2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)

How Often and What StackOverflow Posts Do Developers Reference in Their GitHub Projects?

Saraj Singh Manes
School of Computer Science
Carleton University
Ottawa, Canada
sarajmanes@cmail.carleton.ca

Olga Baysal
School of Computer Science
Carleton University
Ottawa, Canada
olga.baysal@carleton.ca

***Abstract*—Stack Overflow (SO) is a popular Q&A forum for software developers, providing a large amount of copyable code snippets. While GitHub is an independent code collaboration platform, developers often reuse SO code in their GitHub projects. In this paper, we investigate how often GitHub developers re-use code snippets from the SO forum, as well as what concepts they are more likely to reference in their code. To accomplish our goal, we mine SOTorrent dataset that provides connectivity between code snippets on the SO posts with software projects hosted on GitHub. We then study the characteristics of GitHub projects that reference SO posts and discover popular SO discussions that happen in GitHub projects. Our results demonstrate that on average developers make 45 references to SO posts in their projects, with the highest number of references being made within the JavaScript code. We also found that 79% of the SO posts with code snippets that are referenced in GitHub code do change over time (at least ones) raising code maintainability and reliability concerns.**

***Index Terms*—StackOverflow, code snippets, GitHub, open-source, code evolution, tags, SOTorrent, GHTorrent**

I. Introduction

Stack Overflow (SO) is the most popular Q&A forum for software developers. According to the SOTorrent data [1] (as of December 2018), Stack Overflow hosts over 42 million posts. This is a top forum for developers to find solutions to development problems they face. Stack Overflow's official website statistics suggest that 50 million developers and engineers visit its website every month. This provides insights into the scale of dependency among developers on Stack Overflow posts as a source of information. These posts may consists of code snippets, text descriptions and links to external sources for further references, thus, providing a good pool of ready-to-use code snippets, informal documentation and discussions on various code-related concepts.

Software developers may use and adapt code snippets from Stack Overflow without concerning about their maintainability and licensing [2]. Moreover, if a buggy code snippet is copied, it may be difficult for developers to refactor and debug that code since they did not write it themselves. Further, if no reference to the code snippet borrowed from Stack Overflow is included in the developers' code, it makes it even harder for code reviewers and other developers to understand that code.

Based on our mining and exploration of SOTorrent, we found around 6.5 million references to Stack Overflow posts in the GitHub projects, with some projects even referencing more than 6,000 SO posts. This scale of interdependence between open source projects and Stack Overflow discussions motivates us to conduct an empirical study investigating the nature of SO references in the GitHub code repositories. In this paper, we study the characteristics of Stack Overflow posts that are referenced in the GitHub code repos, as well as the characteristics of the GitHub projects that refer to the SO posts. To conduct our study, we linked SOTorrent, an official data dump of Stack Overflow, to the publicly available GitHub data dump, called GHTorrent [3]. Such mapping of SO to the GitHub projects can allow us to gain insights into rich information about developers and their projects such as projects' commit history, team size, etc.

In this paper, we address the following research questions:

- *RQ1: How often do GitHub developers reference SO posts in their code?*
- *RQ2: What are the characteristics of GitHub projects referring to the SO discussions?*
- *RQ3: What types of SO discussions are most popular in the GitHub projects?*
- *RQ4: Do SO discussions with code snippets evolve over time?*

The first two research questions can offer insights into whether GitHub developers borrow code or concepts from SO, and if so, how often they do this and what GitHub projects are more likely to include references to SO posts in their codebase. While RQ3 can help us to understand what concepts developers reference most. Finally, RQ4 addresses the possibility of bug migration from one platform to another and thus provides insights into the problem of code snippet maintainability.

II. Methodology

To investigate our research questions, we mined two large datasets, (1) the MSR 2019 Mining Challenge, i.e., SOTorrent [1], and (2) GHTorrent [4]. We now describe these datasets, provide details on how we map them, and how we extract various characteristics of the GitHub projects such as team size, language tags, as well as attributes of the SO posts including tags.

2574-3864/19/$31.00 ©2019 IEEE
DOI 10.1109/MSR.2019.00047

A. Datasets: SOTorrent and GHTorrent

SOTorrent [1] is an open dataset based on the official Stack Overflow data dump. SOTorrent provides access to the version history of SO content at the level of a whole post and an individual post block [5]. A post block typically includes code snippets and the textual content of the post discussion and is dependent on the author's formatting of the post. SO provides a version history of post blocks, as well as a link to external resources, for example, a link to the GitHub project file that has a reference to some SO post. This linkage of the SO post references with the GitHub project files allows us to map SOTorrent to GHTorrent [3]. We describe this mapping process in Section II-B.

GHTorrent [4] is a queryable, offline mirror of GitHub. Many researchers have already mined GHTorrent in their work, e.g., analyzing pull request development [4], [6], studying social or gender diversity in GitHub teams [7]–[9]. GHTorrent provides data about project development, pull requests, commit history, size of projects, etc. In this work, we leverage GHTorrent for extracting the characteristics of the projects having references to the SO discussions.

B. Mapping SOTorrent with GHTorrent

SOTorrent and GHTorrent are created and being maintained independently. There is no foreign key that can be used to run a join operation for mapping two datasets. While SOTorrent mentions a "path URL" to the GitHub's raw dump file, unfortunately, it can not be used as a foreign key. Yet, we can try to extract such key from this "path URL". Below is an example of the GHUrl entry in `PostReferenceGH` table:

https://raw.githubusercontent.com/RationalAsh/freeIMU/master/debug/decode_float.py.

From this URL, we can identify a pattern such as "https://raw.githubusercontent.com/<user-name>/<project-name>/<branch-type>/<relative-path-to-file>". We then can extract "<user-name>/<project-name>" as the user and project names from this pattern. This "derived" project name can then be used to compute a join operation on the `PostReferenceGH` table in SOTorrent with the `Projects` table in GHTorrent with the join condition specifying the "derived" name in `PostReferenceGH` to be the same as the name of the project in GHTorrent.

C. Data Pre-processing

By mining SOTorrent, we extracted 6,039,434 references to external sources (mainly GitHub) with the total of 439,646 unique GitHub repositories having references to SO. Figure 1 shows the number of GitHub projects that reference SO posts. To minimize potential noise in the data, we tried to eliminate outliers by removing the lower 5% and upper 5% of the data points, i.e., projects that have too few references to SO (e.g., 189,349 GitHub projects include only one reference to SO) or too many references (e.g., over 600). After filtering out these outliers, our final dataset contains 556 GitHub projects with the total unique references to 21,537 SO posts. One average, a GitHub project contains 176 references to SO posts, while the minimum and maximum numbers are 15 and 565 references, respectively.

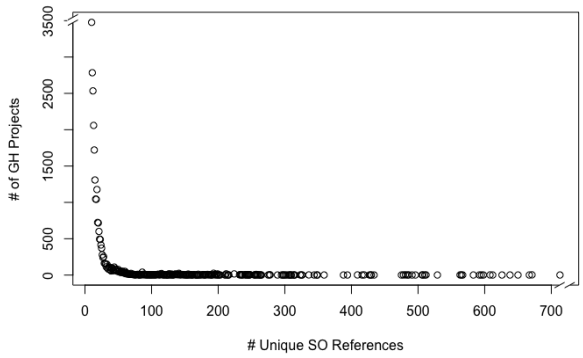

Fig. 1: Number of SO references in GitHub projects.

D. Determining Project's Team Size

To extract the team size of each project from GHTorrent, we considered the number of commits (i.e., contributions) of each project. We identified all unique authors of all commits and consider their count as our "team size" metric. We should note here that after 2014 GitHub had disabled its API to retrieve information about its project members, thus `Project` table in GHTorrent has not been updated since making this a threat to validity as discussed in Section IV.

E. Tags

To extract post tags for our analysis, we mined `Post` table (about 93GB of uncompressed XML data) of SOTorrent. Our data from GHTorrent contains references not only to the "question" posts but also to the "answer" posts. However, tags for the posts in SOTorrent are associated with the "question" posts only. Thus, we extracted tags for the "answer" posts by gathering the tags from the corresponding parent "question" post. We then classify post tags into two categories: *language tags* and *concept tags*. Such classification would allow us to better understand what type of SO posts GitHub developers reference in their code.

1) Language Tags: To eliminate any ambiguity, we use Wikipedia [10] as a reference for identifying all programming languages and their abbreviations. Any tag matching a language name or its abbreviation is classified as a *language tag*. Some cases were resolved manually, e.g., <C++> and <C++11> tags are both identified as C++ language.

2) Concepts: Any tag that is not a language tag is considered to be a *concept tag*. However, we soon noticed that such classification is not quite accurate as we found a large number of concepts related to frameworks or platforms, e.g., <android> being the most frequent one in our dataset. Since we are more interested in the concepts related to code development, we manually filtered out tags that were relevant to platforms, frameworks or tools.

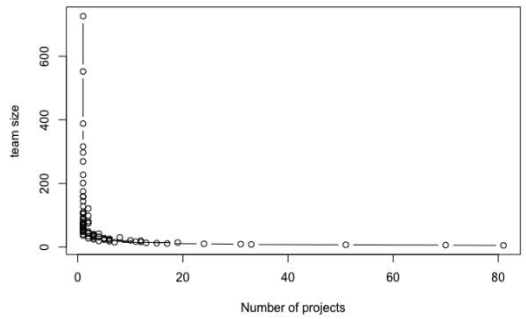

Fig. 2: GitHub team size distribution.

III. Results

Mapping and combining two very large data sources such as SOTorrent and GHTorrent provide a wide research landscape for data mining. For the scope of this paper, we limit our data mining and analysis to four research questions.

A. RQ1: How often do GitHub developers reference SO posts in their code?

Figure 1 shows the distribution of SO references per developer across GitHub projects. We can see that most of the projects have around 30 SO references per developer (on average, each developer has 45 references to SO in their code), while some developers reference SO discussions more than 250 time in their code. Moreover, we determine that the average number of SO references per project is 176 (median is 144 references, while first and third quartiles are at 76 and 241 references, respectively). This shows that open source projects in GitHub frequently reference SO discussion posts.

B. RQ2: What are the characteristics of GitHub projects referring to the SO discussions?

1) Team Size: Figure 2 shows the team size distribution on GitHub. The team size varies between 5 to 726 developers. Please note that only actual contributors to code are counted as developers. Median team size is around 45 developers, while first and third quartiles are at 24 and 84 developers, respectively. More interesting results can be observed when we correlate the team size with the SO references per developer on a team (as shown in Figure 3). The larger the team, the more SO references are present in code. We speculate that as the team size increases the more developers are involved in discussing the referenced SO posts, this can also affect the stability of the code snippet or concept borrowed from Stack Overflow.

2) Language: The language of a project is extracted from the file extension that includes a reference to a SO post. Results are also mapped to the GHTorrent's `Projects` table with the language field. Some GitHub projects have code written in multiple languages. Thus, we believe that looking at the language of the file that has a SO reference gives a more accurate picture of what code (in terms of a programming language) is more likely to borrow code snippets or concepts from Stack Overflow.

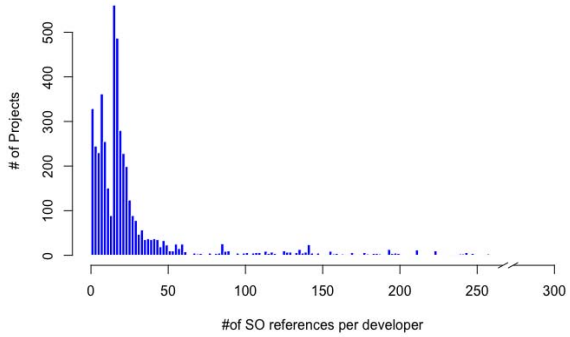

Fig. 3: How often do developers refer to SO? [Note: the # of SO references are normalized by team size].

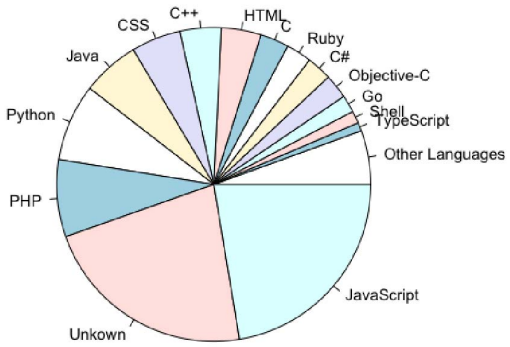

Fig. 4: Language distribution in GitHub projects.

Figure 4 shows the language distribution in GitHub projects with SO references. The majority of the GitHub code repositories that reference SO discussions are related to languages like JavaScript, PHP, Python, Java, CSS, C++, HTML, C, Ruby, C#, etc.

C. RQ3: What types of SO discussions are most popular in the GitHub projects?

We answer this research question by considering the attributes of SO posts — what makes post a popular post in GitHub projects? This analysis is based on the tags of SO posts that we classified as *language* or *concept* tags. As mentioned earlier, tags related to frameworks or platforms are excluded from the analysis.

1) Language Tags: We first observed that majority of the SO posts in our dataset refer to the projects with the matching language tag. 79.5% (1,344,623) of all SO references in our dataset have the same language tag as the language of the project's code. The rest of 20.5% of the SO references we categorized as *concepts* or conceptual references, i.e., when code snippets were adapted to another language domain. We also noticed that the language distribution in projects is similar to the language tag distribution of the SO posts, as shown in Figure 5, with the exception of *typescript* and *json*.

2) Concepts: To better understand what concepts are being borrowed from Stack Overflow, we look at top 30 most popular posts with non-programming language tags. Figure 6 shows

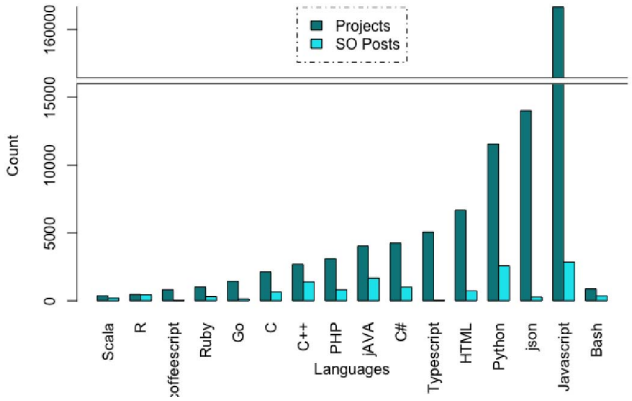

Fig. 5: Distribution of language tags in SO posts and GitHub projects.

most frequent *concepts* that are borrowed from SO by GitHub developers with top popular concepts being related to Linux OS, string and regular expressions.

D. RQ4: Do SO discussions with code snippets evolve over time?

Investigating the possibility of bug migration from Stack Overflow to GitHub projects can offer some insights about maintainability of copied or adapted code snippets. To answer this, we consider all SO posts and their evolution which can be tracked by analyzing the versions of code blocks of those posts. In our dataset, around 27% of the SO posts which are referenced in GitHub are of an *answer* type, while the rest 73% of the posts are *questions*. Developers are more likely to refer to the question posts as a proxy for referencing the overall discussion on a specific SO topic. However, to determine the evolution of code snippets, we consider accepted answers regardless of whether developer referenced a question or an answer post in their projects. Thus, we focus on the versions of code snippets in these answer posts.

We found that 13,821 SO posts in our dataset (i.e., 64% of all SO posts) contain code snippets. Figure 7 demonstrates how many times code snippets change over time. Out of all posts, the code snippets have changed at least once in 10,858 posts (79%) and more than 20 times in 1,193 (9%) posts. The most striking finding in our work is the extend of the code snippets that evolve, i.e., 79% of code snippets in Stack Overflow are changed over time. This finding sheds some light into the evolution of code snippets. One of the implications of this work is to improve developer awareness of the code snippet maintainability. We advise developers who adapt code snippets from SO to keep a close eye on those snippets and discussions in Stack Overflow as they evolve over time.

IV. Threats to Validity

The main threat is related to the lack of the updated data on the project contributors in GHTorrent. Instead, we extracted the team size from `commits` table of GHTorrent that we believe is accurate in identifying the actual code contributors.

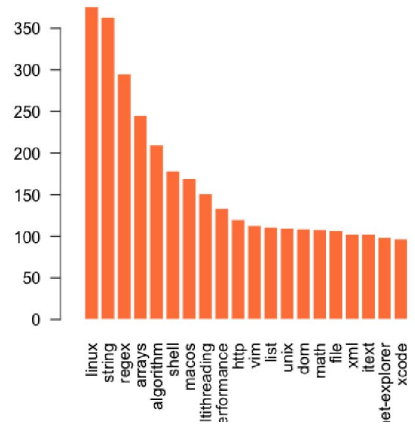

Fig. 6: Most popular SO concepts being borrowed by GitHub developers.

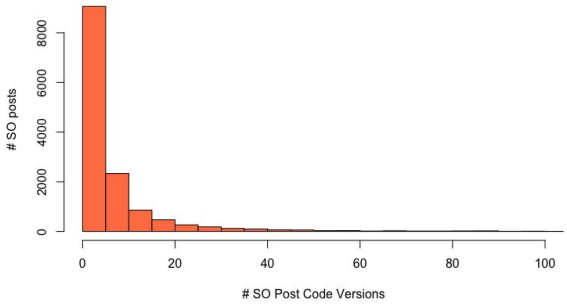

Fig. 7: Evolution of code snippets in SO posts.

Yet, for 55% of GitHub projects we found no any commit information. We don't know why commit data is missing for these projects. We tried to extract additional commit data from `pull_request` table but it was also not reliable as not every commit happens via a pull request.

Another threat is related to the way we treat GitHub projects and their forked repos, i.e., each project is considered and analyzed independently without considering its origin. If a project has been forked from another project, they carry over the same codebase. Therefore, our dataset may have some duplicated SO references from the forked children.

V. Conclusions

This work studies whether and how Stack Overflow posts are referenced by GitHub developers in their open source projects. We have conducted our analysis of SO references in GitHub projects by mapping two very large datasets such as SOTorrent and GHTorrent. Our preliminary findings demonstrate that GitHub developers do reference SO discussions in their code and thus allowing us to study such phenomenon. We found that developers are more likely to reference programming language related discussions that match the language of their code. We also observed that 79% of posts with code snippets evolve over time. Our future work will focus on better understanding of the evolution of code snippets on SO and how it may affect project development and maintenance.

REFERENCES

[1] S. Baltes, C. Treude, and S. Diehl, "SOTorrent: Studying the Origin, Evolution, and Usage of Stack Overflow Code Snippets," in *Proceedings of the International Conference on Mining Software Repositories (MSR 2019)*, 2019.

[2] R. Abdalkareem, E. Shihab, and J. Rilling, "On Code Reuse from StackOverflow," *Inf. Softw. Technol.*, vol. 88, no. C, pp. 148–158, 2017.

[3] G. Gousios, "The GHTorrent Dataset and Tool Suite," in *Proceedings of the 10th Working Conference on Mining Software Repositories*, 2013, pp. 233–236.

[4] G. Gousios, M. Pinzger, and A. v. Deursen, "An Exploratory Study of the Pull-based Software Development Model," in *Proceedings of the International Conference on Software Engineering*, 2014, pp. 345–355.

[5] S. Baltes, L. Dumani, C. Treude, and S. Diehl, "SOTorrent: Reconstructing and Analyzing the Evolution of Stack Overflow Posts," in *Proceedings of the International Conference on Mining Software Repositories*, 2018, pp. 319–330.

[6] G. Gousios, A. Zaidman, M.-A. Storey, and A. van Deursen, "Work Practices and Challenges in Pull-based Development: The Integrator's Perspective," in *Proceedings of the 37th International Conference on Software Engineering - Volume 1*, 2015, pp. 358–368.

[7] 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*, 2015, pp. 3789–3798.

[8] B. Vasilescu, A. Serebrenik, and V. Filkov, "A Data Set for Social Diversity Studies of GitHub Teams," in *Proceedings of the 12th Working Conference on Mining Software Repositories*, 2015, pp. 514–517.

[9] B. Vasilescu, V. Filkov, and A. Serebrenik, "StackOverflow and GitHub: Associations between Software Development and Crowdsourced Knowledge," in *2013 International Conference on Social Computing*, 2013, pp. 188–195.

[10] Wikipedia, "List of programming languages," https://en.wikipedia.org/wiki/List_of_programming_languages, 2019-02-03, [Online; accessed 28-October-2018].