VANT 149 Research Project Proposal: Amount of Developers Influenced Size of OSS Projects

Kewei Li and Yuhang Zeng

Vantage College, University of British Columbia

Abstract

Open Source Software (OSS) has recently being increasingly popular. OSS are usually hosted on OSS communities such as GitHub, SourceForge etc. GitHub has become one of the biggest OSS community, by September 2017, GitHub had almost 24 million users and 67 million repositories (GitHub, 2017). This study was devoted to model the relationship pattern between the amount of developers and the size of OSS project on GitHub platform, since there exists no recent study focused on this relationship. In this study, a program was developed to create this model by analyzing repositories newly built on GitHub between September 2016 and September 2017. Repositories in this range were grouped into 17 groups by the amount of developers and the average of project size in each group were plotted into the model. This study concluded a quantitate formula describing the relationship between the amount of developers influenced the size of OSS projects. This verification of this relationship can potentially benefit the OSS communities, software companies or personal developers, by giving them a general understanding of how the amount of developers influenced the size of software project.

Amount of Developers Influenced Size of OSS Projects

Introduction

With the development of Internet and computer technology, the demand of application of programs are getting more and more complex. These complex applications also lead program designing problem more and more complex. With this developing difficulty, Open-source software (OSS) had become increasingly popular these days. OSS (can also be called as repositories) is a kind of software "under a special software license that allows users to use, change, and/or improve the software's source code, and to redistribute the software either before or after it has been modified" (Mayhew, 2015). People can contribute towards a same project, and everyone can do what they are good at. The appearance of OSS solved the increasingly developed demand, the complexity coding load may be hard to deal with individually but the workload can be simplified if a number of developers are involved. One of the most famous OSS community is GitHub. GitHub is an online platform which is widely used among all multi-person collaboration needed fields such as teamwork program develop. It stores the projects on its server so that multi people can work together because all the people who work with it can have access and edit. Other features such as access control, bug tracking, feature requests, task management, and wikis are offered for every project. By September 2017, GitHub owned almost 24 million users and 67 million repositories (GitHub, 2017), making it one of the largest source code database in the world. Most of previous research in the field of OSS, has a lot of limitations (will specify in Literature Review section) and no recent study had focused on modeling the relationship between the number of contributor and the size of OSS project. Thus, in order to fill the gap of knowledge, in this study, a program was developed to create this model by analyzing repositories newly built on GitHub between September 2016 and September 2017. A hypothesis had been made before the research starts: with the increase of amount of developers, the size of repositories would increase. This study expects a quantitively formulated model for this relationship.

Literature Review

With the increasingly popular of Open Source Software (OSS), and its public access permit, it had drawn researchers' interest. A series of previous research had been done and give the current study a general idea. A 2016 study is focused on how non-core developer's participation related to the prospect of the project (Krishnamurthy, Jacob, Radhakrishnan & Dogan, 2016). The authors did the following experiment: 2600 archives have been analyzed and 7 factors was controlled (number of users, code complexity, license restrictiveness, age of project, number of bugs, size of project and programming language). This 2016 study has limitations in the number of archives and also controlled too many factors. The current study was designed to take much more amount of sample archive size and will not control those factors, except the age of project.

Another previous study worth mention is a study done by 2006, focused on how motivations of contributors of OSS are being related (Roberts, Hann & Slaughter, 2006). This study analyzed projects under Apache license from 1999 to 2002, by using an original theoretical model with Structural Equation Model (SEM) (Roberts, Hann & Slaughter, 2006). This 2006 study was using data from 1999-2002, which is relatively an old data set which made it a limitation of the study. To avoid the same limitation, in this study, we decided to use new data between 2016 and 2017.

Objective

Quantitively model the relationship of the amount of developers influenced the size of OSS projects.

Methods

Around 600 repositories newly created between September 2016 to September 2017 have been analyzed. The analysis was done with the following process: generate a user tree by using GitHub API and get each users' own repositories and append them all to a list, then download all the repositories, the natural logarithm of file size of each repositories will be regard as the size of repositories, and the number of contributor will be pulled out from the meta data by using GIT command: [shortlog].

In order to model the pattern between the amount of developers and the size of repositories, the amount of developers had been grouped into 17 groups as following: 1~3; 3~6; 6~10; 10~15; 15~21; 21~28; 28~36; 36~45; 45~55; 55~65; 65~75; 75~85; 85~100; 100~130; 130~145; 145~190 and 190+. Since we have not enough time and storage resource to analysis and store the repositories that are too large, and based on the raw data, we found that there is no much need to record a specific amount of developers when the amount of developers are more than 190, since those data are highly separately distribute and are not going to benefit creating a model. The size of repositories used the natural logarithm of file size, since the natural logarithm can respond to skewness of large values and can plot a better, user friendly model. And the average value of each group of repositories will be used in the model.

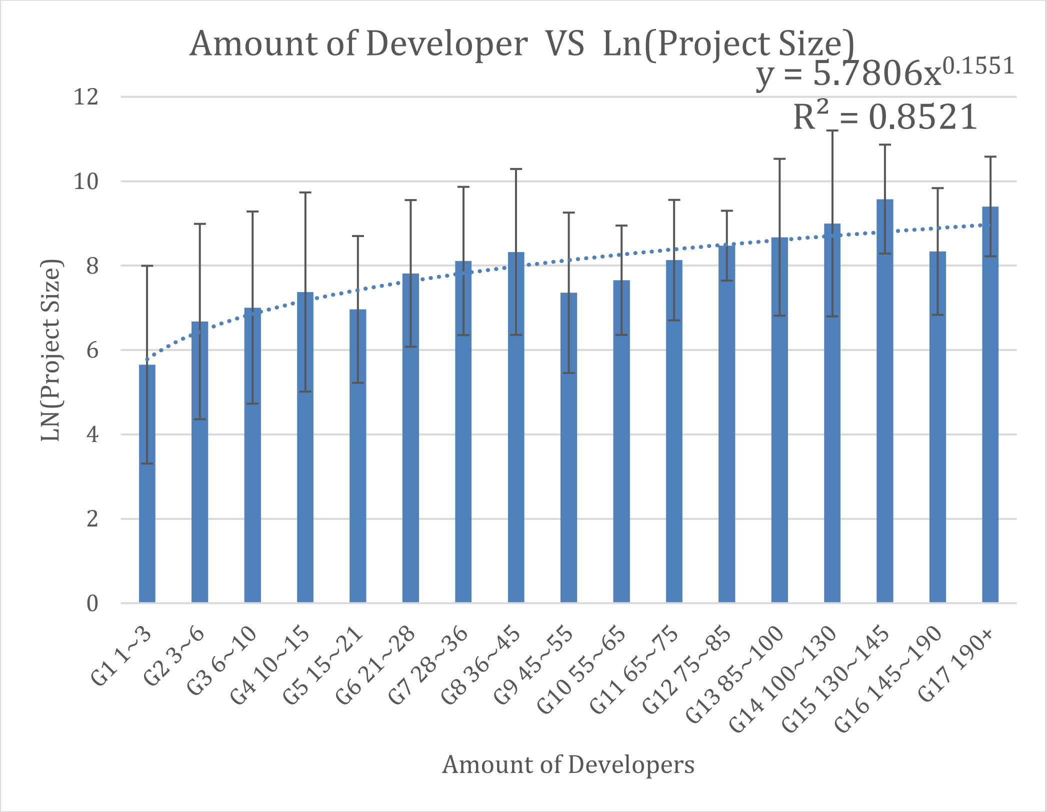
After finishing the modeling, a coefficient between size of project (in terms of kB) and lines of code need to be setup for applications of this model. This coefficient was set as follow:

many style guides for computer programming define the maximum character per line of source code as 80 characters. And considering not all lines in the code can fulfill this maximum limit, the average amount of characters per line can be estimate as 50% of 80 character (80*0.5=40). And 1 kilobyte (kB) = 1024 characters = 1024/40 lines of code = 25.6 lines of code.

Hence, this coefficient could be set as 25.6.

Results and Calculation

A summarized data is attached below, with 17 groups of developer data corresponding to the average of natural logarithm of project sizes and trend line was generated by excel.



The model can be used with the following steps:

1. Estimate the line of code you probably need => (X).

2. Use natural logarithm on the size of project (X/coefficient=X/25.6): Ln(X/25.6) =>(y).
3. Plug (y) into the model: y=5.7806 x^ 0.1551
4. Figure out the (x). which integer parts indicated which group it would fall into among all 17 groups in the model, and the fractional part will indicate the exact place it would at in this indicated group. E.g. if the x=8.3, the integer part indicates the group: 8 (36-45 developers); and the fractional part indicates the accurate number: 36+0.3*(45-36) =38.7. (On average, projects of this size would need around 38 developers).

Discussion

The results indicated that with the increase of amount of developers, the size of repositories increased and by trying match with different kind of trend line, the model fit a power trend line the best (y=5.7806 x^ 0.1551). Which full filled the expectation of this study and also proved the hypothesis.

However, there still exist some limitations. One is the sample choices, according to the raw data which used to generate this model, the model is good at estimate the project size between 788kB and 4500kB or in terms of amount of developers, between 3 to 85 developers. Since the data projects that were too small might be highly dispersed; and some of the large repositories have to be abandoned in this study, due to the limit of time and resource, which makes there exist not enough data of large projects to conclude a solid result. Another limitation is the sample size, due to the limit time for this project, the samples used in this study is

The results can be improved by analyzing more research sample especially more large repositories, which could fill in the leak of large repositories in the current data set of this study. As for future studies, different time range may be used and more samples may be testified to gain other interesting models.

Conclusion

This study found a quantitively formula model of how amount of developers influenced the size of repositories: $y=5.7806 x^{\wedge} 0.1551$. This result proved the hypothesis and fulfilled the expectation of this study.

References

GitHub. (2017). *GitHub Octoverse 2017 | Highlights from the last twelve months*. Retrieved from https://octoverse.github.com/

Krishnamurthy, R., Jacob, V., Radhakrishnan, S., & Dogan, K. (2016). Peripheral developer participation in open source projects: an empirical analysis. *ACM Transactions on Management Information Systems (TMIS), 6*(4), 14.

Mayhew, S. (2015). open source software. In A Dictionary of Geography. Retrieved from http://www.oxfordreference.com/view/10.1093/acref/9780199680856.001.0001/acref-9780199680856-e-4035

Microsoft. (2018). *Choosing the best trendline for your data - Office Support*. Retrieved from https://support.office.com/en-us/article/choosing-the-best-trendline-for-your-data-1bb3c9e7-0280-45b5-9ab0-d0c93161daa8

Roberts, J. A., Hann, I. H., & 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), 984-999.