Leveraging Predictions From Multiple Repositories to Improve Bot Detection

Natarajan Chidambaram*, Alexandre Decan*, Mehdi Golzadeh*
*Software Engineering Lab, University of Mons, Belgium
Email: {natarajan.chidambaram,alexandre.decan,mehdi.golzadeh}@umons.ac.be

***Abstract*—Contemporary social coding platforms such as GitHub facilitate collaborative distributed software development. Developers engaged in these platforms often use machine accounts (bots) for automating effort-intensive or repetitive activities. Determining whether a contributor corresponds to a bot or a human account is important in socio-technical studies, for example to assess the positive and negative impact of using bots, analyze the evolution of bots and their usage, identify top human contributors, and so on. BoDeGHa is one of the bot detection tools that have been proposed in the literature. It relies on comment activity within a single repository to predict whether an account is driven by a bot or by a human. This paper presents preliminary results on how the effectiveness of BoDeGHa can be improved by combining the predictions obtained from many repositories at once. We found that doing this not only increases the number of cases for which a prediction can be made, but that many diverging predictions can be fixed this way. These promising, albeit preliminary, results suggest that the "wisdom of the crowd" principle can improve the effectiveness of bot detection tools.**

I. Introduction

Social coding platforms like GitHub promote collaboration and interaction between contributors [1]. Along with this opportunity for engagement, developers also face some workload in performing error-prone and repetitive task such as conducting regular dependency checks, deploying release, testing code, reviewing code, merging pull requests and so on [2]. While small projects can thrive under the guidance of a lone developer [3], the complexity and workload of larger projects makes it difficult to keep up with the pace of maintaining high-quality software releases. To reduce this workload and to automate effort-intensive and repetitive activities, automated tools/bots (machine accounts that works without human intervention) are used [4]–[10]. These bots might also influence the software development process, either positively or negatively, depending on the role that they are assigned and the way that they are being used [11].

Identifying the presence of these bots is not only useful for researchers conducting socio-technical studies but also for practitioners and funding organizations to identify contributors and to accredit them. The research literature already lists a few approaches to identify bots in software repositories, such as BIMAN [12], BoDeGHa [13] or BoDeGic [14]. BIMAN [12] 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. BoDeGHa [13] analyses comments posted in issues and pull requests to detect bots, based on the assumption that bots tend to frequently use a limited set of comment patterns. BoDeGic [14] transposes this approach to commit messages, assuming that bots tend to have a limited set of commit message patterns. Golzadeh et al. [15] proposed a probabilistic model based on NLP techniques to detect bot activity at the level of individual comment in issues and pull requests. In a recent work [16], they also compared 5 different approaches to detect bots in software repositories, including the above ones, and found that none of them is accurate enough to capture all bots.

Our goal is to improve the detection of bots active in issue and pull request comments, that is, to improve BoDeGHa. BoDeGHa is a tool that, given the name of a GitHub repository, predicts for each contributor with enough activity in the repository whether this contributor corresponds to a *bot* or a *human* contributor. If a contributor has not made enough comments, BoDeGHa classifies it as *unknown*.

Although BoDeGHa has been shown to perform well in detecting bots [13], it may still wrongly classify some contributors. Because BoDeGHa works at the repository level, this means that a same contributor active in multiple repositories may lead to diverging predictions, this is, it may be classified as *bot* in some repositories and as *human* in some other ones. For example, while BoDeGHa identifies the well-known *dependabot* bot correctly in many different repositories, it identifies it as a *human* contributor in *artichoke/rand_mt* because the 24 comments made by *dependabot* in this repository exhibit 10 different comment patterns, corresponding to the behaviour usually observed for human contributors. At the same time, BoDeGHa classifies the same bot as *unknown* in *cossacklabs/themis* because it only has 9 comments in this repository. Similarly, a human contributor can be sometimes classified as a bot. For example, in the GitHub repository *rust-lang/libc* we found a human contributor[1] that is detected as *bot* because most of his/her comments follow a single comment pattern of the form "*bors r+*". On the other hand, this contributor is correctly classified as *human* in *crossbeam-rs/crossbeam* and *rust-lang/rust* for example.

In this paper, we investigate how frequently such situations occur in GitHub repositories. We quantify how frequently do contributors have diverging predictions (that is, predicted as *bot* and *human* by BoDeGHa), and how frequently they have incomplete predictions (that is, predicted as *unknown*

[1] Name is hidden to comply with GDPR regulations.

by BoDeGHa). We provide preliminary insights on a novel approach to improve the accuracy of BoDeGHa by leveraging predictions from multiple repositories. We evaluate to which extent diverging and incomplete predictions can be fixed based on the wisdom of the crowd principle. More specifically, we address the following research questions:

RQ0: How frequently are contributors active in multiple repositories? We observe that one third of the contributors are active in multiple repositories.

RQ1: How frequently do contributors have diverging or incomplete predictions? More than half of the contributors identified at least once as bots have diverging or incomplete predictions.

RQ2: To which extent can we fix diverging predictions? We show that an approach based on the wisdom of the crowd principle is effective at fixing diverging predictions.

RQ3: To which extent can we complete predictions? We show that the same approach is promising to address incomplete predictions.

II. Dataset

BoDeGHa bot identification tool takes as input a GitHub repository and outputs whether the contributors in this repository correspond to bot or human contributors. Since our goal is to improve the performance of BoDeGHa by leveraging predictions from multiple repositories, we need a large collection of GitHub repositories having their contributors active in multiple repositories. Following the advice of Kalliamvakou et al. [17] we need to avoid repositories that have been created merely for experimental or personal reasons, or that only show sporadic traces of activity. Good candidate datasets are collections of repositories associated to the collaborative development of open source software packages for specific programming languages.

We collected the GitHub repositories associated to the software packages that are distributed through the Cargo package manager, for the Rust programming language. In October 2021, 68,621 Rust packages were available on Cargo and 38,886 of them (i.e., 56.7%) have an associated repository on GitHub. Since we need bots to be active in the repositories to conduct our empirical study, and since bots are more likely to be present in larger and more mature projects, we excluded packages that do not even refer to their homepage or to their documentation. This left us with 22,156 packages. Given that BoDeGHa relies on the comments made in issues and pull requests to identify bot contributors, we excluded repositories having less than 100 issues or pull requests. At the end of the data extraction process, the dataset contains 1,039 GitHub repositories accounting for 147,426 pairs of contributor/repository.

III. Findings

RQ0: How frequently are contributors active in multiple repositories?

Since we aim to improve bot detection by leveraging predictions made on multiple repositories, we need contributors to be active in more than a single repository. This question aims to quantify how frequently contributors are active in multiple repositories. The 147,426 pairs of contributor/repository in our dataset correspond to 57,757 distinct GitHub accounts, already indicating that some contributors are active in more than one repository. Only 8,532 contributors out of these 57K (14.8%) have enough commenting activity in at least one repository for BoDeGHa to be applied. For each of these 8,532 contributors (i.e., each distinct GitHub account), we counted the number of repositories that each contributor was active in. Table I reports on the number and proportion of contributors in function of the number of repositories they are active in.

TABLE I: Number and proportion of contributors in function of the number of repositories they are active in

# repositories →	1	2	3	4 or 5	6 - 9	10+
# contributors	5,671	1,530	496	385	239	211
% contributors	66.5%	17.9%	5.8%	4.5%	2.8%	2.5%

We observe that while most contributors (5,671 out of 8,532, 66.5%) are active in a single repository only, around one third of the contributors (2,861, i.e., 33.5%) are active in multiple repositories. We will focus on those 2,861 contributors since they correspond to those for which BoDeGHa will produce several, potentially diverging (i.e., *bot* and *human*) or incomplete (i.e., *unknown*) predictions. These 2,861 contributors are active in a total of 1,010 distinct repositories.

RQ1: How frequently do contributors have diverging or incomplete predictions?

We applied BoDeGHa on each of the 1,010 repositories identified in $RQ0$ in order to get the predictions for each of the 2,861 contributors active in two or more repositories. Under the hood, BoDeGHa downloads up to 100 pull request or issue comments for each contributor active in the repository. Only the comments made during the last five years (i.e., after December 2016) are considered. BoDeGHa then analyses these comments and predicts whether the contributor corresponds to a *bot* or a *human* contributor based on several features including the repetitiveness of comments and the number of comment patterns. If a contributor has less than 10 comments, BoDeGHa classifies it as *unknown*. At the end of this process, we have a total of 41,542 predictions of which 1,146 correspond to *bot*, 10,227 to *human* and 30,169 to *unknown*. The high proportion of *unknown* predictions (73%) indicates that most contributors have less than 10 comments in the considered repositories.

Since our focus is on improving bot detection, we select contributors that were classified *bot* at least once. Out of the initial 2,861 distinct contributors active in at least two repositories, 229 (8%) were classified *bot* at least once. Among them, 106 (46%) were consistently classified *bot* in all the repositories they were active in. Out of the 123 remaining contributors having been predicted as *bot* at least once, 60 have diverging predictions (i.e., they were also classified as *human*) and 63 have consistent but incomplete predictions (i.e., they were also classified as *unknown*).

To assess to which extent bot detection can be improved by leveraging predictions from multiple repositories, we need to determine the correct type (i.e., bot or human) of each account. Two authors of this paper manually and independently checked the 3,086 predictions for the 229 contributors that were at least once predicted as bot to determine their actual type, following an inter-rater agreement process. The first step of this process ended up with an agreement on 95% of the cases. The remaining ones were discussed together, ending up with an agreement on all of them. With this process, we found that BoDeGHa incorrectly predicted *bot* in 110 cases and incorrectly predicted *human* in 31 cases. Table II summarizes the number of actual bot and human contributors we found, as well as the number of *bot*, *human* and *unknown* predictions obtained for them.

TABLE II: Number of actual bot and human contributors, and their number of *bot*, *human* and *unknown* predictions

		predictions		
	contributors	# *bot*	# *human*	# *unknown*
actual bot	142	1,110	31	413
actual human	87	110	288	1,134
total	229	1,220	319	1,547

RQ2: To which extent can we fix diverging predictions?

Previous research question revealed that many contributors have different predictions depending on the repository BoDeGha is applied on. In this research question, we propose an approach based on the wisdom of the crowd principle to fix these diverging predictions. More specifically, if one assumes that BoDeGHa is more often correct than wrong in predictions, then, given a contributor having multiple predictions, we can assume that the most frequent prediction (either *bot* or *human*) is correct, while the less frequent one is not. Let WoC-P be such bot detection model. WoC-P stands for *Wisdom of the Crowd principle for Predictions* and works on top of BoDeGHa by automatically replacing the less frequent predictions of a contributor by the most frequent ones. Ties are arbitrarily resolved as *human*.

We applied both BoDeGHa and WoC-P on the 84 contributors that have at least two predictions of which one is *bot*. Figure 1 shows, for each contributor, the number of *human* predictions, the number of *bot* predictions, and whether it is an actual bot or human. To permit distinguish overlapping points, we added a jitter of 0.25 on both axes. The diagonal line illustrates the WoC-P model: any contributor above the line will be consistently predicted as a bot (i.e., the *human* predictions are replaced by *bot* predictions), while any contributor below will be consistently predicted as a human (i.e., the *bot* predictions are replaced by *human* predictions).

As can be observed from the figure, the approach proposed by WoC-P seems promising, most of the contributors having mostly predictions corresponding to their actual type. Only five human contributors have a higher number of *bot* predictions than *human* predictions. These contributors will be consistently but wrongly predicted as *bot* by WoC-P.

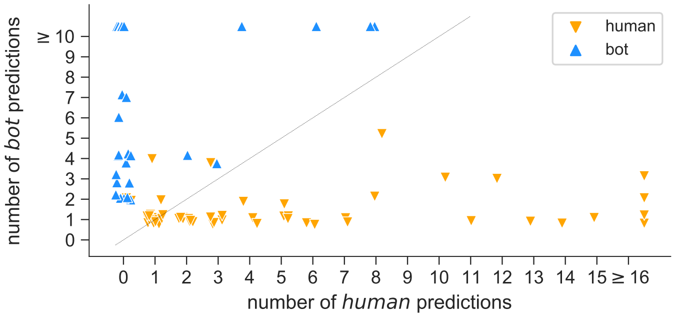

Fig. 1: Number of *bot* and *human* predictions, each point is a contributor

TABLE III: Score comparison between BoDeGHa and WoC-P

	TP	TN	FP	FN	Acc	Prec	Recall	F1
BoDeGHa	928	288	79	31	91.7	92.2	96.8	94.4
WoC-P	959	348	19	0	98.6	98.1	100.0	99.0

To assess to which extent BoDeGHa can be improved by WoC-P, we evaluated both models on the 84 contributors. Table III reports on the resulting number of true positives (TP), true negatives (TN), false positives (FP), false negatives (FN) as well as on the accuracy (Acc), precision (Prec), recall and F1 scores of the two models.

We observe that WoC-P actually improves the predictions made by BoDeGHa. WoC-P replaced a total of 101 predictions out of 1,326 (i.e., 7.6%): 65 *bot* predictions were correctly converted to *human* predictions, while 36 *human* predictions were converted to *bot* predictions, among which 31 correspond to actual bots. This leads the number of false negatives to drop from 31 to 0, and the number of false positives to decrease from 79 to 19. These 19 incorrect predictions correspond to the five human contributors above the diagonal line in Figure 1. As a consequence, WoC-P has higher accuracy, precision, recall and F1 scores compared to BoDeGHa.

RQ3: To which extent can we complete predictions?

So far, we relied on the wisdom of the crowd principle, using the most frequent prediction to fix the less frequent predictions. This question aims to determine whether a similar approach can be followed to fix *unknown* predictions as well.

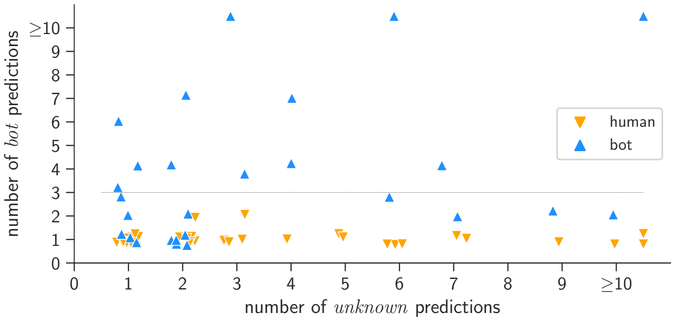

Fig. 2: Accounts predicted as bot/unknown vs ground truth

Figure 2 shows the number of *unknown* and *bot* predictions for the 63 contributors that were either predicted *bot* or *unknown* (i.e., that have no *human* predictions). We observe that the situation is more delicate than for $RQ2$. Indeed, many actual human contributors are among the contributors having only *bot* and *unknown* predictions. Converting the *unknown* predictions to *bot* predictions for these 33 human contributors would only increase the number of incorrect predictions for them. For instance, while converting the 184 *unknown* predictions of the 30 bots increases the number of correct predictions from 336 to 520, doing the same for the 158 *unknown* predictions of the 33 human contributors increases the number of incorrect predictions from 35 to 193.

Nevertheless, we observe that most of these human contributors have a low number of *bot* predictions compared to the actual bot contributors. For instance, there are 17 bots and no human having three or more *bot* predictions. On the other hand, all human contributors and "only" 13 bots have one or two *bot* predictions. Converting only the *unknown* predictions of contributors having three or more *bot* predictions would increase the number of correct predictions from 318 to 460, without increasing the number of incorrect predictions. However, since this threshold of "3+ *bot* predictions" is obtained by observation, it cannot be integrated into the WoC-P model without prior validation on another dataset.

IV. Conclusion

Bots are being more and more used to automate some of the effort-intensive and repetitive activities that are part of the software development process in GitHub repositories. Identifying bots is not only useful for researchers conducting socio-technical studies, but also for practitioners and funding organizations to identify contributors and to accredit them. BoDeGHa is one of the few tools that were proposed in the literature to detect bots in software repositories. Although, BoDeGHa works well at the repository level, it may still produce diverging or incomplete predictions for contributors being active in multiple repositories.

This paper gave preliminary insights on a novel approach based on the wisdom of the crowd to improve the performance of BoDeGHa. Based on a dataset of around one thousand repositories, we investigated how frequently contributors are active in multiple repositories, how frequently BoDeGHa produces diverging or incomplete predictions, and to which extent these predictions can be fixed. We found that around one third of contributors are active in the commenting activity of multiple repositories. We found that these contributors led BoDeGHa to produce a significant amount of diverging and incomplete predictions. We proposed WoC-P, a new model relying on individual predictions made by BoDeGHa. Although BoDeGHa is already one of the best bot identification tools [16], WoC-P improved its accuracy by 6.9% for the considered cases. This suggests that relying on the wisdom of the crowd principle is a promising approach to improve bot detection models across repositories. We also investigated to which extent such an approach can be used to fix incomplete predictions. The initial insights we obtained are encouraging, but they need to be validated on a different, larger dataset.

In a near future, we will consider extending BoDeGHa by integrating WoC-P to allow practitioners to benefit from this improved bot detection model. We plan to evaluate and compare other approaches that exploit more data to detect bots, for example, by modifying the input and data processing part of BoDeGHa so that it can be applied on several repositories at once (i.e., aggregating comments from multiple repositories instead of aggregating predictions made from these repositories). We also plan to evaluate to which extent other tools than BoDeGHa can benefit from the approach proposed in this paper.

Acknowledgement

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 O.0157.18F-RG43 and T.0017.18.

References

[1] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb, "Social coding in GitHub: Transparency and collaboration in an open software repository," in *International Conference on Computer Supported Cooperative Work*, 2 2012, pp. 1277–1286.

[2] M. Wessel, B. M. de Souza, I. Steinmacher, I. S. Wiese, I. Polato, A. P. Chaves, and M. A. Gerosa, "The power of bots: Characterizing and understanding bots in OSS projects," *Proc. ACM Hum.-Comput. Interact. (CSCW)*, vol. 2, 11 2018.

[3] J.-G. Young, A. Casari, K. McLaughlin, M. Z. Trujillo, L. Hébert-Dufresne, and J. P. Bagrow, "Which contributions count? analysis of attribution in open source," *International Conference on Mining Software Repositories (MSR)*, pp. 242–253, 6 2021.

[4] M. A. Storey and A. Zagalsky, "Disrupting developer productivity one Bot at a Time," *ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE)*, pp. 928–931, 2016.

[5] C. Lebeuf, M. A. Storey, and A. Zagalsky, "Software Bots," *IEEE Software*, vol. 35, no. 1, pp. 18–23, 2017.

[6] B. Lin, A. Zagalsky, M. A. Storey, and A. Serebrenik, "Why developers are slacking off: Understanding how software teams use Slack," *ACM Conference on Computer Supported Cooperative Work (CSCW)*, pp. 333–336, 2016.

[7] M. Wessel, B. M. De Souza, I. Steinmacher, I. S. Wiese, I. Polato, A. P. Chaves, and M. A. Gerosa, "The power of bots: Understanding bots in OSS projects," *The ACM International Conference on Human-Computer Interaction*, 2018.

[8] U. Farooq and J. Grudin, "Human-computer integration," *Interactions*, vol. 23, no. 6, pp. 26–32, Oct. 2016.

[9] S. Urli, Z. Yu, L. Seinturier, and M. Monperrus, "How to design a program repair bot: Insights from the repairnator project," *International Conference on Software Engineering (ICSE)*, pp. 95–104, 2018.

[10] I. Beschastnikh, M. F. Lungu, and Y. Zhuang, "Accelerating software engineering research adoption with analysis bots," *International Conference on Software Engineering: New Ideas and Emerging Results Track*, pp. 35–38, September 2017.

[11] M.-A. Storey and A. Zagalsky, "Disrupting developer productivity one bot at a time," in *International Symposium on Foundations of Software Engineering (FSE)*. ACM SIGSOFT, 2016, pp. 928–931.

[12] T. Dey, S. Mousavi, E. Ponce, T. Fry, B. Vasilescu, A. Filippova, and A. Mockus, "Detecting and characterizing bots that commit code," in *17th International Conference on Mining Software Repositories (MSR)*. ACM, 2020, pp. 209–219.

[13] M. Golzadeh, A. Decan, D. Legay, and T. Mens, "A ground-truth dataset and classification model for detecting bots in GitHub issue and PR comments," *Journal of Systems and Software*, vol. 175, 2021.

[14] M. Golzadeh, A. Decan, and T. Mens, “Evaluating a bot detection model on git commit messages,” in *19th Belgium-Netherlands Software Evolution Workshop (BENEVOL)*, vol. 2912. CEUR Workshop Proceedings, 2020.

[15] M. Golzadeh, A. Decan, E. Constantinou, and T. Mens, “Identifying bot activity in GitHub pull request and issue comments,” in *3rd International Workshop on Bots in Software Engineering (BotSE)*. IEEE/ACM, 2021, pp. 21–25.

[16] M. Golzadeh, A. Decan, and N. Chidambaram, “On the accuracy of bot detection techniques,” in *2022 IEEE/ACM Fourth International Workshop on Bots in Software Engineering (BotSE)*. IEEE, 2022.

[17] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, “The promises and perils of mining GitHub,” in *11th Working Conference on Mining Software Repositories (MSR)*, ser. MSR 2014. ACM, 2014, pp. 92–101. [Online]. Available: https://doi.org/10.1145/2597073.2597074