807.

Vasilescu, B., Capiluppi, A., & Serebrenik, A. (2012). Gender, representation and online participation: A quantitative study of Stackoverflow. In International Conference on Social Informatics.

Wang, G., Gill, K., Mohanlal, M., Zheng, H., & Zhao, B. Y. (2013). Wisdom in the social crowd: an analysis of quora. In Proceedings of the 22nd international conference on World Wide Web, International World Wide Web Conferences Steering Committee, pp. 1341-1352.

Wang, X. J., Tu, X., Feng, D., & Zhang, L. (2009). Ranking community answers by modeling question-answer relationships via analogical reasoning. In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 179-186.

Weinberger, K., Blitzer, J., & Saul, L. (2006). Distance metric learning for large margin nearest neighbor classification. Advances in neural information processing systems, Vol. 18, pp. 1473.

Xuan, H., Yang, Y., & Peng, C. (2013). An Expert Finding Model Based on Topic Clustering and Link Analysis in CQA Website. Journal of Network & Information Security, Vol. 4 No. 2, pp. 165-176.

Yang, L., Bao, S., Lin, Q., Wu, X., Han, D., Su, Z., & Yu, Y. (2011). Analyzing and Predicting Not-Answered Questions in Community-based Question Answering Services. In AAAI.

Zhang, H. (2004). The optimality of naive Bayes. A A, Vol. 1 No. 2, pp. 3.

Zhang, W., Pang, L., & Ngo, C. W. (2012). FashionAsk: pushing community answers to your fingertips. In Proceedings of the 20th ACM international conference on Multimedia, ACM, pp. 1345-1346.

Zhang, Z., Li, Q., & Zeng, D. (2010). Evolutionary community discovery from dynamic multi-relational CQA networks. In Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on, IEEE, Vol. 3, pp. 83-86.

APPENDIX A. INTER-RATER AGREEMENT FOR CONTENT APPRAISAL FEATURES

Completeness (between evaluator 1 and 2)

Kappa value = 0.768

Completeness		Evaluator 2				
		1	**2**	**3**	**4**	**5**
Evaluator 1	**1**	4	0	0	0	0
	2	1	21	0	0	0
	3	3	11	38	1	0
	4	0	3	1	59	5
	5	0	0	3	3	27

Table A1. Evaluator 1 & 2 on Completeness.

Completeness (between evaluator 1 and 3)

Kappa value = 0.707

Completeness		Evaluator 3				
		1	**2**	**3**	**4**	**5**
Evaluator 1	**1**	4	0	0	0	0
	2	1	18	3	0	0
	3	3	2	39	9	0
	4	0	0	3	61	4
	5	0	4	1	8	20

Table A2. Evaluator 1 & 3 on Completeness.

Completeness (between evaluator 2 and 3)

Kappa value = 0.781

Completeness		Evaluator 3				
		1	2	3	4	5
Evaluator 2	1	5	0	3	0	0
	2	3	22	0	7	3
	3	0	0	42	0	0
	4	0	1	1	61	0
	5	0	1	0	10	21

Table A3. Evaluator 2 & 3 on Completeness.

Complexity (between evaluator 1 and 2)

Kappa value = 0.796

Complexity		Evaluator 2				
		1	2	3	4	5
Evaluator 1	1	7	4	2	0	0
	2	3	29	5	0	0
	3	0	4	68	4	0
	4	0	0	0	43	4
	5	0	0	0	0	7

Table A4. Evaluator 1 & 2 on Complexity.

Complexity (between evaluator 1 and 3)

Kappa value = 0.726

Complexity		Evaluator 3				
		1	2	3	4	5
Evaluator 1	1	6	5	2	0	0
	2	6	27	4	0	0
	3	1	8	63	4	0
	4	0	1	3	42	1
	5	0	0	0	0	7

Table A5. Evaluator 1 & 3 on Complexity.

Complexity (between evaluator 2 and 3)

Kappa value = 0.836

Complexity		Evaluator 3				
		1	2	3	4	5
Evaluator 2	1	6	0	0	4	0
	2	0	34	3	0	0
	3	1	5	69	0	0
	4	5	0	0	42	0
	5	1	2	0	0	8

Table A6. Evaluator 2 & 3 on Complexity.

Language error (between evaluator 1 and 2)

Kappa value = 0.703

Language error		Evaluator 2				
		1	2	3	4	5
Evaluator 1	1	136	6	2	0	0
	2	4	16	5	2	0
	3	0	0	7	0	0
	4	0	0	0	2	0
	5	0	0	0	0	0

Table A7. Evaluator 1 & 2 on Language error.

Language error (between evaluator 1 and 3)

Kappa value = 0.780

Language error		Evaluator 3				
		1	2	3	4	5
Evaluator 1	1	131	7	5	1	0
	2	2	25	0	0	0
	3	0	0	7	0	0
	4	0	0	0	2	0
	5	0	0	0	0	0

Table A8. Evaluator 1 & 3 on Language error.

Language error (between evaluator 2 and 3)

Kappa value = 0.749

Language error		Evaluator 3				
		1	**2**	**3**	**4**	**5**
Evaluator 2	**1**	130	7	1	2	0
	2	0	21	1	0	0
	3	2	2	10	0	0
	4	1	2	0	1	0
	5	0	0	0	0	0

Table A9. Evaluator 2 & 3 on Language error.

Presentation (between evaluator 1 and 2)

Kappa value = 0.729

Presentation		Evaluator 2				
		1	**2**	**3**	**4**	**5**
Evaluator 1	**1**	3	0	0	0	0
	2	1	12	1	0	0
	3	0	1	58	2	0
	4	0	0	6	64	6
	5	0	4	5	7	10

Table A10. Evaluator 1 & 2 on Presentation.

Presentation (between evaluator 1 and 3)

Kappa value = 0.703

Presentation		Evaluator 3				
		1	**2**	**3**	**4**	**5**
Evaluator 1	**1**	3	0	0	0	0
	2	2	9	0	3	0
	3	0	2	56	3	0
	4	0	0	6	65	5
	5	0	3	7	5	11

Table A11. Evaluator 1 & 3 on Presentation.

Presentation (between evaluator 2 and 3)

Kappa value = 0.858

Presentation		Evaluator 3				
		1	2	3	4	5
Evaluator 2	1	3	0	0	1	0
	2	1	9	0	7	0
	3	1	0	69	0	0
	4	0	5	0	67	1
	5	0	0	0	1	15

Table A12. Evaluator 2 & 3 on Presentation.

Politeness (between evaluator 1 and 2)

Kappa value = 0.752

Politeness		Evaluator 2				
		1	2	3	4	5
Evaluator 1	1	62	8	5	0	0
	2	2	44	5	0	0
	3	0	0	30	2	0
	4	0	1	6	7	3
	5	0	0	0	0	5

Table A13. Evaluator 1 & 2 on Politeness.

Politeness (between evaluator 1 and 3)

Kappa value = 0.696

Politeness		Evaluator 2				
		1	2	3	4	5
Evaluator 1	1	60	6	7	2	0
	2	5	45	1	0	0
	3	1	4	27	0	0
	4	0	4	2	4	7
	5	0	0	0	0	5

Table A14. Evaluator 1 & 3 on Politeness.

Politeness (between evaluator 2 and 3)

Kappa value = 0.806

Politeness		Evaluator 3				
		1	2	3	4	5
Evaluator 2	1	57	2	2	0	3
	2	4	49	0	0	0
	3	2	8	35	0	1
	4	3	0	0	6	0
	5	0	0	0	0	8

Table A15. Evaluator 2 & 3 on Politeness.

Subjectivity (between evaluator 1 and 2)

Kappa value = 0.778

Subjectivity		Evaluator 2				
		1	2	3	4	5
Evaluator 1	1	6	2	6	0	0
	2	3	33	3	0	0
	3	0	4	35	3	0
	4	0	2	6	57	0
	5	0	0	0	1	19

Table A16. Evaluator 1 & 2 on Subjectivity.

Subjectivity (between evaluator 1 and 3)

Kappa value = 0.751

Subjectivity		Evaluator 3				
		1	2	3	4	5
Evaluator 1	1	3	2	9	0	0
	2	2	28	7	2	0
	3	0	6	33	3	0
	4	0	0	0	65	0
	5	0	0	0	2	18

Table A17. Evaluator 1 & 3 on Subjectivity.

Subjectivity (between evaluator 2 and 3)

Kappa value = 0.848

Subjectivity		Evaluator 3				
		1	**2**	**3**	**4**	**5**
Evaluator 2	**1**	4	0	1	3	1
	2	0	36	0	5	0
	3	1	0	46	3	0
	4	0	0	2	58	1
	5	0	0	0	3	16

Table A18. Evaluator 2 & 3 on Subjectivity.

Overall average kappa value = **0.765**

Content appraisal feature	evaluator 1 & evaluator 2	evaluator 2 & evaluator 3	evaluator 1 & evaluator 3	average kappa coefficient
Completeness	0.768	0.781	0.707	0.752
Complexity	0.796	0.836	0.726	0.786
Language	0.703	0.749	0.780	0.744
Presentation	0.729	0.858	0.703	0.763
Politeness	0.752	0.806	0.696	0.751
Subjectivity	0.778	0.848	0.751	0.792
average kappa coefficient	0.754	0.813	0.727	**0.765**

Table A19. Overall average Cohen's kappa coefficient for the evaluation of content appraisal features.

APPENDIX B. ACCURACY AND AUC FROM 10-FOLD CROSS-VALIDATION

Logistic regression	Without feature selection		With feature selection	
	Accuracy	AUC	Accuracy	AUC
Fold-1	0.730	0.800	0.753	0.803
Fold-2	0.743	0.817	0.740	0.815
Fold-3	0.750	0.819	0.733	0.814
Fold-4	0.733	0.830	0.740	0.824
Fold-5	0.767	0.829	0.763	0.826
Fold-6	0.743	0.832	0.747	0.830
Fold-7	0.737	0.836	0.730	0.829
Fold-8	0.750	0.848	0.757	0.845
Fold-9	0.700	0.760	0.707	0.753
Fold-10	0.697	0.784	0.680	0.781
Average	**0.735**	**0.816**	**0.735**	**0.813**

Table B1. Accuracy and AUC from 10-fold cross-validation for logistic regression.

SVM	Without feature selection		With feature selection	
	Accuracy	AUC	Accuracy	AUC
Fold-1	0.727	0.800	0.743	0.805
Fold-2	0.727	0.814	0.733	0.817
Fold-3	0.750	0.819	0.733	0.815
Fold-4	0.747	0.830	0.747	0.824
Fold-5	0.767	0.829	0.760	0.826
Fold-6	0.733	0.825	0.737	0.822
Fold-7	0.747	0.836	0.743	0.829
Fold-8	0.743	0.847	0.760	0.844
Fold-9	0.703	0.761	0.703	0.754
Fold-10	0.697	0.788	0.687	0.784
Average	**0.734**	**0.816**	**0.735**	**0.813**

Table B2. Accuracy and AUC from 10-fold cross-validation for SVM.

Decision tree	Without feature selection		With feature selection	
	Accuracy	AUC	Accuracy	AUC
Fold-1	0.743	0.777	0.733	0.775
Fold-2	0.757	0.807	0.743	0.791
Fold-3	0.720	0.789	0.740	0.797
Fold-4	0.767	0.806	0.777	0.824
Fold-5	0.717	0.780	0.737	0.792
Fold-6	0.723	0.788	0.733	0.787
Fold-7	0.743	0.800	0.743	0.811
Fold-8	0.703	0.756	0.700	0.758
Fold-9	0.693	0.747	0.693	0.749
Fold-10	0.710	0.744	0.720	0.750
Average	**0.728**	**0.780**	**0.732**	**0.784**

Table B3. Accuracy and AUC from 10-fold cross-validation for decision tree.

Naïve Bayes	Without feature selection		With feature selection	
	Accuracy	AUC	Accuracy	AUC
Fold-1	0.663	0.775	0.703	0.792
Fold-2	0.523	0.758	0.733	0.791
Fold-3	0.567	0.805	0.730	0.792
Fold-4	0.557	0.733	0.733	0.775
Fold-5	0.557	0.780	0.693	0.766
Fold-6	0.543	0.777	0.757	0.804
Fold-7	0.587	0.770	0.727	0.786
Fold-8	0.543	0.767	0.723	0.799
Fold-9	0.640	0.709	0.647	0.719
Fold-10	0.563	0.709	0.677	0.738
Average	**0.574**	**0.759**	**0.712**	**0.777**

Table B4. Accuracy and AUC from 10-fold cross-validation for naïve Bayes.

k-NN	Without feature selection		With feature selection	
	Accuracy	AUC	Accuracy	AUC
Fold-1	0.647	0.727	0.670	0.752
Fold-2	0.707	0.773	0.697	0.771
Fold-3	0.707	0.749	0.717	0.770
Fold-4	0.707	0.779	0.690	0.761
Fold-5	0.697	0.792	0.703	0.776
Fold-6	0.770	0.797	0.753	0.805
Fold-7	0.697	0.756	0.723	0.765
Fold-8	0.747	0.806	0.720	0.798
Fold-9	0.613	0.720	0.647	0.714
Fold-10	0.650	0.727	0.657	0.720
Average	**0.694**	**0.763**	**0.698**	**0.763**

Table B5. Accuracy and AUC from 10-fold cross-validation for *k*-NN.

APPENDIX C. ROC CURVES FROM 10-FOLD CROSS-VALIDATION

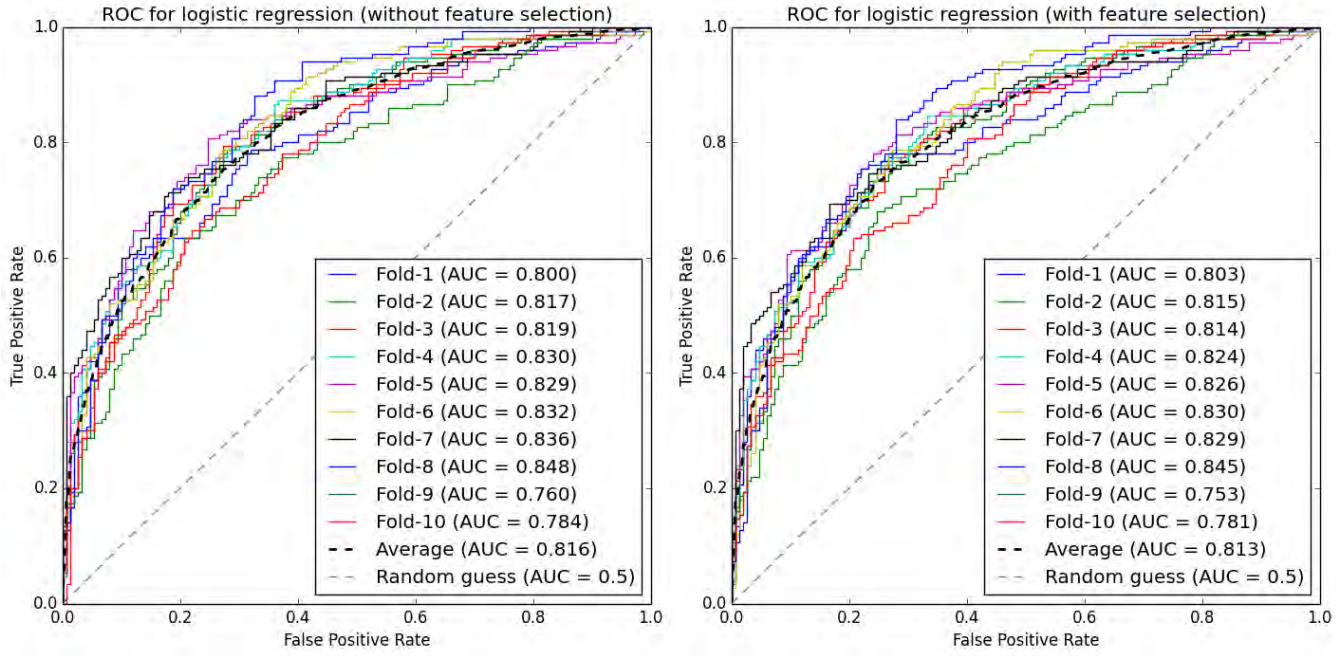


Figure C1. ROC curves for logistic regression

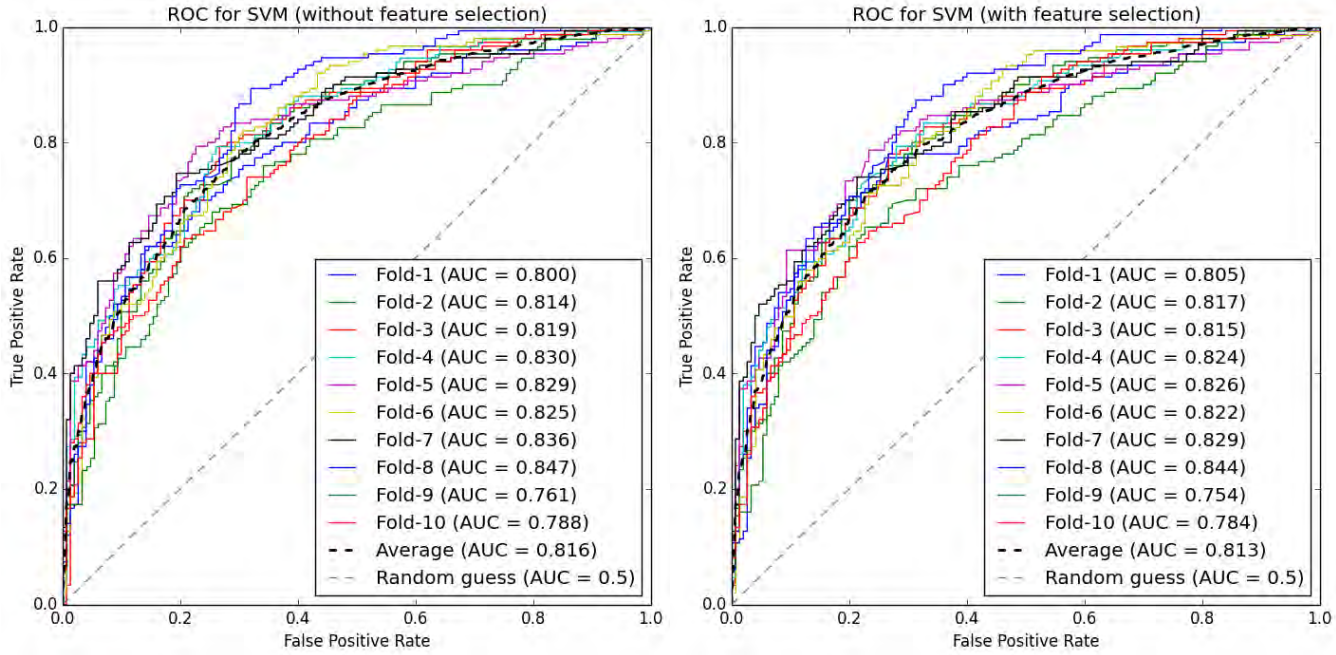


Figure C2. ROC curves for SVM

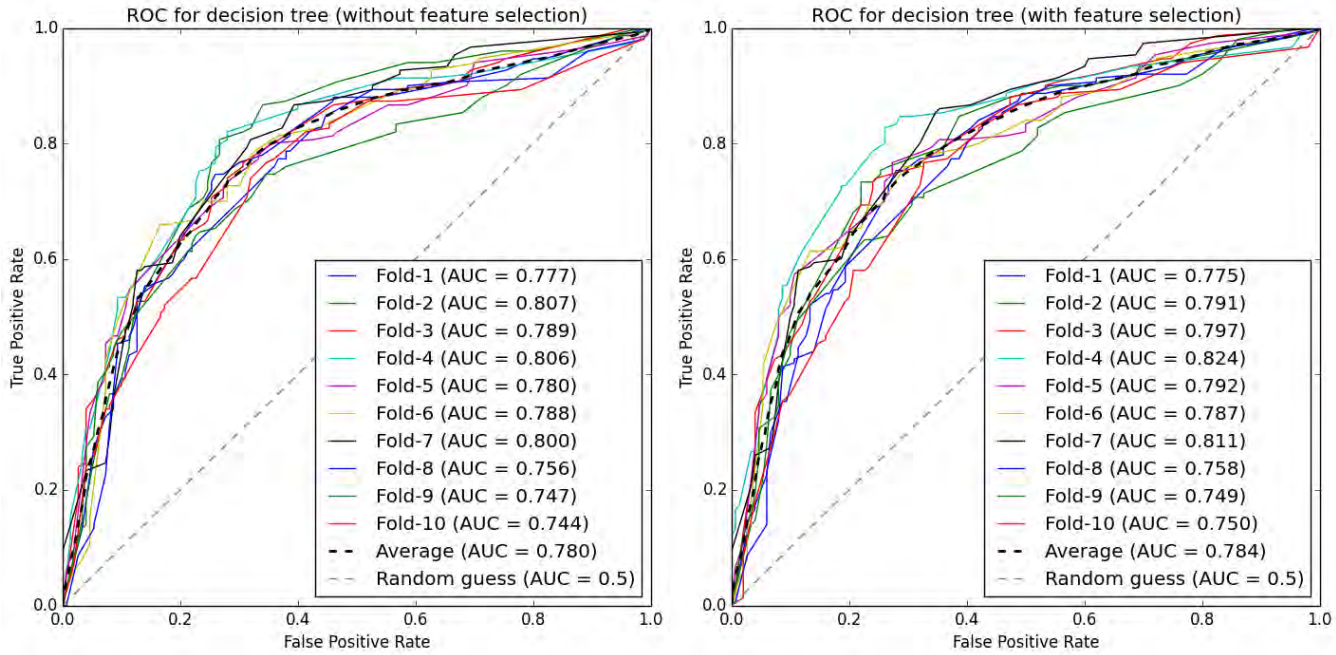


Figure C3. ROC curves for decision tree

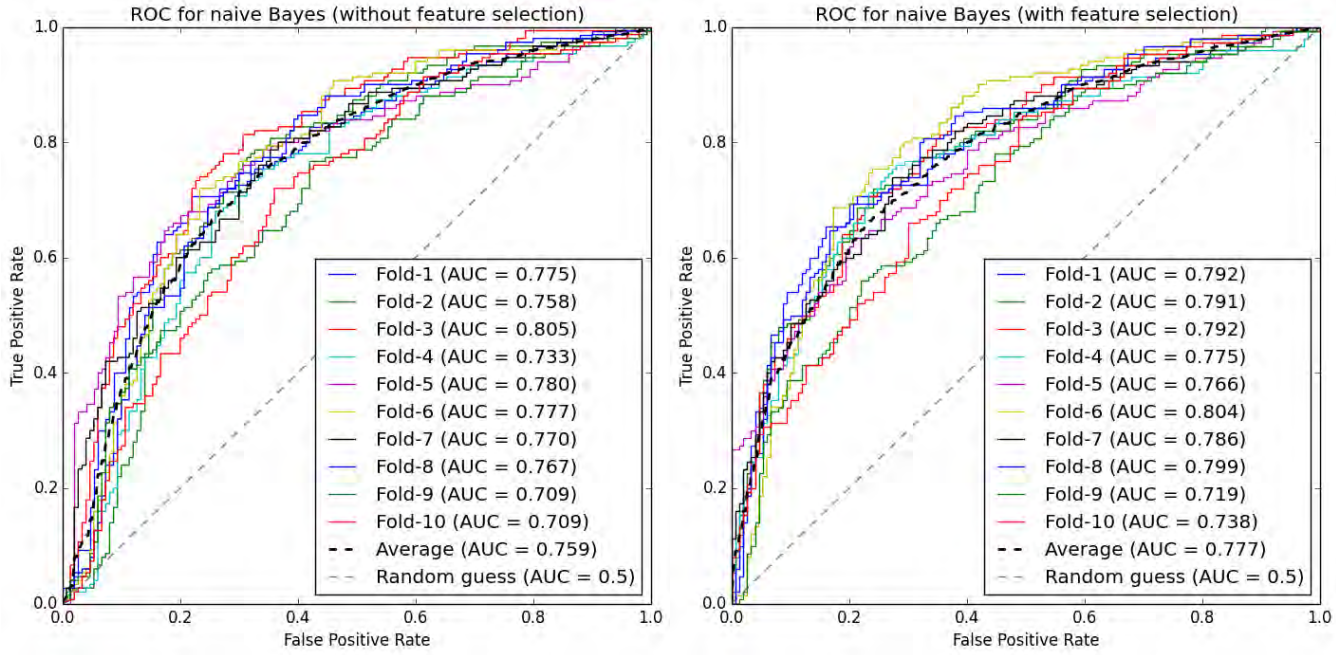


Figure C4. ROC curves for naïve Bayes

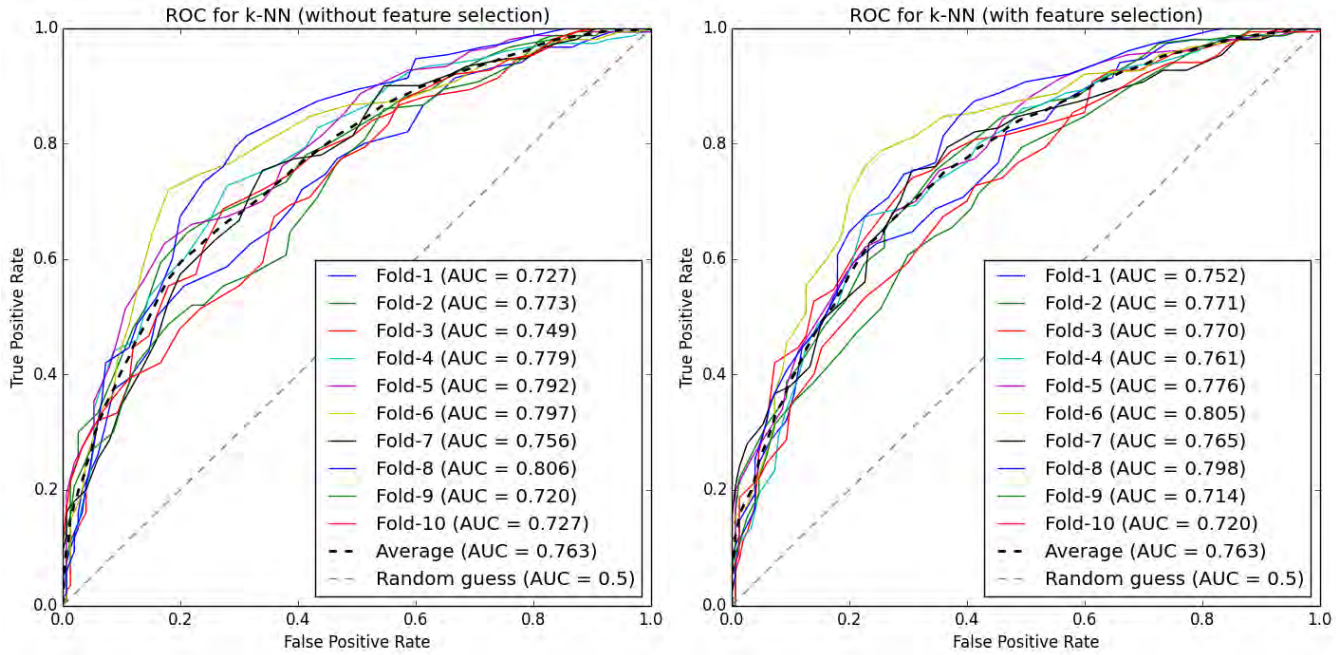


Figure C5. ROC curves for *k*-NN

APPENDIX D. RESEARCH INTEGRITY CERTIFICATES

NANYANG TECHNOLOGICAL UNIVERSITY

Wee Kim Wee School of Communication and Information (WKWSCI)

This certificate is awarded to

GARY KUEN WEN HAO

for the completion of Research Integrity Course Module in

Engineering and Technology Track

09 Mar 2014
Date

Prof Charles T. Salmon
Acting Chair, WKWSCI

NANYANG TECHNOLOGICAL UNIVERSITY

Wee Kim Wee School of Communication and Information (WKWSCI)

This certificate is awarded to

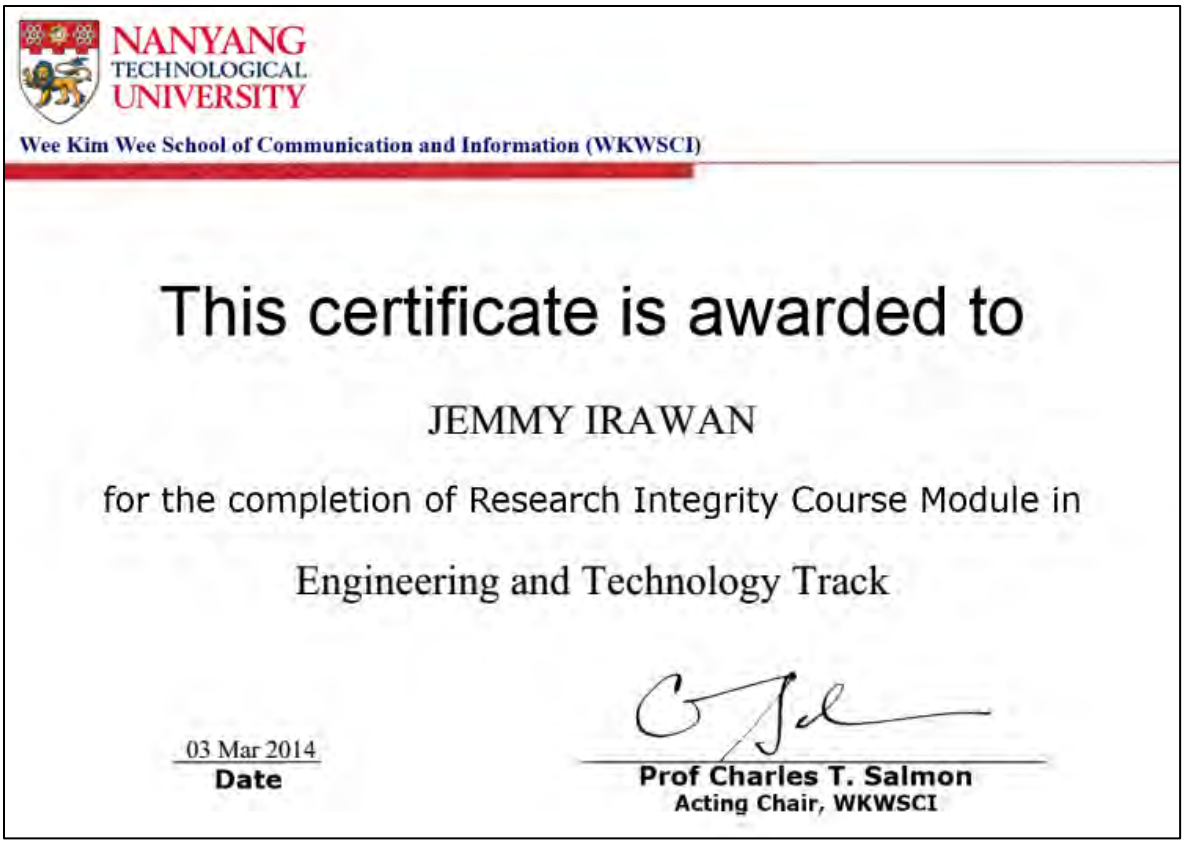
ZHOU SHU

for the completion of Research Integrity Course Module in

Engineering and Technology Track

20 Apr 2014
Date

Prof Charles T. Salmon
Acting Chair, WKWSCI


NANYANG TECHNOLOGICAL UNIVERSITY

Wee Kim Wee School of Communication and Information (WKWSCI)

This certificate is awarded to

JEMMY IRAWAN

for the completion of Research Integrity Course Module in

Engineering and Technology Track

03 Mar 2014
Date

Prof Charles T. Salmon
Acting Chair, WKWSCI

APPENDIX E. DECLARATIONS OF AUTHORSHIP

Wee Kim Wee School of Communication and Information
Declaration of Authorship

Name	Gary Kuen Wen Hao
Matriculation No	
Course Code	CI6229
Course Title	CRITICAL INQUIRY IN INFORMATION SYSTEMS
Lecturer/Tutor	A/P Alton Chua Yeow Kuan
Submission Date	5th May, 2014

Plagiarism and Collusion

Plagiarism : to use or pass off as one's own, the writings or ideas of another without acknowledging or crediting the source from which the ideas are taken.

Collusion : submitting an assignment, project or report completed by another person and passing it off as one's own (as defined in the NTU Honour Code (http://academicintegrity.ntu.edu.sg).

I understand the nature of plagiarism to include the reproduction of someone else's words, ideas or findings and presenting them as my own without proper acknowledgement.

I understand that there are many forms of plagiarism which include direct copying or paraphrasing from someone else's published work (either electronic or hard copy) without acknowledging the source; using facts, information and ideas derived from a source without acknowledgement; producing assignments (required to be independent) in collaboration with and/or using the work of other people; and assisting another person to commit an act of plagiarism.

I understand that the work submitted may be reproduced and/or communicated by the University or a third party authorized by the University for the purpose of detecting plagiarism.

Penalties for Plagiarism and Collusion

The penalties associated with plagiarism reflect the seriousness with which NTU view cheating, and its commitment to academic integrity. This could include the award of a failing grade for the assignment (or the course), or expulsion from the University. This policy applies to all work submitted, including oral presentations and/or written work.

Keep a copy of the Assignment

Be sure to make a copy of your work. If you have submitted your assignment electronically, also make a backup copy.

Declaration

I declare that this assignment is my own work, unless otherwise referenced, as defined by the NTU policy on plagiarism. I have read the NTU Honour Code and Pledge.

Signed GARY KUEN WEN HAO Date 3rd May 2014

Wee Kim Wee School of Communication and Information
Declaration of Authorship

Name	Zhou Shu
Matriculation No	
Course Code	CI6229
Course Title	CRITICAL INQUIRY IN INFORMATION SYSTEMS
Lecturer/Tutor	A/P Alton Chua Yeow Kuan
Submission Date	5th May, 2014

Plagiarism and Collusion

Plagiarism : to use or pass off as one's own, the writings or ideas of another without acknowledging or crediting the source from which the ideas are taken.

Collusion : submitting an assignment, project or report completed by another person and passing it off as one's own (as defined in the NTU Honour Code (http://academicintegrity.ntu.edu.sg).

I understand the nature of plagiarism to include the reproduction of someone else's words, ideas or findings and presenting them as my own without proper acknowledgement.

I understand that there are many forms of plagiarism which include direct copying or paraphrasing from someone else's published work (either electronic or hard copy) without acknowledging the source; using facts, information and ideas derived from a source without acknowledgement; producing assignments (required to be independent) in collaboration with and/or using the work of other people; and assisting another person to commit an act of plagiarism.

I understand that the work submitted may be reproduced and/or communicated by the University or a third party authorized by the University for the purpose of detecting plagiarism.

Penalties for Plagiarism and Collusion

The penalties associated with plagiarism reflect the seriousness with which NTU view cheating, and its commitment to academic integrity. This could include the award of a failing grade for the assignment (or the course), or expulsion from the University. This policy applies to all work submitted, including oral presentations and/or written work.

Keep a copy of the Assignment

Be sure to make a copy of your work. If you have submitted your assignment electronically, also make a backup copy.

Declaration

I declare that this assignment is my own work, unless otherwise referenced, as defined by the NTU policy on plagiarism. I have read the NTU Honour Code and Pledge.

Signed ZHOU SHU Date 3rd May 2014

Wee Kim Wee School of Communication and Information
Declaration of Authorship

Name	Jimmy Irawan
Matriculation No	
Course Code	CI6229
Course Title	CRITICAL INQUIRY IN INFORMATION SYSTEMS
Lecturer/Tutor	A/P Alton Chua Yeow Kuan
Submission Date	5th May, 2014

Plagiarism and Collusion

Plagiarism : to use or pass off as one's own, the writings or ideas of another without acknowledging or crediting the source from which the ideas are taken.

Collusion : submitting an assignment, project or report completed by another person and passing it off as one's own (as defined in the NTU Honour Code (http://academicintegrity.ntu.edu.sg).

I understand the nature of plagiarism to include the reproduction of someone else's words, ideas or findings and presenting them as my own without proper acknowledgement.

I understand that there are many forms of plagiarism which include direct copying or paraphrasing from someone else's published work (either electronic or hard copy) without acknowledging the source; using facts, information and ideas derived from a source without acknowledgement; producing assignments (required to be independent) in collaboration with and/or using the work of other people; and assisting another person to commit an act of plagiarism.

I understand that the work submitted may be reproduced and/or communicated by the University or a third party authorized by the University for the purpose of detecting plagiarism.

Penalties for Plagiarism and Collusion

The penalties associated with plagiarism reflect the seriousness with which NTU view cheating, and its commitment to academic integrity. This could include the award of a failing grade for the assignment (or the course), or expulsion from the University. This policy applies to all work submitted, including oral presentations and/or written work.

Keep a copy of the Assignment

Be sure to make a copy of your work. If you have submitted your assignment electronically, also make a backup copy.

Declaration

I declare that this assignment is my own work, unless otherwise referenced, as defined by the NTU policy on plagiarism. I have read the NTU Honour Code and Pledge.

Signed JIMMY IRAWAN Date 3rd May 2014