TrustQ: A Category Reputation Based Question&Answer System

Yuhua Lin and Haiying Shen

Department of Electrical and Computer Engineering
Clemson University, Clemson, South Carolina 29631, USA

ABSTRACT

Question and Answering (Q&A) systems aggregate the collected intelligence of all users to provide satisfying answers for questions. A well-developed Q&A system should provide high question response rate, low response delay and good answer quality. Previous works use reputation systems to achieve the goals. However, these reputation systems evaluate a user with an overall rating for all questions the user has answered regardless of the question categories, thus the reputation score cannot accurately reflect the user's ability to answer a question in a specific category. In this paper, we propose TrustQ, a category reputation based Q&A System. TrustQ evaluates users' willingness and capability to answer questions in different categories. Considering a user has different willingness to answer questions from different users, TrustQ lets each node evaluate the reputation of other nodes answering its own questions. User *a* calculates user *b*'s final reputation by considering both user *a*'s direct rating and the indirect ratings on user *b* from other nodes. The reputation values facilitate forwarding a question to potential answerers, which improves the question response rate, response delay and answer quality. Our trace-driven simulation on PeerSim demonstrates the effectiveness of TrustQ in providing good user experience in terms of response rate and latency, and the answer quality.

Keywords: Question & Answer systems; Reputation systems

1. INTRODUCTION

Recent years have witnessed rapid prevalence of online Question and Answering (Q&A) systems such as *Yahoo! Answers*, *Naver KiN*, *Microsoft Live Q&A*, *Google Answer*, and *Mahalo*. Q&A systems have significantly changed the way we seek information. Compared with traditional web search engines, Q&A systems tend to provide answers to a broader range of questions.[1] For example, users (user and node are interchangeable terms in this paper) may ask for restaurant suggestions, or advice on his career development from users with related knowledge. Response rate, response delay and answer quality are three important issues affecting the performance of a Q&A system.

It is a common case that users will not receive answers for their questions, or suffer from long delay before they receive answers. This is normally due to the lack of incentives in answering questions. Hsieh *et al.*[2] studied *Microsoft Live Q&A* and reported that the average time of receiving an answer is about 3 hours, and 20% of the questions never receive an answer. Users hope to receive satisfying answers to their questions. However, it is difficult to match a question to a user who has the expertise to answer it. Reputation system is a common tool to facilitate the recommendation of reliable potential answerers.[3–7] In this system, user ratings are collected based on the quality of answers they provide, and a computation engine is used to compute each user's reputations. The questions are then forwarded to users under the guidance of the reputation scores. The rationale of these reputation systems is that a user's reputation score is an indicator of the quality of answers the user can provide. However, there are a variety of question categories and different knowledge fields in modern Q&A systems. A user excels in one question category may not be familiar with another category. Existing reputation systems evaluate a user with an overall rating for all questions the user has answered regardless of the question categories,

Further author information: (Send correspondence to Haiying Shen)
Yuhua Lin: E-mail: yuhual@clemson.edu, Telephone: +1 864 650 3021
Haiying Shen: E-mail: shenh@clemson.edu, Telephone: +1 864 656 5931

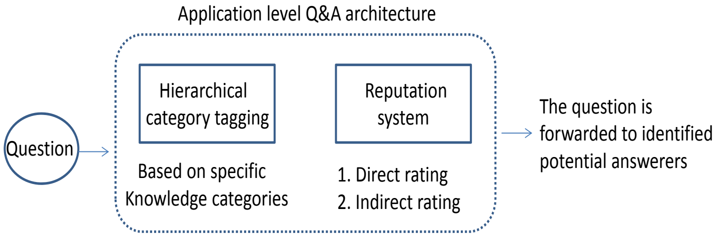

Application level Q&A architecture
Question
Hierarchical category tagging
Reputation system
Based on specific Knowledge categories
1. Direct rating
2. Indirect rating
The question is forwarded to identified potential answerers

Figure 1: An overview of the TrustQ Q&A system.

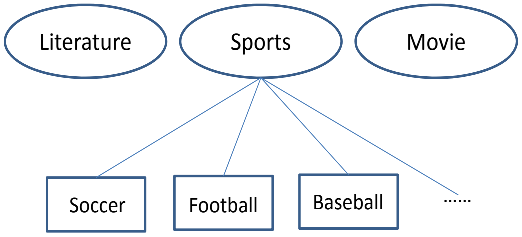


Figure 2: Hierarchical category classification.

thus the reputation score cannot accurately reflect the user's ability to answer a question in a specific category. Therefore, forwarding a question to users with high reputation scores regardless of the question's category is not effective in finding a suitable answerer. In this paper, we propose TrustQ, a category reputation based Q&A System. TrustQ evaluates the reputations of users' willingness and capability to answer questions in different categories. Considering a user has different willingness to answer questions from different users, TrustQ lets each node evaluate the reputation of other nodes answering its own questions. User a calculates user b's final reputation by considering both user a's direct rating and the indirect ratings on user b from other nodes. The reputation values facilitate forwarding a question to potential answerers, which improves the question response rate, response delay and answer quality.

The remainder of this paper is arranged as follows. Section 2 presents an overview of the related work. Section 3 presents a detailed description of our design. Section 4 presents the experimental results. Section 5 concludes this paper with remarks on our future work.

2. RELATED WORK

Recent years have witnessed rapid prevalence of online Q&A systems[8–11] in our daily lives. Facebook launched a Q&A application in July, 2010, which facilitates users to post and answer questions on the online social network. Early works in Q&A system research community focus on analyzing some of the large-scale Q&A sites, including *Yahoo! Answers* and *Naver KiN*.[12,13]

Besides the studies on the basic characteristics of Q&A systems, researchers also attempted to improve the question answering rate and answer quality. Some works apply reputation systems to locate credible answerers.[3–7] Guo *et al.*[3] proposed to improve the recommendation of answer providers by discovering latent topics of question. They applied the Latent Dirichlet Allocation (LDA) model to discover latent topics of questions and answers, and latent interests of users, then recommended question answerers for new questions based on similarity between question topics and answerer interests. Tausczik *et al.*[4] proposed a reputation model that measures users' offline and online reputations in an online mathematics community. By considering the number of publications, earned points, authoritativeness and social connectedness, this model is effective to identify high quality author submissions. Bian *et al.*[5] developed a semi-supervised coupled mutual reinforcement framework for Q&A system reputation model. This framework requires relatively small amount of labeled data to initialize the training process. Chen *et al.*[6] incorporated social network element when calculating a user's rating in a Q&A systems since a user's rating is influenced by other users' interactions with this user. Hong *et al.*[7] studied two popular ranking schemes: HITS and PageRank, Based on these themes, they built two user reputation models in question answering systems. They showed that the reputation models are effective in improving answering quality. These systems calculate a general reputation score for every user as an indicator of whether the user is reliable in providing high-quality answers. TrustQ is distinguished from these works in a way that it provides reputation scores for each user in different question categories, which more accurately reflects a user's reliability in answering questions. Thus, TrustQ improves the question response rate, response delay and answer quality.

3. SYSTEM DESIGN

3.1 An Overview of TrustQ

An overview of TrustQ is shown in Figure 1. It has two main parts: hierarchical category tagging and reputation system. As current Q&A systems, TrustQ has categories such as *sports*, *literature* and *movie*. Hierarchical category means that each category is further classified to different sub-categories, then each sub-category is further classified to different group, and so on. In this paper, we use two level classification as an example. We define *theme* as a further classification of a category; for example, the *sports* category has themes such as *soccer*, *football* and *basketball*.

In TrustQ, when a user launches a question, (s)he determines the question's main category and detailed theme in the hierarchical category tagging stage. This question is then labeled with tags describing its category and theme. Previous study[14] shows that less knowledgable users always choose more knowledgable users as their contacts. Also, when a user trusts another user, it becomes a fan of that user. Therefore, a user's contacts are the best candidates of potential answerers. Thus, TrustQ first identifies the contacts of the asker who have registered with interests in these category and theme. Then, TrustQ selects the users that have the highest reputation values in these category and theme as the potential answerers. These reputation values reflect the user's ability and willingness to answer questions in each category and theme. The reputation system updates users' reputation ratings periodically.

3.2 Question Category Determination

A Q&A system may receive a tremendous amount of various questions every day. The selected potential answerers for a question must be knowledgable in the category and theme of the question. Therefore, it is crucial to determine the category and theme of each question and user by the knowledge area, which helps find potential answerers that are interested in a given question. To this end, TrustQ uses a hierarchical category tagging method. It has two levels of tags for questions: category and theme, as shown in Figure 2. Categories such as *sports*, *literature* and *movie* are information domains that a question may belong to. When a user registers for the TrustQ Q&A system, (s)he needs to specify the categories and themes that (s)he is interested in. When a user asks a question, (s)he need to go to the corresponding category and theme to ask the question. The TrustQ Q&A system can retrieve the category and theme of a new question, which helps identify the potential answerers for the question. We use q_k to denote a question; c_u be a category; t_{uv} be a theme belonging to category c_u. Then, a question belonging to category c_u is represented by $q \in c_u$, and a question belonging to theme t_{uv} is represented by $q \in t_{uv}$.

Multiple category classification levels provide a fine-grained classification, which enables more accurate matching between potential answerers and questions. For example, for a question about soccer, the identified potentials answerers with interest *soccer* may be able to provide more accurate answers than those with interest *sports*. The two-level hierarchical category can be easily extended to more levels.

3.3 Category based Reputation Management

Reputation systems in the Q&A systems help users judge other users' trustworthiness or expertise, which can be used to identify potential answerers for a question. TrustQ considers a number of factors in reputation calculation.

(1) A user's received reputation rating on a question reflects the user's trustworthiness on answering the questions in this question's category and theme. Therefore, a user's reputation should be evaluated based on different categories and themes.

(2) A user's reputation also reflects the user's willingness to answer questions. Previous study[14] shows that a user in the contact lists of high-reputed nodes should be trustable. Therefore, in TrustQ, potential answerers are selected from the asker's contacts and the asker (i.e., fan) rates its contacts. Thus, node a considers its reputation evaluations on node b (i.e., direction reputation) and the reputation evaluation from b's fans on b (i.e., indirection reputation) to calculate user b's reputation.

(3) The reputation of a user should be updated periodically, and the reputation value in an older time period should have a lower weight in the final reputation value calculation.

3.3.1 Direct Reputation

User a evaluates user b's direct trust based on the responses that user a has received from user b periodically. Higher satisfactions on the responses in terms of the willingness, quickness and quality of the responses lead to higher trust values. As in current Q&A systems, an asker always rates its received answers in a range (e.g., [0,1]). We use r_{ab}^k to denote user a's rating on user b's answer for user a's question q_k. r_{ab}^k is in a range of [0,1]. $r_{ab}^k = 0$ if user b has not provided an answer during a reputation evaluation period. In the end of each reputation evaluation period, user a classifies its questions based on categories and themes. For each category c_i, user a calculates the average reputation values of the questions in category c_i (i.e., $q \in c_i$). In each category c_i, user a calculates the average reputation values of the questions in each theme t_{ij} (i.e., $q \in t_{ij}$). Finally, user a generates two vectors for user b to represent the reputations for different categories and for different themes: $R_{ab}^c = (r_{ab1}^c, r_{ab2}^c, ...r_{abn}^c)$ and $R_{ab}^t = (r_{ab1}^t, r_{ab2}^t, ...r_{abm}^t)$. Every element in R_{ab}^c and R_{ab}^t is calculated by evaluating the answers belonging to a specific category or theme. In the following, for simplicity, we use r_{ab} to denote the average of user a's ratings on user b's answers on a category or a theme during a reputation evaluation period.

3.3.2 Indirect Reputation

When user a evaluates user b's reputation, it considers the direct reputation evaluations of user b's fans on user b as *indirect reputations.* After user a' calculates user b''s category reputation vector $R_{a'b'}^c$, and theme reputation vector $R_{a'b'}^t$, it sends the two vectors to other fans of user b. In order to make the exchange of reputation values safe and accurate, TrustQ should withstand some common types of network attacks, such as Man-in-the-Middle attack and data modification. Various approaches such as PKI[15] have been well developed to prevent these attacks, so this issue is not the focus of our paper. After a user, say user a, receives the indirect reputations, it calculates the indirect reputations on its contacts. In the calculation, user a considers its different degrees of social closeness of the fan (i.e., rater) and its contact (i.e., ratee). S_{kb} denotes the social closeness of rater k towards ratee b, $0 < S_{kb} < 1$, a larger value of S_{kb} means a closer social relationship. Previous work[16] shows that socially closer nodes tend to give higher rating to each other, so we aim to reduce the impact of social relationship while evaluating a user's reputation. As a result, user a calculates the trusted indirect reputation of user b (denoted by r'_{ab}) by:

$$r'_{ab} = \frac{\sum_{k \in F_b} r_{kb} \times (1 - S_{kb}^2)}{\sum_k r_{kb}}, \tag{1}$$

where F_a denotes the set containing all user b's fans, S_{kb} is the social closeness between k and b. Finally, user a stores its indirect reputations on user b on different categories (denoted by r'^c_{ab}) and on different themes (denoted by r'^t_{ab}) in two vectors: $r'^c_{ab} = (r'^c_{ab1}, r'^c_{ab2}, ...r'^c_{abn})$ and $r'^t_{ab} = (r'^t_{ab1}, r'^t_{ab2}, ...r'^t_{abm})$.

3.3.3 Overall Reputation

Next, for each category or theme, user a calculates the overall reputation of user b by combing the direct reputation (r_{ab}) and the indirect reputation (r'_{ab}) using Equation (2).

$$R_{ab} = \alpha r_{ab} + (1 - \alpha) r'_{ab}, \tag{2}$$

where α and $1 - \alpha$ are the weights placed on each reputation.

We use T to denote a period of time for reputation evaluation. To consider a user's question answering behaviors in both the previous and current time periods in reputation evaluation, TrustQ applies an exponential decay factor, $\phi = e^{-\lambda T}$, on the reputation in the previous time period. The decay constant λ is set to 1 to make a moderate decrease as time elapses. That is, after user a calculates user b's reputation in the current time period (denoted by $R_{ab}^{current}$), it updates user b's reputation by:

$$R_{ab}^{new} = \phi R_{ab}^{old} + (1 - \phi) R_{ab}^{current}, \tag{3}$$

When selecting question answerers, asker a will check the reputation value (R_{ab}^{new}) of each of its contacts in the question's theme and choose the contacts with the highest reputation value. If none of its contacts have reputation values in the question's theme, asker a evaluates the overall reputations of its contacts indirectly from the reputation values of the questions' category and the category's other themes. Assume the category and the theme of the question are c_u and t_{uv}, respectively, and $\mathcal{T}_u$ includes all question themes under category c_u. Then, user a calculates the overall reputation of contact b based on Equation (4).

$$R_{ab} = \sum_{k\in(\mathcal{T}_u\setminus t_{uv})} \beta_k \times r^t_{abk} + \gamma \times r^c_{abu} \tag{4}$$

$\gamma \in (0, 1)$ is the weight of category reputation, $\beta_k \in (0, 1)$ is the weight of theme reputation for theme t_{uk}. After this reputation calculation, user a chooses the contacts with the high R_{ab} as potential question answerers.

4. PERFORMANCE EVALUATION

We conducted trace-driven experiments on PeerSim.[17] The data set we used is crawled from *Yahoo! Answers* from Aug. 17 to Oct. 19, 2011, which includes: 1) personal information of 119,175 users such as best answer rate (which is the percentage of a user's answers that are chosen by the askers as best answers), number of followers (i.e., fans) and contacts for each users, 2) general information of 119,174 questions such as the categories they belong to and the answers they draw. According to *Yahoo! Answers*, the questions are grouped by 26 categories, including "Travel", "Environment", and 148 themes including "Air Travel" and "Australia" under category "Travel". In the simulation, we deployed 10,000 nodes as users on Q&A system; the users are selected from the trace data who have more than 6 contacts. Follower and contact relationships are set based on user information from the trace data. Each user has 1 to 4 randomly selected question categories (interests), and has 1 to 5 themes under each question category. The expertise level of each user in a category or a theme is chosen from 1 to 10. The expertise level indicates a user's ability to answer questions, higher level in a specific question theme represents higher proficiency in answering questions belonging to the theme. In order to have more capable answerers in the system, in additional to v number actual answerers in the trace, we also randomly selected $10v$ users from the users who have interest in the question's theme as capable answerers. After receiving a question, if a user is a capable answerer of this question, (s)he will respond after a delay randomly chosen from [1,30] minutes. A user can answer up to 2 questions within every 30 minutes. An asker will rate each answer with scores based on the answerer's expertise level. If an answer is received from a user with level l expertise, then the asker will rate this answer with $l/10$ score. In order to generate answering activities and cumulate reputation scores for users, we executed a warm-up process by launching 20,000 questions. During the test, user reputation was updated every 30 minutes. The simulation contains a 12 hours process, within every 30 minutes, a number of randomly users post questions and these questions are forwarded to their contacts.

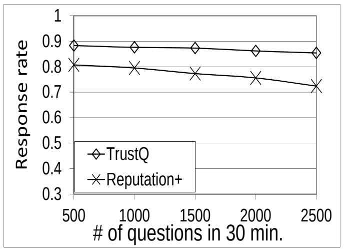


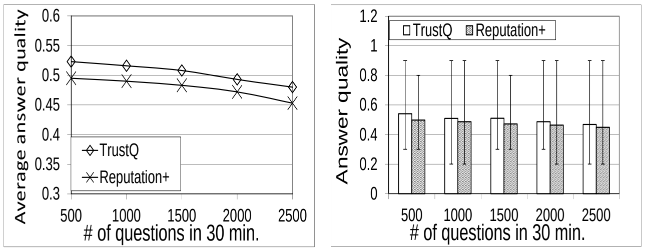
Figure 3: The question response rate.

In our proposed TrustQ Q&A system, when a user posts a question, the contacts of this user are sorted by their reputation scores on the question's theme and category. The question is forwarded to 3 contacts with the highest reputation scores. If no answer returns after 30 minutes, this question is forwarded to the next 3 contacts, and then the question forwarding operation is terminated. We compared our proposed TrustQ Q&A system with *Reputation+*. In *Reputation+*, each user's reputation is represented by averaging the reputation scores he/she earns in all question categories (we call it general reputation score). A user's question is forwarded to the users with 3 top reputation scores in its contacts. In both *Reputation+* and TrustQ, if no answer returns after 30 minutes, a question will be forwarded to the next 3 contacts with the highest reputation scores, and then the question forwarding operation is terminated.

We are interested in the following metrics:

- Response rate. The percentage of questions that can receive at least one answer.[18]
- Answer quality. The rating of answers given by the askers.
- Response latency. The time spans from a question is launched until it draws the first answer.

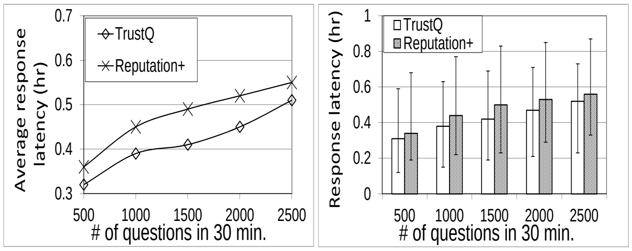
Figure 3 shows the question response rate when there are different number of new questions posted in the system within 30 minutes. We see that as the number of posted questions within 30 minutes increases, the



(a) The average quality.

(b) The median, 5th and 95th percentile quality.

Figure 4: The quality of answers.



(a) The average latency.

(b) The median, 5th and 95th percentile latency.

Figure 5: The latency of answer responses.

question response rate of both *Reputation+* and TrustQ exhibit slight decrease. As each user can answer a limited number of questions within a time period, when there are more questions, the probability of forwarding a question to a user that is unable to answer the question increases, thus generating even lower question response rate. We also see that *Reputation+* yields less response rate than TrustQ. In *Reputation+*, new questions are always forwarded to users with high general reputation scores regardless of the question category. Users with high general reputation scores may not have the expertise to answer a specific question. Thus, *Reputation+* leads to lower question response rate. Our proposed TrustQ system is effective in providing high question response rate under different question arrival rates due to the reason that the questions are forwarded to users with expertise in the question's specific category and theme. Therefore, the receivers are more likely to answer the questions they receive, leading to higher question response rate.

We calculated the average answer quality by averaging all answer scores received from askers. Figure 4(a) shows the average answer quality as a function of the number of questions posted per 30 minutes. Figure 4(b) shows the 5th percentile, median and 95th percentile of answer quality when new questions are posted in the system at different rates. If no answer is provided by an asker, the score for this answer quality is 0. Figure 4(a) shows that TrustQ produces about 0.04 increase in average answer quality than *Reputation+*. Figure 4(b) shows that TrustQ is advantageous in maintaining high answer quality by reaching 0.9 answer quality at the 95th percentile, while the 95th percentile answer quality sometimes drops to 0.8 in *Reputation+*. Also, the median answer quality of TrustQ is higher than that of *Reputation+*. These experimental results are caused by the reason that each question is forwarded to potential answerers with high reputation in the question's specific area in TrustQ, while a high general reputation score in *Reputation+* does not guarantee sufficient expertise in resolving questions in each specific question category. Figure 4(b) also shows that as the number of questions posted per 30 minutes increases, the average answer quality in both system decreases. With more questions, the number of question exceeding the receivers's response capacity also increases, thus leading to lower answer quality.

Figure 5(a) depicts the average latency of receiving answers as a function of the number of questions posted per 30 minutes. Figure 5(b) depicts the 5th percentile, median and 95th percentile latency of receiving answers. From both figures, we see that as questions are posted at a higher rate in the system, the latency of receiving answers increases in both methods. This is due to the reason that users can respond to a limited number of questions within every time period. Figure 5(a) shows that TrustQ yields response latency 0.05-0.1 hours shorter than that in *Reputation+*. Figure 5(b) shows that at both the 1th and 95th percentiles, *Reputation+* generally yields response latency 0.1-0.2 hours longer than that in TrustQ, which indicates the advantage of TrustQ in finding answerers that are willing to answer questions. Thus, TrustQ outperforms *Reputation+* in reducing answering latency due to the reason that it considers users' experience in answering questions in different categories when selecting potential answerers. Therefore, a questions is likely to be forwarded to suitable answerers in the first forwarding attempt, and there is no need to forward the question to the next group of contacts. Also, the questions receivers are likely to answer questions quickly. *Reputation+* has a relatively low answering rate when the question is forwarded to the first 3 contacts with high general reputation scores, and it needs extra latency to forward the question to the next 3 contacts. Also, since the question receivers may not answer the questions quickly since they may not have the expertise on the question's category. The above experimental results indicate the effectiveness of TrustQ in providing good user experience in terms of response rate and latency, and the answer quality compared to *Reputation+*.

5. CONCLUSIONS

The rapid growth of Q&A system makes it an important way of knowledge discovery. However, as a Q&A system generally serves a large amount of users and tens of thousands of new questions are posted in the system every day, forwarding questions to users who are willing and able to provide satisfying answers is crucial in maintaining the performance of Q&A systems. This paper proposes TrustQ, a category reputation based Q&A System. TrustQ evaluates users' reputation towards every knowledge category and theme, and forwards questions to a number of high reputable users in the question's knowledge category and theme. The advantage of TrustQ is verified by experiments on PeerSim. In our future work, we will study using effective incentives to further improve answer quality and response rate.

Acknowledgements

This research was supported in part by U.S. NSF grants IIS-1354123, CNS-1254006, CNS-1249603, CNS-1049947, CNS-0917056 and CNS-1025652, Microsoft Research Faculty Fellowship 8300751, and the United States Department of Defense 238866.

REFERENCES

[1] Lee, U., Kang, H., Yi, E., Yi, M., and Kantola, J., "Understanding Mobile Q&A Usage: An Exploratory Study," in [*Proc. of CHI*], (2012).

[2] Hsieh, G. and Counts, S., "mimir: a market-based real-time question and answer service," in [*Proc. of CHI*], (2009).

[3] Guo, J., Xu, S., Bao, S., and Yu, Y., "Tapping on the Potential of Q&A Community by Recommending Answer Providers," in [*Proc. of CIKM*], (2008).

[4] Tausczik, Y. R. and Pennebaker, J. W., "Predicting the Perceived Quality of Online Mathematics Contributions from Users' Reputations," in [*Proc. of SIGCHI*], (2011).

[5] Bian, J., Liu, Y., Zhou, D., Agichtein, E., and Zha, H., "Learning to Recognize Reliable Users and Content in Social Media with Coupled Mutual Reinforcement," in [*Proc. of WWW*], (2009).

[6] Chen, W., Zeng, Q., Liu, W., and Hao, T., "A User Reputation Model for a User-Interactive Question Answering System: Research Articles," *Concurrency and Computation: Practice and Experience* (2007).

[7] Hong, L., Yang, Z., and Davison, B. D., "Incorporating Participant Reputation in Community-Driven Question Answering Systems," in [*Proc. of CSE*], (2009).

[8] Yahoo!Answers, "http://answers.yahoo.com/, [Accessed in March 2014],"

[9] Answers, G., "http://answers.google.com/, [Accessed in March 2014],"

[10] Quora, "www.quora.com/, [Accessed in March 2014],"

[11] StackOverFlow, "www.StackOverFlow.com/, [Accessed in March 2014],"

[12] Zhang, L. A. J., Bakshy, E., and Ackerman, M., "Knowledge Sharing and Yahoo Answers: Everyone Knows Something," in [*Proc. of WWW*], (2008).

[13] Nam, K., Ackerman, M., and Adamic, L., "Questions in, knowledge in?: a study of naver's question answering community," in [*Proc. of CHI*], (2009).

[14] Li, Z., Shen, H., and Grant, J., "Collective intelligence in the online social network of yahoo!answers and its implications," in [*Proc. of CIKM*], (2012).

[15] Blaze, M., Feigenbaum, J., and Keromytis, A., "Keynote: Trust management for public-key infrastructures (position paper).," in [*Proc. of Security Protocols Workshop*], (1998).

[16] Li, Z., Shen, H., and Sapra, K., "Leveraging Social Networks to Combat Collusion in Reputation Systems for Peer-to-Peer Networks," in [*Proc. of IPDPS*], (2011).

[17] "Peersim: A peer-to-peer simulator." http://peersim.sourceforge.net/, [Accessed in March 2014].

[18] Janes, J., Hill, C., and Rolfe, A., "Ask-an-expert services analysis," *JASIST* **52**(13), 1106–1121 (2001).