Distinguishing Bots From Human Developers Based on Their GitHub Activity Types

Natarajan Chidambaram[1], Alexandre Decan[1,2] and Tom Mens[1]

[1]University of Mons (UMONS), Mons, Belgium
[2]F.R.S.-FNRS Research Associate

Abstract

Development bots are being used by maintainers of GitHub repositories to perform repetitive or error-prone tasks. While multiple approaches have been proposed in the past to identify such bots in GitHub repositories, they are either inaccurate, not general enough (e.g., based exclusively on specific types of activities) or require a large amount of data that makes it difficult to use them in practice. In this paper, we show that information about the activities performed by contributors in the different software repositories to which they contribute can be used as a basis to distinguish bots from human contributors. These activities can belong to one of 24 distinct activity types such as *deleting a branch*, *creating a tag*, *publishing a release*, *opening an issue*, *commenting on a pull request*, and so on. Relying on a dataset of 830K+ activities performed by 713 contributors to GitHub repositories, we identify a number of features allowing to distinguish bots from human contributors. These features are the number of activity types, distribution of inequality in the time taken to perform these activities, distribution of inequality in the number of activity types performed in each repository, ratio between the number of owners and repositories, and the median time taken to switch between repositories to perform another activity. In follow-up work we aim to leverage these features to provide a bot identification model that considers all the activities performed by contributors.

Keywords

bot activity, collaborative software development, distinguish bot and human

1. Introduction

The use of collaborative software development practices in social coding platforms (such as GitHub, GitLab, Gitea and the likes) has become omnipresent [1, 2]. These social coding platforms allow project contributors to perform various activity types in their software repositories, such as *opening an issue*, *opening a pull request*, *commenting on a pull request*, *pushing a commit* and *reviewing code*.

Social coding platforms also enabled the use of development *bots* that, just like human contributors, can perform various activities in software repositories [3, 4, 5]. The main distinction is that their activities are automated, typically based on some external trigger or schedule. Bots enable the automation of error-prone and repetitive tasks to allow human contributors to focus on other more cognitive activities [5]. Some *bots* are even among the top contributors in the software repositories to which they contribute [6].

While the use of bots in open source software repositories can alleviate maintainer workload, their presence poses challenges for empirical software engineering researchers that aim to study socio-technical aspects of software development. A prerequisite for considering bots is the ability to identify their presence in software development activities. While there exists several bot detection models such as BIMAN [7], BoDeGHa [8] or BotHunter [9], they are usually based on limited number of activity types and features (e.g., repetitive patterns in the comments they create in issues and pull requests, commmit messages, or the presence of *bot* in their name) or they require a significant amount of data to reliably distinguish bots from humans in practice.

In this article, we identify a series of features to quantitatively distinguish bots from human contributors based on relevant information such as their activity types, the number of repositories they are involved in, the time it takes to carry out or shift between activity types, and so on. Such a set of distinguishing features will be used to create a bot identification model to efficiently and reliably identify whether a repository contributor is a bot or a human based on the output of a single call to the GitHub Events API. The distinguishing features could also be used to classify bots into different categories (i.e., different types of bots), to understand the role that bots are playing in collaborative software development, and the positive or negative impact they may have on the collaborative development process.

Leveraging on a GitHub activity dataset that we have created in earlier work [3], the identification of distinguishing features in the current article will be based on a dataset of 713 GitHub repository contributors, of which 305 bots and 408 humans, accounting for a total of 830K+

SATToSE'23: Seminar on Advanced Techniques & Tools for Software Evolution, June 12–14, 2023, Salerno, Italy
natarajan.chidambaram@umons.ac.be (N. Chidambaram); alexandre.decan@umons.ac.be (A. Decan); tom.mens@umons.ac.be (T. Mens)
0000-0002-5824-5823 (A. Decan); 0000-0003-3636-5020 (T. Mens)

activities belonging to 24 different activity types that are performed in 46,190 GitHub repositories from 25th November 2022 to 9th March 2023.

2. Related Work

On bot-based studies. Many empirical studies on bots have been conducted in the past. Some of these studies involved analysing the working style of bots [10], characterising bots [7], studying their impact in software projects [11], understanding the difference in response by software project maintainers to the activities performed by bot and human contributors [12], identifying the bot persona that would play a role in developers preference for a bot [13], and so on.

Erlenhov et al. [10] carried out a qualitative study to identify three different personas among software engineering bots focusing on autonomy, chat interfaces, and smartness. Dey et al. [7] characterized the bots that commit code into three types “Continuous Active Bots" (active uniformly over 24 hours), “Synchronous Active Bots" (active during working hours) and “Spike Activity Bots" (active for a few specific hours). They applied this characterization on 454 bots that made at least 1000 commits. Wessel et al. [11] qualitatively analyzed the unexpected impacts of adopting a code review bot in software projects. They interviewed 12 practitioners and concluded that adopting a code review bot increases the number of monthly merged pull requests, decreases monthly non-merged pull requests, and decreases communication among developers. Wyrich et al. [12] empirically analysed the difference in interaction between pull request (PR) created by bots and humans. By analysing the PR created/commented by 1.79M human and 4,654 bot contributors, they concluded that the PRs created by humans were answered faster and 72.53% of PRs were merged, whereas bots received a response after a longer time and only 37.38% were merged. Ghorbani et al. [13] qualitatively analysed the developers perception of bots used for pull request and identified seven themes that reflect this perception: attitude, autonomy, persona, task, feelings, project norm, and role. Through the interviews, they concluded that autonomy and persona exerts more influence on developers perception of bots and conducted surveys to further understand the influence that bot autonomy and persona would play in developers preference for a bot. Further, they gave recommendations to be adopted while designing development bots.

On bot identification tools. Several bot identification tools have been proposed to identify the presence of bots in software projects. Dey et al. [7] developed BIMAN that combines three different approaches to recognize bots in commits: (i) the presence of the string *bot* at the end of the author name, (ii) repetitive commit messages, and (iii) features related to files changed in commits. Golzadeh et al. [8] developed BoDeGHa that uses a classification model to identify bots that are involved in commenting issues and pull requests. They further extended this approach to git commit messages [14]. Chidambaram et al. [15] extended BoDeGHa by leveraging the predictions from multiple repositories to determine the type of account. Abdellatif et al. [9] developed BotHunter, that extends BoDeGHa by integrating additional information from the account’s profile and from the commits, issues and pull requests the account is involved in. We are not aware of any bot identification tool that consider all the activity types performed by a software project contributors, justifying the relevance of our current study that covers a wider range of activities.

All the above-mentioned tools and studies consider a *restricted* set of activity types (e.g., committing code, commenting on pull requests and issues) performed by bots in software repositories. Since bots perform many more activity types, considering all their activities would help to increase bot identification accuracy, while at the same time enabling the detection of bots that are performing tasks not considered by existing bot identification tools.

3. Methodology

3.1. Dataset

The goal of this article is to identify features allowing to distinguish between bots and human contributors based on the activities and activity types that they are performing in the GitHub software repositories they are contributing to.

To do so, in previous work we created a dataset of bot and human activities in GitHub [3]. The dataset was made publicly available on Zenodo [16] and contains 833K+ activities made by 385 bots and 616 humans. We identified 24 meaningful high-level GitHub activity types such as pushing commits, opening pull requests, commenting on issues, creating or deleting branches and so on [3]. This activity data was generated from events that were obtained by iteratively querying the GitHub Events API[1] from 25th November 2022 to 9th March 2023. Table 1 summarises this initial dataset.

Fig. 1 gives the proportion of contributors and the maximal number of activities performed by contributors present in the dataset. We can observe that the initial dataset contains some infrequent contributors that performed very few activities. Such contributors are unlikely to be helpful in determining distinguishing features between bots and humans, since they do not have a number of activities performed by them in software repositories

[1]https://docs.github.com/en/rest/activity/events

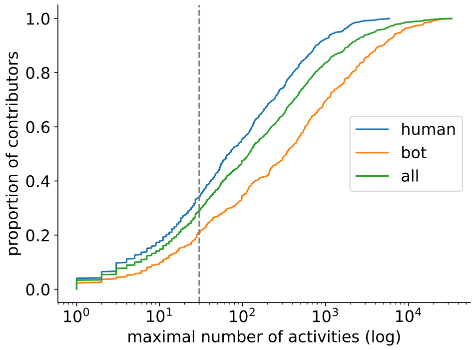

Figure 1: Maximal number of activities performed by the contributors (bots or humans). The vertical dashed line marks the minimum threshold of 30 activities per contributor that we used to include contributors in our analysis.

that is high enough to be considered. Hence, in this exploratory research, we decided to exclude such contributors by selecting a minimum threshold of 30 activities as inclusion criterion (i.e., at least roughly one activity every three days on average). The vertical dashed line in Fig. 1 indicates this threshold. In future work, we will evaluate the performance of the bot identification model in function of the number of considered activities.

Table 1

Initial and final dataset of considered contributors and activities, and top five activity types along with their number of activities in the final dataset

		dataset	
		initial	**final**
bot	**# contributors**	385	305
	# activities	649,755	648,752
human	**# contributors**	616	408
	# activities	184,056	181,751
total	**# contributors**	1,001	713
	# activities	833,811	830,503

Based on this threshold of 30 activities, we excluded 80 bots and 208 humans from the dataset, and their 3,308 corresponding activities. This left us with 305 bots and 408 humans, accounting for a total of 830,503 activities. Table 1 summarises the final dataset that will be used for our analysis.

3.2. Statistical method

Whenever appropriate, for the features that will be studied in Section 4 we carry out statistical tests to compare the distribution values for bots and human contributors, using the non-parametric Mann-Whitney U test (a.k.a. Wilcoxon rank-sum test) since most of the considered distributions are not normally distributed. We will reject the null hypothesis that the two distributions are equal using a significance level of $\alpha = 0.001$ after controlling for family-wise error rate using the Bonferroni-Holm method [17]. For each test for which the null hypothesis can be rejected, we also compute the effect size using Cliff's delta (δ) [18]. Following the interpretation by Romano et al. [19], we consider the effect size to be *negligible* if $\delta < 0.147$, *small* if $0.147 \leq \delta < 0.33$, *medium* if $0.33 \leq \delta < 0.474$ and *large* if $0.474 \leq \delta$.

When visualizing the distributions of some metrics, we will make use of boxen plots (rather than box plots) to provide a good representation of the (often skewed) distribution of the data [20]. In a boxen plot, data representation starts at the median value and extends (i.e., draws a level line) by half of the remaining data to be covered from the current level. For example, from 50% (median), the next level towards the top will be at (50+25)%, the next level at (50+25+12.5)% and so on, similarly on the lower side, the next level will be at (50-25)%, the next at (50-25-12.5)% and so on until it reaches the point where the outliers start. This facilitates the visualization and interpretation of skewed distributions. For similar reasons we hide the outliers in these boxen plots.

Finally, a couple of features are based on the statistical dispersion of some metric. We will rely on the well-known Gini coefficient to measure this dispersion [21]. The Gini coefficient is a widely used social and economic indicator to cope with unevenly distributed data. Its value is comprised between 0 and 1, where a value close to 0 reflects an equal distribution while a value close to 1 expresses a maximal inequality among individuals.

4. Features to Distinguish Bots and Humans

In this section, we present our intuitions on the differences between bot and human activities. Then, we propose a metric capturing each intuition and we show that this metric effectively differs for bots and humans.

4.1. Number of activity types

Intuition: Bots are involved in less activity types than humans.

We expect bots to be mostly involved in a specific set of activities. For example, a bot that keeps dependencies up-to-date will create pull requests only, and is unlikely to push commits directly to the repository or to comment some issue. On the other hand, we expect human contributors to perform a wider range activities across the repositories to which they contribute. For example, human contributors close or merge incoming pull requests in the repositories they maintain while they create pull requests and issues in the external repositories they contribute to. To verify this hypothesis, we computed for each contributor the number of activity types performed. Fig. 2 shows the distribution of this number of activity types, distinguishing between bot and human contributors.

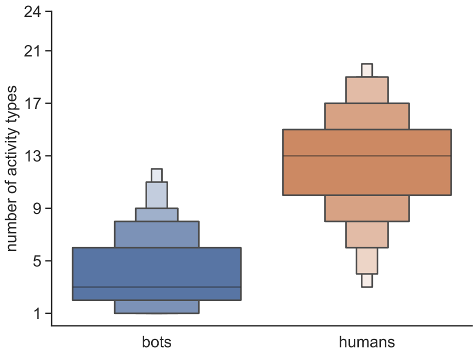

Figure 2: Boxen plots of the distribution of number of activity types performed by bots and human contributors

We observe a clear visual difference in line with our intuition, with a median number of 3 activity types for bots compared to 13 for humans. To confirm that there is a significant difference between the number of activity types performed by bots and human contributors, we performed a Mann-Whitney U test between the two distributions. The null hypothesis is rejected ($p < \alpha$) indicating that there is a statistically significant difference between the number of activity types performed by bots and human contributors in software repositories. The effect size turned out to be *large* (δ=0.88).

Conclusion: Bots tend to be involved in less activity types than human contributors.

4.2. Specialisation of activity types across repositories

Intuition: Bots are more consistent than humans in performing their intended activity types across the repositories they contribute to.

Not only bots are involved in less activity types than human contributors, but we expect them to be more consistent in performing these activity types across the multiple repositories to which they contribute compared to that of humans. Consider again the case of a bot keeping dependencies up-to-date. Such a bot, regardless of the repository it is deployed in, will consistently have one activity type across the repositories (i.e., creating pull requests). On the other hand, depending on the repository, a human contributor may be involved in a limited number of activity types (e.g., opening an issue) or a larger number of activity types (e.g., pushing commits, closing or merging pull requests, closing and commenting issues, creating releases).

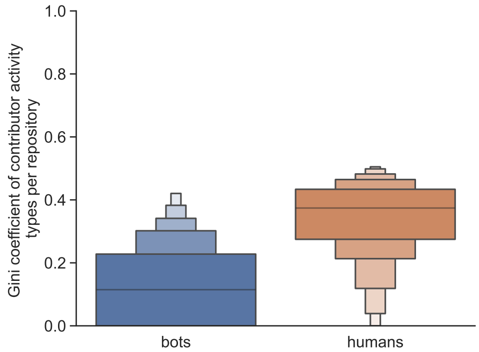

Figure 3: Boxen plots of the distribution of Gini coefficient for the number of activity types performed per repository by bot and human contributors

To capture this intuition, we computed for each contributor the Gini coefficient of its number of activity types across repositories. Fig. 3 shows the distribution of these Gini coefficients. One can visually observe that the number of activity types performed by bots are more equally distributed (i.e., Gini is closer to 0) across repositories than for human contributors. To statistically confirm our intuition, we performed a Mann-Whitney U test between both distributions. The null hypothesis was rejected ($p < \alpha$) indicating a statistical difference between the two populations, with a *large* effect size ($\delta = 0.76$).

Conclusion: The number of activity types performed by bots tend to be more equally distributed across repositories than the number of activity types performed by humans.

4.3. Number of repositories

Intuition: Bots are active in more repositories than humans, and tend to be used within repositories belonging to the same organizations/owners.

We expect bots to be active in more repositories than humans as they do not have any workload restrictions. We also expect bots to be active in more repositories belonging to the same owner or organization since it would make sense to deploy a bot in all the repositories of the same organization or owner as long as the bot is satisfactory. For humans, on the other hand, we expect to observe a more diverse behaviour, in the sense that human contributors might choose to go outside the boundaries of the repository owner or of its organization in order to contribute to external repositories owned by another user or organization.

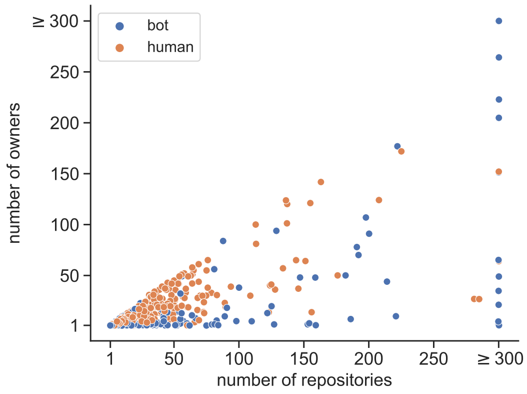

Figure 4: Number of repositories versus number of owners that the contributors are involved in. A jitter of 0.25 is applied on both the axes.

For each contributor, we computed the number of repositories and the number of distinct owners (or organizations) the contributor is active in. Fig. 4 shows a scatter plot of the number of repositories versus the number of distinct owners. To ease visualizing the contributors, we added a jitter of 0.25 on both axes, and we grouped contributors that are active in more than 300 repositories or owners. Overall, 21 bots and 2 human contributors are involved in more than 300 repositories, and 10 of these bots (and no human contributor) are involved in repositories belonging to more than 300 distinct owners or organizations. On the other hand, among the 158 contributors (i.e., 22.16%) that are active in repositories belonging to a single owner or organization, 88% are bots. Among the 81 contributors (i.e., 11.36%) active in repositories belonging to exactly two owners, 72% are bots. This suggests that bots tend to be involved in repositories belonging to a smaller number of distinct owners.

Since the number of distinct owners is upper bounded by the number of repositories, and since the number of repositories is upper bounded by the number of activities (one cannot be active in more repositories than its number of activities), and in order to capture the differences observed from Fig. 4, we computed for each contributor its *owner ratio* as the ratio between the number of distinct owners and the number of repositories the contributor is active in. This is, owner ratio $= \frac{\#\text{owners}}{\#\text{repositories}}$. An *owner ratio* close to 1 indicates that nearly all the repositories in which the contributor is active in belong to different owners or organizations. On the other hand, a ratio close to 0 indicates that most of the repositories the contributor is active in belong to the same owner or organization.

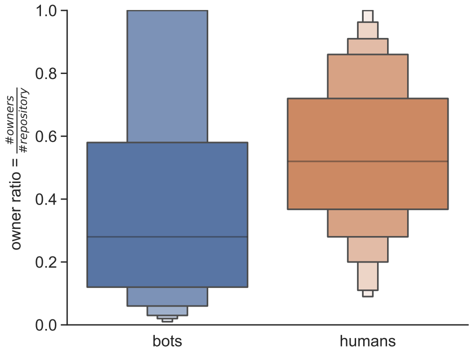

Figure 5: Boxen plots of the distribution of owner ratio between bots and humans

Fig. 5 shows the distribution of this owner ratio for bots and human contributors. We observe that the ratio is higher for humans than for bots, indicating that human contributors tend to be involved in repositories belonging to different owners or organizations, while bots tend to be involved in repositories belonging to a more limited set of owners and organizations. We confirmed this difference between the owner ratio of bots and humans by performing a Mann-Whitney U test. The null hypothesis was rejected ($p < \alpha$) with a *medium* effect size ($\delta = 0.33$), therefore indicating a statistically significant difference between the owner ratio of bots and humans.

Conclusion: The owner ratio is lower for bots than humans, indicating that humans tend to be active in repositories belonging to wider range of owners or organizations than bots.

4.4. Time to switch between repositories

Intuition: Bots can switch between repositories faster than humans.

The above intuition is based on the idea that bots are not suffering from context switching to the same extent as human contributors. As automated tools, bots can easily be active in multiple repositories at the same time, while we expect human contributors to focus their workload on one repository at a time. As a consequence, we expect the time required to go from one repository to another one to be greater for human contributors than for bots.

To capture this intuition, we computed for each contributor the time to switch between two repositories.

More specifically, we computed the time difference between any two consecutive activities made in two distinct repositories. Since a contributor can switch between repositories multiple times, we aggregated these time differences for each contributor by computing the median value of these differences.

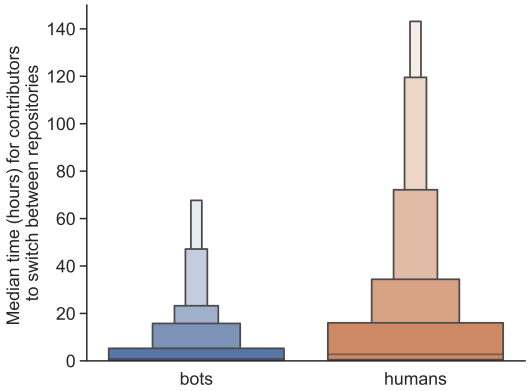

Figure 6: Boxen plots of the distribution of the median time (in hours) taken by bots and humans to switch between repositories

Fig. 6 shows the distribution of the median time (in hours) taken by contributors to switch from one repository to another. The figure indicates that bots usually take less time to switch between repositories compared to human contributors. For instance, the median values are 0.85 hours and 2.77 hours, respectively for bots and humans. To statistically confirm this difference, we performed a Mann-Whitney U test between the two distributions. The null hypothesis is rejected ($p < \alpha$) with a *small* effect size (δ = 0.28), confirming that bots take less time than human contributors to switch between repositories.

Conclusion: Bots take less time to switch between repositories.

4.5. Time between consecutive activities

Intuition: Bots tend to perform activities more regularly than humans.

Bots are not subject to the same limitations as human contributors in performing their tasks. For example, bots do not have to sleep or eat, and can afford to work at any time of the day. Their tasks are usually triggered by external events (such as a newer version of a dependency) or by activities made by other developers (e.g., a pull request is submitted to the repository and a bot evaluates its code quality). Since developers can be spread around the world, a bot does not really have a fixed schedule. In contrast, humans are more likely to concentrate their work during the day (or during the evenings or the weekends if they are not professional developers). As a result, we expect the work rhythm (i.e. the time between two consecutive activities) to be much more regular for bots than for humans. To capture this intuition, and to measure the regularity of the activities of each contributor, we computed the Gini coefficient of the time difference between consecutive activities. The lower the value, the more the contributor carries out activities on regular time intervals.

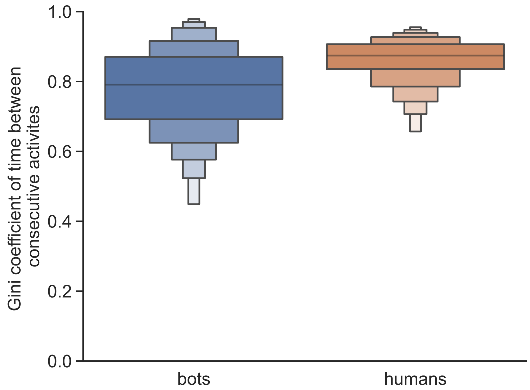

Figure 7: Boxen plots of the distribution of Gini coefficient of the time between consecutive activities

Fig. 7 shows the distribution of these Gini coefficients, distinguishing between bot and human contributors. We observe that, as expected, the Gini coefficient of the time between consecutive activities is higher for humans than bots, indicating that humans carry out their activities on a less regular basis than bots. We performed a Mann-Whitney U test to see whether the two populations exhibit a statistically significant difference. The null hypothesis was rejected ($p < \alpha$) with a *medium* effect size (δ = 0.42), confirming the observed difference.

Conclusion: Bots perform their activities on a more regular basis than human contributors.

5. Conclusion

Bots are used in software repositories to perform automation of error-prone and repetitive tasks, and they are frequently among the top contributors in some software repositories. While the use of bots in open source software repositories can alleviate maintainer workload, their presence poses challenges for empirical software engineering researchers that aim to study socio-technical aspects of software development. A prerequisite for considering bots is the ability to identify their presence in software development activities. This paper is a first step towards building a bot identification model taking into account the various activities that contributors do on GitHub.

In this paper, based on a dataset of 830K+ activities made by 305 bots and 408 human contributors, we proposed five features that can be used to distinguish bot and human contributors, and we showed through statistical tests that these features effectively differ between these two types of contributors. We found that bots tend to be specialized in the sense they are involved in a smaller number of activity types than humans. We showed that bots tend to equally distribute their intended activity types across multiples repositories, and that bots are usually involved in repositories belonging to a smaller set of distinct owners or organizations. We also found that bots take less time to switch between the repositories they are involved in and that they tend to perform their activities more regularly than humans do.

In future work, we plan to rely on these features to create and evaluate a classification model distinguishing between bots and humans based on their activities. Since contributor activities are obtained from the events returned by the GitHub Events API, and since this API returns at most 300 events, we will also evaluate the impact of the number of activities that need to be provided to the model on its accuracy. If it turns out that the model is still accurate enough based on at most 300 activities, we will implement the model as part of a tool that can be used in practice on a large scale without requiring access to other data sources than a simple call to the GitHub API.

6. Acknowledgments

This work is supported by DigitalWallonia4.AI research project ARIAC (grant number 2010235), as well as by the Fonds de la Recherche Scientifique – FNRS under grant numbers F.4515.23, O.0157.18F-RG43 and T.0149.22.

References

[1] L. Dabbish, C. Stuart, J. Tsay, J. Herbsleb, Social coding in GitHub: Transparency and collaboration in an open software repository, in: International Conference on Computer Supported Cooperative Work (CSCW), ACM, 2012, pp. 1277–1286. doi:10.1145/2145204.2145396.

[2] J. M. Costa, M. Cataldo, C. R. de Souza, The scale and evolution of coordination needs in large-scale distributed projects: implications for the future generation of collaborative tools, in: SIGCHI Conference on Human Factors in Computing Systems, ACM, 2011, pp. 3151–3160. doi:10.1145/1978942.1979409.

[3] N. Chidambaram, A. Decan, T. Mens, A dataset of bot and human activities in GitHub, in: International Conference on Mining Software Repositories (MSR), IEEE, 2023.

[4] Z. Wang, Y. Wang, D. Redmiles, From specialized mechanics to project butlers: The usage of bots in open source software development, IEEE Software 39 (2022) 38–43. doi:10.1109/MS.2022.3180297.

[5] M. Wessel, B. M. De Souza, I. Steinmacher, I. S. Wiese, I. Polato, A. P. Chaves, M. A. Gerosa, The power of bots: Understanding bots in OSS projects, International Conference on Human-Computer Interaction (CHI) (2018). doi:10.1145/3274451.

[6] M. Golzadeh, T. Mens, A. Decan, E. Constantinou, N. Chidambaram, Recognizing bot activity in collaborative software development, IEEE Software 39 (2022) 56–61. doi:10.1109/MS.2022.3178601.

[7] T. Dey, S. Mousavi, E. Ponce, T. Fry, B. Vasilescu, A. Filippova, A. Mockus, Detecting and characterizing bots that commit code, in: International Conference on Mining Software Repositories (MSR), ACM, 2020, pp. 209–219. doi:10.1145/3379597.3387478.

[8] M. Golzadeh, A. Decan, D. Legay, T. Mens, A ground-truth dataset and classification model for detecting bots in GitHub issue and PR comments, Journal of Systems and Software 175 (2021). doi:10.1016/j.jss.2021.110911.

[9] A. Abdellatif, M. Wessel, I. Steinmacher, M. A. Gerosa, E. Shihab, BotHunter: An approach to detect software bots in GitHub, in: International Conference on Mining Software Repositories (MSR), 2022, pp. 6–17. doi:10.1145/3524842.3527959.

[10] L. Erlenhov, F. G. d. O. Neto, P. Leitner, An empirical study of bots in software development: Characteristics and challenges from a practitioner's perspective, in: Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE), ACM, 2020, pp. 445–455. doi:10.1145/3368089.3409680.

[11] M. Wessel, A. Serebrenik, I. Wiese, I. Steinmacher, M. A. Gerosa, Quality gatekeepers: investigating the effects of code review bots on pull request activities, Empirical Software Engineering 27 (2022) 108. doi:10.1007/s10664-022-10130-9.

[12] M. Wyrich, R. Ghit, T. Haller, C. Müller, Bots don't mind waiting, do they? comparing the interaction with automatically and manually created pull requests, in: International Workshop on Bots

in Software Engineering (BotSE), 2021, pp. 6–10. doi:10.1109/BotSE52550.2021.00009.

[13] A. Ghorbani, N. Cassee, D. Robinson, A. Alami, N. A. Ernst, A. Serebrenik, A. Wasowski, Autonomy is an acquired taste: Exploring developer preferences for GitHub bots, in: International Conference on Software Engineering (ICSE), IEEE, 2023.

[14] M. Golzadeh, A. Decan, T. Mens, Evaluating a bot detection model on git commit messages, in: Belgium-Netherlands Software Evolution Workshop (BENEVOL), volume 2912, CEUR Workshop Proceedings, 2020.

[15] N. Chidambaram, A. Decan, M. Golzadeh, Leveraging predictions from multiple repositories to improve bot detection, in: International Workshop on Bots in Software Engineering (BotSE), IEEE, 2022. doi:10.1145/3528228.3528403.

[16] N. Chidambaram, A. Decan, T. Mens, A dataset of bot and human activities in GitHub, 2023. doi:10.5281/zenodo.7740521.

[17] S. Holm, A simple sequentially rejective multiple test procedure, Scandinavian Journal of Statistics 6 (1979) 65–70. URL: http://www.jstor.org/stable/4615733.

[18] N. Cliff, Dominance statistics: Ordinal analyses to answer ordinal questions, Psychological bulletin 114 (1993) 494. doi:10.1037/0033-2909.114.3.494.

[19] J. Romano, J. D. Kromrey, J. Coraggio, J. Skowronek, L. Devine, Exploring methods for evaluating group differences on the NSSE and other surveys: Are the t-test and Cohen’s d indices the most appropriate choices, in: Annual Meeting of the Southern Association for Institutional Research, 2006.

[20] H. Hofmann, H. Wickham, K. Kafadar, Letter-value plots: Boxplots for large data, Journal of Computational and Graphical Statistics 26 (2017) 469–477. doi:10.1080/10618600.2017.1305277.

[21] R. Dorfman, A formula for the gini coefficient, The Review of Economics and Statistics 61 (1979) 146–149. doi:10.2307/1924845.