Effect of the Diversity of Interests on GitHub Team Productivity

Yifan Zhang
ivana.zhang@mail.utoronto.ca

I. Introduction

THIS purpose of this project is to understand the role of diversity of individual users and whole team members on the productivity of the repositories in Open Source Software(OSS) Collaboration environment like Github. Team diversity is one of the fundamental issues in social and organizational studies that has been broadly researched on. OSS projects on GitHub, like Wikipedia, highly rely on collaboration and naturally embraced the diversity. It is interesting to study whether team members have diverse interests tend to be more productive in GitHub projects.

The project is based on the paper Diversity of editors and teams versus quality of cooperative work: experiments on Wikipedia[1], written by M. Sydow, K. Baraniak, and P. Teisseyre. In the paper, the authors proposed an original diversity measure to quantify the diversity of interests of the editor in Wikipedia. The interest profile of each editor is defined as the interest distribution vector over the set of all categories. And the diversity of interests(or equivalently versatility) of the editor is defined as the entropy of interest profile.

In order to immigrate this Wikipedia-based project to GitHub database, I select the following features to analyze: domain versatility, language versatility, and productivity.

1) Versatility
   The Github repositories are classified into different domains base on their functionalities and labeled with their main programming languages. In this project, I calculate the domain versatility and language versatility for each committer and calculate both versatilities for each repository. Similar to the main domain of the Wikipedia article, the domain of the Github projects could reflect the interest of GitHub users. Also, mastering different programming languages can also indicate user interest in different fields. For example, language like HTML and CSS are strongly related to Front End, and language like python and R will be the first choice for data analysis.
2) Productivity
   For productivity, an important indicator of a successful Github project, we select commits number to quantify the productivity of GitHub teams according to Rajdeep[2]. A commit occurs when a developer uploads the altered source code file, where the Concurrent Versioning System (CVS) tool updates the changed files automatically. CVS commits reflect meaningful changes to the source code, which is reasonable to treat the number of commits as an indicator of successful technical refinement.

According to the core paper, the author conducts two main experiments on Wikipedia dataset. The first one is to experimentally study the dependence between the editors' versatility and the quality of articles they co-edit. In my own project, we report the experiment to analyze the relationship between an individual GitHub user's Productivity and his Versatility. The second experiment makes a further step and conducted on whole teams of authors and apply a logistic regression model on analyzed data. Since productivity is a quantitative variable instead of a categorical variable, so We use linear regression instead to evaluate the relationship between diversity and productivity within the GitHub repository teams.

II. RELATED WORK

The impact of diversity on collaboration has been studied and broadly theoriesd on virtual communities over the past 40 years. In terms of collaboration pattern, there are two competing theories describing efficient team organization: modularity and integrity. David Parnas, who introduced the modularity pattern, believed that the co-dependence between different components should be eliminated. But in integral mode, the team members must have diverse knowledge or skills, which lead to lower specialization and more diverse. The aim of the project is to

study whether modular or integral collaboration pattern is more successful in GitHub repositories developing based on the work of the core paper[1].

M. Sydow et al [3] used statistical and machine learning techniques, which are very effective tools to investigate such dependencies, to investigate how these measures influence the work quality. They demonstrate on Wikipedia data that interest diversity of an editor seems to be correlated with the quality of the articles they co-edit. They also extend the concept of interest diversity on whole teams of authors and study how it impacts the work quality compared to their productivity and experience. Finally, they also demonstrate that it is possible to use statistical machine learning tools to predict the quality of Wikipedia articles using some attributes that model the level of editors diversity (and some other attributes) which can be interpreted as an additional statistical signal that diversity positively affects work quality in Wikipedia.

J Chen et al [4] found that increased diversity in experience with Wikipedia increases productivity and decreases withdrawal up to a point. Beyond that point, productivity remains high, but members are more likely to withdraw. They employed Hierarchical Linear Models for analysis.

Vasilescu and Daryl [5] also investigated in using mixed effects, multiple linear regression models to analyze the relationship of gender and tenure diversity to productivity and turnover when controlling for team size and other confounds. As for gender diversity, the index is calculated by Blau index[6], which is a well-established diversity measure for categorical variables. Different kinds of tenure diversity are taken into consideration, including account tenure, commit tenure, and project tenure. According to their finding, both gender and tenure diversity have a significant, positive effect on productivity.

III. DATA

A. Filtering data on Google BigQuery

The GHTorent project[7] uses GitHub API to collect more than 900GB of raw data and 10GB of metadata, which enables researchers to retrieve scalable, queriable, offline mirror of data through the Github REST API. The dataset used in this project is ght_2018_04_01 from GHTorrent-bq on Google Bigquery, including projects, project_members, project_languages, project_commits, commits, commit_comments, commit_parents, followers and etc. In the projects table, there are 83624114 records of GitHub repositories. Since some of Github repositories are not suitable for the analyzation, I clean and filter the repositories by following standards.

1) Since projects created less than 90 days don't have sufficient data for analyzing diversity and commits, I selected projects created before Oct 2017.
2) Remove inactive repositories, which having strictly less than 10 commits in total and no commit record within the lastest 90 days.
3) Since I need to infer the repositories' domain by its description, I excluded the repositories which lack of description and language.
4) Since I interested in the effects of diversity on team productivity, so I remove very small projects (less than 5 team members).

Even though most of the repositories records are filtered out, this step is paramount to ensuring sufficient variance in the data set.

B. Inferring domain of repository

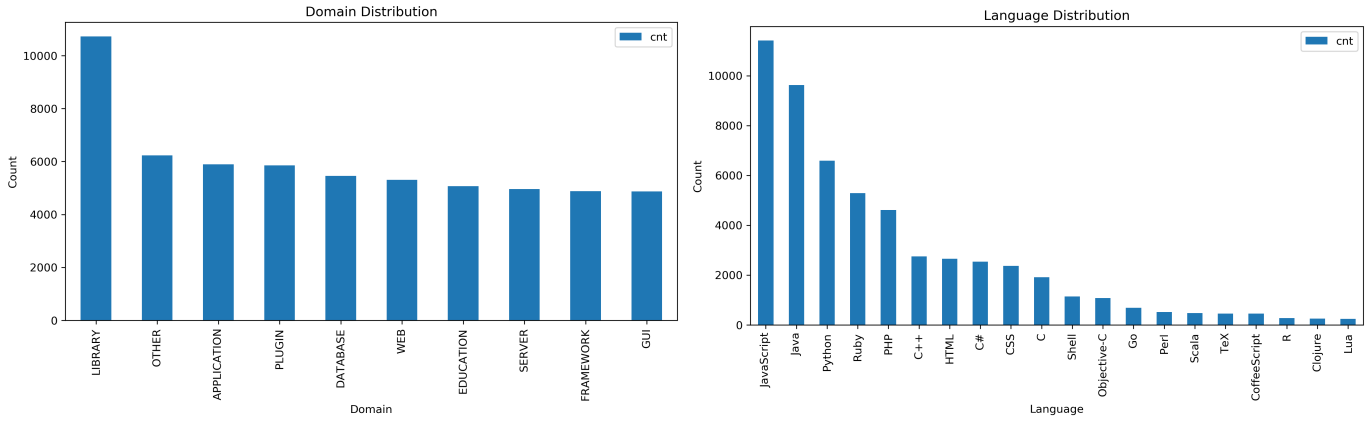
Since Github repositories are not directly tagged with its domains, I use Latent Dirichlet Allocation(LDA) to infer the domain of the projects and generate 10 domains for GitHub repositories based on the name and description of the GitHub repositories.

Before apply LDA model, I concatenate the name and description of each repository as the full-description in order to extract the information of the repository at the largest extent. All full-descriptions are tokenized and cleaned. The processed full-description are feed to CountVectorizer and IDF to generate IDF sparse matrix.

Topic modeling is a type of statistical modeling for discovering the abstract topics that occur in a collection of documents. Latent Dirichlet Allocation (LDA) is an example of the topic model and is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model, modeled as Dirichlet distributions. Since auto-detected domains only include several project-specific keywords, domain names are inspected manually.

TABLE I
DATASETS

Name	Description
project_id	The id of the project
name	The name of the project
language	The main language of the project, including 'PHP', 'Objective-C', 'JavaScript', 'Ruby', 'C', 'C++', 'Shell', 'Python', 'Java' etc.
domain	The domain of the project, including: Domain 0: LIBRARY, Domain 1: FRAMEWORK, Domain 2: PLUGIN, Domain 3: DATABASE, Domain 4: SERVER, Domain 5: OTHER, Domain 6: GUI, Domain 7: EDUCATION, Domain 8: WEB, Domain 9: APPLICATION
total_commits	The total commits number of the project.
total_num	The member number of the project team.
team	The uid of the project team, represent as a string separate by comma.



(a) Domain distribution

(b) Language distribution

Fig. 1. Database Overview.

C. Exploring the Dataset

After processing the raw data provided by GHTorrent, the dataset we use in this project has 7 columns which could be summarized in Table I. There are totally 59244 records in the dataset.

As shown in Fig. 1(a), the distribution of domain is relatively balanced. The LIBRARY domain is the most frequent domain, which is over 10000 records, while GUI is the least frequently domain, which is less than 6000 records. The distribution of language is long-tailed and I visualize the top 20 languages in in Fig. 1(b). It is noticeable to observe that JavaScript is the most popular language with over 10000 records in this Dataset. JAVA, Python, Ruby, and PHP are also really popular according to the result.

D. Calculating Versatility of Individuals and Teams

The measure of interest of diversity proposed in the core paper is based on the entropy term in information theory, which is the biggest innovation of the paper and the definition is not restricted within Wikipedia but suitable to all open-collaboration work.

In the core paper, the author calculated the diversity of interest of users in Wikipedia based on the pre-defined Wikipedia article categories and their contributions to each category. To clarify, let X denote a group of Wikipedia editors, who participate in editing Wikipedia articles. And each article can be mapped to one or more categories from a pre-defined set of categories $C = \{c_1, c_2, ..., c_k\}$ that represent k topics. $t(x)$ denote the total amount of textual content (in bytes) that x contributed to all articles editor co-edited and let $t_i(x)$ denote the total amount of textual content that editor) contributed to the article belonging to a specific category. The interest profile of the editor can be defined as the interest distribution vector over the set of all categories:

$$ip(x) = (p_1(x), p_2(x), ..., p_i(x)) \tag{1}$$

TABLE II
AVERAGE OF DOMAIN VERSATILITY AND LANGUAGE VERSATILITY

productivity level	Description	Count	Average of domain versatility	Average of language versatility
A	total_commits <1000	160147	0.2337085076642322	0.16029239239985488
B	1000<=total_commits<2000	21784	0.4754517293078308	0.32245284431419446
C	2000<=total_commits<3000	11940	0.5548705775313284	0.3752255224312286
D	3000<=total_commits<5000	13834	0.6263895049087452	0.41936311225969
E	5000<=total_commits<10000	16037	0.7330170125686415	0.49278561777226876
F	total_commits>10000	38281	1.166505228422436	0.738431739565536

where $p_i(x) = t_i(x)/t(x)$.

Finally, the diversity of interests (or versatility) of x is defined as the entropy of interest profile of x:

$$V(x) = H((p_1, p_2, ..., p_k)) = \sum_{1 \le i \le k} -p_k log(p_k) \tag{2}$$

While in this project, I consider two kinds of versatility: Domain Versatility and Language Versatility. And the contribution to each domain category or language category denotes the number of commits that the user contribute to each category. For example, assume a user in Github has contributed to 3 projects in domain LIBRARY, FRAMEWORK, and APPLICATION respectively. And total commit number for each domain are 50, 25, and 125. Then the domain interest profile of this user is

$$ip(x) = (\frac{1}{4}, \frac{1}{8}, 0, 0, 0, 0, 0, 0, 0, \frac{5}{8}) \tag{3}$$

And the domain versatility of the user is

$$V(x) = -p_0 log(p_0) - p_1 log(p_1) - p_9 log(p_9) = 1.29875 \tag{4}$$

The team versatility is calculated by the average of all team members' versatility.

IV. MODELING AND RESULTS

A. Experiments Concerning Individual

In this section, we study the dependency between the users' versatility and the productivity of the project they worked on. We analyzed six groups of users and denotes them in Table II. Notice that the six groups represent a rising graded "hierarchy" of productivity from A to F. It is noticeable that the average of the versatility of domain and language are both increasing from A to F level, which indicates there is a positive relationship between versatility and productivity.

For each of the six groups, we computed some statistics concerning versatility measure, including mean, median and quartiles. From the Fig. 2, we could observe a slight increase in versatility when user's productivity add up from level A to level F. In both graph, we could notice a positive connection between versatility and the productivity of the user. In general, more versatility the user are, he or she will be more productive. Also by comparing two graph, we can notice that language versatility seems to have a stronger connection with productivity than domain versatility. However, except for category F, the median of other categories is zero which indicates that most of the users with relatively low productivity might focus on only one domain or master of one language.

B. Experiment Concerning Teams

In this experiment, we use the aggregate data from the previous section and built an additional linear regression model based on analyzed data. Firstly, we separate repositories into three categories regarding their team size. For each category, we select per_commit as the response variable to reflect the productivity of the GitHub team and select domain versatility and language versatility as exploratory variables. In other words, we aim to predict the productivity of the repository teams by their versatilities. Table shows the result of linear regression regrading to three kinds of team size respectively.

We could observe that Root Mean Square Deviation(RMSD, a measure of the differences between values predicted by a model or an estimator and the values observed) is decreasing while the size of group increasing. We could

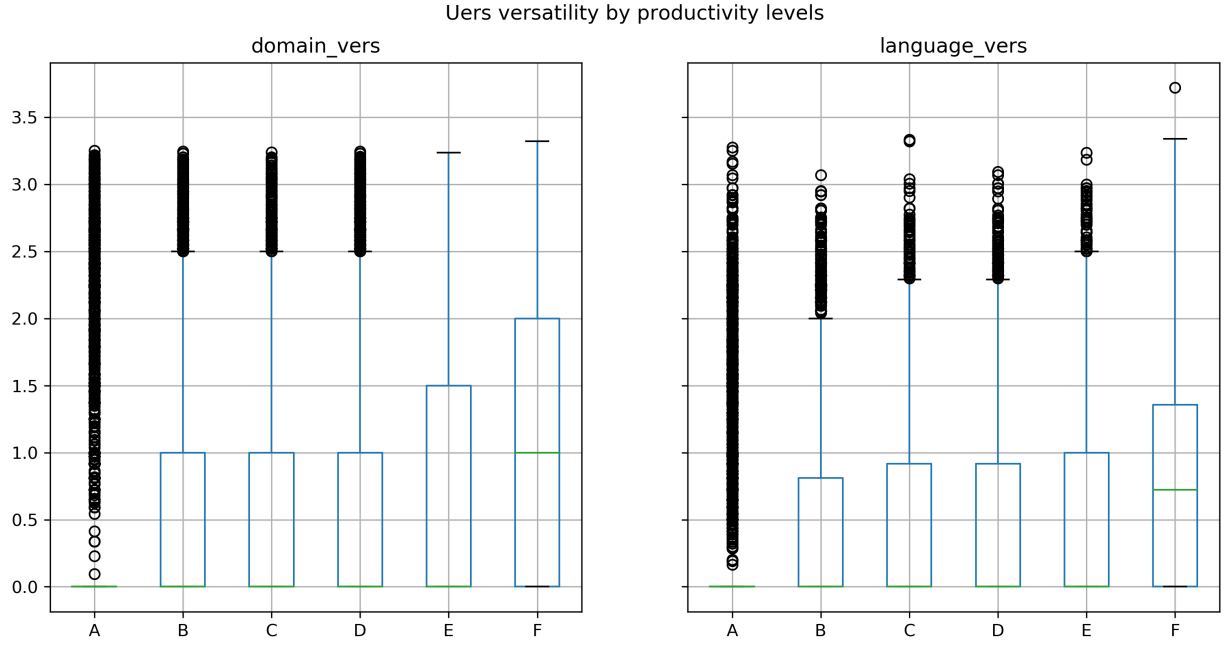


Fig. 2. Simulation Results.

TABLE III
LINEAR REGRESSION MODEL FOR TEAMS

Team size	Coefficients	Intercept	train_RMSE	test_RMSE
small_team	[3.730176678246042, 2.090636085447597]	47.873306	230.220384	194.357750
medium_team	[-0.7141944373278759, -1.6987954758988302]	48.664503	191.990099	223.458552
large_team	[-7.605424938512877, 3.139847160775286]	27.378889	89.431752	47.673887

speculate that as group size increases, the team versatility will also increase and lead to higher productivity. However, We need to admit that the resulting model is not ideal for predicting productivity since the collaboration among Wikipedia projects and collaboration among GitHub projects might have different mechanism and characteristics.

For Wikipedia editor, high diversity of interest reflects the editor's erudite and passion for Wikipedia editing work. Diverse knowledge and familiar collaboration could help editors develop high-quality Wikipedia articles. As Baraniak mentioned in the core paper[1], user with broader expertise might act as ties between community subgroups, which is a crucial factor for maintaining a group coherence.

However, for Github user, commit to a different domain of projects or different language of projects has no strong relation to their work quality. For Github users, expertise in a specific domain turns to be more productive than users who have been involved in many fields.

V. CONCLUSION

In this project, we have presented a large scale of versatility as it relates to GitHub repositories' productivity. As shown in the result, the individual's versatility has a strong correlation to his or her productivity. This confirms the modular theory that a developer with diverse skills and interest will improve outcomes. On the other hand, the relationship between team versatility and productivity is not as ideal as the core paper. There is not a clear pattern in versatility and productivity regarding to different team size. We could speculate that the specialized users could make a huge difference on group productivity rather than diverse users.

In future work, it would be interesting to investigate versatility in different perspectives like gender and location. Also, we observe that the dataset ght_2018_04_01 provided by GHTorrent is not complete. For example, in users table, no user is created in the year 2010, 2014, 2015, 2016. And in projects table, no project is updated in the year 2018. Therefore, we doubt that the faults and incompletency of dataset might affect the result of the experiments. A better dataset should be selected for analyzation in the future.

REFERENCES

[1] K. Baraniak, M. Sydow, J. Szejda, and D. Czerniawska, "Studying the Role of Diversity in Open Collaboration Network: Experiments on Wikipedia." Springer, Cham, 2016, pp. 97–110. [Online]. Available: http://link.springer.com/10.1007/978-3-319-28361-6_8

[2] R. Grewal, G. L. Lilien, and G. Mallapragada, "Location, location, location: How network embeddedness affects project success in open source systems," *Management science*, vol. 52, no. 7, pp. 1043–1056, 2006.

[3] M. Sydow, K. Baraniak, and P. Teisseyre, "Diversity of editors and teams versus quality of cooperative work: experiments on wikipedia," *Journal of Intelligent Information Systems*, vol. 48, no. 3, pp. 601–632, jun 2017. [Online]. Available: http://link.springer.com/10.1007/s10844-016-0428-1

[4] J. Chen, Y. Ren, and J. Riedl, "The effects of diversity on group productivity and member withdrawal in online volunteer groups," in *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*. New York, New York, USA: ACM Press, 2010, p. 821. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1753326.1753447

[5] B. Vasilescu, D. Posnett, B. Ray, M. G. van den Brand, A. Serebrenik, P. Devanbu, and V. Filkov, "Gender and Tenure Diversity in GitHub Teams," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. New York, New York, USA: ACM Press, 2015, pp. 3789–3798. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2702123.2702549

[6] P. M. Blau, *Inequality and heterogeneity: A primitive theory of social structure*. Free Press New York, 1977, vol. 7.

[7] G. Georgios, "The GHTorent dataset and tool suite," *Proceedings of the 10th Working Conference on Mining Software Repositories*, p. 438, 2013. [Online]. Available: https://dl.acm.org/citation.cfm?id=2487132