StackOverflow Data Analytics

Ashwin Bhide
Georgia Institute of Technology
Atlanta, Georgia
ashwin.bhide@gatech.edu

Chaitanya Bapat
Georgia Institute of Technology
Atlanta, Georgia
chai.bapat@gatech.edu

Nidhi Menon
Georgia Institute of Technology
Atlanta, Georgia
nmenon34@gatech.edu

Sneha Venkatachalam
Georgia Institute of Technology
Atlanta, Georgia
sneha30@gatech.edu

Vaibhav Tendulkar
Georgia Institute of Technology
Atlanta, Georgia
vtendulkar@gatech.edu

1 INTRODUCTION

StackOverflow, a flagship site of StackExchange network is an online question-answer community providing rapid access to knowledge and expert peers. Research on StackOverflow shows certain shortcomings in terms of user engagement and question answering.

There is an increasing trend in number of unanswered questions on StackOverflow, and percent distribution of answered questions is skewed according to question domain and other factors. Currently, StackOverflow displays unanswered questions by tags instead of routing questions to specific users. The limitation of this approach is that there is no intelligent algorithm that makes use of extracted features to identify experts and route the pertinent questions to them.

2 PROBLEM DEFINITION

(1) To analyze StackOverflow to find trends in data, and understand community user engagement and dynamics
(2) To design a question routing and expert-recommendation system that will enable experts to provide satisfactory answers to questions
(3) To determine whether a question will be satisfactorily answered on StackOverflow from the features of the question.

3 SURVEY

StackOverflow a flagship site of the StackExchange network is an online question and answer community providing rapid access to knowledge and expert peers. To understand the reason StackExchange is the most-used website for QA, we referred to the paper[1] by Lena et al. The reason behind StackOverflows prominence is the active involvement of software developers in content moderation and presence of game mechanics[6] to make the forum competitive. Moreover, the use of StackOverflow has close relation with the the time software developers spend on actually answering questions versus coding. Bogdan et al[2] tried to explore such an interplay between StackOverflow and software development reflected by code changes in Github. The results highlighted that the more active Github committers provided more answers while asking fewer questions. Similarly, Octay et al [3] suggest using Quasi-experimental Designs to analyse human behaviour on Stack Overflow and determine the cause-and-effect relationships, involving approaches which we could incorporate for analyzing long-lasting value of questions.

User engagement plays a major role in the success of online communities. Laura et al.[4] establish a positive correlation between user reputation scores and their contribution to a diverse tags while Bogdan et al [5] assessed the representation and social impact of gender in Stack Overflow, which can be used to target sections of the community for increased user participation. Gharibi et al. [6] talk about a system that recommends unachieved badges to users based on their behavior, to increase user engagement by calculating the correlation between unachieved badges and users previously awarded badges. In [7], the authors used unsupervised learning to categorize mined Stack Overflow questions and also defined a ranking algorithm to rank questions to understand issues faced by web developers. Asaduzzaman et al. [15] analyze stackoverflow data to see why unanswered questions remain unanswered. These methods could be incorporated into our project in order to extract more valuable information helping the software community to understand and address such issues.

Another aspect contributing to the growth of StackOverflow is its expert finding mechanism. Morakot et al. [8] propose a feature based prediction approach to predict who will answer a given question using Random Forests classification technique, suggesting a social-network based approach to exploit the relational attributes of the community. Thiago et al in [9] study the behavior of the experts in the online communities using TF-IDF measure to rank topics of expertise and introduces a recommendation approach based on trending topics. Zhou et al. in [11] consider the question routing problem as a classification task, deriving a variety of features about the question, user and the relationship between them. Guo et al. in [12] suggest a generative model to discover latent topics and interests to recommend answer providers. Ashton et al in [16] talk about relationships and temporal characteristics to calculate long-lasting value of answers and identifying questions without satisfactory answers. These can be combined with the linear-regression based recommendation model in [10] proposed by Chang et al. that finds domain experts and detects low quality answers to be routed to groups of users rather than a single expert, for improved performance. Jun et al. in [13] analyzed the network representing the asker-helper interactions in an online community and concluded that it produced a different bow-tie structure than what is associated with the graph of World Wide Web. Whereas, Pedro et al. [14] come up with a rankSLDA algorithm which combines supervised ranking with topic modeling for recommending questions to users based on their domain-based ranking.

4 PROPOSED METHOD

4.1 Intuition

Currently, Stackoverflow relies on a upvote-based system to display unanswered questions to potential answerers, and users get to see the most upvoted questions when they log in. We propose an approach to extract features to identify experts in particular domains and route relevant questions to them. This approach will be successful as routing the question to experts will help the questioner get the correct answer quickly.

Our system relies on the intuition that if a user has answered a question in a particular domain satisfactorily in the past, there is a higher probability that he will be able to answer similar questions belonging to the same domain in future. To implement our approach, we rely on parameters such as question tags, readability score etc. We also built a classifier to predict whether a question will be answered using features such as readability score, number of code fragments, external links etc. The intuition behind including the readability score in the question vector is that higher readability score implies that the user will be able to understand the question better and hence provide a more satisfactory answer.

Similarly, greater the number of code fragments and the word count of the question, greater is the amount of time and effort a user has to put into answering that question which might dissuade some users from doing so.

4.2 Description

Data extraction and cleaning:

We used the raw data from StackExchange Data Archive[17]. Since the available data was huge, we wrote SQL queries to extract required data fields like tags, view-counts, bounties, etc. in CSV format. The cleaning process involved filling in the missing values using OpenRefine, removing HTML tags using Python to ensure that cleaned text was passed on to the analysis phase. Fields such as the number of lines of code and the readability score of the post were calculated by using nltk library in Python. Higher readability score indicated greater likelihood of users understanding and answering the question. Using the readability library, we calculated text readability using algorithms like Flesch-Kincaid, Coleman-Liau, Dale-Chall, SMOG, Automated Readability Index, Flesch Reading Ease and averaged outputs to find the final readability score of the post.

Question Recommendation:

A lot of users ask questions on StackOverflow every day, in all these questions, the answerers may get lost in the haystack of questions while looking for questions they would likely know the answer to, and would want to answer. To help them navigate through this haystack, we propose a solution in which we recommend questions to users based on their past activity of what questions they have given answers to. Rather than have them navigate through the haystack looking for the proverbial needle, we try to present them with the needle without them having to put in too much effort. We used K-means to cluster questions based on the tags they have been given. K in our model is the unique number of tags in our dataset. We keep track of the number of answers a user has given in a particular domain and the average answer score per domain. When a new question comes in, we query to see what cluster the question belongs to and then check the top 5 answerers for that cluster and recommend the question to them. For the above approach we have used user features such as user id, question id, tags, answer score.

We used Random Forest classification algorithm to make a prediction based on a set of features of questions whether or not a question has been answered. The feature vector for questions included tags associated with the question, length of the question (word count), readability score, number of code fragments and the number of external links in the question. The intuition behind including readability score, word count, number of code fragments and number of external links has been explained earlier.

Our approach was divided into two phases :

- Learning phase : In this phase, we trained the random forest classifier by feeding it a set of known question vectors and y values indicating whether the question was answered or not. We constructed a forest with 10 trees and we did not impose a limit on the depth of the trees. We used Gini index to determine which feature will be used to split the tree.
- Testing phase : We set aside a fraction of our data which was not used for training the algorithm. Once the learning phase was completed, we gave question vectors to the algorithm to try and classify whether the question was answered or not. The algorithm predicts whether a question with the given features is likely to be answered. To test the accuracy of the algorithm, we compare the output of the algorithm to the actual label of the question (whether it was actually answered or not).

Data analysis and visualization:

We divided the data analysis process into two parts. In the first part, we analyzed user engagement on StackOverflow by monitoring the number of questions asked, and the number of upvotes and downvotes they received. A similar study was conducted on the answers posted to these questions. In the second part, we focused on the major issue faced by StackOverflow users- unanswered questions. The data we scraped indicated an increasing trend in the number of unanswered questions over the past few years which defeats the core purpose of the question-answer community. Another study revealed the increase in the number of bounties that users place on their questions in hopes of receiving immediate answers. These analyses laid the foundation of our project, giving us the objective of building a system that could help improve StackOverflow. These visualizations included in this report were generated using Tableau.

5 INNOVATION

(1) Inclusion of Readability score : Intuition behind including readability score was that a higher readability score implies easier understanding of question. It in turn enables more users to answer the question with greater accuracy.
(2) Intelligent question routing mechanism driven by question tags & user skill match.
(3) Expert-recommendation system that enables experts to provide satisfactory answers to questioners while ensuring valuable user contribution.

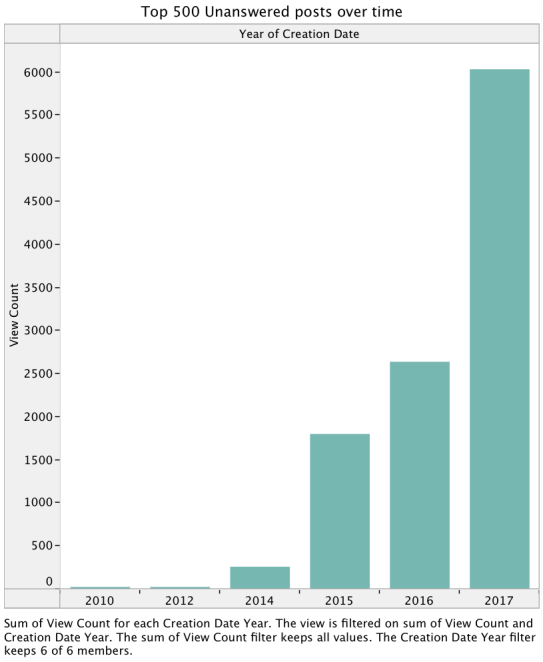

Figure 1: Top 500 Unanswered posts over time

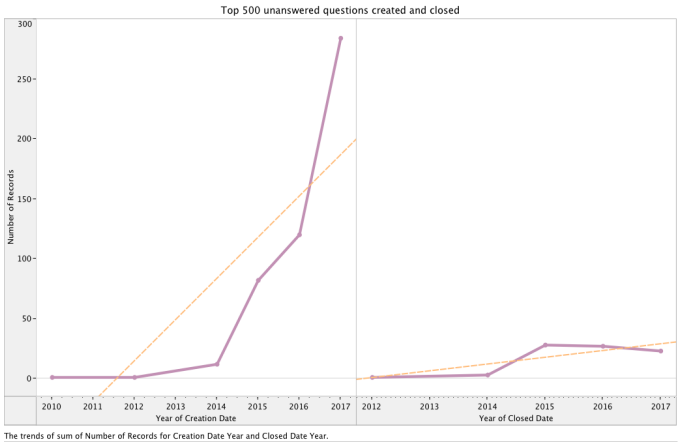

Figure 2: Top 500 unanswered questions created and closed

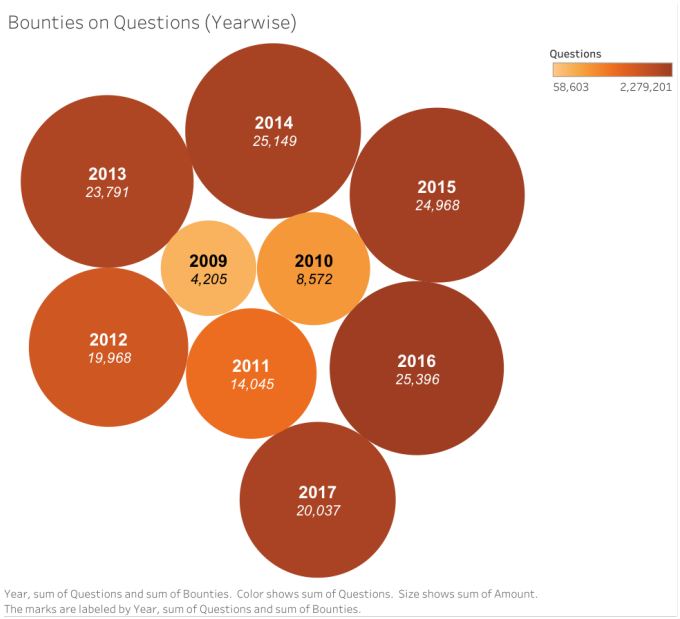

Figure 3: Bounties on Questions(Yearwise)

6 EXPERIMENTS AND EVALUATION

6.1 Description of testbed :

(1) Via Visualizations
  (a) What is the trend in unanswered questions?
  (b) How increasingly important is it becoming for users to place bounties on questions?
  (c) What is the trend in user engagement in terms of question-answers and upvotes-downvotes?
(2) Via Machine Learning Model
  (a) Which user is most likely to answer a given new question correctly?
  (b) Which users are the top answerers for a given domain?

6.2 Observation & Analysis

6.2.1 ***Kmeans Clustering Evaluation***. We clustered users based on the questions they have answered in the past, by using features such as tags, answer score, number of answers etc. After clustering users, when we get a new question, we find the cluster that it belongs to, and then recommend users in the cluster as potential answerers to the question. We played around with different features that the UCI Irvine dataset contained, and tried to find a feature set that made sense to solve our problem. We could not actually route questions on StackOverflow, so we had to come up with another way of checking if our recommendations were making sense. So, we looked up the StackOverflow profiles of the recommended users, and checked to see if they have answered a relatively high number of questions in that domain to see if our results were making sense. We found that in the majority of cases (37/50), the users we recommended made good candidates to answer the question.

6.2.2 ***Random Forest Evaluation***. Evaluating the accuracy and F1 score of random forest classifier using 10-fold cross validation test

Accuracy	0.958374628345
F1 score	0.978737138067

We used the random forest classifier to predict whether a question had been answered or not, using the question vector created. We saw an average accuracy of 0.958374628345. We think the data may create a little bias as a majority of the questions have already

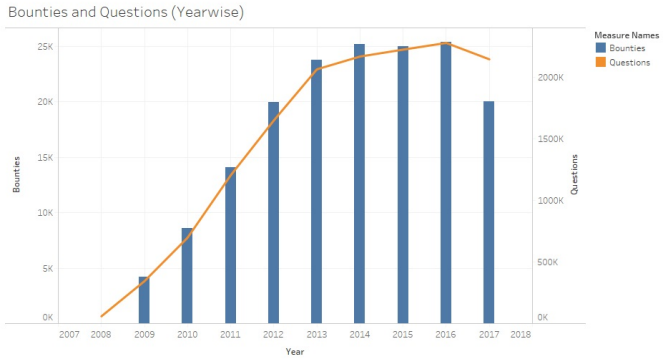

Figure 4: Bounties and Questions(Yearwise)

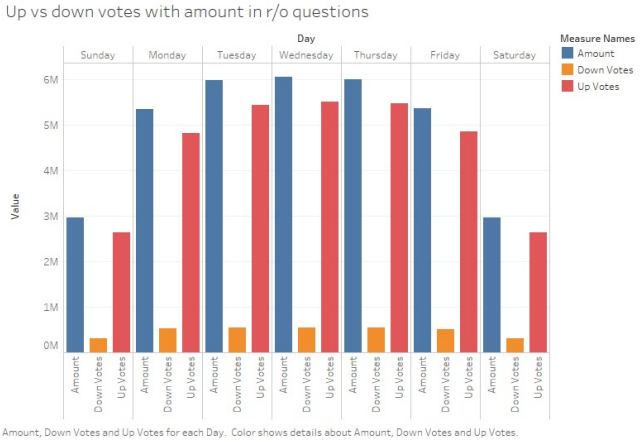

Figure 5: Ups vs down votes with amount inr/o questions

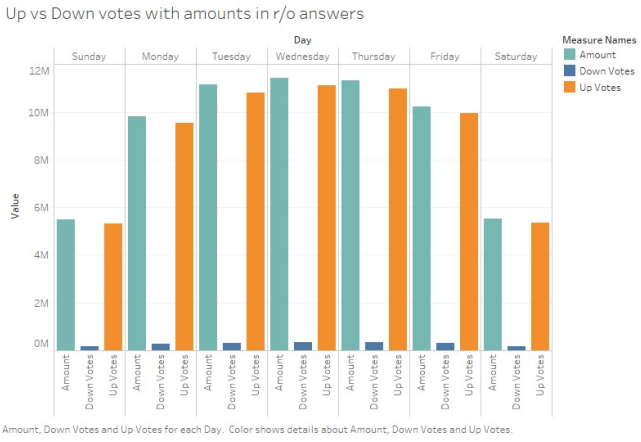

Figure 6: Ups vs down votes with amount in r/o answers

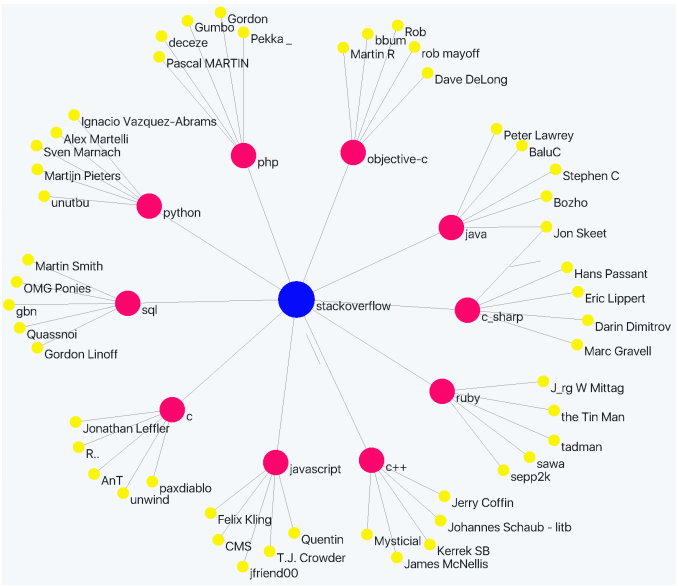

Figure 7: Visual Representation of K-means

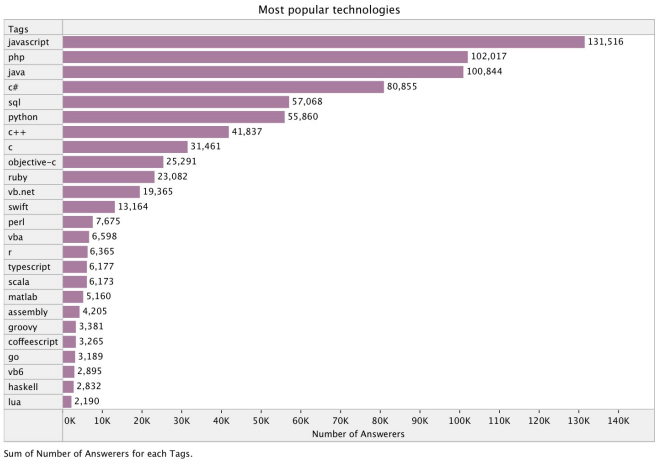

Figure 8: Most popular technologies on StackOverflow

been answered. Hence, using an updated dataset may have made more sense for this experiment.

6.2.3 *Visualization Observations.*

(1) Motivation:
   (a) Trends in unanswered questions: Fig. 1 shows top 500 unanswered posts by their view counts for the years 2010 to 2017. Fig. 2 depicts the trends in the number of unanswered questions created and closed over the years.
   (b) Analysis of bounties on questions: Figures 3 and 4 depict the year-wise trends in the amount of bounties placed on questions. The number of questions, number and amount of bounties are depicted visually by color, labels and size.

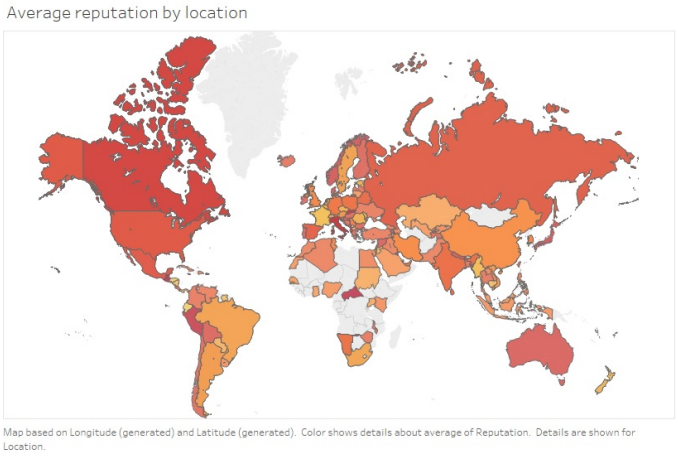

Figure 9: Average reputation by location

Trend lines represent the increasing number of bounties over the years.

These statistics provided motivation towards addressing the problem of increase in the number of unanswered questions. The visualizations helped in analyzing factors such as increasing numbers of bounties and closure of such questions, which further prompted us to propose a suitable solution to this dilemma.

(2) User Engagement Analysis:
   (a) Analysis of upvotes and downvotes of questions and answers for each day: Fig. 5 and 6 represent the trends in upvotes and downvotes on StackOverflow on a per-day basis, by monitoring weekly statistics of the number of questions and answers posted, and the number of upvotes and downvotes they received.
   (b) Choropleth map of reputation by region: Fig. 9 is a choropleth representation that displays average reputation of users by location worldwide. Color shows details about average of reputation with darker colors representing greater reputation.
(3) Visual representation of K-means clustering: We have envisaged the K-means clustering approach with the help of an Argo visualization, which represents cluster centers corresponding to the most popular technologies on Stackoverflow this year. The process of predicting users who have answered the most number of questions in the cluster is depicted for each domain. Fig. 7 depicts clusters of potential answerers for each domain cluster.

In the first part, we analyzed user engagement on StackOverflow by monitoring the number of questions asked, and the number of upvotes and downvotes they received. A similar study was conducted on the answers posted to these questions. In the second part, we focussed on the major issue faced by StackOverflow users-unanswered questions. The data we scraped indicated an increasing trend in the number of unanswered questions over the past few years which defeats the core purpose of the question-answer community. Another study revealed the increase in the number of bounties that users place on their questions in hopes of receiving immediate answers. These analyses laid the foundation of our project, giving us the objective of building a system that could help improve StackOverflow.

Model driven analysis using k-means clustering accomplished grouping of questions based on the tags and the score. We evaluated the model by testing it with outsample data to confirm that the tags retrieved from the given unknown question helped to find top users related to the particular tag. Here for example, we considered top 100 tags and were able to train the model with 250,000 questions based on the features - question id, tag id, answerer id and answer score.

User engagement analysis performed on data revealed that despite receiving a huge number of views, questions on StackOverflow remain unanswered for long periods of time (see Fig. 1). Fig. 2 shows an increasing trend, both in creation of questions and in closing of unanswered questions by users, due to which, in recent years, more users are forced to place huge bounties on questions requiring immediate answerers, as depicted in Fig. 3 and Fig. 4. These results drew our attention to the need for expert-recommendation and question-routing systems. Drilling down to a period of one week, we also visually analysed user activity in terms of upvotes and downvotes for questions and answers on StackOverflow in Fig. 5 and Fig. 6.

7 CONCLUSION AND DISCUSSION

StackOverflow is a very popular question-answering community today with millions of users worldwide. As newer technologies emerge, the database and the number of users are increasing rapidly. More often than not, a vast chunk of questions go unnoticed and unanswered in this deluge of data. While StackOverflow has officially not implemented any corrective measure yet, some papers have suggested simple methods like linear regression to tackle this problem.

We suggest approaches that are better at accuracy than the ones mentioned in these papers. For k-means clustering, we chose cosine similarity as our hyperparameter whereas for random forest we consider multiple factors like readability score and number of code fragments. One possible enhancement in our method could be the removal of inactive users from the pool of StackOverflow users considered by our algorithms.

There is much scope for further improvement of the StackOverflow platform by incorporating more features that can improve the user experience. One such feature is predicting the time until answer for a question posted by a user. This is likely to depend on the subject matter of the question, since some programming communities are more active than others.

8 DISTRIBUTION OF TEAM MEMBER EFFORT

All team members have contributed similar amount of effort.

9 REFERENCES

[1] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, BjoĬĹrn Hartmann, Design Lessons from Fastest QA Site in West

[2] Bogdan Vasilescu,Vladimir Filkov, Alexander Serebrenik, StackOverflow and GitHub: Associations Between Software Development and Crowdsourced Knowledge

[3] H. Oktay, B. J. Taylor, D. D. Jensen, Causal Discovery in Social Media Using Quasi-Experimental Designs

[4] Laura MacLeod, Reputation on Stack Exchange: Tag, You are It!

[5] Bogdan Vasilescu, Andrea Capiluppi, Alexander Serebrenik - Gender, Representation and Online Participation: A Quantitative Study of StackOverflow

[6] Gamified Incentives: A Badge Recommendation Model to Improve User Engagement in Social Networking Websites

[7] K. Bajaj, K. Pattabiraman, A. Mesbah, Mining Questions Asked by Web Developers,

[8] Morakot Choetkiertikul, Daniel Avery, Hoa Khanh Dam, Truyen Tran and Aditya Ghose , Who will Answer my Question on Stack Overflow?

[9] Thiago B. Procaci, Bernardo Pereira Nunes, Terhi Nurmikko-Fuller, Sean W. M. Siqueira, Finding Topical Experts in Question & Answer Communities

[10] S. Chang and A. Pal, Routing Questions for Collaborative Answering in Community Question Answering, IEEE/ACM International Conference, 2013

[11] T. C. Zhou, M. R. Lyu, I. King, A Classification-based Approach to Question Routing in Community Question Answering,

[12] J. Guo, S. Xu, S. Bao, and Y. Yu, Tapping on the Potential of Q&A Community by Recommending Answer Providers,

[13] Jun Zhang, Mark S. Ackerman, Lada Adamic, Expertise Networks in Online Communities: Structure and Algorithms

[14] Jose San Pedro, Alexandros Karatzoglou, Question Recommendation for Collaborative Question Answering Systems with RankSLDA

[15] Muhammad Asaduzzaman, Ahmed Shah Mashiyat, Chanchal K. Roy, Kevin A. Schneider, Answering Questions about Unanswered Questions of Stack Overflow

[16] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, Jure Leskovec, Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow

[17] StackExchange, 2017 Retrieved from https://data.stackexchange.com/