PACIS 2018 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

6-26-2018

Road to Success: How Newcomers Gain Reputation in Open Source Community

Shahab Bayati
The University of Auckland, shbayati@yahoo.com

K. Dharini Amitha Peiris
The University of Auckland, a.peiris@auckland.ac.nz

Road to Success: How Newcomers Gain Reputation in the Open Source Community

Completed Research Paper

Shahabedin Bayati
University of Auckland
New Zealand
s.bayati@auckland.ac.nz

K. Dharini Amitha Peiris
University of Auckland
New Zealand
a.peiris@auckland.ac.nz

Abstract

Open source software (OSS) relies on volunteers' and newcomers' contributions to survive. Open source development is a socio-technical process which forces newcomers to have both technical and social skills for their success. Many newcomers face social and technical challenges while they contribute to OSS projects. Such challenges may lead to some discontinuing their contributions. This paper investigates the newcomers' success and the reputation achieved in a social coding platform, by analyzing initial activities carried out by them in an open source community. By applying exploratory data analysis on GitHub data, we found the important role of social skills in newcomers' reputation across the OSS community. Result of this exploration shows the concurrent contribution to multiple projects increases the chance of popularity in this social coding environment. Also, we have analyzed the main project attributes that successful newcomers contributed to, during the early stages of their involvement. The research outcomes of this study are useful for newcomers to make better decisions and for project leaders to be more supportive.

Keywords: Newcomers, Open Source Software, Reputation Analysis, Exploratory Data Analysis, Social Coding

Introduction

Open source projects rely on volunteers' commitment to the projects (Schilling 2012; Von Krogh et al. 2003). To have a successful and active community, project organizers need to have more contributors ranging from newcomers to experienced developers. However, many of these newcomers lose their motivations and do not continue their jobs in OSS projects due to various socio-technical barriers (Steinmacher et al. 2015; Steinmacher et al. 2014). Volunteers' motivations in OSS projects are studied widely in the context of information systems (Moqri et al. 2015; Roberts et al. 2006). Gaining reputation inside the community is mentioned as a motivational factor for OSS contributors. In this study, we explored newcomers' activities and their reputation in OSS community as a motivational factor to find what makes a newcomer successful.

The main motivations of open source contributors are discussed in the literature (Crowston et al. 2006; Moqri et al. 2015; Von Krogh and Von Hippel 2006). These motivations are categorized as intrinsic, extrinsic and ideological factors. However, many of OSS newcomers are not successful in their joining process into OSS community due to the lack of technical knowledge about projects architecture, code complexity, coding standards (Steinmacher and Gerosa 2014). Also, many of them cannot find a good task or project to start with. Many of newcomers face social problems in their interactions with other project members. These issues are investigated in past OSS literature. To have a better understanding of newcomers situations in OSS community we took the socio-technical lens to check their initial project selection and behaviour. This analysis, will help newcomers to make a better decision about their behaviour and provide recommendations for project organizers to make the project atmosphere more *"newcomers oriented"*.

To support newcomers in their joining process, some large OSS projects and organization such as OpenHatch (2017) have provided programmes to help newcomers. Mentorship programmes are a common approach in OSS to guide newcomers (Panichella 2015). Some tools and recommender systems help newcomers to find a better a task to start with (Wang and Sarma 2011), mentors to get supportive guides (Steinmacher et al. 2012) and documents and code to get familiar with project structure (Steinmacher and Gerosa 2014). However, the main concept which is not studied well is what makes a newcomer successful and what the successful ones have done during their joining process. This study contributes to OSS literature by focusing on newcomers' success factors by comparing their initial activities and project selection.

The main research questions of this study are "What are the factors of being successful newcomers in open source community?" and "Which project attributes have more effect on newcomers' success?" To answer this question we captured data on newcomers in a social coding platform and analysed their initial activities and their reputation inside the OSS community. We used GitHub data as the largest community of open source contributors with more than 65 million projects and 25 million members. We have applied the organizational socialization theory for newcomers' adjustment which reveals the effect of newcomers' personal characteristics, initial activities and project structure as the effective factors on their adjustment and outcome in an organization (Bauer and Erdogan 2011). We argued that OSS project operates as an online organization and measured its attributes to know the project environment. Also, we considered the newcomers' reputation as a satisfactory result of the commitment to OSS project. Through the lens of self-determination and social capital theories, newcomers are willing to get a good reputation in online communities (Tsai and Pai 2014). This reputation will increase their commitment in the community. Our study is founded on these theories.

Compared with other studies that discussed newcomers barriers and problems in joining OSS projects, this study helps newcomers to have a better understanding of how to select the right project and how to contribute to the community to be successful. It will also help project owners to know the main features of successful newcomers to align their strategies with the needed requirements to hire suitable candidates. They can also know the main collaborative barriers newcomers' contributions will have and plan to optimize combining their scripts. We have shown how successful newcomers are different from others, which can help others to follow similar pathways to achieve a good reputation.

This paper is structured as follows, in the next section we provide related works. Data preparation and analysis is discussed in section 3 and 4. Discussions are presented in section 5. The study is concluded in the last section.

Literature Review

The motivations of open source developers for contributing in the OSS community is a factor studied in past information systems literature (Crowston et al. 2006; Von Krogh and Von Hippel 2006). Intrinsic motivators (driven by internal rewards), extrinsic motivators (driven by external rewards), community motivators (driven by service to community) and ideology motivators (driven by one's views) are the main motivators of developers to join OSS projects (Krishnamurthy et al. 2014; Moqri et al. 2015; Roberts et al. 2006). The role of volunteers and newcomers in the success of OSS projects (Von Krogh et al. 2003), and OSS success factors and measures (Grewal et al. 2006; Subramaniam et al. 2009), are well researched in the past. How newcomers found a project across the millions of available OSS projects in online repositories, however requires more investigation. Previous social ties with core members are considered as a factor of project selection (Hahn et al. 2008). Reputation analysis with socio-technical perspective is mentioned as a project selection approach (Dabbish et al. 2013). It is discussed in the literature of social coding that project members and their expertise can be evaluated through their socio-technical activities (Tsay et al. 2014). Therefore, it can be argued that socio-technical activities of developers can affect their reputation. The licence type is also considered as an effective factor for project selection (van Osch et al. 2011). These few studies only focused on general project selection features. However, to the best of our knowledge, there is no published study on how newcomers in OSS community evaluate and select a project and which project features improve the chance of newcomers' success. Generally, for a newcomer, it is more difficult to find and select a suitable project as there is a lack of knowledge and expertise in the new environment.

The important role of newcomers and their main challenges and barriers in the OSS community is studied in the literature with various perspectives (Steinmacher et al. 2015; Steinmacher et al. 2014; Von Krogh et al. 2003). A comprehensive literature review of these barriers are explored in past literature (Steinmacher et al. 2015).

Technical problems, out-dated documents, lack of knowledge, and social immaturity are considered as the main issues for newcomers in OSS community. Some studies designed artefacts to help a newcomer to find a task, source code or mentors to have a successful joining process (Cubranic et al. 2005; Malheiros et al. 2012). Newcomers' performance is related to socialization experiences they got in mentorship programmes and with community members (Carillo et al. 2017). The initial behaviours of newcomers have an impact on their later success (Zhou and Mockus 2015). This study is different from other studies in the field of newcomers in OSS as it investigates the newcomers' characteristics and initial activities in open source community to see what will affect their future success. It also finds the main project factors, which affect the success of OSS newcomers and it shows the factors that differentiates popular newcomers from others through a socio-technical lens.

Developers' success and reputation in the OSS community were analysed in different studies. Developers' experience and their quality of coding are realized as effective factors on their reputation in Ohloh (OpenHub) by considering peer evaluations (Kudos) as reputation measurement (Cai and Zhu 2016). Developers' reputation is measured by the centrality of a developer in OSS social network and its effect has been investigated on the projects' success (Bosu and Carver 2014). The effect of developers' homophily in terms of organization, country, and programming language is measured through Ohloh Kudos as reputation measures in OSS (Hu et al. 2012). Few studies on GitHub social coding platform look for contributors' reputation in terms of the number of followers. Moqri et al. argued the number of followers over time affect developers' commitment in GitHub (Moqri et al. 2015). The important role of highly followed developers (rock-stars) is mentioned in (Blincoe et al. 2016). They suggested that other contributors make the decision based on what the highly reputable contributors do (Blincoe et al. 2016). In this study, we have used the number of followers as a measure of reputation which is borrowed from the literature of social coding and applied it in the context of newcomers' reputation analysis through their socio-technical behaviour.

Data Collection and Pre-processing

We focused on GitHub as a social coding environment where OSS developers can collaborate and contribute to the OSS community. GitHub provides a REST API for public access to projects histories, developers' profiles and any socio-technical collaboration in this platform across OSS projects. GhTorrent (Gousios 2013) is a public archive of GitHub data which uses big data techniques to map various types of event in the form of relational data dumps. We used the MySQL data dump for March 2016 this data dump contains data since 2008.

To collect newcomer related data we have developed an application in C#.Net to query over GhTorrent data dump. We assume anyone registered to the GitHub as a newcomer and we selected all the activity of them for a period of a year. To find successful newcomers, we have counted the number of followers they gained in a year. GitHub developer will follow each other when they want to be aware of their activities (Blincoe et al. 2016; Moqri et al. 2015). *"Following"* is a social activity which is unidirectional. We used following as a reputation indicator. A developer with more followers are considered as more successful in this study.

We analysed community newcomers on different days across a year and we found that the number of followers gained by newcomers over a year follow the *power-law distribution*. Almost 70% of newcomers cannot gain any followers over a year, 15% only have one follower and only 2% have more than 10 followers. Figure 1 shows this distribution for a day and a month.

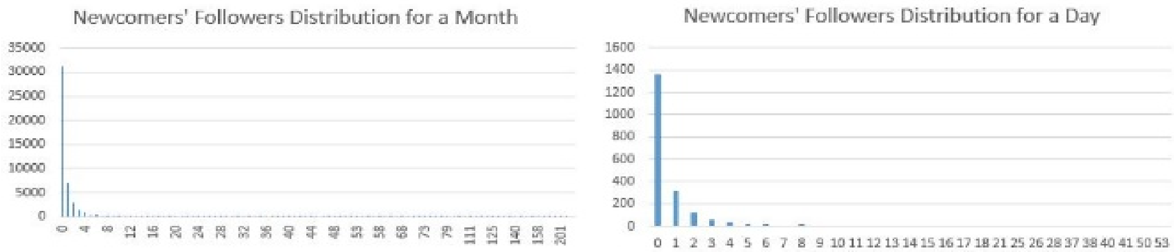

Figure 1. Newcomers' followers' distribution after a year of joining in GitHub for different periods

Based on our exploratory analysis on newcomers' reputation we categorized them into four different classes. Table 1 describes each category. Then for each set of these newcomers, we summarized their activities in all the projects they have contributed to, to see the main differences among them. As the aim of this study is supporting newcomers in their initiative activities while they join other projects we only focused on those who participated in the projects, which are not owned by the developers, and compared their activity rates.

Table 1. Newcomers Categorization

Super Reputable (SR)	Top 1% (14 or more Followers in a year)
High Reputable (HR)	Top 1-10% (3-13 Followers in a year)
Low Reputable (LR)	Top 10-30% (1-2 Followers in a year)
Not Reputable (NR)	Others (No Followers in a year)

In our analysis we have listed all the projects these newcomers have any type of collaboration, including technical (Code Commits, Pull Requests, Issue reporting and issue resolution) or social collaboration, which consist of any conversation and comments they made on others technical events. Then we analysed these projects socio-technical activities and compared them based on different categories shown in Table 1.

Sample set of Newcomers

We have sampled the list of newcomers in different categories (Table 1). Our sample contains 473 newcomers which are categorized as 61 from SR category, 93 from HR, 133 from LR and 186 from NR. Table 2 briefly illustrates each category. It shows that SR newcomers contributed in more projects than others. Also, this group follows more GitHub members than others. We have already categorized them by the number of followers but SR has more followees. The interesting point is that SR has less commitment than other groups which is aligned with the result of (Blincoe et al. 2016) which depicted commitments are not an effective factor in developer popularity. To quantify newcomers' technical commitments to project we have counted the number of code commits they have submitted to the repositories. Also, it can be understood that newcomers in SR and HR are more interested in coordination activities such as reviewing pull requests and commenting on them. In GitHub. any changes to the main repository is going through submitting a pull request by developers and merging and accepting the request by the coordinators and admins. This table (Table 2) focused on newcomers and their activities. Based on the outcome of this table and with a socio-technical lens, a newcomer needs to be more active on the social side rather than in technical commitment, to be popular in the OSS community.

Table 2. Sample data summary

	SR	HR	LR	NR
Newcomers	61	93	133	186
Total Contributed Projects	281	336	50	85
Project/Newcomer	4.61	3.61	0.38	0.46
Avg Followers	27.61	6.97	1.37	0
Avg Followee	22.70	8.77	2.52	0.43
Avg Commits	5.83	6.87	29.92	6.34
Avg Pull Request Comment	0.06	0.59	0.04	0.02

Figure 2 shows how different classes of newcomers are distributed around the world. It can be figured out that U.S has the largest population of newcomers in all four categories with almost the same number of members. Table 3 gives more details on country level separation for each category 45% of the Newcomers' data set used in this study prepared their geo-located data. It also consisted 213 records of countries that the newcomers are from.

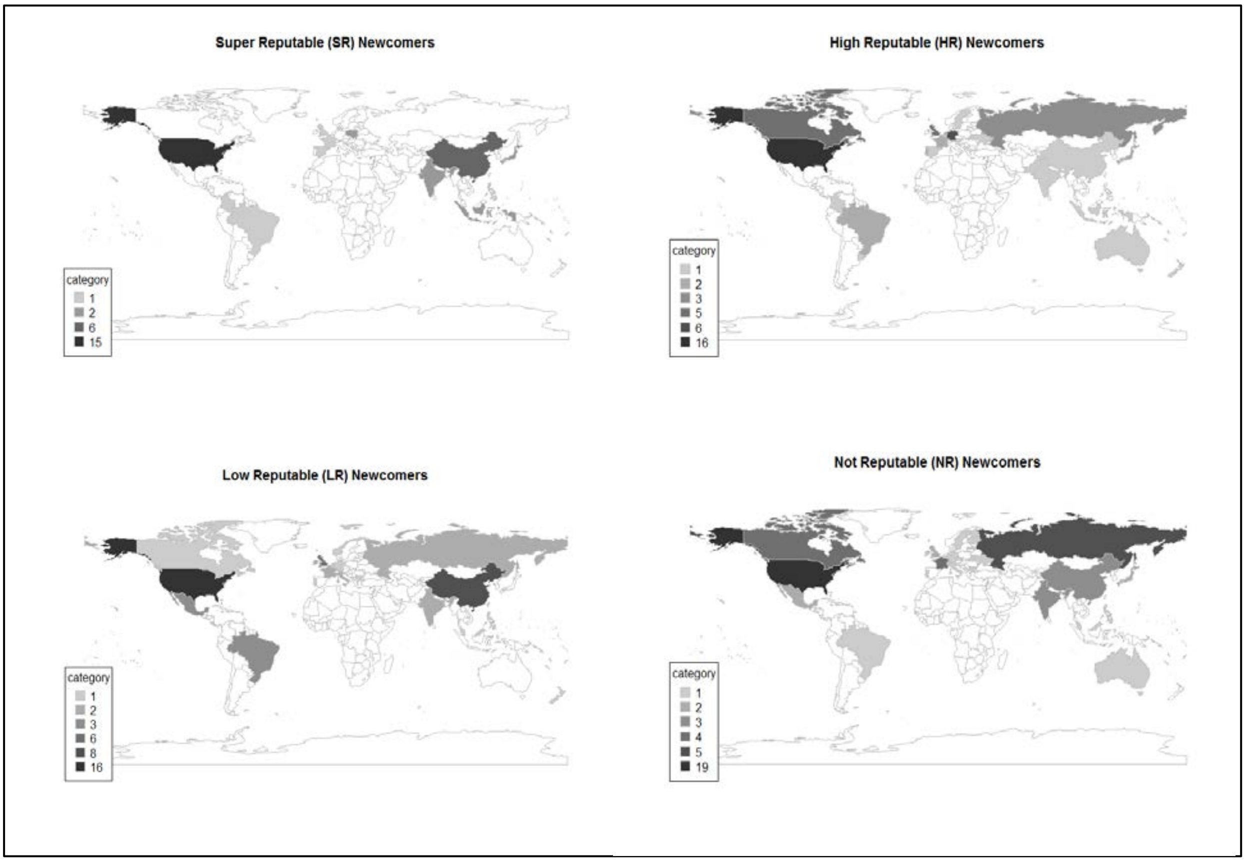

Figure 2. Newcomers' distribution around the world in the sample dataset (SR, HR, LR, and NR)

Table 3. Top countries in the sample dataset based on each category

	US	**CN**	**GB**	**CA**	**RU**	**FR**	**DE**	**IN**	**BR**	**JP**	**MX**	**PT**	**BE**	**ID**	**IT**	**NL**	**PH**
SR	15	6	1			1		2	1	2				2			1
HR	16	1	5	5	3	2	6	1	2	2		2	1	1		2	
LR	19	8	6	1	2	2	1	2	3		3	1	1		2	1	1
NR	19	3	2	4	5	4	1	3	1	2	2	1	1		1		1
Total	66	18	14	10	10	9	8	8	7	6	5	4	3	3	3	3	3

Newcomers of the SR category have more chance to be hired by the organization as it is revealed in table 4. This chance is reduced when we go to the bottom categories. This may prove that by being employed by an organization a newcomer may have more chance to be seen by others. There is only a newcomer who is employed by 4 different organizations and belong to SR. 30% SR newcomers are employed by organizations. Only 2% of LR members belong to one organization.

Table 4. Organizational Employment

Organization Employment Count	**0**	**1**	**2**	**3**	**4**	**Total**
SR	43	12	2	3	1	61
HR	69	20	3			92
LR	130	3				133
NR	181	5				186

Newcomers contributed to different types of projects and table 5 summarizes the differences in project characteristics across our four groups of newcomers. A total of 752 records of project information are available in our dataset. This data is gathered for 3 years from the joining date of newcomers in GitHub. Finding the relationship among projects and predefined categories of newcomers is not an easy task. In the next section, we applied machine learning techniques to ascertain the most important factors that can lead to better project selection by newcomers.

Table 5. Contributed projects

	SR	**HR**	**LR**	**NR**
Avg Commits	744.1	651.7	336.3	870.3
Avg Watchers	456.73	272.59	1007.42	375.99
Avg Issues	184.86	259.78	222.04	303.84
Avg Commit Comments	1.97	4.67	1.88	3.08
Avg Committers	29.61	64.46	18.24	27.2
Avg Member Committers	16.49	1.27	1.86	1.95
Avg Issue Comment	614.02	902.92	648.86	747.71
Avg Pull Request Comment	50.12	33.72	16.7	58.17
Avg Age	241.20	182.80	348.66	338.26

In our dataset, we have projects in 33 different programming languages. As this number of languages make the process of decision making difficult, we have categorized them based on the popularity of languages in GitHub. Table 6. Shows this categorization. However, a total of 86 projects do not provide the main programming languages used in GitHub (752-666= 82).

Table 6. Programming Language Categorization

	Languages	SR	HR	LR	NR	SUM
LangCat_1	Java, JavaScript, Python, Ruby	131	170	24	55	380
LangCat_2	C, C++, C#, PhP, CoffeeScript, Shell, Scala, Viml, Objective-C, Perl	80	99	19	22	220
LangCat_3	CSS, HTML, Puppet, Go, R, Clojure, Groovy, Assembly, TeX, TypeScript, Matlab	35	14	5	4	58
LangCat_4	MakeFile, AwK, BitBake, Dart, Lua, Nemerle, PLpgSQL	1	5	1	1	8
	SUM	247	288	49	82	666

Data Analysis

In this section, the main data analysis part is conducted on the captured dataset of newcomers in GitHub. We have applied mining software repositories (MSR) techniques on our data to find out the main differences among different classes of newcomers and their contributed projects. Firstly, we have analysed what are the most important features defining newcomers' reputation in social coding environment. For this purpose, we have applied random forest technique (Ghasemkhani et al. 2015).

Figure 3 shows the result of random forest analysis. There are two different diagrams in this figure.

- Mean decrease accuracy reveals that how worse the model performs in terms of accuracy if we do not include the variable.
- Mean decrease Gini is a mathematical model to show how the purity of leafs decreases in decision trees in the absence of the variable.

Results show the importance of Reported Issues, Commits, Pull Requests and Activity Count in defining characteristics of newcomers to be set in one of the categories. Table 7 lists the Newcomers' activities.

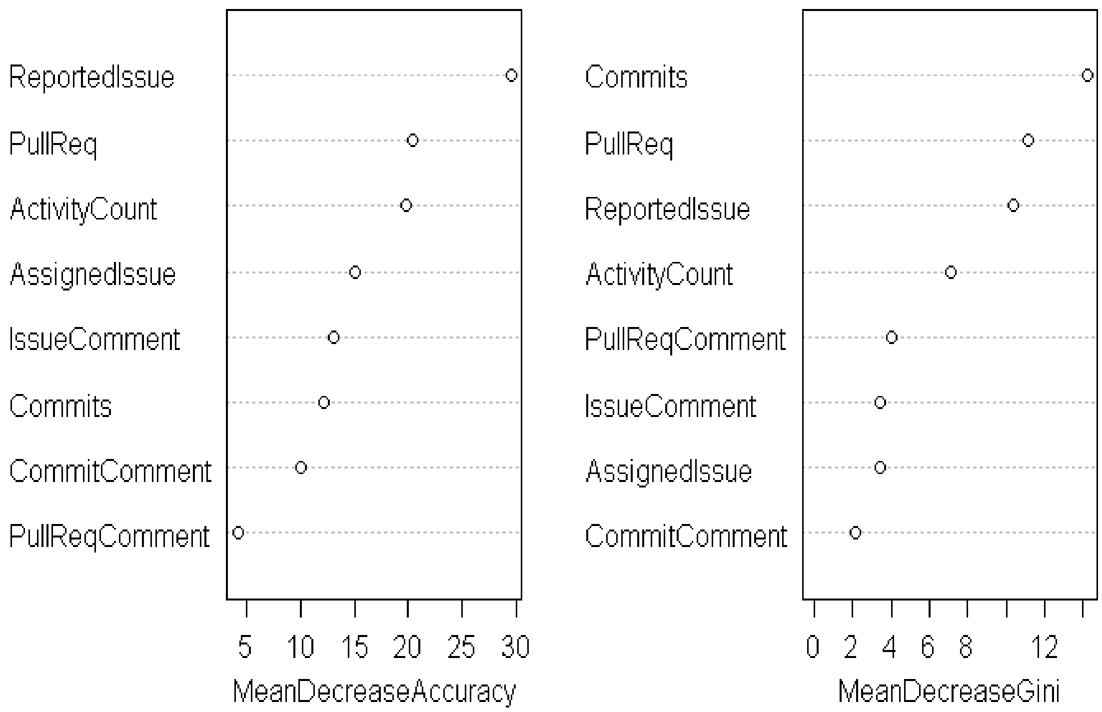

Figure 3. Main influential newcomers' activities

Table 7. Newcomers' socio-technical activities features list

Attributes	Description
ReportedIssue	Number of Issues reported by the newcomer
PullReq	Number of Pull Requests submitted by the newcomer
ActivityCount	Number of Different type of activities done by the newcomer range from (1-7)
AssignedIssue	Number of Issues assigned to the newcomer
IssueComments	Number of Comments posted by the newcomer on issues
CommitComment	Number of comments posted by the newcomer on submitted commits
Commits	Number of commits done by the newcomer
PullReqComments	Number of Comments posted by the newcomer on submitted Pull Requests

We captured a list of project characteristics that newcomers contributed to, during their initial period in GitHub. For each selected project we captured 3-years of data. The list of attribute abbreviations and their descriptions are listed in Table 8.

Our data mining process is started with a feature selection process by applying the Random Forest BI technique. Then we continue the process using an exploratory analysis through visualization of the main factors. Lastly, we used a Correlation Matrix and Regression Trees to provide a decision plan for newcomers, to use in their project selection, in order to have a better chance at being successful. We have implemented all these steps using the R packages, following CRISP-DM Methodology (Wirth and Hipp 2000).

Table 8. Project attributes

Attributes	Description
Forks	Number of times project forked
Members	Number of Members added to the project
Commits	Number of Commits applied to the project
Issues	Number of reported issues
Watchers	Number of users watched the project
PullReq	Number of submitted Pull Requests to the project
CommitCmnt	Number of comments on project commits
PRCmnt	Number of comments on Pull Requests
PRIssueCmnt	Number of Comments on issues related to Pull Requests
IssueCmnt	Number of Comments on reported issues
Committers	Number of project committers
MemCommitters	Number of project committers who are member
Age	Number of days passed after project creation
Lang	Programming Language category

Projects' Feature Ranking

We applied the Random Forest technique to our dataset to find the most important factors among the list of factors mentioned in Table 8. Figure 4 provides the result of feature ranking. By comparing two ranking charts we have selected the top 7 features (MemCommitters, Age, Members, Committers, Commits, Watchers, and Lang). Figure 3 illustrates the effect of omitting each feature on the accuracy of decision making (Ghasemkhani et al. 2015).

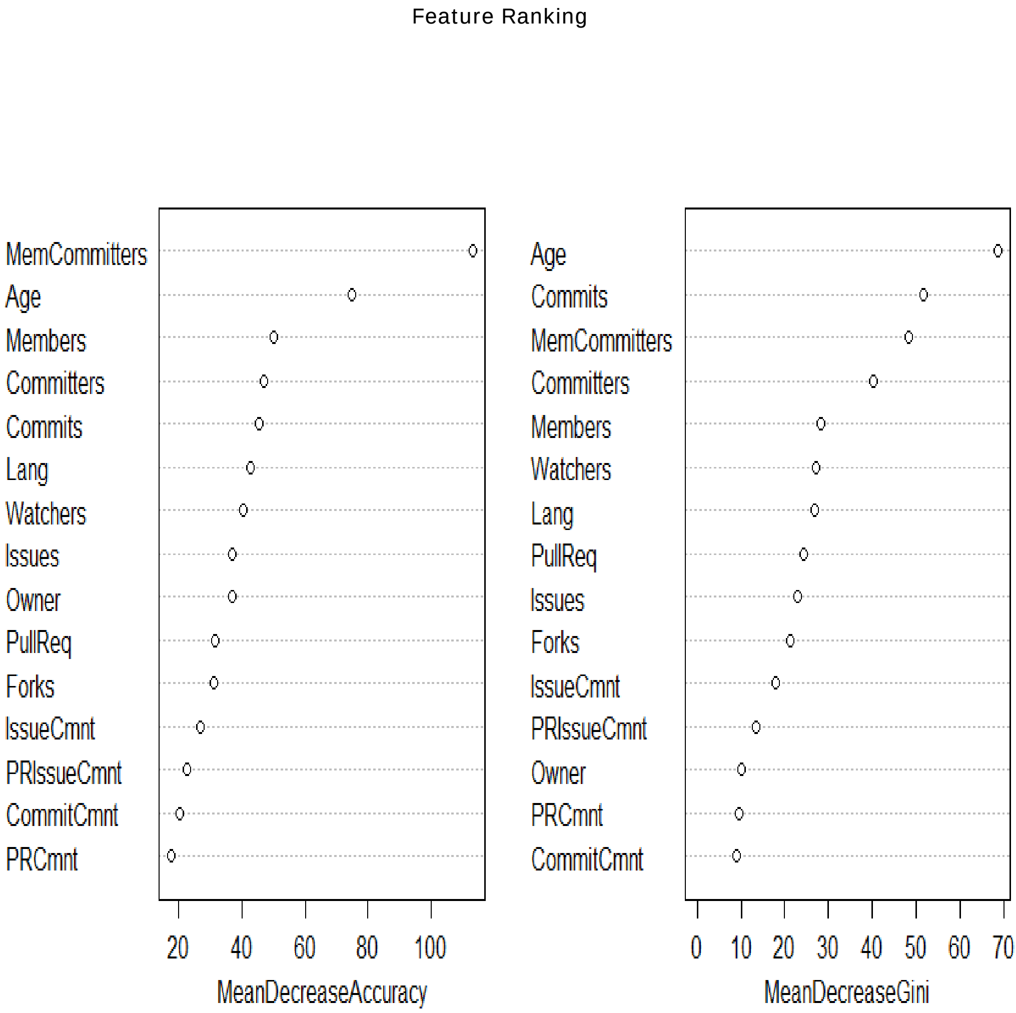

Figure 4. Feature ranking by Random Forest.

Exploratory Visualization

In this part, we visualized the effect of each selected factor on the popularity class of newcomers. Boxplots in Figure 5, illustrates the relationships of each class of newcomers, along with the main factors. In this analysis we could see that joining old projects is not a good decision as the age range for NR group is higher than others. Age represents the number of days passed from the project creation date. However, joining projects with a higher rate of member committers is an effective technique. Member committers are the portion of project members who are participating in technical coding by adding and modifying the current codebase. Also, more technical commitment is not always a factor of popularity. An interesting result which requires more in-depth analysis is that popular projects are not a good point to start one's career with. Newcomers in NR and LR are joined more popular projects rather than two other groups. In terms of project size, it can be drawn from this result that bigger projects have less chance for newcomers to get reputation. Project size is measured by the number of members of a project.

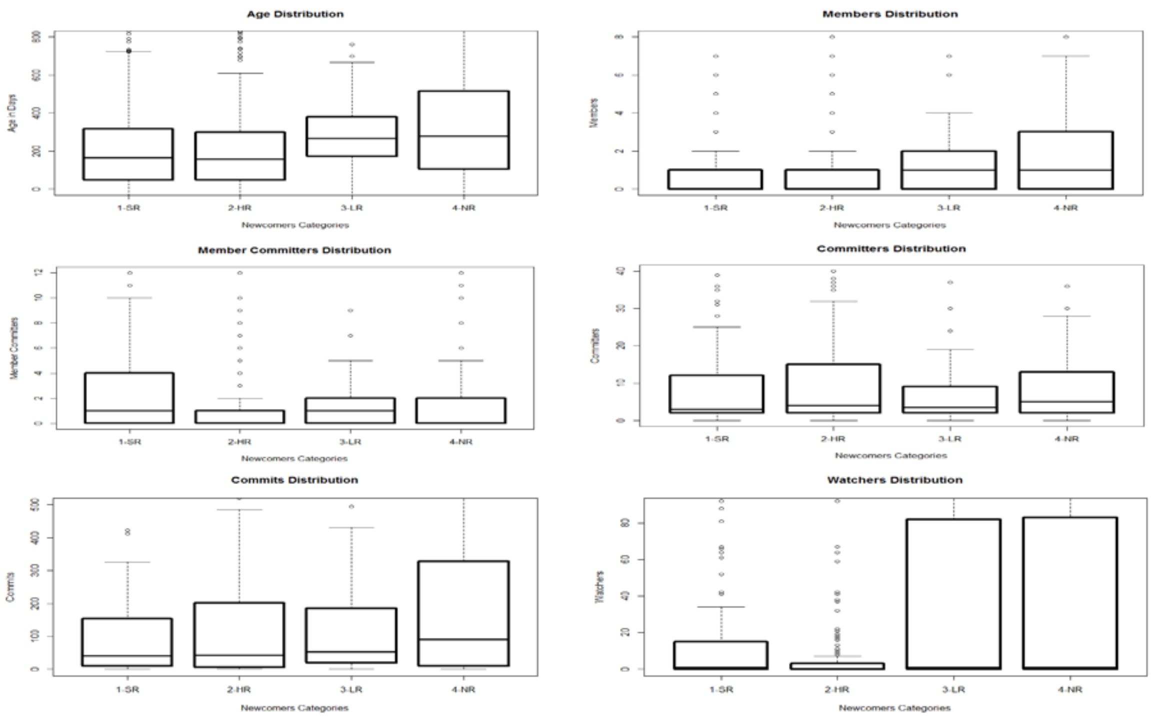

Figure 5. Exploratory visualization of main factors effect on project distribution (Age, Members, MemCommiters, Committers, Commits, and Watchers). Boxes left to right (SR, HR, LR, and NR)

It was also noticed that there was no high correlation among main factors (the correlation matrix is provided in figure 6 with nothing over .6), which validates our feature selection process.

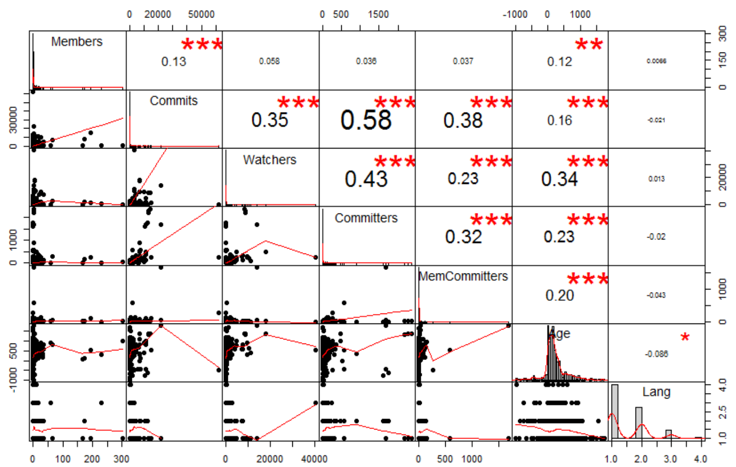

Figure 6. Correlation Matrix and Factors Distributions

Discussion

This study focused on the OSS newcomers and investigates their social identity through the lens of reputation analysis by mining historical data of GitHub newcomers. The analysis shows not only technical commitment that can help a newcomer to gain popularity but also, more importantly, they need to be more social. They need to participate in reviewing process and follow other members. It is recommended to contribute just not to one project. More projects they contribute to, the more members they will know. Such newcomers with the possibility of being hired by organizations have more chance to be seen. Also, we have mined the projects newcomers contributed to, and found that items like project age and committers from project core members, can help newcomers in selecting a project, in order to better manage their reputation. We found that contributing to popular projects did not have a positive effect on newcomers' reputation.

To the best of our knowledge, this is the first study that looks for newcomers' contribution in GitHub to see which project types and activity can make them popular. However, similar to most studies it suffers from some limitations. Focusing on the limited number of newcomers and projects is the main issue of this study although we have categorized our data based on a large sample distribution, it can be improved by more data environments. We have counted all the commits in with same value, however in the real world commits are different. Similar results were seen for the following. Having a reputable follower is different from the general member but we consider all the same value. Longitudinal data of developer activities over time can give a better understanding rather than the case we have mentioned. In addition, this study only focused on the followers' count as a factor of success which is biased on the social side of the social coding environment. However, other metrics such as sustained contribution and expertise can be considered as developer success measures in future research. This study is an exploratory study carried out on a big data set of newcomers. Confirmatory studies can be extended on the outcomes of this research.

Conclusion and Future Works

This paper investigates GitHub social coding platforms to find out newcomers' reputation factors. The result of data mining on GitHub newcomers shows the importance of both social skills and parallel project contributions as being very important factors for newcomers. A successful newcomer with technical commitment, should participate in code review and commenting activities as well. Also, we have shown that newcomers who joined old and popular projects have less chance to get popular. Analysing the members and their participation rate in coding activities can help newcomers to select better projects to survive and even thrive. These results are the outcome of applying data mining techniques to cross-sectional data of a small portion of GitHub Newcomers. Using Big Data Analytics techniques can help us to analyse more newcomers' activities with a longitudinal dataset as an extension of this study. Also, we can consider other success factors such as expertise and sustainable commitments in addition to reputation analysis carried out in this study.

References

2017. "Openhatch Is a Non-Profit Dedicated to Matching Prospective Free Software Contributors with Communities, Tools, and Education.", from OpenHatch.Org

Bauer, T. N., and Erdogan, B. 2011. "Organizational Socialization: The Effective Onboarding of New Employees,").

Blincoe, K., Sheoran, J., Goggins, S., Petakovic, E., and Damian, D. 2016. "Understanding the Popular Users: Following, Affiliation Influence and Leadership on Github," Information and Software Technology (70), pp. 30-39.

Bosu, A., and Carver, J. C. 2014. "Impact of Developer Reputation on Code Review Outcomes in Oss Projects: An Empirical Investigation," Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement: ACM, p. 33.

Cai, Y., and Zhu, D. 2016. "Reputation in an Open Source Software Community: Antecedents and Impacts," Decision Support Systems (91), pp. 103-112.

Carillo, K., Huff, S., and Chawner, B. 2017. "What Makes a Good Contributor? Understanding Contributor Behavior within Large Free/Open Source Software Projects–a Socialization Perspective," The Journal of Strategic Information Systems).

Crowston, K., Howison, J., and Annabi, H. 2006. "Information Systems Success in Free and Open Source Software Development: Theory and Measures," Software Process: Improvement and Practice (11:2), pp. 123-148.

Cubranic, D., Murphy, G. C., Singer, J., and Booth, K. S. 2005. "Hipikat: A Project Memory for Software Development," IEEE Transactions on Software Engineering (31:6), pp. 446-465.

Dabbish, L., Stuart, C., Tsay, J., and Herbsleb, J. 2013. "Leveraging Transparency," IEEE software (30:1), pp. 37-43.

Ghasemkhani, H., Reichman, S., and Westerman, G. 2015. "Using Predictive Analytics to Reduce Uncertainty in Enterprise Risk Management,").

Gousios, G. 2013. "The Ghtorent Dataset and Tool Suite," Proceedings of the 10th Working Conference on Mining Software Repositories: IEEE Press, pp. 233-236.

Grewal, R., Lilien, G. L., and Mallapragada, G. 2006. "Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems," Management Science (52:7), pp. 1043-1056.

Hahn, J., Moon, J. Y., and Zhang, C. 2008. "Emergence of New Project Teams from Open Source Software Developer Networks: Impact of Prior Collaboration Ties," Information Systems Research (19:3), pp. 369-391.

Hu, D., Zhao, J. L., and Cheng, J. 2012. "Reputation Management in an Open Source Developer Social Network: An Empirical Study on Determinants of Positive Evaluations," Decision Support Systems (53:3), pp. 526-533.

Krishnamurthy, S., Ou, S., and Tripathi, A. K. 2014. "Acceptance of Monetary Rewards in Open Source Software Development," Research Policy (43:4), pp. 632-644.

Malheiros, Y., Moraes, A., Trindade, C., and Meira, S. 2012. "A Source Code Recommender System to Support Newcomers," Computer Software and Applications Conference (COMPSAC), 2012 IEEE 36th Annual: IEEE, pp. 19-24.

Moqri, M., Qiu, L., Bandyopadhyay, S., and Horowitz, I. 2015. "The Effect of “Following” on Contributions to Open Source Communities,").

Panichella, S. 2015. "Supporting Newcomers in Software Development Projects," Software Maintenance and Evolution (ICSME), 2015 IEEE International Conference on: IEEE, pp. 586-589.

Roberts, J. A., Hann, I.-H., and Slaughter, S. A. 2006. "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects," Management science (52:7), pp. 984-999.

Schilling, A. 2012. "Links to the Source-a Multidimensional View of Social Ties for the Retention of Floss Developers," Proceedings of the 50th annual conference on Computers and People Research: ACM, pp. 103-108.

Steinmacher, I., and Gerosa, M. A. 2014. "How to Support Newcomers Onboarding to Open Source Software Projects," IFIP International Conference on Open Source Systems: Springer, pp. 199-201.

Steinmacher, I., Silva, M. A. G., Gerosa, M. A., and Redmiles, D. F. 2015. "A Systematic Literature Review on the Barriers Faced by Newcomers to Open Source Software Projects," Information and Software Technology (59), pp. 67-85.

Steinmacher, I., Wiese, I. S., Conte, T., Gerosa, M. A., and Redmiles, D. 2014. "The Hard Life of Open Source Software Project Newcomers," Proceedings of the 7th international workshop on cooperative and human aspects of software engineering: ACM, pp. 72-78.

Steinmacher, I., Wiese, I. S., and Gerosa, M. A. 2012. "Recommending Mentors to Software Project Newcomers," Proceedings of the Third International Workshop on Recommendation Systems for Software Engineering: IEEE Press, pp. 63-67.

Subramaniam, C., Sen, R., and Nelson, M. L. 2009. "Determinants of Open Source Software Project Success: A Longitudinal Study," Decision Support Systems (46:2), pp. 576-585.

Tsai, H.-T., and Pai, P. 2014. "Why Do Newcomers Participate in Virtual Communities? An Integration of Self-Determination and Relationship Management Theories," Decision Support Systems (57), pp. 178-187.

Tsay, J., Dabbish, L., and Herbsleb, J. 2014. "Influence of Social and Technical Factors for Evaluating Contribution in Github," Proceedings of the 36th international conference on Software engineering: ACM, pp. 356-366.

van Osch, W., Adelaar, T., and Pith, M. 2011. "So Many Developers, So Many Projects: Toward a Motivation-Based Theory of Project Selection," AMCIS.

Von Krogh, G., Spaeth, S., and Lakhani, K. R. 2003. "Community, Joining, and Specialization in Open Source Software Innovation: A Case Study," Research Policy (32:7), pp. 1217-1241.

Von Krogh, G., and Von Hippel, E. 2006. "The Promise of Research on Open Source Software," Management science (52:7), pp. 975-983.

Wang, J., and Sarma, A. 2011. "Which Bug Should I Fix: Helping New Developers Onboard a New Project," Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering: ACM, pp. 76-79.

Wirth, R., and Hipp, J. 2000. "Crisp-Dm: Towards a Standard Process Model for Data Mining," Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining, pp. 29-39.

Zhou, M., and Mockus, A. 2015. "Who Will Stay in the Floss Community? Modeling Participant's Initial Behavior," IEEE Transactions on Software Engineering (41:1), pp. 82-99.