Mining the Network of the Programmers: A Data-Driven Analysis of GitHub

Yezhou Ma, Huiying Li, Jiyao Hu, Rong Xie, Yang Chen
School of Computer Science, Fudan University, China
{yzma14, huiyingli13, jyhu13, xieronglucy, chenyang}@fudan.edu.cn

ABSTRACT

GitHub is a worldwide popular website for version control and source code management. In addition, since its users can follow each other, it also forms a *professional social network* of millions of users. In this work, we perform a data-driven study for analyzing the GitHub network. By introducing a distributed crawling framework, we first collect profiles and behavioral data of more than 2 million GitHub users. To the best of our knowledge, this is the largest and latest public dataset of GitHub. Then, we build the social graph of these users and conduct a thorough analysis of the network structure. Moreover, we investigate the user behavior patterns, particularly the patterns of the "commit" activities. Finally, we utilize machine learning methods to discover important users in the network with a high accuracy and a low overhead. Our inspiring findings are helpful for GitHub to provide better services for its users.

CCS CONCEPTS

• **Human-centered computing → Social network analysis**;

KEYWORDS

GitHub, professional social networks, PageRank, machine learning, spatial-temporal analysis

ACM Reference format:
Yezhou Ma, Huiying Li, Jiyao Hu, Rong Xie, Yang Chen. 2017. Mining the Network of the Programmers: A Data-Driven Analysis of GitHub. In *Proceedings of ChineseCSCW '17, Chongqing, China, September 22-23, 2017,* 4 pages.
https://doi.org/10.1145/3127404.3127431

1 INTRODUCTION

Online social networks (OSNs) such as Facebook [13], Twitter [7] and Foursquare [3] have provided great platforms for communication and interaction among people [6]. Much work has been done through analysis of these networks. In this work, however, we focus on the construction and analysis of a *professional social network* built from the data of GitHub. GitHub is a popular international website for version control and source code management. Users can push their codes to GitHub, and collaborate with other users on projects conveniently. These technical operations make GitHub a professional version control and source code management platform. Furthermore, users can follow and send messages to others and star other their repositories. These social features also make the GitHub network a professional social network.

We are also keen on revealing GitHub user behavior patterns and discovering important users in the GitHub network. To reach our goal, we first build the social graph of GitHub based on the crawled data of more than 2 million users. Then we analyze its structural features. Furthermore, we investigate the temporal and spatial patterns of user behavior, especially about the "commit" operations. Finally, we build a model to predict important users in the network using machine learning methods. We can achieve an F1-score of 0.7834.

Overall, our main contributions can be summarized as below:

(1) We implement a distributed crawler and collect more than 2 million users' data from GitHub. This dataset has been the latest and largest compared to other work thus far.
(2) We perform a series of structural, temporal and spatial analysis of the massive data we collected. We gain insights on the interactions and the "commit" behavior of GitHub users.
(3) We apply machine learning techniques to the discovery of important users with a high accuracy and low overhead. We use only individual users' features instead of the knowledge of the whole network, which makes the discovery much more efficient.

Our inspiring findings facilitate a better understanding of connections between programmers, content publishing activities and the discovery of important users in the professional social network. It is helpful for GitHub to provide better services for users.

The rest of the paper is structured as below: We first introduce our data collection method in Section 2. Then we build and analyze the GitHub network in Section 3, and investigate the content publishing patterns in Section 4. Next we predict important users in the network by using machine learning methods in Section 5. Finally we introduce related work in Section 6 before we conclude our work in Section 7.

2 DATA COLLECTION

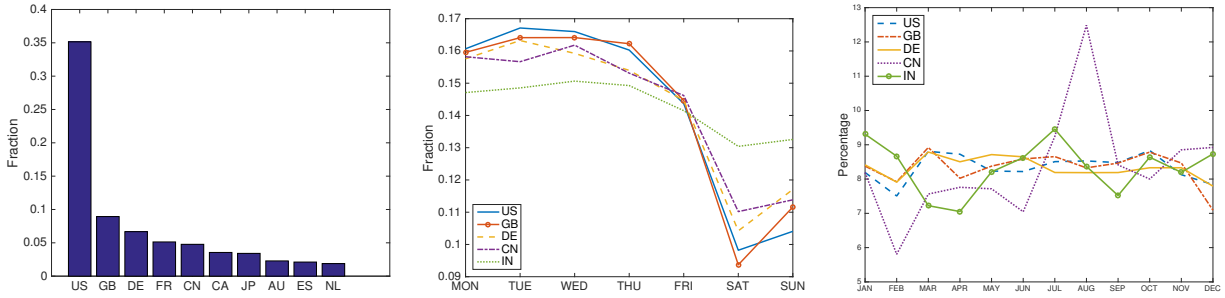
As in [5], we implemented a distributed crawler which served to fetch data from each user's profile page on GitHub's websites. The crawler is composed of two parts, i.e., a *scheduler* and a number of *crawling workers*. The scheduler maintains a MySQL database to record workers' progress information. Starting with a randomly selected active user, our distributed crawler uses the Breadth-First Search (BFS) algorithm to obtain a number of GitHub users' profile pages and the lists of their followers and followings. The crawler drops repeated users automatically.

ChineseCSCW '17, September 22-23, 2017, Chongqing, China

ACM ISBN 978-1-4503-5352-6/17/09.
https://doi.org/10.1145/3127404.3127431

Attribute	Definition	Value
Nodes	Number of nodes.	2,006,356
Edges	Number of edges.	10,034,342
Nodes with 0 in-degree	Number of nodes which do not have edges pointing to them.	555,231
Nodes with 0 out-degree	Number of nodes with no edges pointing to others.	517,422
Nodes with positive in&out-degree	Number of nodes whose both in and out-degree is more than 0.	934,228
Directed edges	Number of directed edges.	10,034,342
Undirected edges	Number of edges in the undirected version of the network	9,070,211
Mutual edges	Number of edge pairs between two nodes	1,928,262
Diameter of graph	The length of the longest "shortest path" in a graph.	19

Table 1: Basic Statistics of the GitHub Network



(a) Percentage of contributions of top 10 countries (b) #commits of five countries in a week (c) #commits of five countries in a year

Figure 1: Pattern of Commits in GitHub

We deploy our distributed crawler to 21 nodes on the North Virginia data center of Amazon Web Services. One of them is the scheduler and the rest are crawling workers. 2,006,356 GitHub users' information were collected from April 2nd to April 13rd, 2016. Note that we respect to the privacy of GitHub users, and only fetch the publicly-accessible information.

3 ANALYSIS OF THE GITHUB NETWORK

We construct a directed and unweighted network $G = (V, E)$ of GitHub users. In this network, each node $v \in V$ represents a GitHub User. If user A follows user B on GitHub, then we add a direct edge pointing from A to B, i.e., $(v_A, v_B) \in E$. Among all the users we collected information from, we find that all of them are in one *weakly connected component*. Basic information of this weakly connected component is shown in Table 1. The diameter of the GitHub network is 19, which is much larger than that of mainstream OSNs, e.g., 13.4 in Facebook [13]. It means that the connections between users in the GitHub network are not as close as those in mainstream OSNs such as Facebook.

4 ANALYSIS OF THE "COMMIT" ACTIVITIES

We perform both temporal and spatial analysis of their behavior, especially of the widely used "commit" behavior. A commit in GitHub is a change to a file or a set of files. A unique ID will be assigned to each commit that allows programmers to keep track of what changes were made when and by whom.

As the key function of GitHub, it is meaningful to investigate the commit behavior of active users. We define users who have more than 50 contributions as *active committers*. At first, we inspect the spatial features of them. We find out that they come from 129 different countries. Among these countries, United States (US), United Kingdom (GB), Germany (DE), France (FR), China (CN), Canada (CA), Japan (JP), Australia (AU), Spain (ES) and Netherlands (NL) are top 10 countries having the largest number of active committers. The fractions of the total number of contributions of these countries are shown in Fig. 1(a).

We further explore the temporal patterns of commit behavior. From Fig. 1(b), we find out that programmers make much fewer contributions on Friday than they do from Monday to Thursday, and Saturday is the day with the fewest contributions in a week. This is in alignment with our common sense that people do not work during weekends. As a result, their productivity decreases since Friday. Besides, we find that compared to programmers in US, GB and DE, programmers in CN and IN tend to have more commits on Saturdays. What is more, programmers in IN almost generate as many commits on Saturdays as they do on weekdays.

We also visualize commits of users in these five countries in the past 12 months (Fig.1(c)). It is shown that number of commits from Chinese programmers drop significantly in February due to the Chinese New Year.

In conclusion, spatial and temporal analysis of user behavior reveal certain real world facts. These findings are helpful for GitHub to gain a better understanding about its users and improve its resource provisioning accordingly. Also, these findings help governors know more about the working load of their citizens as well as the contributions of the programmers in their countries, which can be referred by policy making.

5 IDENTIFYING IMPORTANT USERS BY MACHINE LEARNING

As in [7, 11], we introduce *PageRank* [10] to evaluate the importance of each user in the network. PageRank has been used by Google to measure the importance of web pages. We run PageRank algorithm on the constructed GitHub network G and get the PageRank value of each node. Our aim is to distinguish between users with highest PageRank scores and ordinary GitHub users. We introduce the following three metrics, i.e., the number of contributions in the last year, the number of starred repositories, and the longest streak period in the last year [1]. As shown in Fig. 2, we can see the users whose PageRank score rank top 2000 stand out among all users in the above mentioned aspects. Therefore, these users are more active in contributing to GitHub. Although all GitHub users have identical functionality in making contributions to projects, the GitHub social network provides a great incentive for users to contribute more.

However, calculating PageRank is a resource-consuming operation because it requires the knowledge of the whole network. It means that to measure the importance of one user, the information all users' social connections is required. However, the GitHub network evolves from time to time. Therefore, getting the latest snapshot is not cost effective due to the overhead of data collection. We then design and build a model using only an individual user's data to predict whether she is an important user or not. In order to gain sufficient data set, we define users having top 1% highest PageRank scores as *important users*, with a total number of 20,000. To balance our dataset, we then randomly pick 20,000 of users from the rest data as *unimportant users*.

5.1 Features Selection

For each user, as listed in Table 2, we select 14 features related to the user's behavior and profiles. We add the age of account to the feature set because we consider that the older the account is the more important it might be. Besides, we can see that important users tend to be active in pushing code or making contributions to the network, thus streak period is included. Also, important users could be famous people in the community who are willing to disclose their home city and emails, which are also included.

Feature	Explanation
repos	#Repositories the user has
forks	Sum of #forks for original repos
max-forked-repo	#Forks of repo with highest #forks
account age	Days since registration
starred-repos	#Repos that have been starred
streak	Length of streak period last year
contributions	#Contributions to any repos
starred	Total #stars the user gets
stars	Total #stars the user gives to others
organizations	#organizations the user joins
curr-streak	Length of current streak period
recent-contributions	#contributions in last year
location	Whether the user has location info
email	Whether the user has public email

Table 2: Selected Features

[1] Streak period is a continuous period of days in which the user keeps making contributions every day.

Rank	Feature	χ^2
1	stars	8950.6433
2	repos	8163.0955
3	recent-contributions	6686.6546
4	contributions	6452.7047
5	starred	6360.2334
6	streak	6259.3827
7	starred-repos	5949.7399

Table 3: Result of χ^2 statistic

Parameter	Value
colsample_bytree	0.8
min_child_weight	1
subsample	0.8
eta	0.1
max_depth	4
gamma	0.2
subsample	1
lambda	0
alpha	0
booster	gbtree
objective	binary:logistic

Table 4: Selection of parameters of XGBoost

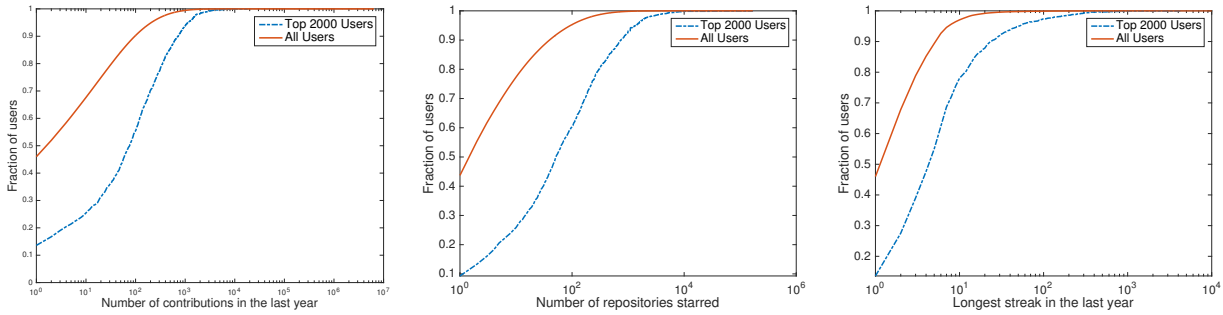
In order to measure the discriminative power of features we selected, we do χ^2 statistics on these features [14]. Table 3 shows the top 7 features. The higher the value is, the more discriminative the feature is for the discovery of important users. We can see that the features having the 5 highest χ^2 values are all user behavior related features such as stars, repos and contributions. It implies that how much a user contributes in GitHub impacts the user's importance in the GitHub professional social network a lot. This finding suggests that those GitHub users who want to be influential should contribute more.

5.2 Model Training and Evaluation

We apply Gradient Boosting algorithm to our machine learning model to tell whether a user is important or not. In practice, we use XGBoost, a scalable tree boosting system, which has been widely applied in recent machine learning competitions like Kaggle [2]. We use precision, recall and F1-score to evaluate the important users' identification results given by our model. Here, precision is the ratio of identified important users that are truly important. Recall is the fraction of truly important users that are identified. F1-score is the harmonic mean of precision and recall. 5-fold cross-validation is used to train and evaluate the model. We tune the parameters of XGBoost, we record the set of parameters leading to the highest F1-core in Table 4. The resulted F1-score is 0.7834. Meanwhile, the corresponding precision and recall values are 0.7699 and 0.7974, respectively. This indicates that by utilizing the individual user's behavior and profile features, we are able to identify important users in the GitHub professional social network with a high accuracy.

6 RELATED WORK

There has been much research on online social networks (OSNs) and massive online collaboration in recent years. Also, researchers have conducted data-driven studies on GitHub.



(a) (b) (c)

Figure 2: Differences between all users and top 2000 users

On OSNs, researchers first focused on some representative OSNs like Facebook [13] and Twitter [7]. They mainly focused on connections and interactions between users, and they did some analysis on degree of nodes, diameter, radius, clustering coefficient, etc. Regarding large-scale online collaboration, much work has been done on Wikipedia. Vuong et al. [12] proposed two Controversy Rank models to solve disputes on some hot articles by leveraging the amount of disputes within an article and the relationships between articles and contributors. In recent years, more methods originally widely used on OSNs have been introduced into research on Wikipedia. Brandes et al. [1] used a novel network (graph) model to encode online interactions, and they proposed several indicators to characterize the graph structure.

Github is known as a *professional social network*. At first, research on GitHub is still more about merely *online collaboration* [4], but it soon came out the research using *social network* approach. Majumder et al. [9] tried to prove that effective team selection requires the team members to be socially close, and they built a model for team formation. Lima et al. [8] did some basic statistics like number of contributors per project and followers per user, and they analyzed social ties and repository-mediated collaboration patterns on GitHub. Using Geocoding, they did some early geographical analysis. In our work, we perform analyses from many new aspects, for example, spatial-temporal analysis of Github commit patterns and detecting important users based on machine learning. These studies have not been conducted by previous work.

7 CONCLUSION

In this paper, we aim to facilitate a better understanding of connections between GitHub programmers, reveal GitHub user behavior patterns and discover important users in the GitHub network. We successfully collect profiles and behavioral data of more than 2 million GitHub users. This is the largest and latest public dataset of GitHub. Then we build the social network of these users and analyze the network structure. Moreover, we investigate users' temporal and spatial behavior patterns especially through the patterns of the "commit" activities. It suggests that programmers show clear working patterns in a week, regardless of countries. Yet, programmers in developed countries tend to do less coding than programmers in developing countries at weekends. Finally, we introduce PageRank to evaluate the importance of users, and utilize machine learning methods to discover important users in the network. The F1-score of the prediction is 0.7834. Our findings can help GitHub provide better services and help researchers gain a better understanding professional social networks.

For future work, we aim help users discover valuable projects according to their interests. Also, we plan to study the evolution of the GitHub network, and examine the user interactions from a dynamic perspective.

ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China (No. 61602122), Natural Science Foundation of Shanghai (No. 16ZR1402200), Shanghai Pujiang Program (No. 16PJ1400700). The first two authors contributed equally to this work. Yang Chen is the corresponding author.

REFERENCES

[1] U. Brandes, P. Kenis, J. Lerner, and D. van Raaij. Network analysis of collaboration structure in wikipedia. In *Proc. of WWW*, 2009.
[2] T. Chen and C. Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proc. of ACM KDD*, 2016.
[3] Y. Chen, Y. Yang, J. Hu, and C. Zhuang. Measurement and Analysis of Tips in Foursquare. In *Proc. of IEEE PerCom Workshops*, 2016.
[4] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. Social coding in github: transparency and collaboration in an open software repository. In *Proc. of ACM CSCW*, 2012.
[5] C. Ding, Y. Chen, and X. Fu. Crowd Crawling: Towards Collaborative Data Collection for Large-scale Online Social Networks. In *Proc. of ACM COSN*, 2013.
[6] L. Jin, Y. Chen, T. Wang, P. Hui, and A. V. Vasilakos. Understanding User Behavior in Online Social Networks: A Survey. *IEEE Communications Magazine*, 51(9):144–150, 2013.
[7] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Proc. of WWW*, 2010.
[8] A. Lima, L. Rossi, and M. Musolesi. Coding Together at Scale: GitHub as a Collaborative Social Network. In *Proc. of AAAI ICWSM*, 2014.
[9] A. Majumder, S. Datta, and K. Naidu. Capacitated team formation problem on social networks. In *Proc. of ACM KDD*, 2012.
[10] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.
[11] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogenous networks. In *Proc. of ACM WSDM*, 2012.
[12] B.-Q. Vuong, E.-P. Lim, A. Sun, M.-T. Le, H. W. Lauw, and K. Chang. On ranking controversies in wikipedia: models and evaluation. In *Proc. of ACM WSDM*, 2008.
[13] C. Wilson, B. Boe, A. Sala, K. P. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. In *Proc. of ACM EuroSys*, 2009.
[14] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *Proc. of ICML*, 1997.