Mobile Application Development in FLOSS Platform: A Collaborative Network Approach

Qiqi Jiang[1], Chuan-Hoo Tan[2], and Kwok-Kee Wei[2]

City University of Hong Kong, Department of Information Systems
Tat Chee Ave 83, Hong Kong SAR
qjiang6@student.cityu.edu.hk
{ch.tan,isweikk}@cityu.edu.hk

Abstract. We are deluged by wide variety of mobile applications (thereafter terms as mobile apps) to run on our mobile devices such as smartphones and tablets. However, few studies have been conducted to investigate the issue of how mobile applications are developed in academia so far. In order to bridge such gap, we collected data on mobile application development projects from the largest FLOSS (Free and open-source software) platform, i.e., SourceForge, longitudinally. Four key indicators, graph density, reciprocity, modularity, and dyad census asymmetry, which had been drawn from previous social network literatures, were adopted to investigate the collaborative network. We found that the interactive collaboration was especially important for mobile application development process. Developers formed centralized/mutual structure for collaboration instead of several collaborative subdivisions.

Keywords: Open source software, Mobile applications development, social network analysis.

1 Introduction

There is no doubt that we are living in a mobile world. We are deluged by wide variety of mobile applications (thereafter terms as mobile apps) to run on our mobile devices such as smartphones and tablets. Gartner predicted that the mobile apps market would reach 9 Billion USDs in 2012 and the number of mobile app development projects will rise four times more than native PC projects in 2015 [1]. Despite such emergence, to-date in academia, few studies have been conducted to investigate the issue of how mobile applications are developed, particularly in an open innovation environment such as FLOSS. The most relevant study we found is a pilot work discussing the developers' opportunities in the mobile age [2], but this work falls short of empirical verification. In order to address this research scarcity and to gain a better understanding of mobile apps development, we collected data on mobile application development projects from the largest FLOSS (Free and open-source software) platform, i.e., SourceForge, longitudinally. We seek to identify factors influencing the success of these mobile application development projects.

P.L.P. Rau (Ed.): CCD/HCII 2013, Part II, LNCS 8024, pp. 228–236, 2013.

We used the data from FLOSS mobile application development projects for two reasons. First, SourceForge stores complete data archive of each project, which allows us to analyze the projects in great details. Second, the learning curve effect, previously identified as an obstacle for FLOSS development participation in previous literature [3], could be weakened in FLOSS mobile application development projects due to the availability of development tools and instrumental instructions such as iOS[1] and Android[2] development toolkits. The backbones of functionalities of each mobile platform have been well packaged and documented, which could enable a newcomer to be familiarized with the development environment fast.

In accordance with the previous studies on FLOSS [4, 5], we adopted the information systems (IS) success as the key dependent variable in this study, which is also a prevalent research topic in IS and project management arenas. To contextualize the IS success of mobile application development project, we used two dimensions of IS success as measurements, namely, project awareness and project usage popularity. For project awareness, we counted the amounts of page views within each project; for project usage popularity, we counted the number of application downloads of each project. Both measurements have been validated as meaningful indicators of IS success in the previous studies [5, 6]. For the independent variables, we have four variables measuring the collaborative network characteristics and a time-lag dependent variable. The four network related variables are graph density (reflecting whether the network is tight or sparse), modularity (measuring the strength of division of a network into subgroups), dyad census asymmetry (operationalized by the number of non-mutual connected pairs of vertexes), and reciprocity (that is, whether the pairs of vertexes tend to form mutual connection), which were all based on prior literature on social network analysis [7, 8, 9, 10, 11]. Furthermore, we included three control variables, i.e., incremental ratio of number of files released, the incremental ratio of project developers, and the age of project in the analysis model.

We utilized the seemingly unrelated regression model to analyze the data and obtained consistent results involving the two dependent variables. The results indicate that the sparser network (lower graph density) and the network with greater strength to form subgroups (higher modularity) are negatively related the projects' success. Higher tendency of mutual connection between vertexes is of positive relationship with projects' success. Likewise, we also observed that the more non-mutual connected vertexes exist in network, the lower projects' success probability. In addition, the results reveal that the lag downloading or page viewing amounts also has positive relationship with projects' success.

2 Prior Investigations on FLOSS

Studies on FLOSS development have been prevalent in the last two decades in various academic areas, including computer sciences organization studies, and IS.

[1] `https://developer.apple.com/devcenter/ios/index.action`
[2] `http://developer.android.com/sdk/index.html`

For instances, a private-collective model of OSS innovation was presented in the seminal work by Von Hippel and Von Krogh [12]; Shah [13] extended such model and identified several additional key elements, such as need of satisfaction, reciprocity, and enjoyment, by studying two open source software (OSS) development communities. Besides the investigations from the institutional perspective [12, 14, 15], prior studies also utilized the econometric models to understand the dynamics of FLOSS. For instances, Subramaniam et al. [5] analyzed 8,627 projects hosted in SourceForge and measured three dimensions, namely number of developers, number of downloads, and number of files released, to understand the OSS projects' success; Sen et al. [16] analyzed 7,720 projects hosted at SourceForge to uncover how the intrinsic elements, such as whether to accept donation, programming language, or operation systems, influence projects' success. In addition, some studies utilize the notions from SNA (social network analysis) to investigate how social relationships or capital influencing on the project success. For instances, Grewal et al. [17] randomly selected and measured the network embeddedness of 10 OSS projects, and they found the embeddedness of different project roles (developers and leaders) had different influences (number of CVS commits and number of downloads); Hahn et al. [18] measured network ties to uncover whether the prior collaborative experiences affected the tendency of future collaboration; and Singh et al. [19] deployed the internal and external cohesion within the development network to estimate the OSS project success which was measured by the number of Concurrent Versions System (CVS) commits.

While these studies provide great insights, they, like any other studies, suffer from several precincts, such as the limitation of data sample [13, 17], simplex measurement [19], and lack of sense of social collaboration [5, 16]. Furthermore, the OSS development for mobile application did not draw serious attention in the previous literatures although both industries and academia had realized the importance of application on mobile platform. All of these antecedents motivate us to shift the attention from gaining general insights on overall OSS development to specific understandings on OSS mobile application development.

3 Collaboration from Network Approach

The notions from SNA are introduced to investigate the issues of collaboration for some time. For instances, Singh et al. [19] used internal cohesion, direct and indirect ties, and external cohesion to estimate the number of CVS commits by examining 2,378 OSS project hosted in SourceForge; Cocaldi et al. [20] used rate of network changes and degree centrality to depict the intrinsic structure of bug-fixing network of an OSS project; Singh [21] used cluster coefficient ratio and path length ratio within a network to predict the success of an OSS project. Due to the page limit, we did not review all the studies utilizing SNA to investigate the OSS collaboration here. After a comprehensive extent of literature review of SNA, we identified four most representative SNA indictors, namely (1) graph density, (2) reciprocity, (3) modularity, and (4) dyad census asymmetry, to evaluate the collaboration networks of the mobile OSS projects hosted in Sourceforge.

Graph density denotes the number of edges within a network [10]. In other words, the graph density determines whether the network is tight or sparse, and the dense graph indicates the number of edges is close to the maximal number of edges. In the context of collaboration, the denser graph indicates the higher extent of mutual cooperation.

Reciprocity measures the tendency of vertex pairs to form mutual linkage between each other [11]. Such measurement can be used to estimate the likelihood of bidirectional connection between vertexes. Hence, the higher value of reciprocity indicates the stronger bidirectional connection within such network. In our context, the reciprocity can be used to measure flatness of the collaborative atmosphere. More specifically, the project leaders assigned the tasks of developers and the project leaders also took the tasks left by the developers. The job boundary between developers and leaders is equivocal.

Collectively, the graph density represents the mutuality from an overall perspective while the reciprocity represents the mutuality in the vertex level, and both of them can be used to depict a centralized collaborative network.

Modularity is designed to measure the strength of division of a network into subdivisions [9]. A higher value of modularity indicates that a network has more obvious structure where vertexes have dense connection within the same subdivision but sparse connection between vertexes across subdivision. More specifically, there are several leaders who had been surrounded by their own ‘troops’. Hence, in the view of collaborative work, such structure could be seen as a kind of decentralized co-working style.

Dyad census asymmetry is obtained in terms of counting the number of non-mutual connected pairs of vertexes. Such index can be seen as an opposite of reciprocity. If a network containing more subdivisions, the mutual connection of vertexes across subdivisions will be unlikely to being high. Hence, the value of dyad census asymmetry within decentralized network should be lower than that of the centralized network. In this regard, we used modularity and dyad census asymmetry to depict a decentralized collaborative network. In the similar vein, the modularity represents the decentralization from an overall level while the dyad census asymmetry represents the decentralization in the vertex level.

The benefits or defects of both centralized/mutual and decentralized collaborative structure can be found in the previous literatures [19, 22, 23, 24, 25]. Hence, we proposed the competing propositions:

Proposition 1 (competing): A centralized/mutual collaborative network can accelerate the project success in the context of FLOSS mobile applications.

Proposition 2 (competing): A decentralized collaborative network can accelerate the project success in the context of FLOSS mobile applications.

4 Methodology and Data Analysis

By collaborating with Sourceforge, we were permitted to access the information of entire OSS projects hosted in Sourceforge. We selected the all the mobile projects incepted from May 2008; the data was collected till July 2011. In total, there were 3,076 mobile projects hosted in Sourceforge during the period. After filtering out the inactive projects and the outliers, there were a total of 383 projects being entered our analysis model. The collaborative networks of these projects were examined and the focal social network indexes were computed. The detailed descriptive analysis is shown in Table 1. In order to control for the bias of longitudinal effect, we did not set a fixed time stamp of T1 to measure number of downloads or number of page views but obtain such values in the middle of project inception and July 2011. Furthermore, the correlation table is shown in Table 2. We can find that all the correlation coefficients between the predictors are less than 0.7. Hence, there is inexistence of multicollinearity in our analysis model.

Table 1. Descriptive Data Analysis

Variable	Definition	Mean	Std. Dev.	Min.	Max.
ND_t2*	Number of downloads at T2	0.5048	0.4003	0.0001	1
PV_t2**	Number of page views at T2	1.5876	0.9493	0.2218	5.4014
ND_t1*	Number of downloads at T1	0.3044	0.3803	0.00004	1
PV_t1**	Number of page views at T1	2.0350	0.9395	0.1663	4.9657
Gd_t2*	Graph Density at T2	0.5780	0.1808	0.0021	0.9949
R_t2***	Reciprocity at T2	0.0387	0.1232	0	1
M_t2**	Modularity at T2	0.0071	0.0246	-0.0085	0.1761
Dca_t2*	Dyad census asymmetry at T2	0.2489	0.1754	0.0004	0.5
Usr_r*	Incremental ratio of project developers	0.4904	0.0525	0.2	0.8
Page_t2	Project age at T2 (by days)	2060.595	1307.571	56	4252
Frs_r*	Incremental ratio of number of released files	0.5256	0.1256	0.0269	0.9790

**Inversed normalization; **Log-transformed normalization; ***square rooted normalization*

In order to control for the contemporaneous cross-equation error correlation between two dependent variables, namely number of downloads and number of page views, we applied the seemingly unrelated regression model to test our competing hypotheses. The results are shown in Table 3.

Table 2. Correlation Table

	ND_t1 (PV_t1)	**Gd_t2**	**R_t2**	**M_t2**	**Dca_t2**	**Usr_r**	**Page_t2**	**Frs_r**
ND_t1 (PV_t1)	1							
Gd_t2	-0.1360 (0.2179)	1						
R_t2	-0.0292 (0.1617)	-0.1048	1					
M_t2	-0.1227 (0.0921)	0.1803	0.0196	1				
Dca_t2	0.3362 (-0.5426)	0.1425	-0.2085	-0.1507	1			
Usr_r	0.0517 (-0.1355)	0.0260	0.0493	0.0781	0.0049	1		
Page_t2	-0.1666 (0.1229)	0.2670	0.0539	0.1560	-0.1899	0.1634	1	
Frs_r	-0.1742 (0.2888)	0.0387	0.0585	0.0479	0.2367	-0.0363	0.0290	1

Model 1 is the baseline model with all control variables. We can find either the number of downloads or the number of page views in the current time point to be positively influenced by such values in the previous lag. The project ages and the incremental ratio of users and released files only positively influenced the times of downloading but not the page views of project. The indicators depicting the centralized/mutual collaborative network were entered in Model 2. Both graph density and reciprocity have positive impact on the project success, namely number of downloads (such value was inversed for normalization) and page views. The effect of released files is not shown after including two focal variables. Model 3 is the full model. It is obvious that the graph density and reciprocity positively influence the project success but the modularity and dyad census asymmetry have negative impact on the project success. Hence, we can reach the conclusion that the proposition 1 is supported. Thus, in the context of mobile OSS projects, a centralized/mutual collaborative network is preferred.

Table 3. Results

	Model 1 Coef. (Std. Err)		Model 2 Coef. (Std. Err)		Model 3 Coef. (Std. Err)	
	ND_t2	**PV_t2**	**ND_t2**	**PV_t2**	**ND_t2**	**PV_t2**
ND_t1	0.7433***	-	0.7564***	-	0.6877***	-
	(0.0309)		(0.0351)		(0.0347)	
PV_t1	-	0.69686***	-	0.6772***	-	0.5449***
		(0.0305)		(0.0348)		(0.0398)
Gd_t2	-	-	-0.2315***	0.9434***	-0.3866***	1.4245***
			(0.0765)	(0.1796)	(0.0748)	(0.1852)
R_t2	-	-	-0.2940***	0.6828***	-0.1713*	0.5007**
			(0.1082)	(0.2532)	(0.1029)	(0.2413)
M_t2	-	-	-	-	0.5806***	-1.9681*
					(0.0814)	(1.2144)
Dca_t2	-	-	-	-	1.5122***	-1.5632***
					(0.5200)	(0.2162)
Usr_r	0.9536***	-0.3939	0.9192***	-0.2290	0.8418***	-0.3389
	(0.2541)	(0.5750)	(0.2553)	(0.5940)	(0.2396)	(0.5655)
Page_t2	0.00004***	0.00002	0.00004***	-0.00001	0.00005***	-0.00005**
	(0.00001)	(0.00002)	(0.00001)	(0.00002)	(0.00001)	(0.00002)
Frs_r	-0.2291**	-0.2610	-0.1423	-0.3953	-0.0042	-0.6142
	(0.1029)	(0.2370)	(0.1066)	(0.2518)	(0.1018)	(0.2403)**
R^2	0.5710	0.5741	0.5856	0.6059	0.6368	0.6468

$*p<0.1$; $**p<0.05$; $***p<0.01$

5 Discussion and Conclusion

Our results suggest that the interactive collaboration is especially important for mobile application development process. Developers formed centralized/mutual structure for collaboration instead of several collaborative subdivisions. Such findings provided both theoretical and managerial implications for either existing or future work. For the researchers, this article examines the mobile application development in the FLOSS platform, which has been inadequately examined. To the best of our knowledge of project management, such collaborative network is analogous to the notions from LMX (leader-member exchange) theory or path-goal theory. Future work may be explored and endeavored in-depth discussion of such topics. For the practitioners, we suggest that project managers should pay more endeavors on establishing interactive collaboration network among the developers instead of overemphasizing the decentralized collaborative structure for developing an OSS mobile application. The intelligence or skills from each person should be centralized instead of locating in her/his own ‘circle(s)’.

Like any other studies, this work also has a number of limitations that afford opportunities for future research. First, we examined the collaborative network of each project. However, we did not consider the case of projects of same project leader(s). Second, we only use two indexes, number of download and number of pages views, to depict the project success. A more holistic measurement of OSS project success is needed in the future research. Third, the developers might have used other IT artifacts, such as forums, mailing lists, or private messages, for communication, even for organizing or allocating the tasks. All such caveats provide researchers with the opportunity to refine this exploration in the future.

References

1. Gartner, Inc., http://www.gartner.com/technology/research/predicts/ (2012)
2. Holzer, A., Ondrus, J.: Trends in mobile application development. In: Hesselman, C., Giannelli, C. (eds.) Mobilware 2009 Workshops. LNICST, vol. 12, pp. 55–64. Springer, Heidelberg (2009)
3. Von Krogh, G., Spaeth, S., Lakhani, K.R.: Community, joining, and specialization in open source software innovation: a case study. Research Policy 32(7), 1217–1241 (2003)
4. Crowston, K., Howison, J., Annabi, H.: Information systems success in free and open source software development: Theory and measures. Software Process: Improvement and Practice 11(2), 123–148 (2006)
5. Subramaniam, C., Sen, R., Nelson, M.L.: Determinants of open source software project success: A longitudinal study. Decision Support Systems 46(2), 576–585 (2009)
6. Stewart, K.J., Gosain, S.: The impact of ideology on effectiveness in open source software development teams. MIS Quarterly 30(2), 291–314 (2006)
7. Coleman, T.F., Moré, J.J.: Estimation of sparse Jacobian matrices and graph coloring blems. SIAM Journal on Numerical Analysis 20(1), 187–209 (1983)
8. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. Proceedings of the National Academy of Sciences 105(4), 1118–1123 (2008)
9. Newman, M.E.J.: Modularity and community structure in networks. Proceedings of the National Academy of Sciences 103(23), 8577–8582 (2006)
10. Wasserman, S., Faust, K.: Social network analysis: Methods and applications. Cambridge University Press (1994)
11. Garlaschelli, D., Loffredo, M.I.: Patterns of link reciprocity in directed networks. Physical Review Letters 93(26), 268701 (2004)
12. Von Hippel, E., Von Krogh, G.: Open source software and the “private-collective” innovation model: Issues for organization science. Organization Science 14(2), 209–223 (2003)
13. Shah, S.K.: Motivation, governance, and the viability of hybrid forms in open source software development. Management Science 52(7), 1000–1014 (2006)
14. Hagedoorn, J., Duysters, G.: External sources of innovative capabilities: the preferences for strategic alliances or mergers and acquisitions. Journal of Management Studies 39(2), 167–188 (2002)

15. Lounsbury, M., Crumley, E.T.: New practice creation: An institutional perspective on innovation. Organization Studies 28(7), 993–1012 (2007)
16. Sen, R., Singh, S.S., Borle, S.: Open source software success: Measures and analysis. Decision Support Systems 52(2), 364–372 (2012)
17. Grewal, R., Lilien, G.L., Mallapragada, G.: Location, location, location: How network embeddedness affects project success in open source systems. Management Science 52(7), 1043–1056 (2006)
18. Hahn, J., Moon, J.Y., Zhang, C.: Emergence of new project teams from open source software developer networks: Impact of prior collaboration ties. Information Systems Research 19(3), 369–391 (2008)
19. Singh, P.V., Tan, Y., Mookerjee, V.: Network effects: the influence of structural capital on open source project success. MIS Quarterly 35(4) (2011)
20. Conaldi, G., Lomi, A., Tonellato, M.: Dynamic models of affiliation and the network structure of problem solving in an open source software project. Organizational Research Methods 15(3), 385–412 (2012)
21. Singh, P.V.: The small-world effect: The influence of macro-level properties of developer collaboration networks on open-source project success. ACM Transactions on Software Engineering and Methodology (TOSEM) 20(2), 6:1–6:27 (2010)
22. Kirsch, L.S.: Portfolios of control modes and IS project management. Information Systems Research 8(3), 215–239 (1997)
23. Boutellier, R., Gassmann, O., Macho, H., Roux, M.: Management of dispersed product development teams: The role of information technologies. R & D Management 28(1), 13–25 (1998)
24. Bresnen, M., Goussevskaia, A., Swan, J.: Embedding new management knowledge in project-based organizations. Organization Studies 25(9), 1535–1555 (2004)
25. Koskela, L., Howell, G.: The underlying theory of project management is obsolete. In: Proceedings of the PMI Research Conference, PMI, pp. 293–302 (2002)