Tracking Hackathon Code Creation and Reuse

Ahmed Imam
University of Tartu
Estonia
ahmed.imam.mahmoud@ut.ee

Tapajit Dey
Lero—the Irish Software Research
Centre, University of Limerick
Limerick, Ireland
tapajit.dey@lero.ie

***Abstract*—Background: Hackathons have become popular events for teams to collaborate on projects and develop software prototypes. Most existing research focuses on activities during an event with limited attention to the evolution of the code brought to or created during a hackathon. Aim: We aim to understand the evolution of hackathon-related code, specifically, how much hackathon teams rely on pre-existing code or how much new code they develop during a hackathon. Moreover, we aim to understand if and where that code gets reused. Method: We collected information about 22,183 hackathon projects from DEVPOST– a hackathon database – and obtained related code (blobs), authors, and project characteristics from the WORLD OF CODE. We investigated if code blobs in hackathon projects were created before, during, or after an event by identifying the original blob creation date and author, and also checked if the original author was a hackathon project member. We tracked code reuse by first identifying all commits containing blobs created during an event before determining all projects that contain those commits. Result: While only approximately 9.14% of the code blobs are created during hackathons, this amount is still significant considering time and member constraints of such events. Approximately a third of these code blobs get reused in other projects. Conclusion: Our study demonstrates to what extent pre-existing code is used and new code is created during a hackathon and how much of it is reused elsewhere afterwards. Our findings help to better understand code reuse as a phenomenon and the role of hackathons in this context and can serve as a starting point for further studies in this area.**

***Index Terms*—Hackathon, Code Reuse, Repository Mining, Commits, Blob Reuse**

I. Introduction

Hackathons are time-bounded events during which individuals form – often ad-hoc – teams and engage in intensive collaboration to complete a project that is of interest to them [1]. Most hackathon projects focus on creating a prototype that can be presented at the end of an event [2]. This prototype often takes the form of a piece of software. The creation of software code can, in fact, be considered as one of the main motivations for organizers to run a hackathon event. Scientific and open source communities, in particular, organize such events with the aim to expand their code base [3], [4]. It thus appears surprising that the evolution of the code used and developed during a hackathon has not been studied yet, as revealed by a review of existing literature. In our paper, we aim to address this knowledge gap by studying 22,183 hackathon projects, identified usingDEVPOST, by leveraging WORLD OF CODE, a dataset of almost all open source projects.

Complete results of an extension of this hackathon project is available at [5], and **the replication package** for our study is available at [6].

II. Research questions

In order to address the knowledge gap mentioned earlier, in this hackathon project we aimed to study the evolution of the code used and created by the hackathon team members from two main perspectives. First, we studied where the code *originates*: While teams will certainly develop original code during the hackathon, it can be expected that they will also utilize existing (open source) code as well as code that they might have created themselves prior to the event, so our first research question that addresses the topic:

RQ$_1$. *Where does hackathon code come from?*

In particular, we focused on the sub-questions:

RQ$_{1a}$. *When was the code created?*

RQ$_{1b}$. *Who were the original creators of the code?*

Second, to understand the impact of hackathon code, i.e. code created during a hackathon event by the hackathon team in the hackathon project repository, on the wider software development community, we aimed to study whether and how it *propagates* after the event has ended. As noted in section I, existing studies do not address the question of whether and where hackathon code gets reused after an event has ended. In fact, hackathons are widely considered as "one-off" events by many. Knowing the answer to this question, thus, would be crucial for understanding the impact of hackathons on the larger open source community. This leads us to also asking the following second research question:

RQ$_2$. *What happens to hackathon code after the event?*

III. Methodology and Results

To address our research questions, we conducted an archival analysis of the source code utilized and developed in the context of 22,183 hackathon projects that were listed in the hackathon database DEVPOST[1]. To track the origin of the code that was used and developed by each hackathon project and study its reuse after an event has ended, we leveraged the open-source database WORLD OF CODE [7], [8], the primary focus of this hackathon event, which allowed us to track the origin of hackathon code and code reuse across almost all open source repositories. In our study, we focused on blob-level code reuse.

[1] https://devpost.com/

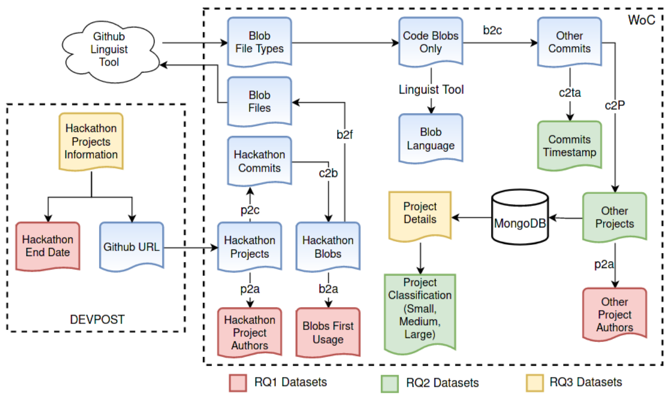

Fig. 1. Data Collection Workflow: Highlighting the different data sources used and the process of gathering the required information from them, and the data used in answering our research questions

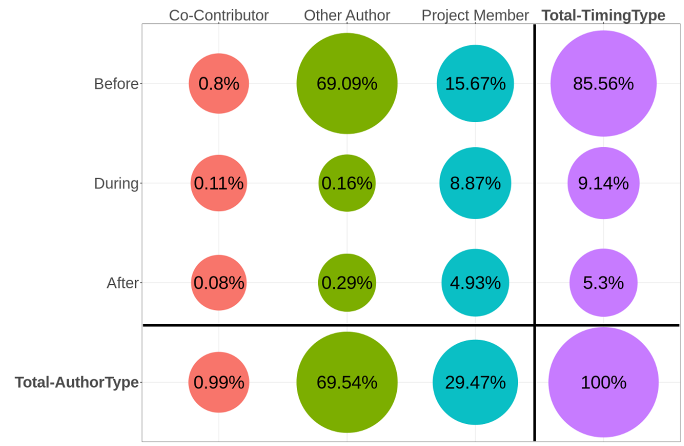

Fig. 2. Plot depicting what percentage of the hackathon code was created by whom and when

A. Data Sources

DEVPOST is a popular hackathon database that is used by corporations, universities, civic engagement groups and others to advertise events and attract participants. It contains data about hackathons including hackathon locations, dates, prizes and information about teams and their projects including the project's GITHUB repositories. It was our primary source for identifying the "hackathons".

However, DEVPOST doesn't have all the information we need for answering our research questions, so we leveraged the WORLD OF CODE dataset for gathering additional information about the projects, authors, commits, and code blobs.

B. Data Collection and Analysis

Here we describe how we collected the data required for answering our research questions, along with details of all the filtering we introduced. An overview of the approach is shown in fig. 1, which also highlights the different data sources and what data was used for answering each research question.

1) Steps to answer RQ1: Our starting point was a list of hackathon projects from DEVPOST and using the GITHUB URLs for the projects, we mapped them to the project names in WORLD OF CODE, which are formatted as `GitHubUserName_RepoName`. We started by collecting all the commits for these projects using *p2c* map, then we collected all the blobs in those commits using *c2b* map. We were also interested in identifying the authors so we collected the authors for the hackathon projects using *p2a* map and the first author who introduced each blob along with the timestamp when the blob was first used using *b2a* map.

It is common in projects to not only have code files but also other file types like images, data, markup, and prose files, so we collected the file information of each blob using *b2f* map which was important to identify non-code blobs and filter them out during our analysis. We utilized the *linguist* tool from GITHUB to find out the file types of the blobs and we filtered only code blobs.

In order to answer our first research question, we started by understanding **"when"** the blob is first used with compared with the hackathon event start and end dates. The DEVPOST dataset doesn't include the start date, so we derived the start date from the end date by assuming the duration of hackathon events of 72 hours which appears reasonable since hackathons are commonly hosted over a period of 48 which are often distributed over three days [9], [10]. We then compared the first timestamp of each blob with the hackathon event start and end dates and we classified the blobs based on time to before, during, and after the hackathon event.

Since we are also interested in **"who"** are the original creators of the blobs and their connection with the hackathon projects in question, we checked if the original author of a hackathon code blob was part of the hackathon team or not. We also wanted to understand if any of the hackathon project members were part of the other projects where the blobs were first introduced (we call the code creators as "co-contributors" in such cases). Since we have the first commit which introduced this blob from the *b2a* map, we collected the projects for these commits using *c2P* map and then the author list of these projects using *p2a* map. We used the approach outlined by [11] for author ID disambiguation to merge all of the different IDs belonging to one developer together, which is a common occurrence, as discussed in [12].

Figure 2 shows the results of our analysis for RQ1, highlighting that 85.56% of the code (in terms of the no. of blobs) in the hackathon project repositories is created before the hackathons, with around 9.14% of the code being created during the events (which is significant considering the duration and team member constraints of the hackathons). Moreover, The members of the hackathon teams created around 29.47% of the code blobs, while 69.54% of the code blobs are created by developers outside the team (mostly authors of some project/package/framework used by the team).

> ***Origin of the Hackathon Code (RQ1)****: Hackathon projects often reuse code in terms of some package/framework. Teams also tend to reuse their own code. Most of the code created during or after the event is created by the hackathon team members.*

2) Steps to answer RQ2: Here our goal was to understand how the hackathon-generated code gets reused after the hackathon event, so our starting point was the result from the RQ1 analysis since we used that as a base for filtering and answering RQ2. We applied a filter to blobs that satisfy two conditions: (a) Blobs are created during the hackathon event and (b) Blobs are created by hackathon project team members. Once these blobs are identified, we start collecting the commits that use these blobs using *b2c* map, and we collected the commit timestamps using *c2ta* map. We also used the project information dataset from a Mongo Database associated with WORLD OF CODE to identify the project size using two variables *(numAuthors, numStars)* which are indications of project size and popularity and were found to have a low correlation (Spearman Correlation: 0.26). We used *Hartemink's pairwise mutual information-based discretization method* [13], which was applied to a dataset with log-transformed values of the number of stars and developers for the projects, to classify the projects into three categories: Small, Medium, and Large. 89.2% of the projects that reused the hackathon code blobs were classified as *Small*, 8.5% were *Medium*, and 2.3% were classified as *Large* projects.

Hackathon code reuse (RQ2): *Around 28.8% of hackathon code blobs got reused in other projects, with 57.73% of the code being used in Small projects, 32.85% in Medium projects, and 9.42% in Large projects. Most of the reused blobs were related to web/mobile apps/frameworks. The temporal dynamics of code reuse show a clear trend of it reducing over time, and then saturating to a stable value.*

IV. FUTURE WORK

There are several ways to extend this research, e.g. considering code clones/snippets while looking for code reuse (e.g. by looking at the associated CTAG tokens - a dataset available in WORLD OF CODE), identifying other factors that affect code reuse, including code quality [14], [15], project popularity [16], [17], the type of Open Source license used, etc. Looking deeper into the code created during the hackathons, it might also be interesting to see to what extent the teams use bots [18], [19] which might aid in the understanding of hackathon code reuse as well. We hope that further studies will explore these and other related topics, and give us a clearer understanding of the impact of hackathons and code reuse.

REFERENCES

[1] E. P. P. Pe-Than, A. Nolte, A. Filippova, C. Bird, S. Scallen, and J. D. Herbsleb, "Designing corporate hackathons with a purpose: The future of software development," *IEEE Software*, vol. 36, no. 1, pp. 15–22, 2019.

[2] M. A. Medina Angarita and A. Nolte, "What do we know about hackathon outcomes and how to support them? - a systematic literature review," in *Collaboration Technologies and Social Computing*. Springer, 2020.

[3] E. P. P. Pe-Than and J. D. Herbsleb, "Understanding hackathons for science: Collaboration, affordances, and outcomes," in *International Conference on Information*. Springer, 2019, pp. 27–37.

[4] A. Stoltzfus, M. Rosenberg, H. Lapp, A. Budd, K. Cranston, E. Pontelli, S. Oliver, and R. A. Vos, "Community and code: Nine lessons from nine nescent hackathons," *F1000Research*, vol. 6, 2017.

[5] A. Imam, T. Dey, A. Nolte, A. Mockus, and J. D. Herbsleb, "The secret life of hackathon code," *arXiv preprint arXiv:2103.01145*, 2021.

[6] "Replication package," https://github.com/woc-hack/track_hack.

[7] Y. Ma, C. Bogart, S. Amreen, R. Zaretzki, and A. Mockus, "World of code: an infrastructure for mining the universe of open source vcs data," in *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, 2019, pp. 143–154.

[8] Y. Ma, T. Dey, C. Bogart, S. Amreen, M. Valiev, A. Tutko, D. Kennard, R. Zaretzki, and A. Mockus, "World of code: Enabling a research workflow for mining and analyzing the universe of open source vcs data," *arXiv preprint arXiv:2010.16196*, 2020. [Online]. Available: https://arxiv.org/pdf/2010.16196

[9] A. Nolte, E. P. P. Pe-Than, A.-A. O. Affia, C. Chaihirunkarn, A. Filippova, A. Kalyanasundaram, M. A. M. Angarita, E. H. Trainer, and J. D. Herbsleb, "How to organize a hackathon - a planning kit," *ArXiv*, vol. abs/2008.08025, 2020.

[10] D. Cobham, K. Jacques, C. Gowan, J. Laurel, S. Ringham *et al.*, "From appfest to entrepreneurs: using a hackathon event to seed a university student-led enterprise," in *11th annual International Technology, Education and Development Conference*, 2017.

[11] T. Fry, T. Dey, A. Karnauch, and A. Mockus, "A dataset and an approach for identity resolution of 38 million author ids extracted from 2b git commits," in *Proceedings of the 17th International Conference on Mining Software Repositories*, ser. MSR '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 518–522. [Online]. Available: https://doi.org/10.1145/3379597.3387500

[12] T. Dey, A. Karnauch, and A. Mockus, "Representation of developer expertise in open source software," *ArXiv*, vol. abs/2005.10176, 2020.

[13] A. J. Hartemink, "Principled computational methods for the validation discovery of genetic regulatory networks," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[14] T. Dey and A. Mockus, "Modeling relationship between post-release faults and usage in mobile software," in *Proceedings of the 14th International Conference on Predictive Models and Data Analytics in Software Engineering*, ser. PROMISE'18. New York, NY, USA: Association for Computing Machinery, 2018, p. 56–65. [Online]. Available: https://doi.org/10.1145/3273934.3273941

[15] T. Dey and A. Mockus, "Deriving a usage-independent software quality metric," *Empirical Software Engineering*, vol. 25, no. 2, pp. 1596–1641, Mar 2020. [Online]. Available: https://doi.org/10.1007/s10664-019-09791-w

[16] T. Dey and A. Mockus, "Are software dependency supply chain metrics useful in predicting change of popularity of npm packages?" in *Proceedings of the 14th International Conference on Predictive Models and Data Analytics in Software Engineering*, ser. PROMISE'18. New York, NY, USA: Association for Computing Machinery, 2018, p. 66–69. [Online]. Available: https://doi.org/10.1145/3273934.3273942

[17] T. Dey, Y. Ma, and A. Mockus, "Patterns of effort contribution and demand and user classification based on participation patterns in npm ecosystem," in *Proceedings of the Fifteenth International Conference on Predictive Models and Data Analytics in Software Engineering*, ser. PROMISE'19. New York, NY, USA: Association for Computing Machinery, 2019, p. 36–45. [Online]. Available: https://doi.org/10.1145/3345629.3345634

[18] T. Dey, S. Mousavi, E. Ponce, T. Fry, B. Vasilescu, A. Filippova, and A. Mockus, "Detecting and characterizing bots that commit code," in *Proceedings of the 17th International Conference on Mining Software Repositories*, ser. MSR '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 209–219.

[19] T. Dey, B. Vasilescu, and A. Mockus, "An exploratory study of bot commits," in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ser. ICSEW'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 61–65.