Identifying Reputation Collectors in Community Question Answering (CQA) Sites: Exploring the Dark Side of Social Media

Pradeep K. Roy
Computer Science and Engineering Department
National Institute of Technology Patna
Bihar-800005, India
Email: pradeep.cse15@nitp.ac.in

Jyoti P. Singh
Computer Science and Engineering Department
National Institute of Technology Patna
Bihar-800005, India
Email: jps@nitp.ac.in

Abdullah M. Baabdullah
Department of Management Information Systems
King Abdulaziz University, Jeddah, Saudi Arabia
Email: baabdullah@kau.edu.sa

Hatice Kizgin
School of Management
Swansea University
Fabian Way, Swansea, SA1 8EN, UK
Email: hatice.kizgin@swansea.ac.uk

Nripendra P. Rana*
School of Management
Swansea University
Fabian Way, Swansea, SA1 8EN, UK
Email: n.p.rana@swansea.ac.uk

*Corresponding author

Abstract
This research aims to identify users who are posting as well as encouraging others to post low-quality and duplicate contents on community question answering sites. The good guys called *Caretakers* and the bad guys called *Reputation Collectors* are characterised by their behaviour, answering pattern and reputation points. The proposed system is developed and analysed over publicly available Stack Exchange data dump. A graph based methodology is employed to derive the characteristic of *Reputation Collectors* and *Caretakers*. Results reveal that *Reputation Collectors* are primary sources of low-quality answers as well as answers to duplicate questions posted on the site. The Caretakers answer limited questions of challenging nature and fetches maximum reputation against those questions whereas *Reputation Collectors* answers have so many low-quality and duplicate questions to gain the reputation point. We have developed algorithms to identify the *Caretakers* and *Reputation Collectors* of the site. Our analysis finds that 1.05% of *Reputation Collectors* post 18.88% of low-quality answers. This study extends previous research by identifying the *Reputation Collectors* and

how they collect their reputation points.

Keywords: Community Question Answering, Reputation Collectors, Expert Users, Stack Exchange, Data Analysis

1. Introduction

Community Questions Answering (CQA) sites such as Yahoo! Answers (YA)[1], Stack Overflow (SO)[2], Stack Exchange[3], Quora[4] etc. are Web 2.0 based services, which allow people to seek information by asking questions and share knowledge by providing answers to questions asked by rest of the community members (Luo et al., 2016; Roy, Ahmad, Singh, Alryalat, Rana, and Dwivedi, 2018). Some CQA sites allow users to ask questions without any topic restriction such as YA, Quora, while other CQA systems are devoted to a specific area such as SO. SO was primarily developed for software developer to make it a useful resource of conceptual or code review questions for them. The content of this site sometimes supplements the official software documentation as well (Serna, Bachiller and Serna, 2017; Treude, Barzilay, & Storey, 2011). Therefore, the quality of content on this site is the most important thing (Aladwani et al., 2017; Hashim and Tan, 2015; Jin, Zhou, Lee and Cheung, 2013). Any compromise with quality of the content on this site will make it useless and people will be afraid using it. As digital and social media platforms and applications continue to disseminate, both positive and negative aspects associated with them are becoming increasingly apparent (AlAlwan et al. 2017; Aswani et al. 2018; Dwivedi et al. 2015, 2016, 2017a, 2017b, 2018; Ismagilova et al. 2017; Kamboj et al. 2018; Kapoor et al. 2017; Kapoor and Dwivedi 2015; Plume et al. 2016; Rathore et al. 2016; Shareef et al. 2017, 2018; Tamilmani et al., 2018). For example, every social media platform such as Facebook, Twitter and YouTube is now facing the problem of some naughty users who are trying to dilute these forums. A number of researchers have started finding the notorious activities on these forums (Garcia and Sikstrom, 2017; Fox and Moreland, 2015; Kaplan and Haenlein, 2010; Krasnova, Widjaja, Buxmann, Wenninger, and Benbasat, 2015; Stieglitz, Mirbabaie, Ross, and Neuberger, 2018). CQA site such as Stack Exchange is not an exception to this list and some people have started posting abusing content, duplicate questions and below-quality answers to this forum. This forum is highly dependent on its quality content, hence posting of any below-quality questions, or duplicate questions and their low-quality answers may be seen an act of negative activity on this site. These activities will kill the very purpose of the forum for which it was being developed and used. Traditionally the quality of the posts (questions as well answers) is evaluated and maintained by the community users only. Users can vote up or vote down a post (questions or answers) to express their views on posts. A sample post of CQA site having different attributes can be seen from Figure 1.

[1] https://in.answers.yahoo.com
[2] https://stackoverflow.com/
[3] https://stackexchange.com/
[4] https://www.quora.com

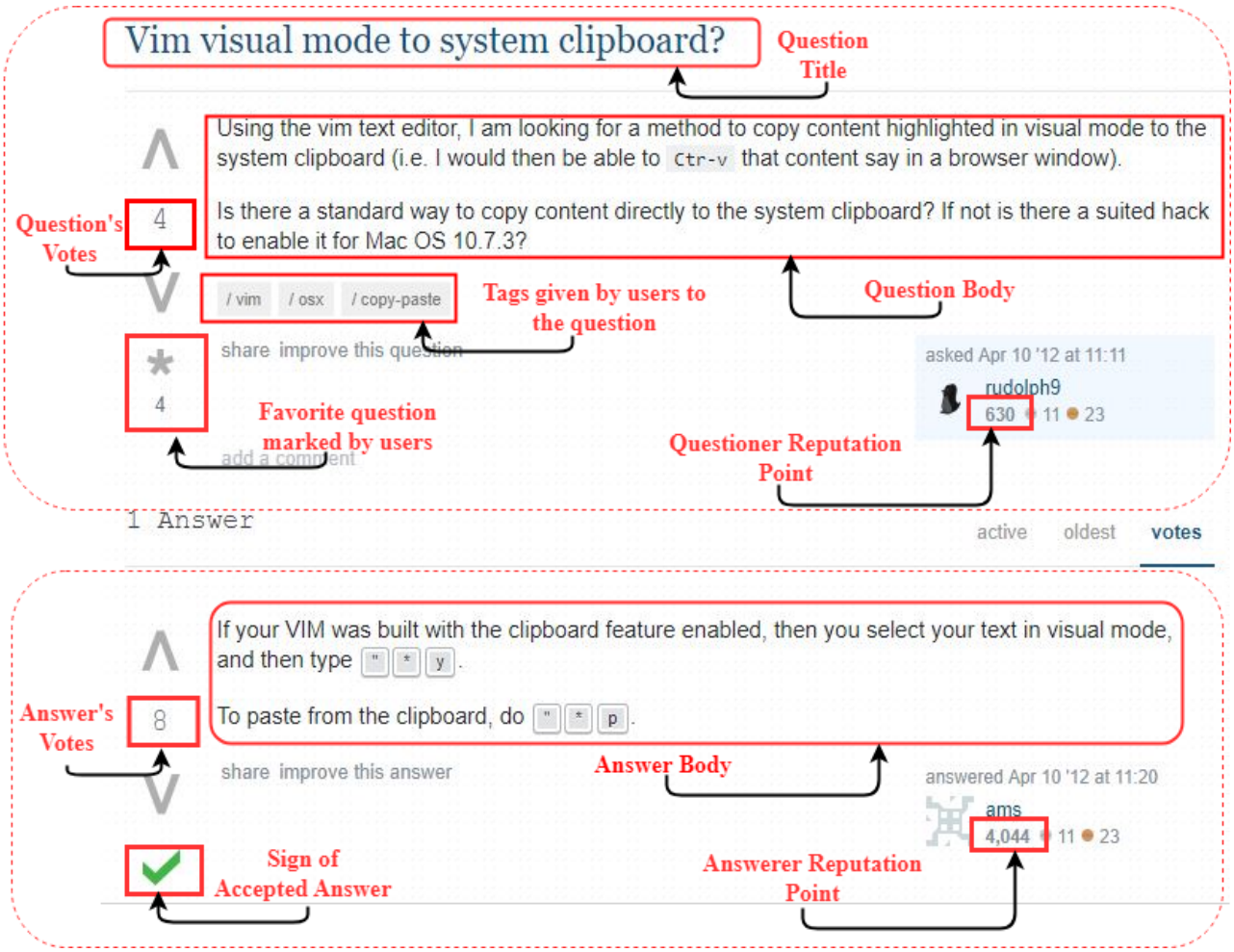

Figure 1: A sample of Stack Exchange post with different attributes.

To encourage user participation in the site, reputation points and badges system are in use.
For example, the questioner gets +2 reputation points by accepting an answer while the answerer gets +15 reputation points. Other activity that a user can do is, he/she can comment on any answers if they are not satisfied with the posted answer. User can vote positive or negative with their satisfactory level. Every activity that a user performs with a posted answer, the reputation point is updated in respective questioner/answerer reputation point. The summary of activities along with their reputation point change is shown in Table 1.
If a questioner has not received any answer for his/her question, and he/she needs the answer immediately then he/she may assign a bounty on that particular question to attract other user to answer the question. If the answer of a bounty questions is accepted, the bounty is added to the answerer's reputation and the same reputation point is subtracted from the questioner's reputation. There is no limit on the bounty, any number of reputation point can be added to the question as bounty. However, a user can earn a maximum of 200 reputation (except bounty) points per day, according to Stack Overflow community policy rules.

~~The users collect certain reputation point if their answer or question is voted up, or answer is accepted by questioner, and so on, the detail scheme of gaining the reputation by users is shown in Table 1.~~

Table 1: Stack exchange reputation scheme for community users

Action	Reputation Change
Answer is voted up	+10

Question is voted up	+5
Answer is accepted	+15 (+2 to acceptor)
Question is voted down	-2
Answer is voted down	-2 (-1 to voter)
Experienced Stack Exchange user	Onetime +100
Accepted answer to bounty	+bounty
Offer bounty on question	-bounty

The user's privilege keeps on increasing as they earn more reputation points. There are five classes of privilege[5] such as (i) *documentation privilege*, (ii) *creation privilege*, (iii) *communication privilege*, (iv) *moderation privilege*, and (v) *milestone privilege.* Among these privileges, the milestone is the highest whereas documentation privilege is the lowest one: i) with *documentation privilege*, a user has authority to approve or reject the changes made on the posts, can comment on the proposed changes and so on. ii) the main authority *creation privilege* is to create a new *tag* for the site, iii) with communication privilege, a user can create gallery chat rooms where only specific users may chat, iv) the main authority of *moderation privilege* is *marked questions as protected*. A protected question prevents answers being added by anonymous and new users, v) finally, a *milestone privilege* makes a user as a trusted user of the site. Hence, they can delete the questions having negative votes, also delete the low-quality answers if there is no hope to improve it further and so on. A user with this privilege may get a special access to the data collected from the community users. A group of users has started abusing the very purpose of reputation by doing activities which are not supposed to be done by genuine members. They do so to gain more privilege and make them visible to other community members. These users are termed as *Reputation Collectors (*Srba and Bielikova, 2016a*)*. *Reputation Collectors* attempt to answer so many questions to earn reputation points without giving much attention to quality of questions. They answer most of the duplicate questions, which appear on Stack Exchange as found in our analysis.

~~The duplicate questions are against the principle of Stack Exchange community rules and places too much of burden on the system and dilute the ideology of the community.~~ The duplicate questions are undesirable as a similar question already exists. Also, it increase the workload of the site moderators and waste the computing and storage space of the site (Ahasanuzzaman, Asaduzzaman, Roy, and Schneider, 2016; Zhang, Lo, Xia, and Sun, 2015). A number of researchers have started to take the problem of duplicate questions (Ahasanuzzaman, Asaduzzaman, Roy and Schneider, 2016; Zhang, Lo, Xia, and Sun, 2015). One way to discourage users posting duplicate questions is by not providing any answers to those questions or simply adding those questions to their master questions. But mere pointing to master question will not fetch any reputation point. Hence, some users answer these question as they are easy to answer (some answers to master question is already there). The duplicate and low-quality questions are very hard to stop as they are mostly asked by new user or users who only ask questions but never contribute by giving answers. They have been termed as "*One day flies*" (Slag, Waard, and Bacchelli, 2015), "Help Vampires" and "Noobs" (Srba et al., 2016a). One day flies, noobs or help vampires ask questions, get their answers and vanish. They are difficult to stop, but it is the *Reputation Collectors* who encourage them to behave this way by answering such repetitive and low-quality questions. If *Reputation Collectors* stop answering their questions, their presence can be automatically controlled. In our opinion, this phenomenon represents the dark side of the CQA site as they are degrading the content quality of the site, thereby making it less useful and flooded with duplicate and low-quality questions. To the best of our knowledge, till now no technique has been proposed to identify these *Reputation Collectors* who are indirectly responsible for motivating the noobs and help vampires by posting the low-quality answers. In this paper, we focus on the identification of *Reputation Collectors* who trick the system to gain reputation.

[5] https://stackoverflow.com/help/privileges

We also characterise the *Reputation Collectors* and *Caretakers* based on their answering activity, type of question answered and reputation point against those answers. These characteristics clearly distinguished *Reputation Collectors.*

The remainder of the paper is organised as follows: Section 2 reviews the related work. The statistics and other details of Stack Exchange are explained in Section 3. Section 4 elaborates our proposed model to characterise *Reputation Collectors* and *Caretakers* work. In Section 5, we present the results of our evaluation. The discussion on the result is written in Section 6. In Section 7, we conclude the research by highlighting the key limitations and future works.

2. Literature Review

The pillar of success of Community Question Answering (CQA) sites is knowledge sharing behaviour by individuals. The motivation and reason for knowledge sharing on various forums have been investigated by several researchers (e.g. Davis and Agrawal, 2018; Lu and Hsiao, 2007). A comprehensive research has been undertaken by (Srba and Bielikova, 2016b), where they covered various research issues of CQA site, including the selection of best answer (Lee, Rodrigues, Kazai, Milic-Frayling, and Ignjatovic, 2009; Li, He, Jeng, Goodwin, and Zhang, 2015; Liu, Liu, and Yang, 2010; Xie, Nie, Jin, Li, and Li., 2015; Yao, Tong, Xie, Akoglu, Xu, and Lu, 2015), expert finding and the topic modelling. They covered 265 different articles to categorise the work done in CQA. Most of the research works deal with expert findings (the bright side of CQA sites), a very limited work has been done on negative aspects of the CQA sites such as duplicity of the post, low-quality question and answers. Ponzanelli, Bacchelli, Lanza, and Fullerton (2014) proposed a system to identify low-quality questions on Stack Overflow. They made a classification system using textual and social features. They reduced the misclassification rate and minimised the review queue for deletion of low-quality questions. Their model achieved the precision of 0.68 with effective review queue reduction of 9%. Saha, Saha, and Perry (2013) reported that the volume of unanswered question has increased rapidly in the last few years, which creates a negative impact on the reputation of CQA sites. They found that the lengthy questions (whose word length is more than average word length) were mostly unanswered. Asaduzzaman, Mashiyat, Roy, and Schneider (2013) reported that the question posted between 8 p.m. and 11 p.m. received the answers more quickly than posting at other time. Chua and Banerjee (2015) found some questions not getting a single answer for a reasonable long time. They suggested that the answer-ability of questions depends on both metadata and the content. To validate the proposed framework they used a case study of Stack Overflow, where 3000 questions were selected and divided equally between those answered and unanswered. Their findings confirmed that the questions asked by new users received quick answers, and the question with clear title, short description, and with few tags attract more answerers compared to the complex questions.

Slag et al. (2015) identified a group of users from Stack Overflow dataset called 'One-day flies', who join the community, ask a question and then never come back again. To find the reason behind this, authors analysed the post of such users and found that, their posts were duplicates of the other post or they did not tag the question accurately, hence, their question receives very less number of user views. Also, due to the duplicate post, their questions are closed either by themselves or by the site moderators. On the other hands, the posts of one-day flies are easy to answer, hence, *Reputation Collectors* are targeting such posts and answer them to increase their reputation. The issue of duplicate questions was captured by (Zhang et al., 2015) as well. They proposed a system called *DupPredictor* to find the similarity between the two questions based on certain factors such as title of question, descriptions and the tag present in the question. Based on the similarity score, they detected if the posted question is a duplicate or not. Ahasanuzzaman et al. (2016) mined the duplicate questions on the Stack Overflow (SO) and confirmed that the quality of content is failing (Srba and Bielikova, 2016a). The SO site is currently handled by some moderators, who manually filter the

duplicate and low-quality questions. Due to manual evolution, many duplicate questions are unidentified and good questions are marked as duplicate (Zhang et al., 2015).

Srba and Bielikova (2016a) analysed the content of CQA site SO for the period of January 2011 to September 2014 and found that the content quality was degrading with the time. They tried to find the reason behind the low-quality content of sites. They categorised the community users in four different types called: 1) Help Vampires-users who ask questions without prior research, 2) *Reputation Collectors*-users answer as many questions as possible (commonly, regardless of their insufficient knowledge of a questions topic) primarily to gain a reputation 3) Noobs-these are low-expertise users who create mainly trivial, poor-quality questions, and 4) *Caretakers*-these are experts who want to keep the system clean with valuable content. The major portion of low-quality contents are posted either by help vampire or by noobs. Also, authors confirmed that the *Reputation Collectors* are motivating these two types of users to post the low-quality content. However, they were silent about identifying such *Reputation Collectors* automatically. Liu, Liu, Zhou, Zhang, and Ma (2017) proposed a system that detected the collusive spanning activity on CQA sites. They collected data from two different sites namely Zhubajie.com and RapidWorkers.com. Users are created a number of accounts on these sites and posts the questions and answers to promote a particular product. They cluster the questions and answers separately, and by using the combined factor graph model (CFGM) classify them. Their model achieved precision, recall and F1-score as 0.85, 0.91, 0.88 for questions and 0.92, 0.84, 0.90 for answers respectively. A summary of some potential researches that focused to find the duplicate, low-quality, and unanswered questions from the various CQA sites are presented in Table 2.

To the best of our knowledge none of the existing work is focused on finding *Reputation Collectors* and their behaviour. None of them provide any methodology to identify the users who are responsible for the stated problem. Also, no thorough analysis has been done regarding the damage these *Reputation Collectors* are doing to a CQA site. Therefore, in this paper, we propose a method to detect these *Reputation Collectors*, so that appropriate steps can be taken to maintain the quality of CQA sites. Also, we conducted a thorough analysis of behaviour of *Reputation Collectors* and the adverse effect they have on a CQA site.

Table 2: Summary of relevant work on duplicate, unanswered questions and *Reputation Collectors* of CQA sites

Source	**Problem statement**	**Approach**	**Results**
Asaduzzaman, M., Mashiyat, A. S., Roy, C. K., & Schneider, K. A. (2013).	Finding unanswered questions on stack overflow	Random Forest, J48 classifier.	Precision: 0.38 and Recall: 0.45.
Saha, R. K., Saha, A. K., & Perry, D. E. (2013).	Cause of unanswered questions in software sites.	J48, KNN, Naïve Bayes, Random Forest.	Precision: 0.88, Recall: 0.91, and F1-score: 0.90.
Ponzanelli, L., Mocci, A., Bacchelli, A., Lanza, M., & Fullerton, D. (2014).	Improving low-quality post detection on stack overflow	Genetic Model.	Precision: 0.68, with effective low-quality question's review queue reduction of 9%.
Chua, A. Y., & Banerjee, S. (2015).	Studying question answerability in stack overflow	Hierarchical logistic regression	Accuracy: 77.50% for predicting of answerability of a question
Slag, R., de Waard, M., & Bacchelli, A. (2015).	Why the vast majority of stack overflow users only posts once.	Analytical	The question posted by the users called *one day flies* are unable to attract answers from the peer users. Hence, such users have very low participations.
Zhang, Y., Lo, D., Xia,	Duplicate question detection in Stack	DupPredictor	Recall rate @20: 63.8%.

X., & Sun, J. L. (2015).	Overflow		
Ahasanuzzaman, M., Asaduzzaman, M., Roy, C. K., & Schneider, K. A. (2016).	Mining Duplicate Questions of stack overflow	BM25, Dupe, DupePredictor and SO search	Recall-rate@20: 66.10%
Srba and Bielikova, (2016)	Why is stack overflow failing?	Analytical	Users like noobs, help vampire, reputation collectors are the main source of low-quality content on SO.
Liu, Y., Liu, Y., Zhou, K., Zhang, M., & Ma, S. (2017).	Detecting collusive spamming activities in community question answering.	CFGM	Precision: 0.85, Recall: 0.91, F1-score: 0.88 and AUC: 0.95.

3. Research Methodology

The major goal of this research is to find out *Reputation Collectors* and *Caretakers*. The methodology to find and verify the *Reputation Collectors* is depicted through a block diagram in Figure 2. The complete methodology is grouped into two parts (i) extracting *Reputation Collectors* and (ii) verifying *Reputation Collectors*. The steps are explained in detail in subsequent paragraphs. Similar steps are followed to find the *Caretakers* where low-quality answer module of Figure 2 is replaced with high-quality answers module.

Data Preparation: The dataset for current research is downloaded from the Stack Exchange archive (Stack Exchange, 2016). It contains 150+ zip files of different topics available on site each containing data of different sub-domain of Stack Exchange consortium of sites. Each zip file contains eight different 'xml' files each containing a different set of information. We extracted data from two xml files (i) User.xml and (ii) Post.xml. From User.xml, we select two fields *userId* and *Reputation* and from Post.xml, *postId*, *owner*, and *AcceptedAnswerId* are selected and stored into a csv file.

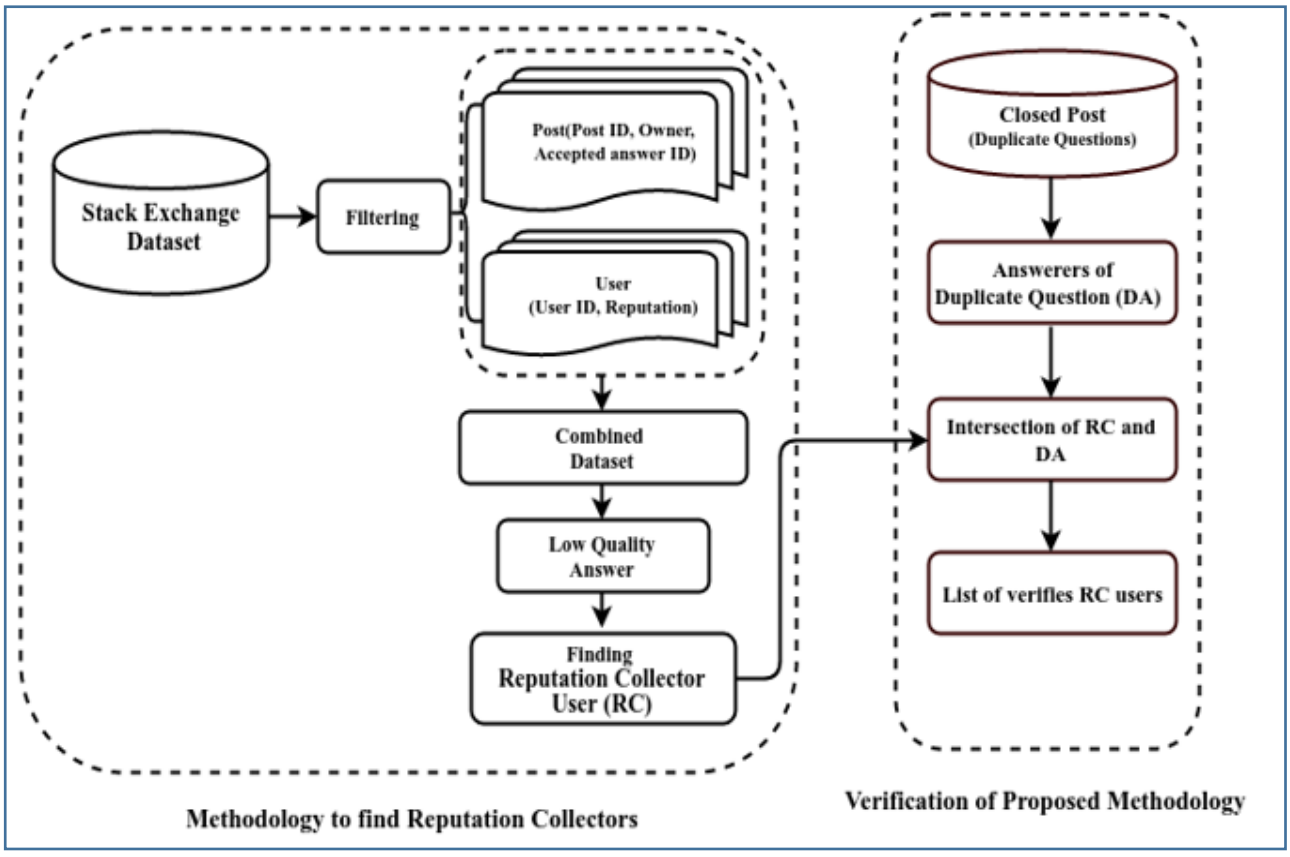

Figure 2: Framework of proposed methodology

Low-quality and high-quality answers: Combined dataset is labelled as low-quality answers and high-quality answers based on user votes. Low-quality answers are those answers, which receive either zero or negative user vote. Whereas high-quality answers characterised by having at least three positive votes from the community users. The reason behind this is an early posted answer may receive some number of votes till another good quality answer appears. Once the other answers become available, the answer having good quality content start attracting more number of votes than that of early posted answer. Hence, we can say an initial answer has a greater chance to obtain one or two votes, however, more than that number of votes are indicates that the answer is really of good quality with respect to the question. Based on rationale, we choose three votes as minimum for considering an answer is a high-quality answer. The distribution of votes on Ask Ubuntu dataset is shown in Figure 3. One can find that 0, 1 and 2 votes are very common but 3 or higher votes are not so.

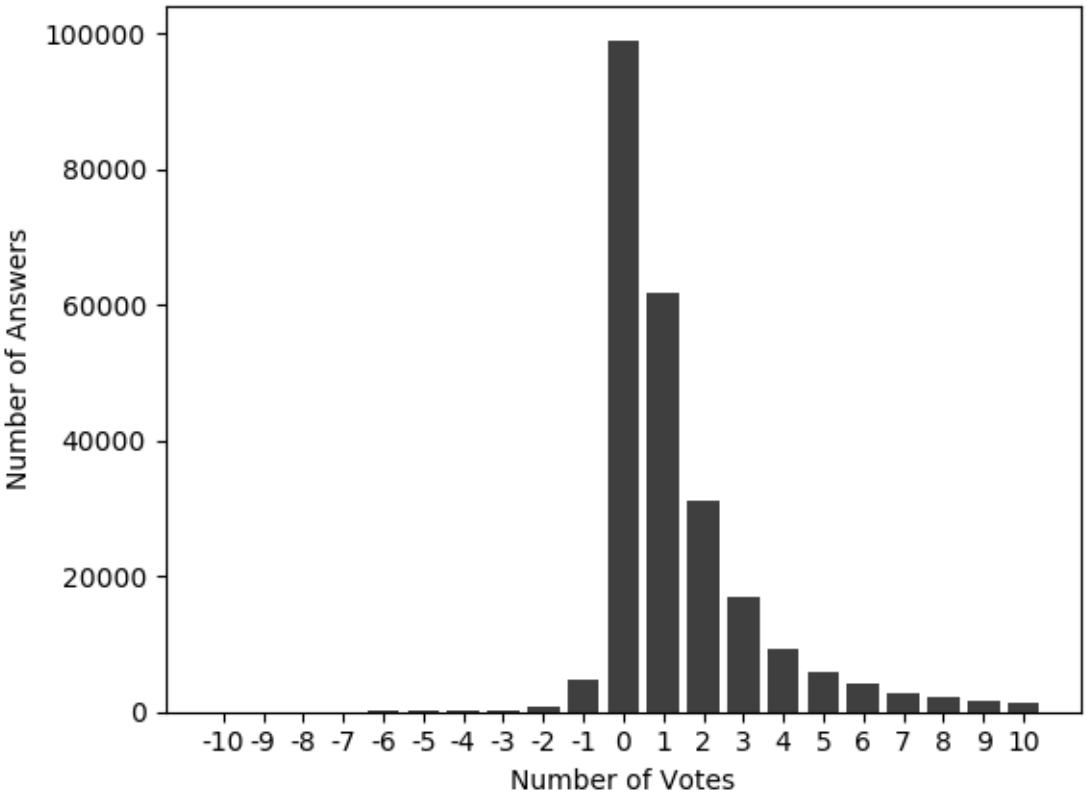

Figure 3: Votes distribution on Ask Ubuntu dataset

The reason behind this is an early posted answer may receive more number of votes till another good quality answer will appear. Once the other answers available then, the answer having good quality content may attract more number of votes than that of early posted answer. Hence we can say an initial answer has a greater chance to obtain one or two votes, however, more number of votes only received if they have good quality content with respect to the question as well as other competitive answers. Based on these, we choose three votes minimum for considering an answer is a high-quality answer.
The answers, which receive either '1' or '2' user votes are of moderate quality answers, which do not play any role in finding the *Reputation Collectors* or the expert user, so we have ignored these answers for our proposed methodology.

3.1 Finding Reputation Collectors

The complete procedure of finding *Reputation Collectors* is provided in Algorithm 1. The input to the algorithm is the csv file containing Questions, Answers, and User information. The output of the algorithm is the tentative *Reputation Collectors*. First, we find out all the answers, which had obtained votes of less than or equal to zero and yet the answer had been accepted. After finding the list of such answers, we found the list of all the users who had given those answers. These are the users who are suspected to be *Reputation Collectors*. Once, we obtain the list of these suspected users, we find the list of all the questions these users have answered and also the list of users who had asked those questions. After obtaining this list of questioners, we find the average reputation of the users who had answered more than 50% of answers from questioners who had obtained less than average reputation points of the users of the site termed as *Reputation Collectors.* Here, it is our assumption that, if a user answers more than 50% of questions posted by the users like noobs, help vampire, and one day flies. Then we can say, such users are not choosing the good quality question to answer, moreover, by answering the simple, duplicate, or low-quality questions mostly they continuously increasing their reputation point.

Algorithm 1. Finding *Reputation Collectors*

Input: Questions, Answers and Users file

Step 1: *ans*:= All answers in *Programmers* dataset
Step 2: *k=0*
Step 3: *suspUsers=[], suspAns=[]*
Step 4: for *i=1* to length(*ans)*:
    if *ans['Score']<=0 and ans['Accepted']=1:*
        suspAns[k]=ans[i]
        suspUsers[k]=ans[i]['Owner']
        k=k+1
Step 5: *qns*=All questions in *Programmers* dataset
Step 6: *low_ans_ques=[]*
Step 7: for j=1 to length(*suspAns*):
    low_ans_ques[j]=suspAns['Owner']['ParentId'] /
    /* ParentId denotes the Question id to question of the selected answer*/
Step 8: *Reputation_Collectors=[], q=0*
Step 9: *users:=* All users and their details in *Programmers* dataset
Step 10: for each *low_ans_ques['Owner']:*
    rep=0
    for *n=1* to length*(low_ans_ques['Owner'])*
        for p=1 to length(*users*):
            if *users['Id']==low_ans_ques['parentId']:*
                r=users['Id']['Reputation']
        rep=rep+r
    *avg_rep=rep/*length*(low_ans_ques['Owner'])*
    q=q+1
    if avg_rep<Avg_Rep_of all users reputation:
        Reputation_Collectors[q]=low_ans_ques['Owner']

Output: List of Reputation Collectors in the array *Reputation_Collectors[]*

3.2 Finding Expert Users or Caretakers

Following a similar way, we also find the *Caretakers* of the site using Algorithm 2. First, we find out all the answers, which had obtained votes of greater or equal to three and the answer had been accepted. After finding the list of such answers, we found the list of all the users who had given those answers. These are the users among who we expect to find *Caretakers*. Once we obtain the list of these users, we find the list of all the questions these users have answered and also the list of users who had asked those questions. After obtaining this list of questioners, we find from the set of users who had answered more than 50% of answers from questioners who had obtained more than average reputation points. As in the case of Reputation collectors (Algorithm 1), for caretaker also, we have assumed the threshold as 50%. That means, if a user answers more than 50% of questions posted by the users whose reputation point is greater than the average reputation point, we classify them as *Caretakers.*

Algorithm 2. Finding Expert Users or Caretakers

Input: Questions, Answers and Users file

Step 1: *ans*:= All answers in *Programmers* dataset
Step 2: *k=0*
Step 3: *suspUsers=[], suspAns=[]*
Step 4: for *i=1* to length(*ans)*:
    if *ans['Score']>=3 and ans['Accepted']=1:*
        suspAns[k]=ans[i]
        suspUsers[k]=ans[i]['Owner']
        k=k+1
Step 5: *qns*=All questions in *Programmers* dataset
Step 6: *high_ans_ques=[]*
Step 7: for *j=1* to length (*suspAns*):
    high_ans_ques[j]=suspAns['Owner']['ParentId']
    /* ParentId denotes the Question id to question of the selected answer*/
Step 8: *Expert_Users=[],q=0*
Step 9: *users:=* All users and their details in *Programmers* dataset
Step 10: for each *high_ans_ques['Owner']:*
    rep=0
    for *n=1* to length*(high_ans_ques['Owner'])*
        for p=1 to length(*users*):
            if *users['Id']==high_ans_ques['parentId']:*
                r=users['Id']['Reputation']
        rep=rep+r
    *avg_rep=rep/*length*(high_ans_ques['Owner'])*
    q=q+1
    if avg_rep>Avg_Rep_of all users reputations:
        Expert_Users[q]=high_ans_ques['Owner']

Output: List of Expert Users in the array *Expert_Users[]*

3.3 Verifying the Reputation Collectors using Duplicate Answers

In the first part of our proposed methodology, we find the list of Reputation *Collectors. Reputation Collectors* are users who are only concerned with gaining reputation and often give low-quality or

repetitive answers. Therefore, according to our intuition *Reputation Collectors* should also target duplicate questions. Thus, duplicate questions give us a method to verify the *Reputation Collectors*. If the predicted *Reputation Collectors* also give answers to duplicate questions, then we can say with conviction that the predicted users are indeed *Reputation Collectors.*

Closed Post: Closed posts are duplicate questions, which are marked as close and they cannot be edited. We collected all the closed posts (Questions) of *Programmer* dataset. A question is closed due to several reasons, such as Duplicate post, off-topic, unclear of what is asked, too broad and primarily opinion based, as explained in the documentation of the stack exchange site (Stack Exchange, 2016; Stack Overflow, 2016). The duplicate posts have a major role in the low-quality content of the site.

Answerers of Duplicate Questions (ADQ): Once the list of closed questions is extracted from the dataset, we find all the answerers who have answered such questions.

1. **The intersection between RC and ADQ:** To verify that the predicted *Reputation Collectors* are indeed targeting duplicate questions, we perform the intersection operation between the list of answerer of the duplicate question (ADQ) and the list of predicted *Reputation Collectors* (RC). If the predicted users also give an answer to duplicate questions, then we can say such users are definitely *Reputation Collectors*. To visually analyse the true nature of the interaction of *Reputation Collectors* with the duplicate questions, we generate the duplicity graph. *Duplicity graph* is the pictorial representation of the interaction between a question and its answerer(s). It is a directed graph arising from a question node and ending at the node representing the answerer of that question. An example of the duplicity graph can be seen in Figure 4. The head of the directed edge represents the answerer of the question, whereas tail represents the question. A part of Figure 4 can be seen from Figure 5. Such a representation of questions and answerers enables us to analyse the user behaviour that is difficult to do by a normal data analysis. The graph is drawn using Gephi tool (Bastian, Heymann, and Jacomy, 2009).

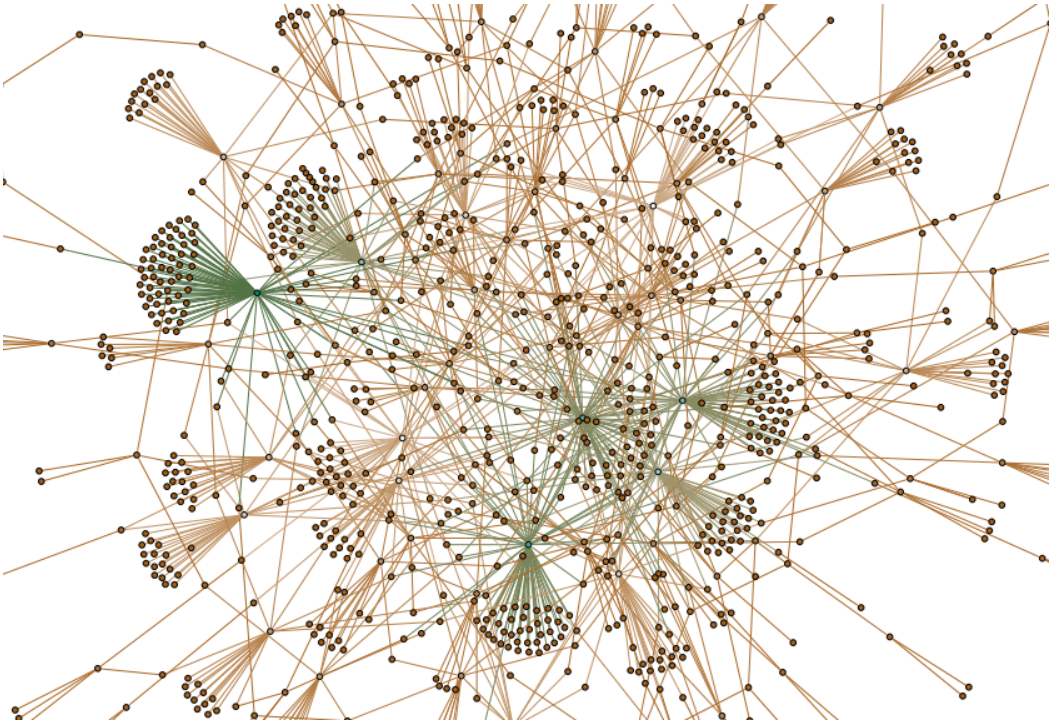

Figure 4: Duplicity Graph of closed post

Table 3: Complete data description of *Programmers* dataset of Stack Exchange

Total Users	176,587
Total Questions	22,653
Total Answers	142,191
Average Reputation of all users	183
Total Low Quality Answers (Votes<=0)	23,791

4. Data Analysis and Results

We used the Stack Exchange data dump released in March 2016. The dataset contained questions, answers and users' information during the period of August 2008 to December 2015 on 150 different topics such as *'Android, Programmers, Mathematics'* to name a few. The testing was done on nine different topics such as *'Ask Ubuntu, Apple, Code Review, DBA, Electronics, Gaming, Mathematics, Physics, and Programmer'*. We started our analysis with Programmers topic dataset because it is one of the biggest datasets. The statistics of the dataset are shown in Table 3. *Programmers* dataset has 176,587 users, which have posted 22,653 questions and 142,191 answers. The average reputation points of all users for this dataset were found to be 183. Our analysis of this dataset shows that 23,791 answers can put into a low-quality because they fetched zero or less than zero votes despite being posted for a long time. These low-quality answers are 16.7% of total answers. Further, we found that 9,686 users are responsible for posting all these low-quality answers. Out of these 9,686 users, 336 users posted low-quality answers, which were accepted by the questioner at least once. This behaviour was strange and such answerers are suspected to be *Reputation Collectors*. We further analysed that these 336 users had posted 6,677 low-quality answers, which were 28% of the total low-quality answers on the dataset. However, we could not classify all these 336 users as *Reputation Collectors,* as some of them might be inexperienced users who were posting low-quality content unintentionally. Among these users, we need to find such users who deliberately sabotage the quality of the CQA site, in order to gain reputation and have more privileges on the site. Such users might target questions asked by noobs type of users. The noobs are the users who have a low level of expertise and post trivial and low-quality questions on the site (Srba and Bielikova, 2016a). Thus, the *Reputation Collectors* are identified from the list of answerers, by selecting such users from the list who have answered more than 50% of questions asked by low reputation users (i.e., with reputation less than the average reputation), and at least five such answers have been accepted. Our analysis yielded 161 such users who satisfy all the criteria (see Table 5).

Table 4: Analysis of data related to duplicate questions

Number of Questions closed due to duplicity	2,123
Number of answers duplicate questions received	8,420
Number of users who answered duplicate questions	3,993
Number of predicted reputation collectors who answered duplicate questions	148 (3.7% of duplicate question answerers)
Number of answers given to duplicate questions by predicted reputation collectors	1,498 (17.8% of all answers to duplicate questions)

Another haunting problem, which is increasing on Stack Exchange is that of duplicate questions. During analysis, we came across some answers given by users whose questions had been closed because such questions were a duplicate of some other questions. We extracted the duplicate questions from our dataset and found that there are 2,123 such questions, which have received a total of 8,420 answers. All such answers were given by 3,993 different users. The complete

statistics of duplicate questions are presented in Table 4.

We checked how many *Reputation Collectors* identified by us had given answers to duplicate questions, and found that out of the 161 *Reputation Collectors,* 148 had given answers to duplicate questions. The predicted *Reputation Collectors* are only 3.7% of such users (users who have posted answers to duplicate questions), but they post 17.8% of all answers to duplicate questions. This supports our assumption that *Reputation Collectors* are not concerned with the quality of answers or questions, but only care about increasing their reputation. To clearly represent the situation, we analyse a part of Duplicity Graph presented in Figure 5.

From the Duplicity Graph shown in Figure 5, we can see that a number of *Reputation Collectors* are targeting one duplicate question. It is interesting to note that so many suspicious users are targeting only one question, and the question has been closed due to Duplicity. This sub graph of Figure 4 accurately captures the aim of Reputation Collectors. Another behaviour of the *Reputation Collectors* can be seen in Figure 6, where a *Reputation Collectors* is targeting a lot of closed questions. This shows that the *Reputation Collectors* is deliberately targeting such questions, to earn easy reputations as shown in Figure 6. The behaviour of *Reputation Collectors* as seen from Figure 5 and Figure 6 prove that such users are a menace to CQA sites, and are only bothered with collecting reputation in order to gain more and more privileges on the site. From Table 5, it can be seen that the total number of *Reputation Collectors* identified by our system is a very small number covering only 0.09% of total users in *Programmers* dataset. Although they are small in number, but they are responsible for 14.8 % of total low-quality answers posted on *Programmers* dataset.

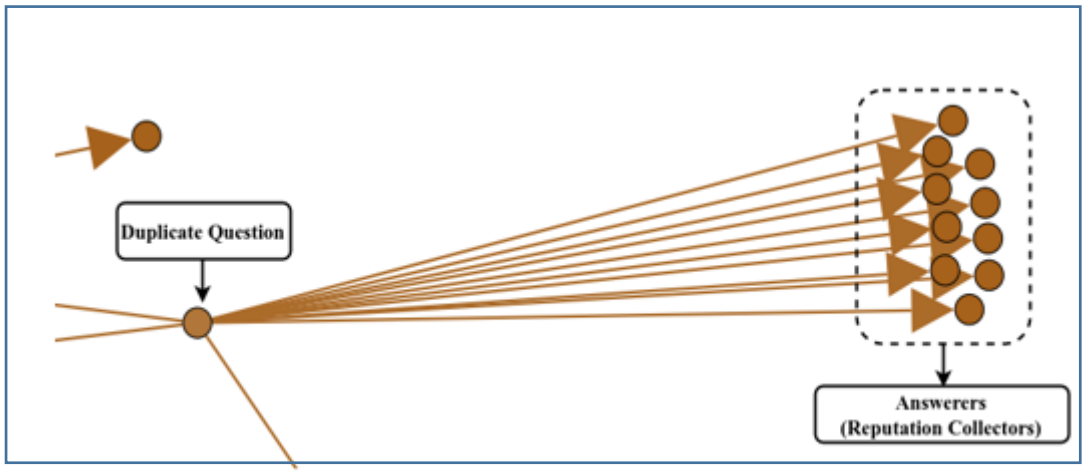

Figure 5: *Reputation Collectors* targeting one duplicate question

Table 5: Analysis of behavior of reputation collectors

Number of predicted *Reputation Collectors*	161 (0.091% of total users)
Number of low-quality answers given by *Reputation Collectors*	3,529 (14.8% of all low quality answers)
Average reputation of all questioners targeted by *Reputation collectors*	58.48
Average reputation of *Reputation Collectors*	494
Average Reputation collected per accepted answer	126

Reputation Collectors behaviour is in contrast to the *Caretakers* on *Programmers* dataset. Our algorithm identified 122 users as *Caretakers*, which is 0.07% of total users. They posted only 5,913 answers and they fetched 240 reputation points against each answer. *Caretakers* mostly answer those questions which are asked by the reputed users, as it reveals that *Caretakers* really answers good quality and challenging questions. The complete statistics of *Caretakers* is shown in Table 6.

Table 6: Analysis of behaviour of expert users

Number of expert users analysed	122 (0.07% of total users)
Number of answers given by expert users	5913 (4.15% of all answers)
Average reputation of questioners targeted by expert users	221.98
Average Reputation of expert users	379.89
Average Reputation collected per accepted answer	240

We compared the behaviour of *Caretakers* with *Reputation Collectors*. Based on the findings of Table 6, some of the contrasting behaviour of *Reputation Collectors* and Caretakers are as follows:

1. The average reputation of all the questioners targeted by the *Reputation Collectors* is 58.48, which is way below the average of all users of programmer's dataset (i.e., 183). The average reputation of all the questioners targeted by the *Caretakers* is 221.98.
2. The average reputation gained by the *Reputation Collectors* per accepted answer is 226, while the average reputation gained per accepted answer by *Caretakers* is 240.
3. Another interesting result is that the average reputation of *Reputation Collectors* is 494, while the average reputation of *Caretakers* is 379.89.

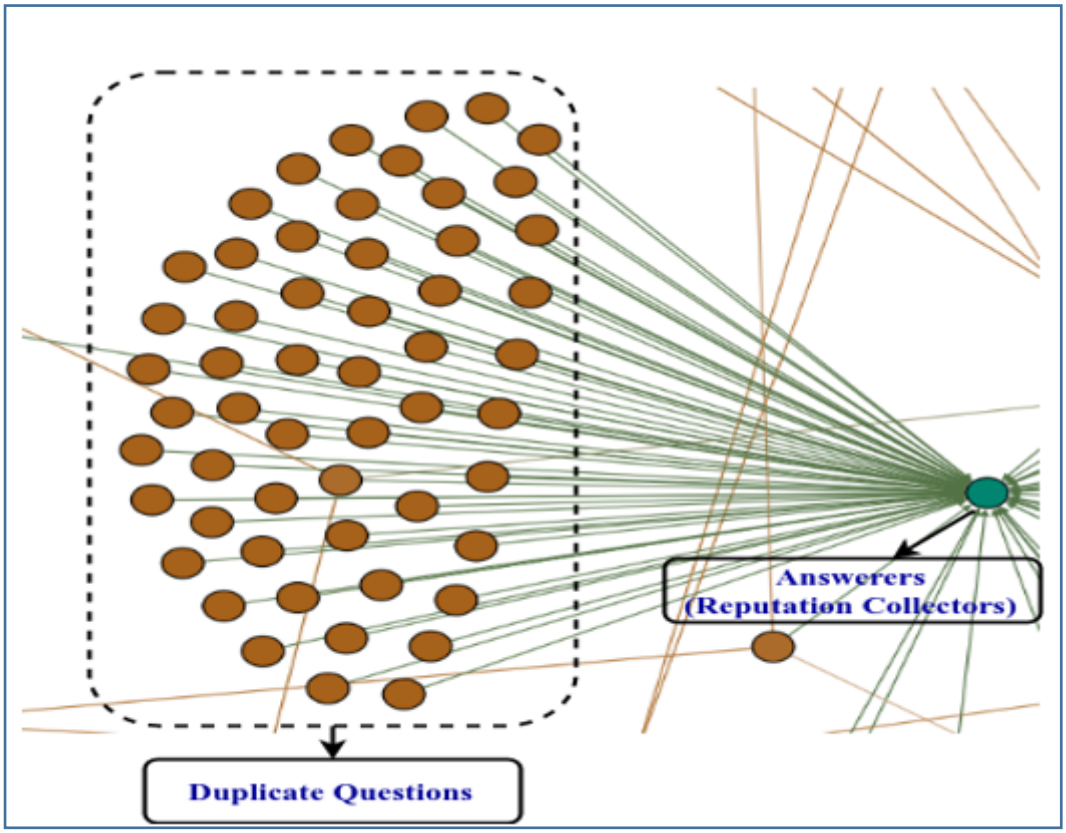

Figure 6: A *Reputation Collector* targeting multiple duplicate answers

The above result confirmed that on *'Programmers'* topic, few users called *Reputation Collectors* are posting a large number of low-quality answers. We further checked the existence of *Reputation Collectors* over the different topics of Stack Exchange. To do this, we select the different topics from the stack exchange, which belong from Science, Electronics, Gaming and others, and found that not only the programmers, but also on an average 1.08% of *Reputation Collectors* are present almost every topic of stack exchange, which posted 18.08% of low-quality answers on an average. The detailed results of the selected topics and the nature of the *Reputation Collectors* are presented in Table 7.

Table 7: Analysis of *Reputation Collectors* across different topics of Stack Exchange

Topic	Total Number of Users	Number of Reputation Collectors	Percentage of low-quality answers given by Reputation Collectors	Average Reputation of Reputation Collectors
Physics	76,666	920 (1.2% of all users)	26.00%	453
Ubuntu	432,187	6,915 (1.6% of all users)	18.48%	571
Mathematics	319,833	5757 (1.8% of all users)	16.24%	703
Code Review	111,765	174 (0.15% of all users)	12.33%	289
Apple	153,488	1980 (1.29% of all users)	15.08%	410
DBA	95,937	1228 (1.28% of all users)	26.00%	313
Programmer	176,587	161(0.091% of all users)	14.08%	494
Gaming	98,941	841 (0.85% of all users)	14.00%	456
Electronics	93,114	1136 (1.22% of all users	27.00%	437

5. Discussion

Our major findings of this research are that on average 18.88% of low-quality answers are posted by a handful 1.05% users who are termed as *Reputation Collectors* over the different topics of the Stack Exchange as shown in Table 7. Through the detailed analysis on the nine different topics, we identified 1.05% *Reputation collectors* are posting 18.88% of low-quality content on the site. The reputation of these users varies from 149 to 3,603 and on average it was 474. This indicates that some of these *Reputation Collectors* have got all the privileges of the site and they can moderate or delete any question and answers. This can be very dangerous for the forum as some of these users have gained all of the controls of the website. Also, these privileged *Reputation Collectors* continue to post low-quality answers, thus encouraging low-quality and duplicate questions. Such privileged users are also equipped to sabotage the content of the site. They cannot only delete good quality posts, but also avoid deleting low-quality posts. The cumulative effect of highly privileged, but ill-intentioned users can be devastating to the site. Recently, researchers have started finding ways to automatically detect low-quality contents and content abusers (Cheng, Danescu-Niculescu-Mizil, and Leskovec, 2015; Kayes, Kourtellis, Quercia, Iamnitchi, and Bonchi, 2015) to help moderators to detect these posts or ban these users. Srba and Bielikova (2016a) did mention the presence of *Reputation Collectors* but they did not mention anything about how to find such users. The current research is extending the analysis of Srba and Bielikova (2016a) by identifying the real *Reputation Collectors.* The current finding is in line with the research of (Huna, Srba, and Bielikova, 2016) where they have stated that reputation usually does not reflect the real value of users' contributions and some users purposefully abuse reputation system. However, our findings indicate that even with the current systems the *Reputation Collectors* and *Caretakers* are easily distinguished. This finding can be supplemented with the research of Ahasanuzzaman et al. (2016) and Zhang et al. (2015) with their duplicate finding systems. Their duplicate question finding system can suggest some duplicate questions, which can even be beneficial for the site moderators to identify that these are duplicate questions and being answered by *Reputation Collectors*.

5.1 Theoretical Contributions

The major theoretical contribution of the current research is the development of algorithms for filtering low- and high-quality contents based on the votes received by those contents. The current

research identifies *Caretakers* and *Reputation Collectors.* The algorithm is developed with the aim to identify a small set of users who are creating and encouraging low-quality content in the investigated forum. The algorithm achieves this objective as it is able to extract just 1.05% of users among the various topics of Stack Exchange (see Table 7) who are identified as troublemakers. The algorithm also supports the findings of Srba and Bielikova (2016a) that even the number of expert users (*Caretakers*) is very less. Our results confirm that it is really very small as they form only 0.07% of total users on the investigated 'Programmer' topic. Other major contributions of the current research is the visualization of the problem of duplicate questions and their answers. To the best of our knowledge, this is the first time that the duplicate questioners and their answerers are represented graphically through Duplicity Graph. The Duplicity Graph helps to visualise the behaviour of the *Reputation Collectors*. This algorithm can be periodically run on the administrative portal of the site to find the suspected users.

5.2 Implications for practice

The findings of this research indicate that only a very few (1.05%) users are doing a major damage on the Stack Exchange site. The motivation of these users is also analysed and we found that they are only running after the collection of reputation points. One of the major practical implications of this research could be a reformulation of reputation system on Stack Exchange. The present finding supports the suggestions of Huna et al. (2016) where they said that one could adjust the reputation system to depict the contribution of the users more accurately, and thus encouraging the users to post good quality content. Since the duplicate question is always discouraged by the Stack Exchange community members, users reputation points, which have been gained by giving answers to duplicate questions, might be revoked. Taking back the reputation points will make the active users first find similar questions and make a link to them instead of posting answers to gain points. The current research categorises the contents into low-quality and high-quality. It also identifies the users posting low-quality and high-quality contents. Another practical implication of this research could be to ask the *Caretakers* to educate and motivate the low-quality content providers to improve their posts. The users consistently posting low-quality contents can also be penalised by taking away some of their reputation points if their low-quality content crosses a threshold. One of our Duplicity graphs (see Figure 5) shows that a particular duplicate question is being answered by so many users. This question may be referred to some *Caretakers* to take a close look into it. The *Caretakers* are not very active, but they answer tricky questions, which are going to be useful for the community for a long time. Our analysis reveals that *Caretakers* fetch on average 240 reputation points from every answer they post. They gain their maximum points through user votes as answer acceptance is a one-time activity and fetches only 15 points. To further motivate these types of answers, these users may be awarded some bonus points if the total votes on an answer go beyond a certain limit. Another interesting way to ensure that good quality questions attract more users is by creating a dynamic reputation system, where the reputation points gained due to an answer, depends on the quality of the question that has been answered as an extension to the work of (Huna et al., 2016). The proposed algorithms are implemented through Python programs and verified on Stack Exchange dataset. Another biggest advantage of the algorithm proposed in this article is that it can be incorporated into the site without making any changes to the core working of the site, or making sweeping changes in existing policies.

6. Conclusion

A group of users on Stack Exchange are posting a lot of low-quality questions and answers and voting up the below average answers. They are so to get more number of reputation point and higher privilege of the site. These users are a threat to the CQA site as they are posting too much

of low-quality content, which will make these sites unuseful and unreliable. In this article, we identified these *Reputation Collectors* by analyzing their posting behaviour. We also extracted the duplicate questions and answers posted against them. We found that a major fraction of the answers to those duplicate questions was given by these so-called *Reputation Collectors.* The low and high-quality contents are separated by their obtained votes and duration of stay in the site. We found from the analysis of out of 23,791 number of low-quality answers, only a fraction of users posting 3,529 number of answers. Similarly, the number of caretakers are also very less compared to the other users, they are only 0.07%, and answering 5,913 of good quality answers out of the 142,191 answers.

6.1 **Limitations and Future Works**

The present study concentrated in the modus operandi of *Reputation Collectors* and *Caretakers* only. This idea can be utilised for characterizing other users also. The algorithms can be extended to other similar sites such as Yahoo! Answers, Quora etc. Since the reputation system is slightly different and hence presented algorithms can be slightly modified to accommodate the new reputation system and verified on these similar sites. The work proposed in this paper uses answer acceptance data to find the initially suspected users. We could have got more insight into the user behaviour if the user voting data is made available on these sites. The low-quality content identification can be further enhanced by taking the textual content, user profile, comments to the answers etc. The duplicity graph can be mined using graph mining algorithm (Tang and Liu, 2010) to yield some more insights such as group activity, average network connections of users asking a question and answering them etc. The duplicity graph can be constructed for different types of questions (Dang, Kelly, and Lin, 2007; Harper, Raban, Rafaeli, and Konstan, 2008; Nam, Ackerman, and Adamic, 2009; Westbrook, 2015) to find out the relationship between duplicate questions and their answerers. It would also be worthwhile to conduct primary research by utilising established theories and models such as Theory of Reasoned Action (e.g. Alryalat et al., 2015; Fishbein and Ajzen, 1975), Technology Acceptance Model (e.g. Alryalat et al., 2016; Davis, 1989), Theory of Planned Behaviour (e.g. Ajzen, 1991; Rana et al., 2016), and Unified Theory of Acceptance and Use of Technology (Dwivedi et al. 2017a, 2017b; Rana et al. 2017; 2016; Venkatesh et al., 2012; 2003) that focuses on understanding user behaviour

References

Ajzen, I. (1991). The theory of planned behavior. Organizational Behavior and Human Decision Processes, 50(2), pp. 179-211.

Ahasanuzzaman, M., Asaduzzaman, M., Roy, C. K., & Schneider, K. A. (2016). Mining Duplicate Questions of Stack Overflow. 13th Working Conference on Mining Software Repositories, 402-412.

Aladwani, A. M. (2017). Compatible quality of social media content: Conceptualization, measurement, and affordances. International Journal of Information Management, 37(6), 576-582.

AlAlwan A, Rana NP & Dwivedi YK, Algharabat R. (2017). Social Media in Marketing: A Review and Analysis of the Existing Literature. Telematics and Informatics, 34(7), 1177-1190.

Alryalat, M.A.A., Rana, N.P., Sarma, H.K.D., and Alzubi, Z.A. (2016). An Empirical Study of Facebook Adoption among Young Adults in a North-Eastern State of India: Validation of Extended Technology Acceptance Model (TAM). I3E2016, Swansea University, UK.

Asaduzzaman, M., Mashiyat, A. S., Roy, C. K., & Schneider, K. A. (2013). Answering questions about unanswered questions of stack overflow. 10th IEEE Working Conference in Mining Software Repositories, 97-100.

Aswani, R., Kar, A.K., Ilavarasan, P.V., & Dwivedi, Y.K. (2018). Search Engine Marketing is not all gold: Insights from Twitter and SEOClerks. *International Journal of Information Management,* 38(1), 107-116.
Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: An open source software for exploring and manipulating networks. ICWSM, 8, 361-362.
Cheng, J., Danescu-Niculescu-Mizil, C., & Leskovec, J. (2015). Antisocial Behavior in Online Discussion Communities. ICWSM, 61-70.
Chua, A. Y., & Banerjee, S. (2015). Answers or no answers: Studying question answerability in Stack Overflow. Journal of Information Science, 41(5), 720-731.
Dang, H. T., Kelly, D., & Lin, J. J. (2007). Overview of the TREC 2007 Question Answering Track. In the Proceedings of Fifteenth Text REtrieval Conference, Washington DC, 7, 63-80.
Davis, F.D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, 13(3), 319-339.
Davis, J. M., & Agrawal, D. (2018). Understanding the role of interpersonal identification in online review evaluation: An information processing perspective. *International Journal of Information Management*, 38(1), 140-149.
Dwivedi, Y. K., Kelly, G., Janssen, M., Rana, N. P., Slade, E. L., & Clement, M. (2018). Social media: The good, the bad, and the ugly (editorial). *Information Systems Frontiers*, 1-5. DoI: https://doi.org/10.1007/s10796-018-9848-5.
Dwivedi, Y.K., Rana, N.P., Jeyaraj, A., Clement, M. & Williams, M.D. (2017a). Re-examining the Unified Theory of Acceptance and Use of Technology (UTAUT): Towards a Revised Theoretical Model. *Information Systems Frontiers*. Available at https://link.springer.com/article/10.1007/s10796-017-9774-y.
Dwivedi, Y.K., Rana, N.P., Janssen, M., Lal, B., Williams, M.D. & Clement, R.M. (2017b). An Empirical Validation of a Unified Model of Electronic Government Adoption (UMEGA). *Government Information Quarterly,* 34(2), 211-230.
Dwivedi, Y.K., Mäntymäki, M., Ravishankar, M.N., Janssen, M., Clement, M., Slade, E.L., Rana, N.P., Al-Sharhan, S. & Simintiras, A.C. (Eds.) (2016). *Social Media: The Good, the Bad, and the Ugly*: 15th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2016, Swansea, UK, September 13–15, 2016, Proceedings (Vol. 9844). Springer.
Dwivedi, Y.K., Kapoor, K.K. & Chen, H. (2015). Social Media Marketing and Advertising. *The Marketing Review,* 15(3), 289-309.
Fishbein, M., and Ajzen, I. (1975). Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research. Reading, MA: Addison-Wesley.
Fox, J., & Moreland, J. J. (2015). The dark side of social networking sites: An exploration of the relational and psychological stressors associated with Facebook use and affordances. Computers in Human Behavior, 45, 168-176. DOI: 10.1016/j.chb.2014.11.083.
Garcia, D., & Sikström, S. (2014). The dark side of Facebook: Semantic representations of status updates predict the Dark Triad of personality. Personality and Individual Differences, 67, 92-96.
Harper, F. M., Raban, D., Rafaeli, S., & Konstan, J. A. (2008). Predictors of answer quality in online Q&A sites. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 865-874.
Hashim, K. F., & Tan, F. B. (2015). The mediating role of trust and commitment on members' continuous knowledge sharing intention: A commitment-trust theory perspective. International Journal of Information Management, 35(2), 145-151.
Huna, A., Srba, I., & Bielikova, M. (2016). Exploiting content quality and question difficulty in CQA reputation systems. In International Conference and School on Network Science, 68-81.
Ismagilova, E., Dwivedi, Y.K., Slade, E.L. & Williams, M.D. (2017). Electronic Word of Mouth (eWOM) in the Marketing Context: A State of the Art Analysis and Future Directions. Springer International Publishing.

Jin, X. L., Zhou, Z., Lee, M. K., & Cheung, C. M. (2013). Why users keep answering questions in online question answering communities: A theoretical and empirical investigation. International Journal of Information Management, 33(1), 93-104.
Kamboj, S., Sarmah, B., Gupta, S. & Dwivedi, Y.K. (2018). Examining branding co-creation in brand communities on social media: Applying paradigm of Stimulus-Organism-Response. *International Journal of Information Management,* 39 (April), 169–185.
Kaplan, A.M., & Haenlein, M. (2010). Users of the world, unite! The challenges and opportunities of Social Media. Business Horizons, 53(1), 59-68.
Kapoor, K.K., Tamilmani, K., Rana, N.P., Patil, P., Dwivedi, Y.K. and Nerur, S. (2017). Advances in Social Media Research: Past, Present and Future. *Information Systems Frontiers*. DOI: https://doi.org/10.1007/s10796-017-9810-y.
Kapoor, K.K. & Dwivedi, Y.K. (2015). Metamorphosis of Indian electoral campaigns: Modi's social media experiment. *International Journal of Indian Culture & Business Management*, 11(4), 496–516.
Kayes, I., Kourtellis, N., Quercia, D., Iamnitchi, A., & Bonchi, F. (2015). The social world of content abusers in community question answering. In Proceedings of the 24th International Conference on World Wide Web, 570-580.
Krasnova, H., Widjaja, T., Buxmann, P., Wenninger, H., & Benbasat, I. (2015). Research note—why following friends can hurt you: an exploratory investigation of the effects of envy on social networking sites among college-age users. Information Systems Research, 26(3), 585-605.
Lee, C. T., Rodrigues, E. M., Kazai, G., Milic-Frayling, N., & Ignjatovic, A. (2009). Model for voter scoring and best answer selection in community Q&A services. In Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. International Joint Conferences on IEEE/WIC/ACM, 1, 116-123.
Li, L., He, D., Jeng, W., Goodwin, S., & Zhang, C. (2015). Answer quality characteristics and prediction on an academic Q&A Site: A case study on ResearchGate. In Proceedings of the 24th International Conference on World Wide Web, 1453-1458.
Liu, M., Liu, Y., & Yang, Q. (2010). Predicting best answerers for new questions in community question answering. International Conference on Web-Age Information Management, 127-138. Springer, Berlin, Heidelberg.
Liu, Y., Liu, Y., Zhou, K., Zhang, M., & Ma, S. (2017). Detecting collusive spamming activities in community question answering. In Proceedings of the 26th International Conference on World Wide Web (pp. 1073-1082). International World Wide Web Conferences Steering Committee.
Lu, H.P., & Hsiao, K.L. (2007). Understanding intention to continuously share information on weblogs. Internet Research, 17(4), 345-361.
Luo, N., Zhang, M., Hu, M., & Wang, Y. (2016). How community interactions contribute to harmonious community relationships and customers' identification in online brand community. International Journal of Information Management, 36(5), 673-685.
Nam, K.K., Ackerman, M.S., & Adamic, L.A. (2009). Questions in, knowledge in?: A study of naver's question answering community. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems, 779-788.
Plume, C.J., Dwivedi, Y.K. & Slade, E.L. (2016). *Social Media in the Marketing Context: A State of the Art Analysis and Future Directions*. 1st Edition, Chandos Publishing Ltd, Oxford, UK.
Ponzanelli, L., Mocci, A., Bacchelli, A., Lanza, M., & Fullerton, D. (2014). Improving low quality stack overflow post detection. IEEE International Conference on Software Maintenance and Evolution, 541-544.
Rana, N.P., Dwivedi, Y.K., Williams, M.D. & Weerakkody, V. (2016). Adoption of online public grievance redressal system in India: Toward developing a unified view. *Computers in Human Behavior*, 59, 265-282.
Rana, N.P., Dwivedi, Y.K., Lal, B., Williams, M.D. & Clement, M. (2017). Citizens' adoption of

an electronic government system: towards a unified view. *Information Systems Frontiers*, 19(3), 549-568.
Rana, N.P., Lal, B., and Slade, E. (2016). Adoption of Two Indian E-Government Systems: Validation of Extended Theory of Planned Behavior (TPB). Americas Conference on Information Systems, San Diego, USA, 2016.
Rathore A.K., Ilavarasan P.V. & Dwivedi, Y.K. (2016). Social Media Content and Product Co-creation: An Emerging Paradigm. *Journal of Enterprise Information Management,* 29(1), 7-18.
Roy, P. K., Ahmad, Z., Singh, J. P., Alryalat, M. A. A., Rana, N. P., & Dwivedi, Y. K. (2018). Finding and Ranking High-Quality Answers in Community Question Answering Sites. Global Journal of Flexible Systems Management, 19(1), 53-68.
Saha, R. K., Saha, A. K., & Perry, D. E. (2013). Toward understanding the causes of unanswered questions in software information sites: A case study of stack overflow. Ninth Joint Meeting on Foundations of Software Engineering, 663-666.
Serna, E., Bachiller, O., & Serna, A. (2017). Knowledge meaning and management in requirements engineering. International Journal of Information Management, 37(3), 155-161.
Shareef, M.A, Mukerji, B., Alryalat, M.A.A., Wright, A. and Dwivedi, Y.K. (2018). Advertisements on Facebook: Identifying the persuasive elements in the development of positive attitudes in consumers. Forthcoming in *Journal of Retailing and Consumer Services*.
Shareef, M.A., Mukerji, B., Dwivedi, Y.K., Rana, N.P., and Islam, R. (2017). Social Media Marketing: Comparative Effect of Advertisement Sources. *Journal of Retailing and Consumer Services*. Available at http://www.sciencedirect.com/science/article/pii/S096969891730591X.
Slag, R., de Waard, M., & Bacchelli, A. (2015). One-day flies on stack overflow-why the vast majority of stackoverflow users only posts once. In Mining Software Repositories (MSR), 12th Working Conference on IEEE/ACM, 458-461.
Srba, I., & Bielikova, M. (2016a). Why is stack overflow failing? Preserving sustainability in community question answering. IEEE Software, 33(4), 80-89.
Srba, I., & Bielikova, M. (2016b). A comprehensive survey and classification of approaches for community question answering. ACM Transactions on the Web (TWEB), 10(3), 18:1-18:63.
Stieglitz, S., Mirbabaie, M., Ross, B., & Neuberger, C. (2018). Social media analytics–Challenges in topic discovery, data collection, and data preparation. International Journal of Information Management, 39, 156-168.
Stack Overflow (2016). What does "closed" mean? Retrieved from http://stackoverflow.com /help/closed-questions, Accessed on 3rd December 2017.
Stack Exchange (2016). The Stack Exchange dataset. Retrieved from https://archive.org/details/ stackexchange, Accessed on 3rd December 2017.
Tamilmani, K., Rana, N.P., Alryalat, M., Alkuwaiter, W., and Dwivedi, Y.K. (2018). Social Media Research in the Context of Emerging Markets: An Analysis of Literature Published in Senior Scholars' Basket of IS Journals. *Journal of Advances in Management Research*, DOI: 10.1108/JAMR-05-2017-0061.
Tang, L., & Liu, H. (2010). Graph mining applications to social network analysis. In Managing and Mining Graph Data, Springer US, 487-513.
Treude, C., Barzilay, O., & Storey, M. A. (2011). How do programmers ask and answer questions on the web?: Nier track. 33rd International Conference on Software Engineering, 804-807.
Venkatesh, V., Morris, M.G., Davis, G.B., and Davis, F.D. (2003). User acceptance of information technology: Toward a unified view. MIS Quarterly, 27(3), 425-478.
Venkatesh, V., Thong, J. Y., & Xu, X. (2012). Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology. MIS Quarterly, 157-178.
Westbrook, L. (2015). Intimate partner violence online: Expectations and agency in question and answer websites. Journal of the Association for Information Science and Technology, 66(3), 599-

615.
Xie, Z., Nie, Y., Jin, S., Li, S., & Li, A. (2015). Answer Quality Assessment in CQA Based on Similar Support Sets. In Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, 309-325.
Yao, Y., Tong, H., Xie, T., Akoglu, L., Xu, F., & Lu, J. (2015). Detecting high-quality posts in community question answering sites. Information Sciences, 302, 70-82.
Zhang, Y., Lo, D., Xia, X., & Sun, J. L. (2015). Multi-factor duplicate question detection in stack overflow. Journal of Computer Science and Technology, 30(5), 981-997.