On the Transferability of a Bot Detection Model from GitHub to GitLab

Cyril Moreau[1]

[1]*Software Engineering Lab, University of Mons, Belgium*

Abstract

Collaborative development platforms like GitHub and GitLab are central to software project lifecycles, but the increasing presence of automated accounts, or development bots, complicates the analysis of contributor behaviour. Existing development bot detection tools are primarily developed and evaluated on GitHub data, raising questions about their transferability to other platforms. This work investigates the transferability of BIMBAS, a state-of-the-art bot detection model for GitHub accounts, to GitLab. To make this transfer possible, we built the necessary tooling to extract and map GitLab user events into activity sequences, and constructed a ground-truth dataset of 593 annotated GitLab accounts (273 bots and 320 humans). Our experiments show that BIMBAS trained on GitHub achieves a weighted F1-score of 0.936 when applied to GitLab accounts. These results demonstrate that, although BIMBAS was designed for GitHub, it can be effectively transferred to GitLab, paving the way for more reliable empirical studies across platforms.

Keywords

GitHub, GitLab, bot identification, transferability, machine learning, automation

1. Context and findings

On collaborative development platforms such as GitHub and GitLab, developers heavily rely on automation mechanisms that support them in managing increasingly complex projects. To deal with repetitive, error-prone and time-consuming tasks such as testing or code reviewing, developers often use automated user accounts, commonly referred to as **development bots** [1]. The presence of these bots complicates empirical studies that rely on analysing human contributor activity, as the automated actions of bots can introduce significant biases [2]. Consequently, it is crucial to accurately identify automated accounts to distinguish human and automated activity.

Several tools for bot detection have been proposed in the literature, such as BoDeGHa [2], and more recently BotHunter [3] and RABBIT [1]. The latter leverages BIMBAS, a machine learning model based on user activity sequences generated through an activity mapping referred to as **rbmap** in this paper. However, these existing approaches have been designed and validated exclusively on public GitHub event data. To our knowledge, no automated bot detection tool has yet been systematically assessed on GitLab, raising the question of their generalisation to other collaborative development platforms, and highlighting the need for solutions dedicated to GitLab.

This work investigates the transferability of the BIMBAS model to GitLab. Transferring such a model is not straightforward, since differences in available events, recorded activities, and platform-specific practices require adapting the activity mapping and constructing an appropriate evaluation dataset. In this work, we therefore adapt the activity mapping to GitLab and build a ground-truth dataset of GitLab accounts, enabling us to evaluate whether a model trained on GitHub can effectively identify bots on GitLab.

Since the rbmap activity mapping does not accurately represent user behaviour and is difficult to adapt, Hourri et al. introduced **ghmap** [4], a more flexible mapping that better represents user behaviour. Building on this, we developed **glmap**, an extension of ghmap for GitLab, which enables BIMBAS to be applied on GitLab accounts. To evaluate the transferability of the model, we also built a new ground-truth dataset of 593 semi-automatically labelled GitLab accounts (273 bots and 320 humans).

BENEVOL 2025

*Corresponding author.

cyril.moreau@umons.ac.be (C. Moreau)

Table 1
Performances of BIMBAS with different training and test mappings on GitHub and GitLab datasets. Precision (P), Recall (R), and F1-score (F1) are reported for bots, humans, and a weighted average.

Mapping		Evaluated on	Bots			Humans			Weighted		
Train	Test		P	R	F1	P	R	F1	P	R	F1
rbmap	*rbmap*	*GitHub*	*.905*	*.891*	*.898*	*.896*	*.910*	*.903*	*.900*	*.900*	*.900*
rbmap	ghmap	GitHub	.888	**.883**	.885	**.887**	.892	.889	.887	.887	.887
ghmap	ghmap	GitHub	**.902**	.880	**.891**	.886	**.907**	**.897**	**.894**	**.894**	**.894**
rbmap	glmap	GitLab	.886	.980	.931	.983	.900	.940	.940	.935	.936
ghmap	glmap	GitLab	**.902**	**.984**	**.941**	**.986**	**.916**	**.950**	**.949**	**.946**	**.946**

As glmap relies on a different strategy than the original rbmap used in BIMBAS, we assessed the impact of this change by creating a variant of BIMBAS trained with ghmap, and comparing it with the original model. As shown in Table 1, performance differences are negligible. When BIMBAS is evaluated with ghmap but trained with rbmap, the weighted F1-score decreases by only 0.13. This gap is further reduced to 0.06 when BIMBAS is trained directly with ghmap. These results suggest that BIMBAS is insensitive to the choice of activity mapping, indicating that variations in the set of activities (such as those provided by GitLab) do not substantially affect its performance.

Concerning transferability to GitLab, one can observe from Table 1 that BIMBAS achieves a weighted F1-score of 0.936 when evaluated on GitLab accounts using glmap. When the model is trained with ghmap, which is conceptually closer to glmap, the score further improves to 0.946. These results indicate that a model trained exclusively on GitHub generalises well to GitLab, and that leveraging an activity mapping aligned with glmap further enhances cross-platform performance.

2. Conclusion

In this work, we introduced glmap, an activity mapping adapted to GitLab, and provided a ground-truth dataset of 593 GitLab accounts in order to evaluate the transferability of the BIMBAS bot detection model on GitLab. Building on these contributions, we showed that BIMBAS, originally designed for GitHub, maintains a weighted F1-score of 0.936, which demonstrates its transferability to GitLab. These findings highlight the feasibility of cross-platform bot detection and suggest many interesting directions for future work, most notably the construction of larger and more diverse datasets and the extension of the approach to additional collaborative development platforms.

References

[1] N. Chidambaram, T. Mens, A. Decan, Rabbit: A tool for identifying bot accounts based on their recent github event history, in: 2024 IEEE/ACM 21st International Conference on Mining Software Repositories (MSR), 2024, pp. 687–691. doi:10.1145/3643991.3644877.

[2] 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) 110911. doi:10.1016/j.jss.2021.110911.

[3] A. Abdellatif, M. Wessel, I. Steinmacher, M. A. Gerosa, E. Shihab, Bothunter: An approach to detect software bots in github, in: 2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR), 2022, pp. 6–17. doi:10.1145/3524842.3527959.

[4] Y. Hourri, A. Decan, T. Mens, A dataset of contributor activities in the numfocus open-source community, in: 2025 IEEE/ACM 22nd International Conference on Mining Software Repositories (MSR), IEEE, 2025, pp. 159–163. doi:10.1109/MSR66628.2025.00035.