Insights into Non-Merged Pull Requests in GitHub: Is there Evidence of Bias Based on Perceptible Race?

Reza Nadri, Gema Rodríguez-Pérez, Meiyappan Nagappan
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
email: {rnadri, gema.rodriguez-perez, mei.nagappan}@uwaterloo.ca,

***Abstract*—Recent studies found that the developer's pull request's quality is not the only factor that correlates with its rejection. Diversity factors such as social, gender, and geographical location also correlate. This paper assists on diversity research with a qualitative study that analyzes whether there is evidence of bias based on perceptible race in the written comments of non-merged pull requests in GitHub. We examine the written reasons left as comments by GitHub developers explaining the rejection of 556 contributions submitted by four perceptible racial groups: Asian/Pacific Islander (API), Black, Hispanic, and White. Our initial results may indicate questionable behavior when rejecting pull requests as perceptible-API get more rejections with no reason than perceptible-White. Furthermore, we have identified that submitters perceptible as Hispanic and Black have 39% of their pull requests rejected because they are seen as unnecessary which is 10-12 percentage points more frequent than the rest of perceptible races.**

I. Introduction

A GitHub survey identified that 30% of GitHub developers are aware of the ethnicity of their team members; and that 30% of GitHub developers have faced some forms of negative experiences because of their country of origin, their language, and their ideology [1]. Furthermore, the 2017 GitHub Open Source Survey [2] reported that 11% of GitHub's respondents had witnessed stereotyping and 3% had experienced stereotyping in Open Source.

These findings motivate the need to understand whether there is racial-based bias from GitHub developers when rejecting pull requests submitted by different racial groups. This understanding will be the first step in helping OSS developers take necessary steps to foster a healthy OSS community. As we can only identify the perceptible race of developers based on their GitHub usernames, we will refer to developers' race in our paper as "perceptible-race".

To shed new light on the perceptible-race issues that may affect rejections of pull requests in GitHub, we conducted a qualitative study to reveal whether there is any evidence of bias based on perceptible-race in the written comments of non-merged pull requests in GitHub. Since this is a qualitative analysis with small and focused samples, we seek in-depth reasoning and quality of results rather than use any statistical tools in the process.

II. Related work

Social psychology theories state that individuals working in groups prefer to collaborate with others similar to them [3]; therefore, members of one's group may be treated better than outsiders. Also, psychological research on dual-process theory states that enough available information about an individual may activate other's stereotypical expectations that may influence their thinking process to make impressions and judgments [4].

Furthermore, individuals' race has been a demonstrated influencing factor in social studies. For example, Black people in US are likely to earn less when compared to White people [5], and they need to send double the number of resumes when compared to White people to get one callback when their names are easily perceptible as Black names [6].

Although previous literature has studied the reasons why pull requests have not been merged [7], [8], to the extent of our knowledge, this is the first exploration study to find any evidence that can suggest bias based on perceptible races. We believe that online collaborative environments such as GitHub may be subject to conscious or unconscious beliefs about various social groups that can be triggered by the perceptible-race derived from one's name.

III. Methods

A. Projects and Pull Requests Selection

Our study used GHTorrent [9] alongside GitHub's developers API to extract data from users and pull requests. To ensure that we only gather information from non-trivial projects, we used reporeapers, a publicly-accessible dataset that assesses GitHub projects as trivial/non-trivial based on best engineering practices [10].

We used the status of the pull request from the GitHub API to identify whether a pull request was merged, non-merged, or open. We considered that a pull request was non-merged when it was closed, and its merge time was null. In total, we extracted $4,029,190$ pull requests from $46,191$ projects. There are different developers participating in a pull request: the *submitter* who submits the pull request, the *closer* who closes the pull requests, and the *merger* who merges the pull request. From the $4,029,190$ pull requests, we removed the pull requests submitted and merged by the same developer and the open and merged pull requests. After removing these pull requests, our dataset has $37,762$ projects and $467,990$ non-merged pull requests from $365,607$ developers.

B. Deriving race from names

We used the registered given names of these $365,607$ developers to identify their perceptible-race, although other approaches like avatar images and public profiles can be used to identify developers' perceptible-race as well. First, we used the *Stanford Named Entity Recognizer (NER)* [11] to identify developers who use real names rather than some abstract username as their give name. Then, we used these real names to infer their perceptible-race using Name-Prism [12]. Name-Prism was trained using more than 74 millions labelled names from 118 countries, and it is the most accurate classification tool to infer race from names with an F1 score of 0.795 [12]. Name-Prism uses six ethnic groups: American Indian and Alaska Native (AIAN), Asian/Pacific Islander (API), Black, Hispanic, White, and 2PRACE (Mixed Race) to build the classifier and produces a confidence rate between 0 and 1 for each group. Note that Name-Prism uses the terminology ethnicity, but these labels are considered in our paper as race, and not ethnicity in other classification systems. Thus, we refer to these labels as race as well.

We assigned a unique perceptible-race to each developer when Name-Prism's confidence rate was equal or higher than 0.8. We removed developers whose Name-Prism's confidence rate was lower than 0.8. We decided to select this conservative threshold to err on the side of caution. Thus, our final dataset contains $37,762$ projects, $314,977$ non-merged pull requests and $105,862$ developers with a perceptible-race. From the $105,862$ developers, 0% are perceptible as AIAN, 11.38% as API, 0.20% as Black, 4.14% as Hispanic, 84.26% as White, and 0% as 2PRACE.

C. Qualitative analysis methodology

We first randomly selected 50 non-merged pull requests from each of the submitter-closer perceptible-race pairs, e.g., all combinations between submitter perceptible as API/Black/Hispanic/White and closer perceptible as API/Black/Hispanic/White. Since some pairs have less than 50 pull requests, this process resulted in 556 pull requests. Notice from Table I that the number of pull requests closed by perceptible Black developers is very few. This is itself a worrying finding that the OSS community is neglecting a particular race of developers.

We then manually analyzed the comments made on these pull requests. The approach used to select non-merged pull requests misidentifies pull requests that were merged in the project [7]. Thus, before analyzing and classifying the reasons why 556 pull requests were non-merged, we identified their real status. To remove subjectivity and bias in this classification, two of the authors of this paper classified the pull requests using an iterative content analysis approach [13]. First, two authors individually analyzed and classified the status and reasons for 76 random pull requests. After that, the authors discussed the names and types of the categories identified and agreed on a common categorization. Second, the authors individually analyzed and classified the status and reasons for another set of 88 random pull requests. This stage validated the previous categories and redefined others. Thus, the authors re-classified the prior 164 pull requests according to the last categories. Finally, the authors individually classified the status and reasons for another 88 pull requests. As they did not find any further mismatch with the names of the categories, they measured their agreement using the Krippendorff's alpha [14]. In this stage, both authors reached a near-perfect agreement of 0.73 classifying the status and a near-perfect agreement of 0.86 classifying the reasons. Therefore, one author analyzed the remaining 392 pull requests.

IV. Results

A. The status of the non-merged pull request

Table I shows the frequency between the 16 perceived racial pairs of submitters and closers and the four different statuses of the 556 non-merged pull requests. Table I illustrates that 19% of the non-merged pull requests were classified as successful, 9% as resolved, 12% as replaced, and 60% as rejected.

1) ***Successful:*** When a pull request ended up being successfully merged in the master branch of the project. In these pull requests, the commit that merged the source code submitted along the pull request sometimes is recognizable in the comments.

The percentages of pull requests (with respect to all pull requests from the developers of a particular perceptible-race) that ended up being merged in the projects was: 18% API (26/147), 23% Black (19/82), 18% Hispanic (25/142), and 19% White (35/185).

This result indicates that perceptible-Black developers have the highest percentage of contributions merged into a project without using the methodology that is recognizable by the GitHub API. Since using this approach to merge pull requests means that the mergers need to explicitly add authorship info, there is a chance that submitters may not get the credit for their contributions.

2) ***Resolved:*** When a pull request was not merged in the master branch of the project, but it was resolved inside the project. In these pull requests, the projects' developers commented that they addressed the changes proposed in the pull request inside the project.

The percentages of contributions that were resolved inside the project for each perceptible-race of submitters was 6% API (9/147), 9% Black (7/82) , 8% Hispanic (12/142), and 11% White (21/185). Similar to the previous category, this approach may allow developers to not give credit to the submitters.

Overall, combining the numbers from successful and resolved categories we can see that perceptible-Black submitters have the highest percentage [(19+7)/82=31%] of contributions accepted this way.

*3) **Rejected**:* When a pull request was truly rejected without any further consideration.

Perceptible-White submitters had the lowest percentage of rejection (101/185=54%) when compared to perceptible-Non-White developers: API (95/147=65%), Black (51/82=62%), and Hispanic (89/142=63%). Furthermore, when looking into same perceptible-race pairs, the pair White-White had the lowest percentage of rejection (18/44=41%). In comparison, the rejection rate for API-API was 66% (31/47) and Hispanic-Hispanic was 71% (35/49), which were similar to the rejection rates with closers from other perceptible-race. Note we do not report on perceptible Black-Black pair as the data is one pull request.

These results indicate that contributions from perceptible-White submitters are rejected less often overall and rejected less often by perceptible-White developers.

*4) **Replaced**:* When a pull request was truly rejected, but it was replaced or closed in favor of another pull request.

Perceptible-White submitters had the highest percentage (28/185=15%) of replaced contributions. In addition, we analyzed the 66 pull request replaced, and found the perceptible-race of the new submitter (for the pull requests where someone else's contributions are taken). We identified that in eleven cases the old submitter was perceptible as Non-white and the new submitter was perceptible as White. This number is the highest for any pair of perceptible-race. This result indicates that contributions from submitters perceptible as Non-White are replaced more often with contributions from developers perceptible as White.

B. Reasons why pull requests were rejected

Table I also shows the frequency between the 16 perceptible-race pairs of submitters and closers and the eight different reasons that explain why the pull requests classified as rejected (336) or replaced (66) were non-merged. From the 402 non-merged pull requests, 8% were stale, 2% were chaotic, 21% had quality issues, 4% were duplicated, 33% were unnecessary, 4% had merge conflicts, 27% had no reason/comment, and 1% were not real pull requests.

*1) **Unnecessary**:* The pull request was non-merged because it was considered unnecessary by the projects' developers. Unnecessary comprise pull requests that (1) were no longer needed, (2) did not fix the issue described, and (3) were irrelevant for a branch and needed to be moved to another branch.

While perceptible-Black submitters or perceptible-Hispanic each had 39% (22/56 and 41/105, respectively) of their non-merged pull requests seen as unnecessary by projects' developers, submitters perceptible as White or API had 29% (38/129) and 27% (30/112), respectively. Furthermore, the percentage of non-merged pull requests between same perceptible-race pairs was: API-API 23% (9/36), Hispanic-Hispanic 28% (11/39), White-White 19% (5/26). We do not discuss Black-Black because of the small sample size (1).

Also, pull requests with Non-White-White submitter-closer relationship were more likely to be deemed unnecessary when compared to pull requests with a White-Non-White submitter-closer relationship: API-White was 29% (10/35), Black-White was 41% (14/34), Hispanic-White was 46% (12/26), while White-API was 27% (9/33), White-Black was 35% (11/31) and White-Hispanic was 33% (13/39).

These results indicate that pull requests from perceptible-Hispanic and perceptible-Black developers are seen more frequently as unnecessary. Furthermore, perceptible-White closers are less likely to deem pull requests from other perceptible-White submitters as unnecessary in comparison to other same perceptible-race pairs. Finally, perceptible-White closers rejected more frequently pull requests as unnecessary from Non-White submitters than the other way around. Finally, perceptible-White closers rejected pull requests as unnecessary from Non-White submitters more frequently than the other way around.

*2) **No reason**:* The pull request was non-merged without any comment or explanation from the project's developers.

Submitters perceptible as API got 30% (34/112) of their contributions non-merged without comments. This percentage was lower for perceptible-Hispanic (28/105=27%), perceptible-White (34/129=26%), and perceptible-Black (13/56=23%). Furthermore, when comparing the non-merged pull requests between same perceptible-race pairs, API-API had 41% (15/36), Hispanic-Hispanic had 41% (16/39), and White-White had 19% (5/26).

These results indicate that perceptible-API submitters get their contributions rejected with no reason more frequently than the rest of perceptible-races. Also, perceptible-White closers rejected contributions without comments less frequent if the submitters are other perceptible-White developers.

*3) **Quality**:* The pull request was non-merged because, according to the project's developers, it did not meet the quality required.

Perceptible-White and perceptible-API submitters got 22% (28/129 and 25/112, respectively) of their contributions non-merged because of quality issues. This percentage was lower for perceptible-Black (12/56=21%), and perceptible-Hispanic (21/105=20%).

These results indicate that perceptible-White and perceptible-API submitters got pull requests non-merged because of quality issues more frequently.

*4) **Stale**:* The pull request was non-merged because the project's developers mention that the contributions did not have activity for a long time.

We identified that 12% (15/129) of the contributions from perceptible-White submitters were stale. This percentage was 8% (9/112) for perceptible-API submitters, 7% (4/56) perceptible-Black, and 4% (4/105) perceptible-Hispanic.

This result indicates that perceptible-White developers submitted more pull requests that were stale than other perceptible-races.

*5) **Merge conflicts**:* The pull request was non-merged because the project's developers identified merge conflicts. The merge conflicts describe any failure when building, integrating, or testing.

The percentage of pull requests with merge conflicts for each perceptible-race was 5% API (6/112), 5% White (7/129), 4% Black (2/56), and 3% Hispanic (3/105).

This result indicates that all perceptible-races have around 3-5% of merge conflicts in their contributions.

*6) **Duplicate**:* The pull request was non-merged because the project's developers identified it as a duplicate.

Perceptible-Hispanics submitters had the highest percentage (7/105=7%) of non-merged contributions because they were deemed as duplicates. This percentage was lower for perceptible-API (3/112=3%), perceptible-Black (1/56=2%), and perceptible-White (4/129=3%).

This result indicates that perceptible-Hispanic developers got more pull request rejected because they were duplicated.

*7) **Chaotic**:* The pull request was non-merged because it was not clear. Some requesters are not familiarized with the pull request process in GitHub and add several changes and commits.

Submitters perceptible as API and Black had the highest percentage (5/112=4% and 2/56=4%, respectively) of non-merged contributions because they were considered chaotic. This percentage was lower for perceptible-Hispanic (0/105=0%), and perceptible-White (2/129=2%).

This result indicates that perceptible-Black and perceptible-API got more pull request non-merged because they were considered as unorganized.

*8) **Not PR**:* The pull request was non-merged because it was not describing a pull request but a checklist or other issues.

V. Takeaways

Our results indicate a worrisome high percentage of contributions from perceptible-Black [(5+29)/50=68%] and perceptible-API [(6+29)/50=70%] developers that were replaced and rejected by perceptible-White developers. This percentage was higher for perceptible-Black and perceptible-API developers than for perceptible-White [(8+18)/44=59%] and perceptible-Hispanic [(4+22)/41=63%]. Furthermore, contributions from perceptible-Non-White developers were more frequently non-merged without a reason and shown as unnecessary than contributions from perceptible-White developers. Finally, perceptible-White closers rejected less contributions with no reason or as unnecessary from perceptible-White Submitters than any other same perceptible-race pair.

While our results indicate that there may be a bias against perceptible-Non-White races, we did not find any explicit racism in the written comments left by GitHub developers. However, we analyzed only one stream of data (pull requests comments). There could be explicit racism/bias in other streams of data like issues, discussion boards, mailing lists or IRCs.

References

[1] B. Vasilescu, V. Filkov, and A. Serebrenik, "Perceptions of diversity on git hub: A user survey," in *2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering*. IEEE, 2015, pp. 50–56.

[2] GitHub, "Open Source Survey," https://opensourcesurvey.org/2017/, 2017, accessed: 2020-04-07.

[3] D. E. Byrne, *The attraction paradigm*. Cambridge, MA: Academic Pr, 1971, vol. 11.

[4] J. S. B. Evans, "In two minds: dual-process accounts of reasoning," *Trends in cognitive sciences*, vol. 7, no. 10, pp. 454–459, 2003.

[5] C. of Economic Advisers (US), D. Council of Economic Advisers, Washington, U. S. A. B. to the President's Initiative on Race, U. S. G. P. Office, and P. I. on Race (US), *Changing America: Indicators of social and economic well-being by race and Hispanic origin*. US Government Printing Office, 1998.

[6] M. Bertrand and S. Mullainathan, "Are emily and greg more employable than lakisha and jamal? a field experiment on labor market discrimination," *American economic review*, vol. 94, no. 4, pp. 991–1013, 2004.

[7] G. Gousios, M. Pinzger, and A. v. Deursen, "An exploratory study of the pull-based software development model," in *Proceedings of the 36th International Conference on Software Engineering*, 2014, pp. 345–355.

[8] I. Steinmacher, G. Pinto, I. S. Wiese, and M. A. Gerosa, "Almost there: A study on quasi-contributors in open-source software projects," in *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*. IEEE, 2018, pp. 256–266.

[9] G. Gousios, "The ghtorent dataset and tool suite," in *2013 10th Working Conference on Mining Software Repositories (MSR)*. New York, NY: IEEE, 2013, pp. 233–236.

[10] N. Munaiah, S. Kroh, C. Cabrey, and M. Nagappan, "Curating github for engineered software projects," *Empirical Software Engineering*, vol. 22, no. 6, pp. 3219–3253, 2017.

[11] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd annual meeting on association for computational linguistics*. Stroudsburg, PA: Association for Computational Linguistics, 2005, pp. 363–370.

[12] J. Ye, S. Han, Y. Hu, B. Coskun, M. Liu, H. Qin, and S. Skiena, "Nationality classification using name embeddings," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. New York, NY: ACM, 2017, pp. 1897–1906.

[13] W. Lidwell, K. Holden, and J. Butler, *Universal principles of design, revised and updated: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Beverly, MA: Rockport Publishers, 2010.

[14] K. Krippendorff, "Computing krippendorff's alpha-reliability," University of Pennsylvania, Tech. Rep., 2011.

TABLE I
FREQUENCY BETWEEN THE STATUS AND REASONS OF THE NON-MERGED PULL REQUESTS (PRS) AND THE DIFFERENT PERCEPTIBLE-RACE PAIRS OF SUBMITTERS AND CLOSERS.

		Status of the 556 non-merged PRs					**Reasons of the 402 rejected (336) and replaced (66) PRs**								
Submitter	**Closer**	**Successful**	**Resolved**	**Replaced**	**Rejected**	**# PRs**	**Stale**	**Chaotic**	**Quality**	**Duplicated**	**Unnecessary**	**Conflict**	**No Reason**	**Not PR**	**# PRs**
API	API	7	4	5	31	47	2	1	5	0	9	4	15	0	36
	Black	1	0	0	4	5	1	0	1	1	1	0	0	0	4
	Hispanic	7	1	6	31	45	1	3	10	1	10	2	10	0	37
	White	11	4	6	29	50	5	1	9	2	10	0	8	0	35
Total # PRs		26	9	17	95	**147**	9	5	25	3	30	6	34	0	**112**
Black	API	5	1	0	5	11	2	0	0	0	1	0	2	0	5
	Black	0	0	0	1	1	0	0	0	0	0	1	0	0	1
	Hispanic	2	2	0	16	20	0	0	6	1	7	0	2	0	16
	White	12	4	5	29	50	2	2	6	0	14	1	9	0	34
Total # PRs		19	7	5	51	**82**	4	2	12	1	22	2	13	0	**56**
Hispanic	API	7	4	8	31	50	1	0	11	3	17	0	6	1	39
	Black	0	1	0	1	2	0	0	0	0	1	0	0	0	1
	Hispanic	7	3	4	35	49	3	0	8	1	11	0	16	0	39
	White	11	4	4	22	41	0	0	2	3	12	3	6	0	26
Total # PRs		25	12	16	89	**142**	4	0	21	7	41	3	28	1	**105**
White	API	8	9	5	28	50	3	1	4	3	9	4	9	0	33
	Black	6	4	8	23	41	3	1	7	1	11	1	7	0	31
	Hispanic	8	3	7	32	50	4	0	9	0	13	0	13	0	39
	White	13	5	8	18	44	5	0	8	0	5	2	5	1	26
Total # PRs		35	21	28	101	**185**	15	2	28	4	38	7	34	1	**129**
Total perceptible-race pairs		**105**	**49**	**66**	**336**	**556**	**32**	**9**	**86**	**16**	**131**	**18**	**108**	**2**	**402**
%		19%	9%	12%	60%	100%	8%	2%	21%	4%	33%	4%	27%	1%	100%