What is the Relationship between Code Review Approval Rate and Release Deadline?

Shuonan Pei
Department of Computing Science
University of Alberta
Edmonton, Canada
spei@ualberta.ca

Zhimao Lin
Department of Computing Science
University of Alberta
Edmonton, Canada
zhimao@ualberta.ca

I. Motivation

Code review is currently a very popular software engineering process in the industry. According to [2], code reviews can avoid 60% of the defects in production. Although code review can improve code quality, it is very time-consuming. Code reviewers need to spend a long time reading the code. However, some research also shows that code reviewers' participation has a positive effect on software quality [4]. We doubt that some code reviewers may reduce code review quality when the release deadline is approaching. They may just simply glance the code and approve it. We want to find out if it is true in this project.

II. Research Question

We define the code review approval rate as the percentage of the number of code reviews that are directly approved without requesting any further changes. We want to answer the following questions after we finish this project:

- What is the relationship between the code review approval rate and release deadline?
- Why the code review approval rate increases or decreases when the deadline is approaching?

III. Expectation

We expect that the code review approval rate suddenly increases while the release deadline is approaching. We believe that reviewers are less criticized when the deadline is approaching so that the code review quality is impacted.

IV. Proposal

This project will analyze the data provided on the Code Review Open Platform(CROP). CROP contains a dataset which includes all the code reviews in a certain time period of 11 open-source projects [1, 3]. Each code review records its time stamp and approval status [3].

After the data collection, we will write a Python script, which can transform the data into a table contains review id, date, whether it is directly approved without any changes, and a URL of the code review record. Then, we can use MATLAB or Microsoft Excel to draw a line diagram to show the relationship between the code review approval rate and date.

V. Evaluation Method

As soon as we generate the line diagram, we can identify the fluctuation of code review approval rate and sample some interesting points on the graph. Then, we can use the URL in the table to manually find out what actually happened during the code review activity. With this information, we can get some insight about why the code review approval rate changes.

VI. Threats to validity

The dataset of this project can be limited. We only have 11 GitHub repositories developed by two communities, Eclipse and Couchbase [1]. It provides data for 11 software systems, accounting for a total of 50,959 code reviews and 144,906 revisions [1]. Additionally, we have to manually evaluate some code reviews, which prevents us from generating a very objective conclusion.

References

[1] "Code Review Open Platform," *Code Review Open Platform*. [Online]. Available: https://crop-repo.github.io/. [Accessed: 14-Feb-2019].

[2] H. Siy and L. Votta, "Does the modern code inspection have value?," Proceedings IEEE International Conference on Software Maintenance. ICSM 2001.

[3] M. Paixao, J. Krinke, D. Han, and M. Harman, "Crop," Proceedings of the 15th International Conference on Mining Software Repositories - MSR 18, 2018.

[4] S. Mcintosh, Y. Kamei, B. Adams, and A. E. Hassan, "The impact of code review coverage and code review participation on software quality: a case study of the qt, VTK, and ITK projects," *Proceedings of the 11th Working Conference on Mining Software Repositories - MSR 2014*, 2014.

It record the revision number.
If a review has been revised once or twice(depend on the repo), it means the review is approved without changes.

→ How will you identify release dates? (need to clearly describe this in your methodology)

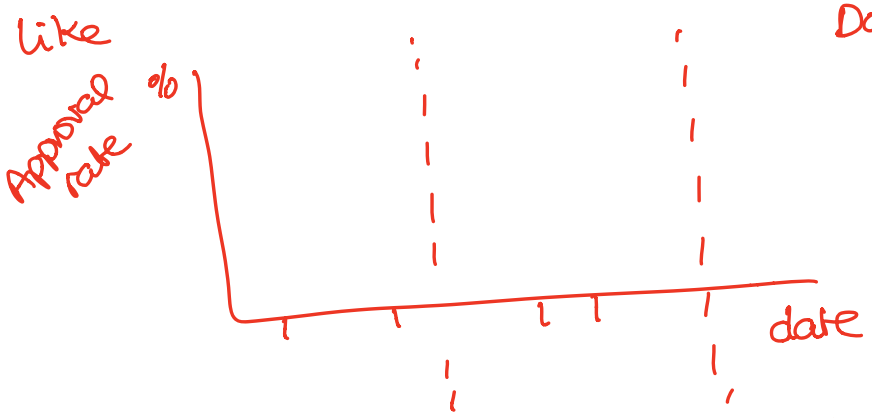
We can find the release date on the Github release page.

→ It might also make sense to compute if there is a correlation between "close to release date" and "approval status" you can use a chi-squared test for this. See

What is this?

https://datascience.stackexchange.com/questions/893/how-to-get-correlation-between-two-categorical-variable-and-a-categorical-variab

→ In terms of your graph, I imagine something like



Is this what you have in mind? Does this mean you are looking at all submitted reviews on a given day? or how exactly will you calculate approval rate? what is the denominator?

Yes

The number of code review in a day.

We can mine the data for the approval rate vs. number of reviews

| Date | How many code reviews are directly approved | Approved after changes | Rejected |

Base on this table, it is easy to calculate the rate in any way that we want.

Questions:

Do I need to have issues/code review/test coverage/CI on our repo?