**ForumHero: Filtering and Analysis of Community Question Answering (CQA)**

**Abstract**

Technology has made an impact in a way that no one has imagined. Community Question Answering (CQA) or Programming Forums have become invaluable resources to people for finding solutions. But the number of active users who actively participate in answering the questions in the community is around 10% of the overall users. This has taken a toll on the consumer experience in many programming forums. Predicting the users who might become inactive in the future would help the community retain the user base and lead to a active forum that can act as an invaluable resource to the internet community. This can be done by extracting various features that can be calculated using the first k posts activity of the user. Using the first k number of posts or by using the dynamics of the programming forums, such as the number of votes generated for the question and the answer posted in the community forum, we can calculate the long-term impact of the question and answer.

**Introduction**

Since its inception, the Internet has been widely used to seek information and to find answers for any question that a user might have. With the evolution of the internet, Question answering sites such as Stack Overflow, Yahoo Answers, Quora, Baidu Knows, Stack Exchange, etc. came into the picture and have become large repositories of valuable information. People pose various questions on these sites which are answered by a community of users. These sites were developed to help and provide users with useful answers to their questions. There are some focused knowledge communities within these sites, answering a wide range of questions ranging from subjective to technical domains, that makes these platforms go beyond search engines where one can search for a simple question and find related results within seconds. As users with deep expertise in their domains answer the questions, the content available within these sites has a long-lasting value. Moreover, Search engines can sometimes redirect the user queries to the information in these sites. So, looking at the big picture, It is important to have a high-quality question-answer pair which adds value to the knowledge base and can help users with similar questions in the future rather than providing an immediate answer to the question.

Most of these sites are crowd-sourced repositories, and to make sure these sites sustain long enough, it is essential to keep the user engaged for as much time as possible. As millions of users use these sites, achieving user engagement, in particular, becomes a challenging task. To overcome this, one needs to make sure that a user is effortlessly being routed to high-quality content available and that his/her questions are being answered by the experts in that domain ensuring a satisfactory answer. As these sites rarely provide monetary benefits, most of the sites try to entice its users by offering virtual rewards for their activities. Stack Overflow, for example, maintains reputation score for a user based on his/her activities which act as a motivation for the community user. Nevertheless, many users leave the forum and to improve the longevity of a site, the factors that have a significant effect on user churn are worth knowing. For example, receiving low-quality responses from the answerers can cause the user to leave the forum. Also, a decrease in the frequency of user posts can act as an indicator for user churn. These factors identify potential user churns and help the sites take the necessary measures to prevent churn. As we move forward, we try to identify and use such factors to predict user churn.

Several users in the community might be looking for information regarding the same/similar questions. In such a case, Identifying question pages that garner huge attention in the community becomes helpful. High activity around a question showcases the user interest towards a particular question and also benefits the answerers in terms of reputation and as multiple answers start pouring in and get endorsements, the question obtains a lasting value.

Several studies focused their work on identifying the factors affecting user churn in Q&A sites. Arguello et al. studied user behavior in Q&A communities and analyzed the factors that led to the success of the communities and more importantly, understand the strategy to prolong user engagement., Anderson et al. worked towards predicting the long lasting value of a question page in these sites and identifying if a question has been answered sufficiently. Oktay et al. studied the patterns of the answer arrivals in stack overflow. Their observation was that answers keep coming even after the question asker had accepted an answer which results in knowledge base beyond the requirements of the questioner. In his work, Tausczik et al. researched to predict the quality of an answer based on the reputation of the user.

In this project, there are three tasks that we aim to do as a part of analyzing the community question answering sites. We try to predict the significant factors for user churn and identify the users who are about to leave the community based on the the initial number of posts a user has made and then based on his/her activity over a period of the time. While this gives the site an opportunity to prevent user churn, predicting the long term value of a question page can help Q&A communities to continue providing high quality content to its users. We achieve these predictions using classifiers such as decision tree, linear SVM and SVM with RBF kernel.

**Problem description**

We focus on identifying the features that are aligned with user churn in Q&A sites and try to predict the users who are about to leave. In addition to this, we identified the question-answer pairs that add value to the sites. Our project dealt exclusively with stack overflow data. Stack overflow has become successful because of the continuous user involvement and the extra step that users take to flag duplicate questions, remove irrelevant information and upvote quality answers. As mentioned before, users are provided with reputation as a form of reward and a user can earn a reputation by answering questions, gaining upvotes for his posts. Posts are ranked in the order of the score they receive after calculating the difference between upvotes and downvotes. But, not every user has the privilege to upvote/downvote. Amateur users need to earn a certain amount of reputation to upvote (15) and to downvote (125). A questioner can offer bounty to answers with high quality. To identify features that are strong indicators of churn, we aim to understand the reasons for a new user to leave the site and for an enthusiastic user to leave the site after a certain number of posts.

We worked on the following tasks to accomplish our goal.

Task 1: Given the initial k posts of the user

Task 2: Given the activity of a user for the first T days

Predict: The likeliness of a user to churn

We considered four features that are most correlated with user churn namely Temporal, Knowledge level, Content, and Frequency and predicted the likeliness of a user to churn.

With variations in k and T, we try to understand how a particular feature is able to differentiate a churn and a non-churn user upon which, we use classifiers to predict the probability of a user to churn. As the trend shows that the majority of the people who churn have made less than 5 posts, it is useful to predict users who churn at early stages and therefore try to mitigate the number of users leaving the community. It is also important to predict expert users who churn after a significant amount of contribution as they have a lot of experience and it can affect the sites content quality. We consider 1≤k≤5 and 16≤k≤20 to be the number of posts and the values of T to be {7,15,30}.

With the understanding of the above features, we then try to perform the following

Task 3: Predict the long-term value of a question page based on the initial activity around the question.

Predicting the question pages that have lasting value and then inferring which properties are associated with them is useful to the maintainers of the site. Such question pages can be shown more often to the experts so that they can contribute to answers that add a long-term value.

**Methodology:**

This section presents a discussion on the methodology adopted to perform the three tasks. The first step is to identify the necessary features, that help in achieving the goal. Once we build the feature sets independently for all the tasks, they are passed to the machine learning models that classifies the training examples (users or questions, subject to the corresponding task). These steps are described in *Feature Selection & Description* and *Training* respectively.

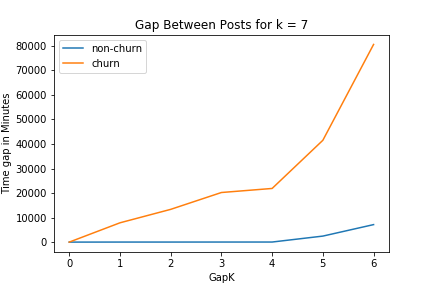
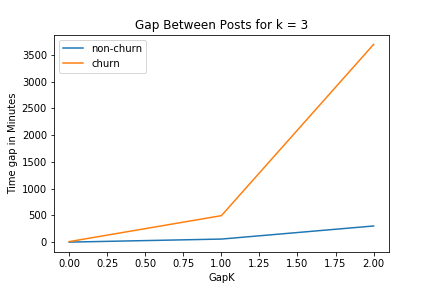
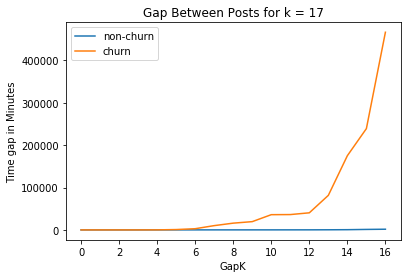
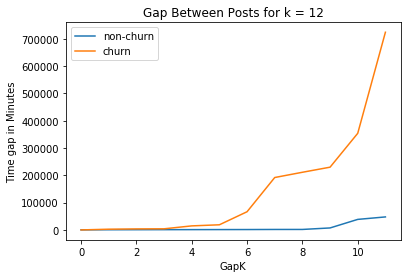
**Feature Selection & Description:**

**Task 1 & Task 2:**

**Temporal Features:**

Temporal features contain information about a user’s account activity. They contain information about a user’s posting pattern in the website (Stack Overflow). With the number of posts and/or the time gaps between them and a user’s activity in the first few days from account creation, we can analyze whether a user will leave the forum or not.   
The following set of features form the temporal features, that were considered in user churn prediction:

1. gap1: This feature is the gap in minutes, between the timestamps of a user’s account creation and his/her’s first post. This feature is used for both Task 1 and Task 2. This is because, the feature value is same irrespective of the k-value or T-value in the corresponding tasks. A very high value of this feature may indicate a possibility of the user to be a churn user.
2. gapK: This feature is a time gap between a user’s k-1th post and kth post for all k and used for Task 1. This feature constitutes the core of temporal feature. This feature captures how timely the user is in his posting and thus show how interested the user is in the site. These k features can be constructed by taking all the posts of interest and finding the time gap between any two adjacent posts. For this to be correct the posts has to be sorted by time first.
3. last\_gap: This feature value in minutes, is obtained from the difference between the timestamps of a user’s last post in the observation period (i.e., the period starting from the date of account creation to T days after that) and the preceding post. This feature is only required for Task 2 and a larger time gap indicates that user has taken significant amount of time to post in the community and this may lead to a possibility of the user being a churn user.

1. time\_since\_last\_post: This feature gives a value that contains the time gap in minutes, between a user’s last post (in the observation period, i.e., T days) and the observation period. This feature is only useful for Task 2 (which is also evident by the description) and a lesser time gap indicates that the user is inclined to posting in the community and is likely that the user continues to post even after the observation deadline. This in turn may mean that the user is possibly a non-churn user.
2. Mean\_gap: As the name suggests, this feature computes the average time gap in minutes from all the posts of a user within the observation period (T days). With the help of this feature, we get to know, how often a user has posted in the observation period and can estimate if he’s going to post anything in the forum after the observation deadline. From this we can classify a user as churn or non-churn. This feature can be constructed by considering all the posts made by a user, sorting them as per the creation date and computing the average of difference between successive posts.

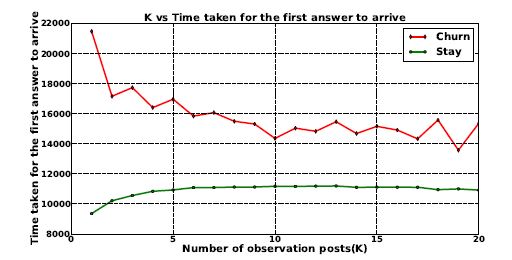
**Frequency Features:**

Frequency features give information about how regularly a user posts (either questions or answers) in the question answering site. If a user posts frequently, then we can possibly say that the user is a non-churn and vice versa.

1. num\_answers: Number of answers among the total posts made by a user (among the k posts for Task 1 or within the T days for Task 2).
2. Num\_questions: Number of questions among the total posts made by a user (among the k posts for Task 1 or within the T days for Task 2).
3. num\_posts: Total number of posts by a user in the observation period. This feature is specific to Task 2, because if a user has more number of posts in a given period, it is likely that the user continues to engage in the site and can be classified to be a non-churn user.

**Knowledge Level Features:**

Using the knowledge level features we can determine whether the domain interests of the user who asked the question match the users/experts of the community who answered the question. The higher the values for the knowledge level features of a user, the lesser the probability that the user will churn. We have considered the following 8 features for predicting the knowledge level of the community.

1. **Accepted\_answerer\_reputation** : Accepted answerer reputation feature value gives out the reputation mean of all those users who have answered a given user’s questions and whose answers were accepted by the user. Higher values for this feature value for a user signifies that the users domain interests match with that of the experts of the community.
2. **Maximum\_reputation\_answerer** : The value for this feature is the mean reputation of all those expert users who have answered a given user’s questions. Similar to the accepted answerer reputation, this feature also signifies that the user’s domain interests match with the experts of the community.
3. **Number\_questions\_answered** : This feature gives out the average number of questions posted by the user that got answered. One can correlate this feature value with the probability of churning of a user easily. For a given user, if the value of this feature is less, the user’s interest/satisfaction level to be active in the community would in general decrease and might result in churning of the user.
4. **Time\_for\_first\_answer** : The reason users post questions in CQA is to seek the help of the people who are having expertise in that domain. If a user’s question has not been answered for a long time, it might be because of the domain interests of the user is no match to that of the users of the community or the question was not given the attention by the experts of the community. If it is due to the first reason, then it would not be a problem to the community. But if it is due to the second reason, the user might not turn up to the CQA in future for a problem which might be of a considerable importance to a lot of people. So, the probability of the user churning would increase with longer the time taken for answering the question of the user. 

The above diagram depicts the Number of posts considered on the x-axis and the time taken for the first answer to arrive on the y-axis. From the diagram, we can infer that the probability of the user churning is more if the time taken for the first answer to arrive is more.

1. **Reputation\_questioner :** Thereputation questioner feature value gives the mean reputation for all users who posted questions that the user in the forum has posted the answer for. Using the reputation of the user ids that posted those questions, we find the mean of their reputation. Value of the mean reputation signifies the quality that the the questions add to the community forum. Higher is the mean reputation, more is the value that the question adds to the community.
2. **Reputation\_answerers :** The value given by the Reputation\_answers feature is the mean of the reputation of all the users that have answered the question posted by each user that posted questions on the community forum. The value from the Reputation\_answers feature can signify the quality of the answers received for that question. Since we are taking into account the reputations of the users who answered the question, which can give an idea of their knowledge in the domain that the question has been asked in.
3. **Number\_answerers\_received :** This feature suggests the mean number of answers received for each question that have been posted by users on the community forum. This also takes into account multiple questions that have been posted by a single user on the forum. The value gathered from this feature indicated the level of user engagement on the forum. With higher value from the feature, we can observe more number of users posting answers to the questions that have been posted by the users.

**Task 3:**

Our objective for this task is to find the most informative question answer pair which can add some value to the knowledge repository of the community forums. So, it is useful to know which attributes of a question helps in finding the best performing question answer pair in the long run. Which then can improve the workflow of the CQA forum moderators in cleaning and organizing the forum. Some ways in which this can improve the forums are by promoting the questions which have all the features of an informative post and asking expert users to answer the post thereby making the post invaluable to the community. The activity of a post in some sense captures the popularity and informativeness of a post, so good accuracies are achievable if features based on community dynamics are considered.

For this task we considered different classes of features as follows:

1. **Questioner features:** These features try to capture the experience of the questioner and could signify how clean and structured the question is.
   1. Questioner reputation: This can be obtained from the users data.
   2. Number of Questioner’s questions and answers: This information can be extracted from the posts data by filtering it by owner user id.
2. **Activity measure features:** This feature captures the activity the question attracted thereby signifying how informative or interesting this feature is to the community.
   1. Number of answers: This can be obtained from the posts data by filtering it by the parent id.
3. **Community dynamics features:** These features capture the kind of response the question garnered from the community.
   1. Number of comments on answer by highest reputation answerer: This can be obtained by first finding the highest reputed answerer’s answer and then taking the comments it got.
   2. Length of highest scoring answer: This can easily be obtained by looking at each answer’s length.
   3. Number of comments on highest scoring answer: This can be obtained by taking the answer with the best score.
4. **Temporal feature:** This feature, in a way captures the newness the question adds to the forum as normal and routine question can get an answer quickly.
   1. Time elapsed between question and best scoring answer.

Now, we use the above set of features to proceed to model training and also look into details of which feature was helpful in determining the long-lasting value of a Question-Answer pair to the CQA forum.

**Training:**

After selecting the features for the corresponding tasks (Task 1, Task 2, Task 3), and building the feature vectors, the next step is to train machine learning models on these features. For the purpose of Task 1 and Task 2, we have selected the following classifiers:

* Decision Tree: Decision trees visually represent a tree with decisions, required to classify an object or to attain a specific goal. The branches of these trees are features of data points, and similar to a tree, using these features we divide the branches until we arrive at a decision-making point.
* SVM with linear kernel (Linear SVM): Linear Support Vector Machines (SVMs), as the name suggests, linearly separates all the training examples, so that a new example can be classified correctly. For a binary classification task (here, classify user as churn and non-churn), the support vectors of examples from both the class labels are separated using a decision boundary and the main goal here is to separate the data points with maximum possible distance between them.
* SVM with RBF kernel: The working of this classifier is similar to that of linear SVM, except that here we use a Radial Basis Function kernel (RBF kernel) to separate data points as to linear separation. The RBF kernel function transforms the points from one space to another, and we solve the regular linear SVM in this transformed space with other similar operations.

Of these three classifiers, decision tree and SVM with RBF have non-linear decision boundaries whereas Linear SVM has a linear decision boundary to classify a user into a churn or non-churn user.

For task 3 in which we try to predict the long-lasting value of a question answer pair, we use the page views as the proxy for the long term value. An alternative variable that can be considered is the number of favorites of a question which is also a good predictor of long term value, as more number of favorites could mean many users saved it for future reference. But we went with the page views because the page views can be from a lot of different sources like search results etc. Whereas, favoriting a question is something only a user can do which can be limiting.

Next, we setup our problem in two different ways, one is given the data about the features we need to predict if the post lies in the top quartile or bottom quartile of the questions set considered in terms of pageviews. Two, we need to predict if the question lies in the bottom half or upper half of the question set considered. For this experiment, we have considered all the questions posted in a particular month and consider all these question as our observation set.

To compare our model for a relative performance measure, we use a baseline model. Our baseline model is the model with features based on crowdsourced attributes like upvotes, downvotes and favorites. So, we construct these crowdsourced features for our data and train a model with the target variables as described above and report the performance.

We also take into account that recognizing the value of a question as early as possible and with as less time restricted data as possible. So, we test our model with varying data size limited by the time they are registered like considering data gathered in one hour after the post and 3, 24, 72 hours respectively.

For model, we considered the decision tree and reported 10-fold cross validation accuracy. As we considered upper half or bottom half and upper quartile or lower quartile as our classes the data classes are equally split making the dataset balanced and does not need any kind of class balancing technique, whereas task 1 and task 2 needed class balancing as the number of churn users are less than the non-churn users.

**Results**

Task 1:

|  |  |  |  |
| --- | --- | --- | --- |
| K  (posts) | Decision Tree | SVM (Linear) | SVM (RBF) |
| 1 | 0.61 | 0.53 | 0.52 |
| 2 | 0.52 | 0.54 | 0.54 |
| 3 | 0.57 | 0.57 | 0.53 |
| 4 | 0.62 | 0.53 | 0.51 |
| 5 | 0.63 | 0.55 | 0.53 |
| 6 | 0.69 | 0.68 | 0.65 |
| 7 | 0.67 | 0.71 | 0.68 |
| 8 | 0.69 | 0.60 | 0.58 |
| 9 | 0.70 | 0.69 | 0.70 |
| 10 | 0.72 | 0.70 | 0.71 |

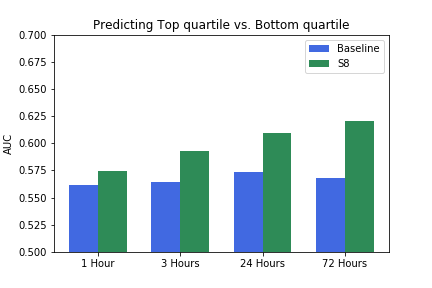
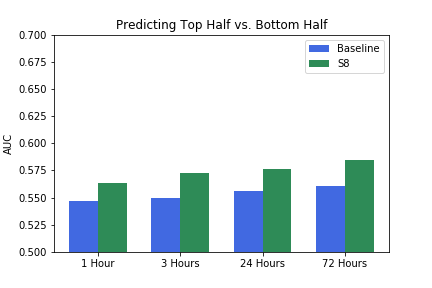
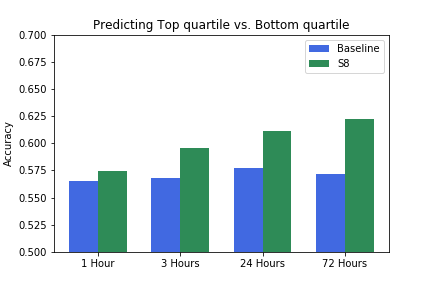
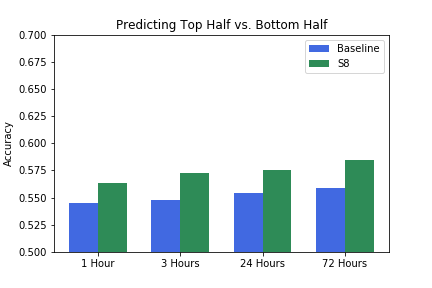
Task 2

|  |  |  |  |
| --- | --- | --- | --- |
| T  (days) | Decision Tree | SVM (Linear) | SVM (RBF) |
| 7 | 0.65 | 0.62 | 0.64 |
| 15 | 0.69 | 0.66 | 0.66 |
| 30 | 0.72 | 0.68 | 0.69 |

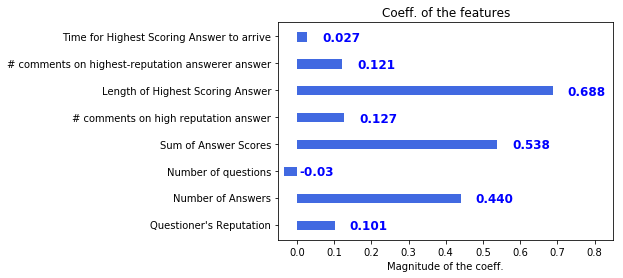
As we can see that decision tree has consistently performed better than Linear SVM and RBF kernel SVM and also in case of Task 1 the difference between their performance decreases as the value of k increases.

**Task 3**

As we can see that our the baseline model has not increased the performance much than just a coin toss which gives 50% accuracy which implies that native crowd sourced features are of little value. Whereas, our model (S8) has both absolute and relative performance in comparison to our baseline model. Also, it is worth noting that our model performed well in case of quartile prediction than with the half. We also used 4 different time windows for our analysis and it can be seen that if time window increases accuracy of our model increases, which means our features have more fidelity with more time window of observation. We also plotted the Area under the curve and has the same characteristics as the accuracy plots.



We further plotted the coefficients of the features (S8) to see which features has more predictive power and the plot is shown in figure below. We can see that the most important features are Length of highest scoring answer, Sum of answer scores, Number of answers. These three features thus has most of the predictive power for our task. Thus, showing that our feature set has indeep proved useful.



**Conclusions and future work**

We started our work focusing on how we can help forums identify inactive users and better divert the resources to help them get engaged with the forum. We also worked on what questions add value in the long run to the knowledge repository of a community based question answer forums like stackoverflow. We saw that the temporal and knowledge level features has both combined gave good accuracies for the churn and non-churn user prediction. Also we have seen that temporal features which summarizes how timely a user posts, has the most predictive power of all the features considered for task 1 and task 2. We saw that for higher values of K the difference between performance of various models decreased, which could be because as K increases the number of features increases thereby increasing the dimensionality and the data could become more linearly separable. For task 3 we have seen that over model has strong absolute and relative performance compared to the crowd sourced feature model. We have predicted the long term views (proxy for long term value) of a question page for an year later given the data of an hour of window after the question is posted and saw our model has done a great job.

As future extension we would like to use the knowledge gained in task 1 and task 2 to better divert the resource to less focus users (churn users) and enhance this process with the help of expert finding to accomodate the need of the user, thereby increasing the users retention and participation of the site. We would also try to divert resources to the questions which would likely be of value in the long run to better enhance the quality of these question and its answers this also can be done by the expert finding and promoting the question to get it answers by prominent users. Another extension is to find if a question is sufficiently answered and this can help with the above process by showing when to stop promoting the question as it is sufficiently answered.

**Contributions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Team Member** | **ASU ID** | **Email ID** | **Contributions** | **Percentage** |
| Sai Nageswara Koushik Dhumantarao | 1213158581 | sdhumant@asu.edu | Task 2 | 16.67% |
| Pujith Sagar Edara | 1212331261 | pedara@asu.edu | Task 2 | 16.67% |
| Pratik Mishra | 1213076694 | pmishr14@asu.edu | Task 1 | 16.67% |
| Yeswanth Narra | 1213523673 | ynarra@asu.edu | Task 3 | 16.67% |
| Ravi Teja Upmaka | 1213086704 | rupmaka@asu.edu | Task 1 | 16.67% |
| Venkata Ravi Teja Reddy Yanamala | 1213420206 | vyanamal@asu.edu | Task 3 | 16.67% |

**References**

1. A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, “Discovering value from community activity on focused question answering sites: A case study of stack overflow,” in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 850–858.
2. J. Pudipeddi, L. Akoglu, and H. Tong, “User Churn in Focused Question Answering Sites: Characterizations and Prediction,” Proc. 23rd Int’l Conf. World Wide Web (WWW 14), 2014, pp. 469–474.
3. W. Li, C. Zhang, and S. Hu, “G-finder: routing programming questions closer to the experts,” in *Proc. OOPSLA,* pp. 62-73, 2010.
4. Y. Yao, H. Tong, F. Xu, and J. Lu, “Predicting long-term impact of cqa posts: A comprehensive viewpoint,” in *KDD’14*, ACM, 2014.
5. Y. Richter, E. Yom-Tov, and N. Slonim, “Predicting customer churn in mobile networks through analysis of social groups,” in *Proc. SDM 10, Soc. Ind. Appl. Math.*, Philadelphia, PA, 2010, pp. 732–741.
6. J. Arguello, B. S. Butler, E. Joyce, R. Kraut, K. S. Ling, , and X. Wang. Talk to me: foundations for successful individual-group interactions in online communities. In CHI, 2006.
7. A. Anderson, D. P. Huttenlocher, J. M. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: a case study of stack overflow. In KDD, pages 850–858, 2012.
8. H. Oktay, B. J. Taylor, and D. Jensen. Causal discovery in social media using Quasi-Experimental designs. SIGKDD Wkshp Soc. Media Analytics, 201
9. Y. R. Tausczik and J. W. Pennebaker. Predicting the perceived quality of online mathematics contributions from users’ reputations. CHI, 2011