Change in User Status And Activity Between Sub-communities in Stack Overflow

Conrad Chan, Alexander Hsu, Changwhan Yea

1. Introduction

Stack Overflow is one of the most popular websites for asking questions related to software development. It has a well-established reputation system that gives users incentives to ask and answer questions, as well as to evaluate content generated by other users. However, the current reputation model is not a good indicator of the user's status because it relies on various user actions that may not necessarily relate to the actual skill level of the user. For example, a user's reputation goes up every time he asks a question, an action that should not necessarily give him higher "status". Moreover, while the reputation on the Q&A site is aggregated from actions in the entire network, we believe that a user's status can be different between sub-communities within Stack Overflow. In this study, we aim to examine the characteristics of different sub-communities in the website and see how they correlate to the difference in users' status across sub-communities. We also come up with another metric, the activity index (explained below), as we believe there may be potentially interesting trends there between sub-communities.

To conduct our research, we established two different user characteristics, status and activity index, for each user within a sub-community. The status represents how the user is respected in the sub-community by other users, and the activity index tries to capture the "contributory or leeching" nature of a user, measuring a user's tendency to ask or answer questions. Similarly, we defined two different ways of measuring similarity between sub-communities, context-based and user-based similarities. With these definitions, we find how the status and activity behavior of users change between two different sub-communities and observed whether they correlate to the level of similarity between the sub-communities.

2. Prior Work Discussion

In the design phase of our research, we looked into three different papers that helped us construct the approach to our study. [1] gives a great overview of Stack Overflow and its reputation score system. It was also insightful in noticing that the timeline of responses to a question take a somewhat "pyramid" format based on expert users. The authors examines how reputation, specifically community involvement, on Stack Overflow correlates with other behavior, such as how users arrive to answer new questions and how their answers are perceived by the community.

In [2], Anderson et al. discusses how similarity in the characteristics of two users affects the

types of evaluations that one user gives to another. The paper found that evaluations are less status-driven when users are more similar to each other and proposes that a certain evaluation can be predicted from a group knowing only the attributes of the members. Anderson et al. provides clear definitions of status and similarities and the reasoning behind them. For Stack Overflow, they specifically explained how simply using the reputation score in the website's database cannot correctly represent status to be used for their research purpose. This paper was referred to establish our definitions of user status along with context-based and user-based similarities.

In [3], Zhang et al. used Z-score, a simple feature-based measure such that a user with a higher score is more likely to be an expert than a user with lower score. A higher Z-score implies that experts answer more questions and ask very few questions. This notion became the basis of activity index which we define later.

3. Data Collection

We obtained a complete trace of all the actions on Stack Overflow from its inception on July 31, 2008 to September 6, 2013, which is publicly available at the community's website. The raw data contained post-level xml data, which we found difficult to directly query on for our purposes. Therefore, we parsed the data and loaded it into a SQLite database. As the data size is extremely big, we created a smaller database that contains 100,000 posts for initial implementation and testing. Since we are primarily interested in looking at user activity under different tags, we designed and created a separate user-level database which we obtain from aggregating posts with tags. Each row in this database contains the user's activity and score under a specific tag.

	Total	Note
Posts	**15,345,130**	**35.71% questions 64.29% answers**
Questions	**5,479,812**	**59.87% accepted answers**
Answers	**9,819,720**	**33.41% were accepted**
Users	**2,121,913**	**48.37% asked questions 32.60% answered**
Votes	**36,435,956**	**91.54% are upvotes**

Table 1. Overview of Stack Overflow Database

4. Retrieval of Top Subcommunities

We determined sub-community based on the tags in each post. For example, if a user wrote a question or answer that has a tag ‘C++’, he or she is part of the ‘C++’ sub-community. Also, if a post has multiple tags ‘C++’ and ‘Java’, then the user who wrote the post is in both the ‘C++’ and ‘Java’ sub-communities. For our purposes, we retrieved the top 20 sub-communities by selecting those with the highest total number of questions and answers posted by their members. Out of the 2,121,913 distinct users on Stack Overflow, 1,011,197 of them are associated with the top 20 sub-communities by either posting a question or a answer in one of the sub-communities. Table 2 shows an overview of the top five sub-communities.

	‘c#’	‘java’	‘php’	‘javascript’	‘android’
Total users	**193,495**	**230,859**	**216,586**	**244,545**	**155,470**
Total questions	**489,497**	**458,254**	**426,287**	**425,852**	**138,214**
Total answers	**1,018,599**	**944,518**	**859,839**	**831,320**	**562,890**
Total accepted answers	**39.111**	**39,873**	**37,578**	**45,622**	**36,554**
Average questions per user	**2.53**	**1.98**	**1.97**	**1.74**	**0.89**
Average answers per user	**5.26**	**4.09**	**3.97**	**3.40**	**3.62**
Average accepted answers per user	**0.20**	**0.17**	**0.17**	**0.19**	**0.24**
Average score per user (upvotes-downvotes)	**10.52**	**8.02**	**5.55**	**6.16**	**6.13**

Table 2. Basic statistics for the top five sub-communities

5. User Characteristics

5-1. Number of Associated Sub-communities

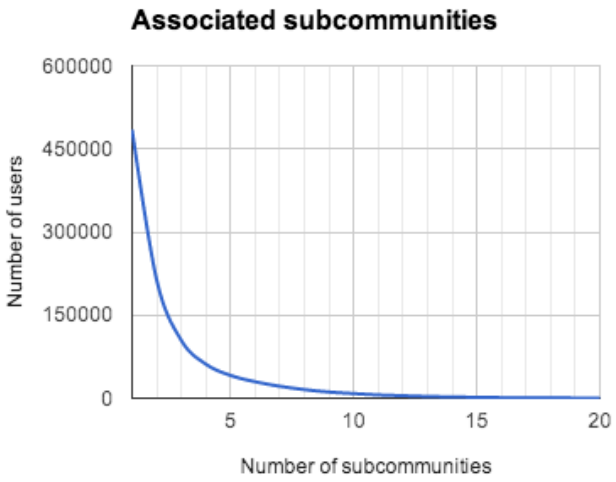

Figure 1. Distribution of number of associated sub-communities

Since our study aims to find status differentiation of users across sub-communities, it is important to understand how the number of different associated sub-communities is distributed amongst users. As expected, the majority of users are associated with only a few sub-communities From the above graph, we can see that the number of users decrease exponentially when the number of associated sub-communities increase. Among the 1,011,197 users in the top 20 sub-communities, 484,714 users have activities in exactly one sub-community, 213,223 users have activities in two sub-communities, while 201 users have activities in all 20 sub-communities.

5-2. Status

We defined the user status of a user in a particular sub-community with the following equation.

$$status = \#\ of\ net\ votes\ /\ \#\ of\ answers$$

net votes are the sum of upvotes and downvotes for answers within a specific sub-community

This score is the most appropriate because the resulting fraction represents how well someone is received in their subcommunity: a large status would mean that the user's posts on a certain topic are well-liked. Using simply the net votes alone is insufficient because an active user who is not necessarily high status could receive a high score simply from a large number of posts.

It is also very important that we considered net votes, rather than upvotes alone. This not only offers a more whole picture of status, but also allows status to be comparable between subcommunities. For example, with upvotes only, the size of a subcommunity may play a role in affecting a user's status because there are more users who could possibly vote up someone's post. Using net votes ensures that while there are more users who could upvote, there are also more that can downvote. The numerator is not expected to fluctuate significantly as a function of size, and the denominator (# of answers) is not

affected by size of the subcommunity either. This allows for a more accurate comparison of status between different communities.

We considered PageRank on a graph, specifically where there is an edge from user A to user B if A upvoted B's post within the subcommunity at hand, to calculate status, but in the end we did not think it made sense to give certain votes more weight than others in the context of StackOverflow. Unlike the web, where the notion of a "high quality page" is subjective in nature and thus, a page should benefit more if linked to from a more "high quality" source, upvotes in StackOverflow are given on a more objective basis, to the answer that most correctly helps the user; the concept of high quality is more rigidly defined. The website prides itself on their objectivity, expecting answers "to be supported by facts, references, or expertise" and rejecting questions that will likely "solicit debate...or extended discussion".

5-3. Activity Index

In order to determine the type of activity that a user usually engages in his or her sub-community, we established a score called the activity index. The definition of the activity index is

$$A - Q / (A + Q)$$

Q: Number of questions asked by user in sub-community
A: Number of answers posted by user in sub-community

This index also varies, for a given user, per sub-community. The index is bounded within the range of -1 to 1, with a value of 1 representing that the user only posts answers and a value of -1 representing that the user only asks questions. Thus, a user with a high activity index in the Java subcommunity is likely to post answers more than questions. The activity index lets us characterize the nature of a user's behavior. As a user characteristic, we computed the activity index of different users in certain sub-communities and find the correlation of the values across different subcommunities. Like status, we made sure to choose a definition that lets a user's activity index in two different subcommunities be comparable. A large subcommunity does not necessarily equate to a change in a specific user's behavior in that subcommunity.

6. Subcommunity Characteristics

6-1. Interaction Index

6-1-1. Limitation of the Bowtie Model Assumption

Coming into the project, we believed that it was valid to assume that sub-communities in Stack Overflow would also follow a bowtie structure similar to what was evident in the Java Forum (which

had 12.3% of users in its core, 54.9% of its users in its in-component, and 13% in its out component) due to similarities between both platforms. On first glance, the interaction indices that we generated ((in% - out%) / core%) seemed to produce valid output. However, closer inspection of the size of each component going into the formula proved otherwise, as no component (in, out, or core) for any subcommunity constituted for over 2% of the subcommunity population (for example, a typical one would have .03% core, 1.3% in, and 0.0% out). In hindsight, this makes logical sense because the Java Forum is less problem-focused than Stack Overflow, and thus general discussion is more permissible; in purely question-answer settings, having a strongly connected core is unlikely, as there is no central community due to the sparse nature of each subcommunity.

6-1-2. Use of Random Deletion

Since our initial findings show that the sizes of in-component and out-component for all subcommunities are insignificant, we used random deletion to better define the structure of a subcommunity. We suspected that the low percentages we detected for each part of the bowtie model could be due to the large size of the subcommunities and wanted to see if different results would come from smaller instances of the graph while preserving the graph's nature. Specifically, we deleted x% (varying x) of all nodes in the network, found the percentage values of each component, and repeated 100 times to find the average of the values. By conducting the process over different x values, we hoped to plot how the values change over the scale and get a better interaction index of the subcommunity than what we originally proposed. This too, however, did not fare well. While percentages were slightly higher for each component for each subcommunity, they were still all falling below 4%. In conclusion, applying an interaction index based on a model that does not fit well with the data would not have produced meaningful results.

6-2. Similarities

6-2-1. Context-based similarity

We looked into how similar two different sub-communities are in terms of the context that their users post. Given two sub-communities A and B, we defined their context-based similarity using Jaccard index.

$$\text{Context-based similarity} = |QA \cap QB| / |QA \cup QB|$$

QA: Set of questions in sub-community A
QB: Set of questions in sub-communityB

Figure 3 shows the context-based similarities of the 'java' sub-community with other top 20

sub-communities. It has a very high similarity with the ‘android’ sub-community, since Java is the language used in the Android platform, while not being very similar contextually to the ‘iphone’, ‘ios’, or ‘ruby-on-rails’ communities. With the top 20 sub-communities, all 190 possible pairs of sub-communities have the context-based similarity value computed. Context-based similarity was used to evaluate how user status and activity index differ between sub-communities.

6-2-2. User-based similarity

We recognized some limitations with our approach when plotting context-based similarity with these; notably, in the way such terms were calculated. Context-based similarity increases when there are many questions in common that are used to create the two subcommunities (denote this set of questions as X). The difference between the status of a user in subcommunities A and B is likely to be low if A and B are context-based similar because his status in each is constructed primarily over largely the same set of questions X. Similar concerns apply for the activity index as well.

To address this concern, we decided to only examine pairs of subcommunities where the context-based similarity is low, to see if there are any conclusions when there is a low overlap in questions belonging to the two subcommunities. This could still potentially yield results to significant questions: i.e. do the status of users remain high even when evaluated in a different subcommunity with different content? Do the same users behave differently when in a very different subcommunity (content-wise)?

Additionally, we also considered other options next, such as the notion of a user-based similarity. This assigns another similarity score to a pair of subcommunities, this time defined by

$$\text{User-based similarity} = |UA \cap UB| / |UA \cup UB|$$

UA: Set of users in sub-community A
UB: Set of users in sub-community B

Unlike the context-based similarity, a user’s difference in status (a fraction that represents his “respect” in his sub-community) in subcommunities A and B is not likely to be attributed to how the user-based similarity was constructed. Similarly, for a user’s difference in activity index between A and B, we are only examining the intersection of users in both subcommunities to begin with when plotting the average difference of activity index in two subcommunities, so the size of this intersection should not play a role and this number will not simply be an artifact of how this user-based similarity was created.

7. Results

For our analysis, we calculated, for every pair of sub-communities, the average of the (difference between the user’s status in sub-community A and sub-community B) over all users in the intersection of A and B. We did a similar calculation for activity index. For the remainder of the pair, we

will refer to these values as the average difference in status and the average difference in activity index, both values that are tied to a certain pair of sub-communities.

7-1. Average Differences in Status and Average Differences in Activity Index Between Subcommunities

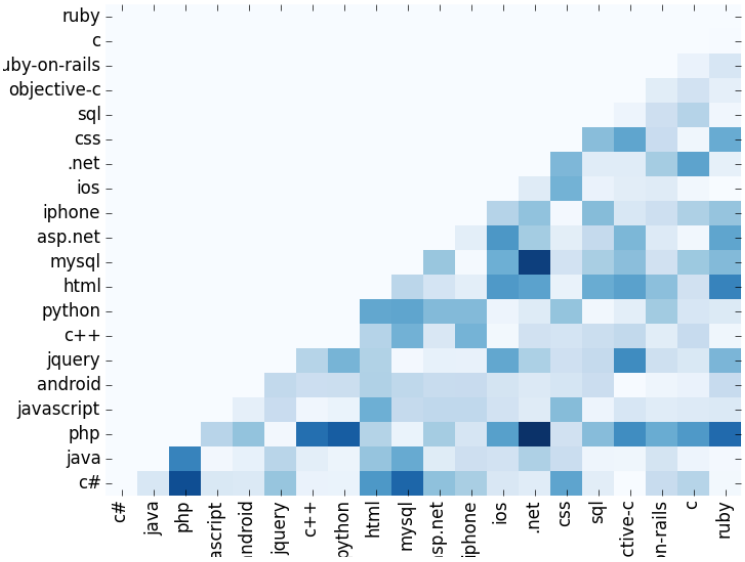

Figure 2. Average differences in activity index between sub-communities

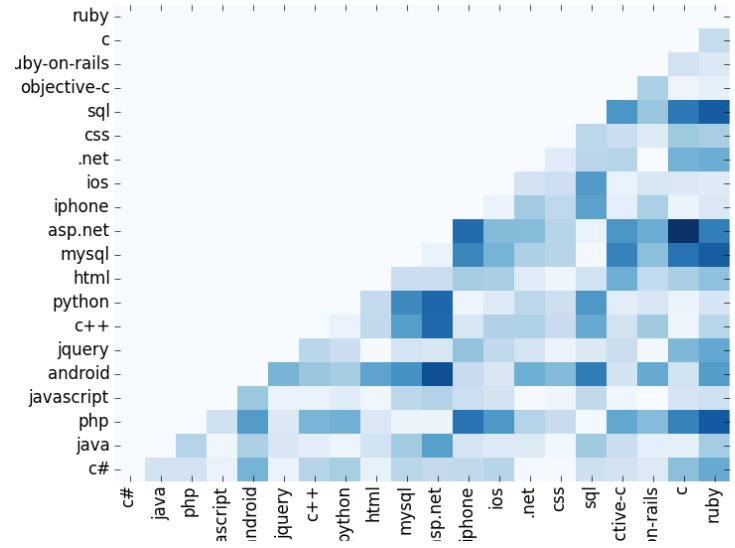

Figure 3. Average difference in status between sub-communities

In each heatmap, the diagonal line is colored with the base color when there the average difference of the respective index is zero. The darker a data point is compared to this base color, the larger the average difference in activity index or status. For example, we can observe that ‘mysql’ and ‘sql’ are very similar to each other in both status and activity index. ‘mysql’ and ‘.net’ are similar in status and varies a lot in activity index. ‘php’ and ‘ruby’ are different in both status and activity index. .Intuitively, these observations make sense in the real world.

7-2. Context-based and User-based Similarities Between Subcommunities

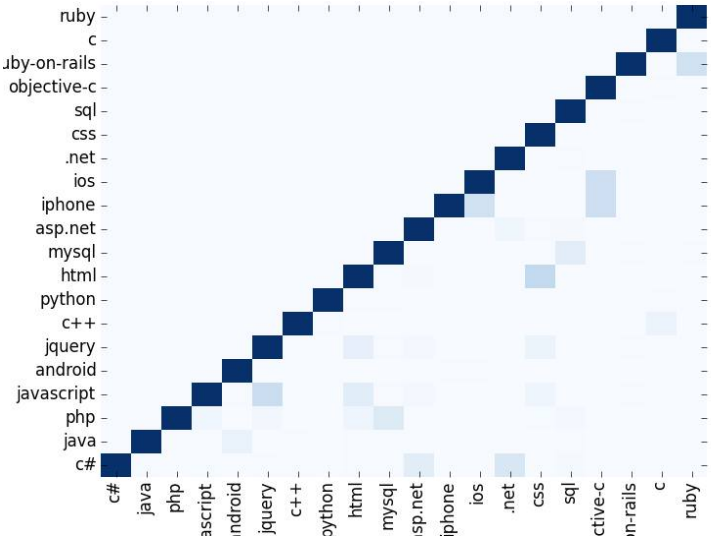

Figure 4. Context-based similarities of subcommunities

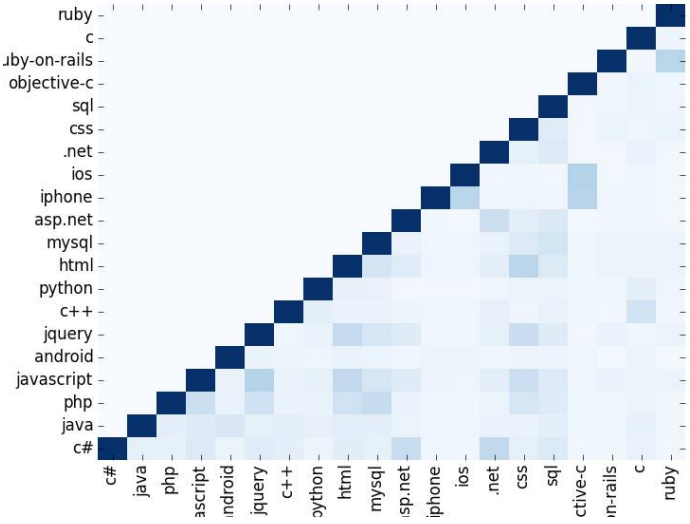

Figure 5. User-based similarities of subcommunities

As we can see from the two graphs above, the context-based and user-based similarities for each pair of sub-communities differ slightly. ‘objective-c’ is similar to ‘ios’ and ‘iphone’ both in terms of context and users, whereas ‘php’ and ‘javascript’ are not similar in context but have a large number of common users. In other words, a user, who is active in the ‘php’ sub-community, is also likely to be active in the ‘javascript’ sub-community. Discovering this served as further confirmation to us that user-based similarities could potentially be a better and more refined metric for similarity than one that is context-based.

7-3 Correlation Between User and Sub-community Characteristics

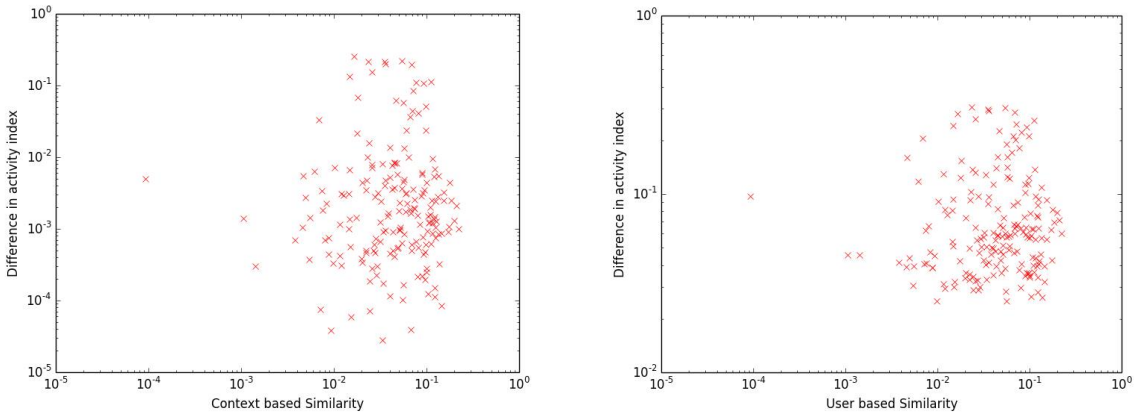

Figure 6. Correlation between average difference in activity and similarities

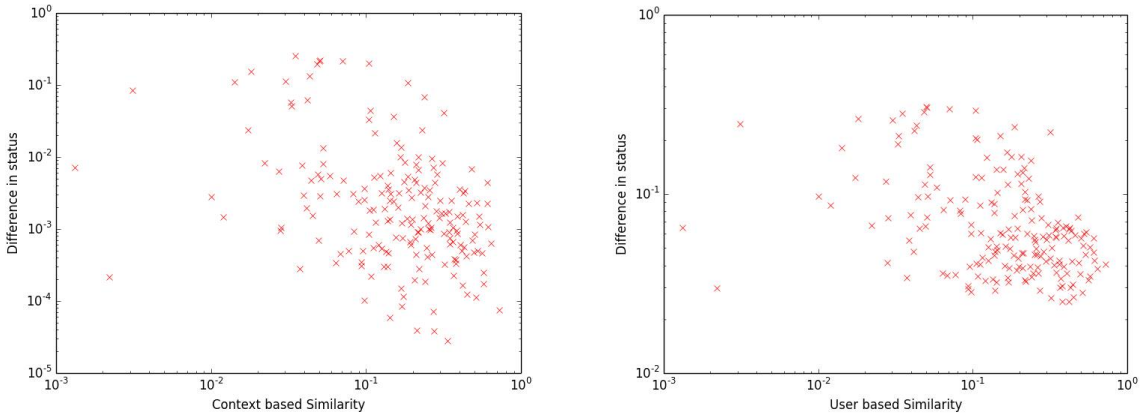

Figure 7. Correlation between average difference in status and similarities

For the above graphs, we took two hundred pairs of the top 20 sub-communities from the above heatmap and plotted 1) the two notions of similarities of the sub-communities against the average

difference in activity index over the users in the intersection, and 2) the two notions of similarities of the sub-communities against the average difference in status over the users in the intersection, leading to four plots total where each data point is tied to a pair of sub-communities.

Negative correlations are noticeable, more strongly when examining the similarities plotted against the average difference of status. As mentioned before, we acknowledge the limitation that the negative correlation between an average difference in status and context-based similarity may be explained as an artifact of how we defined those terms; however, the negative correlation between the average difference in status and the user-based similarity on the other hand is still somewhat interesting. What this means is that the less users there are that are involved in two topics, the more likely those users will be perceived differently in those two topics. For example, users in both jQuery and Java sub-communities, two sub-communities with a low user-based similarity score, are more likely to be perceived more differently in each than the status difference seen for users in both the Php and Javascript sub-communities. This could suggest that status does not necessarily carry over smoothly when a user posts about a "dissimilar" topic that is not usually associated with the user's area of expertise.

8. Conclusion & Future Work

In this paper, we have come up with definitions of sub-communities, status and activity index based on features of StackOverflow: i.e. number of questions and number of answers. We also looked into different kind of ways to define similarities between two sub-communities- one based on the intersection of questions posted, and one based on the intersection of users that post in each topic. We made best efforts to ensure that in coming up with different metrics for behavior that conclusions drawn would not simply be due to artifacts of construction. We acknowledged the limitations of a context-based similarity compared to our notions of status and activity index, but still used it in case valid conclusions could still be draw for two sub-communities with a low context-based similarity.

Once terms were defined, we were ready to look for potential trends. We noticed a negative correlation between similarity and difference in status, as well as between similarity and difference in activity index. We noticed from our data that a user's status in sub-community B is more likely to differ from his status in sub-community A the lower the user-based similarity of the sub-communities were. While the negative correlation discovered may not necessarily be all that is needed for such a bold conclusion (due to potential weaknesses in how we defined our terms along with other flaws), it serves as a step in the right direction and rationalizes further study in looking at different behaviors for users in different sub-communities. For future work, we could look into other definitions for status and activity index. Also, we feel that we should have done more preprocessing of the data, and it might have given us better results. By filtering out less active users and posts that receive fewer view counts, we can focus our analysis on the relatively mainstream posts and users. With the current infrastructure, we can adopt machine learning algorithms to predict whether a given user will act similarly or differently in two given

sub-communities.

References

1. Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2012. Discovering value from community activity on focused question answering sites: a case study of stack overflow. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

2. Jun Zhang, Mark S. Ackerman, and Lada Adamic. 2007. Expertise networks in online communities: structure and algorithms. In Proceedings of the 16th International conference on World Wide Web.

3. Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2012. Effects of user similarity in social media. In Proceedings of the 5th ACM international conference on Web search and data mining