Making Changes

Yegor Bugayenko

Lecture #3 out of 8
80 minutes

The slidedeck was presented by the author in this YouTube Video

Bram Adams

"We found that 33% of the patches makes it into a Linux release, and that most of them need 3 to 6 months for this."

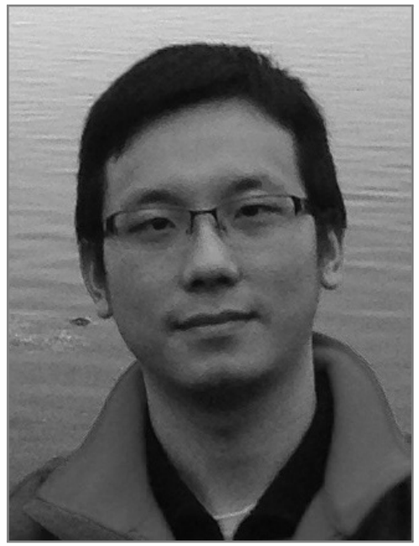
— Yujuan Jiang, Bram Adams, and Daniel M. German. Will My Patch Make It? And How Fast? Case Study on the Linux Kernel. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR)*, pages 101–110. IEEE, 2013. doi:10.1109/MSR.2013.6624016

1. Make small pull requests [Bugayenko, 2020].

Dabbish Laura

"Pull requests with many comments were much less likely to be accepted, moderated by the submitter's prior interaction in the project."

— Jason Tsay, Laura Dabbish, and James Herbsleb. Influence of Social and Technical Factors for Evaluating Contribution in GitHub. In *Proceedings of the 36th International Conference on Software Engineering*, pages 356–366, 2014a. doi:10.1145/2568225.2568315



Yue Yu

"Our preliminary models show that pull request review latency is complex, and depends on many predictors. Naturally, the size of the pull request matters: the shorter it is the faster it will be reviewed."

— Yue Yu, Huaimin Wang, Vladimir Filkov, Premkumar Devanbu, and Bogdan Vasilescu. Wait for It: Determinants of Pull Request Evaluation Latency on GitHub. In *Proceedings of the 12th Working Conference on Mining Software Repositories*, pages 367–371. IEEE, 2015. doi:10.1109/MSR.2015.42

Amiangshu Bosu

"We found that the more files that are in a change, the lower the proportion of comments in the code review that will be of value to the author of the change."

— Amiangshu Bosu, Michaela Greiler, and Christian Bird. Characteristics of Useful Code Reviews: An Empirical Study at Microsoft. In *Proceedings of the 12th Working Conference on Mining Software Repositories*, pages 146–156. IEEE, 2015. doi:10.1109/MSR.2015.21

Caitlin Sadowski

"A correlation between change size and review quality is acknowledged by Google and developers are strongly encouraged to make small, incremental changes (with the exception of large deletions and automated refactoring)."

— Caitlin Sadowski, Emma Söderberg, Luke Church, Michal Sipko, and Alberto Bacchelli. Modern Code Review: A Case Study at Google. In *Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice*, pages 181–190, 2018. doi:10.1145/3183519.3183525

2. Don’t group your changes [Bugayenko, 2020].

Carolyn D. Egelman

"Google categorizes CRs into specific sizes, these sizes are indicated as part of the code review tool and in the notification to the reviewer of the code change... The general advice is to split change requests for easier and quicker reviews when possible."

— Carolyn D. Egelman, Emerson Murphy-Hill, Elizabeth Kammer, Margaret Morrow Hodges, Collin Green, Ciera Jaspan, and James Lin. Predicting Developers' Negative Feelings About Code Review. In *Proceedings of the 42nd International Conference on Software Engineering*, pages 174–185, 2020. doi:10.1145/3377811.3380414

3. Insist on code reviews and merges... politely.

Marco Ortu

"Our results show that valence (expressed in comments received and posted by a reporter) and joy expressed in the comments written by a reporter are linked to a higher likelihood of issues to be merged. On the contrary, sadness, anger, and arousal expressed in the comments written by a reporter, and anger, arousal, and dominance expressed in the comments received by a reporter, are linked to a lower likelihood of a pull request to be merged."

— Marco Ortu, Giuseppe Destefanis, Daniel Graziotin, Michele Marchesi, and Roberto Tonelli. How Do You Propose Your Code Changes? Empirical Analysis of Affect Metrics of Pull Requests on GitHub. *IEEE Access*, 8(1):110897–110907, 2020. doi:10.1109/ACCESS.2020.3002663

What is “valence”?

Valence, also known as hedonic tone, is a characteristic of emotions that determines their emotional affect (intrinsic appeal or repulsion). Positive valence corresponds to the "goodness" or attractiveness of an object, event, or situation, making it appealing or desirable. Conversely, negative valence relates to “badness” or averseness, rendering something unappealing or undesirable. — Wikipedia

Rahul Iyer

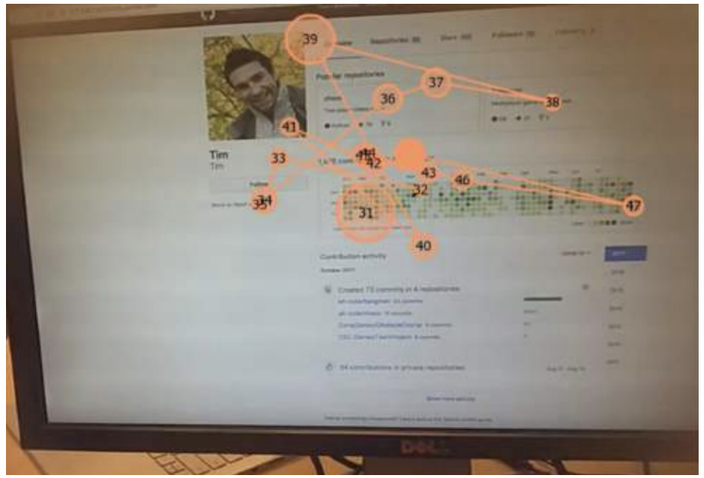
"The larger the difference in personality traits between the requester and the closer, the more positive effect it has on pull request acceptance."

— Rahul N. Iyer, S. Alex Yun, Meiyappan Nagappan, and Jesse Hoey. Effects of Personality Traits on Pull Request Acceptance. *IEEE Transactions on Software Engineering*, 47(11):2632–2643, 2019. doi:10.1109/TSE.2019.2960357

Denae Ford

"We observe that both social and technical aspects are being taken into consideration when deciding upon pull request acceptance. Moreover, we observe that many more social aspects are being considered during the experiment than reported during the post-experiment survey."

— Denae Ford, Mahnaz Behroozi, Alexander Serebrenik, and Chris Parnin. Beyond the Code Itself: How Programmers *Really* Look at Pull Requests. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society*, pages 51–60. IEEE, 2019. doi:10.1109/ICSE-SEIS.2019.00014



Source: Denae Ford, Mahnaz Behroozi, Alexander Serebrenik, and Chris Parnin. Beyond the Code Itself: How Programmers *Really* Look at Pull Requests. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society*, pages 51–60. IEEE, 2019. doi:10.1109/ICSE-SEIS.2019.00014

GitHub is increasing size of the avatar images and emphasizing a developer's "personal brand" by spotlighting features such as the contribution heat map. In the future, platform designers must be more mindful in balancing the power of signals that can amplify bias or harm against users, while still providing the mechanisms for users to freely evaluate the merits of potential code contributions.

> 4. Be a leader and a boss of a pull request — be the one who cares.

Jason Tsay

"We found that the level of a submitter's prior interaction on a project changed how politely developers discussed the contribution and the nature of proposed alternative solutions."

— Jason Tsay, Laura Dabbish, and James Herbsleb. Let's Talk About It: Evaluating Contributions Through Discussion in GitHub. In *Proceedings of the 22nd International Symposium on Foundations of Software Engineering*, pages 144–154, 2014b. doi:10.1145/2635868.2635882

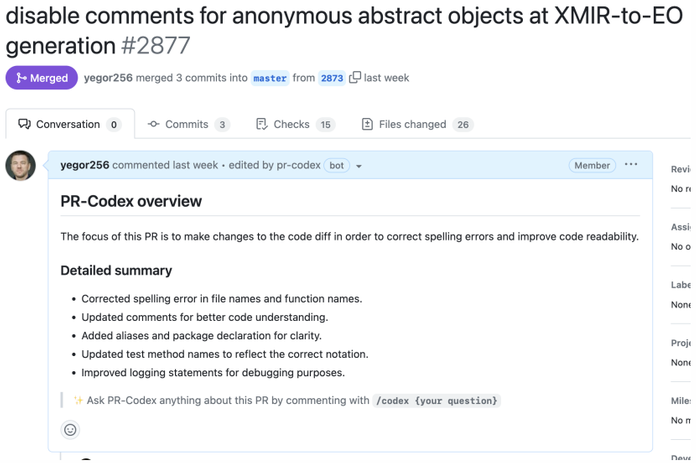
5. Mostly explain “why” you make changes, not “what” you change

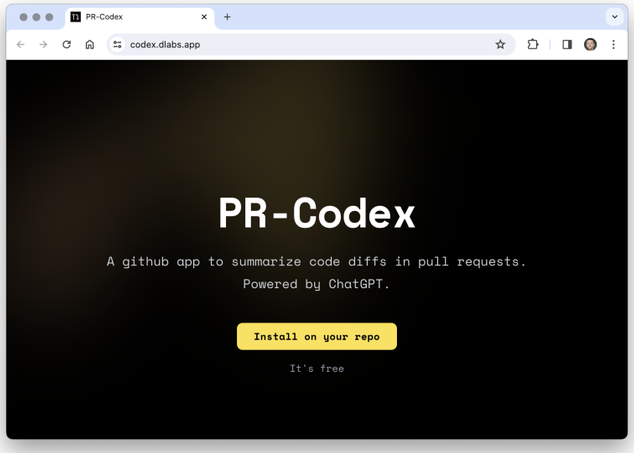
TABLE II
STATISTICS OF OUR COLLECTED PULL REQUESTS

Type	Empty-desc PR	Trivial-desc PR	Long-desc PR	PR with only 1 valid commit	PR with >20 valid commits	Long-source PR	Adequate PR	Total
Number	114,466	61,547	20,516	83,803	2,438	8,399	41,832	333,001

***Long-desc PR** and **Long-source PR** refer to the PRs for which the target sequence and the source sequence do not meet the length constraints, respectively.

Source: Zhongxin Liu, Xin Xia, Christoph Treude, David Lo, and Shanping Li. Automatic Generation of Pull Request Descriptions. In *Proceedings of the 34th International Conference on Automated Software Engineering*, pages 176–188. IEEE, 2019. doi:10.1109/ase.2019.00026


disable comments for anonymous abstract objects at XMIR-to-EO generation #2877
Merged
yegor256 merged 3 commits into master from 2873 last week
Conversation 0
Commits 3
Checks 15
Files changed 26
yegor256 commented last week • edited by pr-codex bot
Member
PR-Codex overview
The focus of this PR is to make changes to the code diff in order to correct spelling errors and improve code readability.
Detailed summary
Corrected spelling error in file names and function names.
Updated comments for better code understanding.
Added aliases and package declaration for clarity.
Updated test method names to reflect the correct notation.
Improved logging statements for debugging purposes.
Ask PR-Codex anything about this PR by commenting with /codex {your question}


PR-Codex
codex.dlabs.app
PR-Codex
A github app to summarize code diffs in pull requests.
Powered by ChatGPT.
Install on your repo
It's free

6. Be prepared for criticism about your style, not functionality.

Jacek Czerwonka

"Only about 15% of comments provided by reviewers indicate a possible defect, much less a blocking defect. Rather, it is feedback related to the long-term code maintainability that comprises a much larger portion of comments provided by reviewers; at least 50% of all."

— Jacek Czerwonka, Michaela Greiler, and Jack Tilford. Code Reviews Do Not Find Bugs: How the Current Code Review Best Practice Slows Us Down. In *Proceedings of the 37th International Conference on Software Engineering*, volume 2, pages 27–28. IEEE, 2015. doi:10.1109/ICSE.2015.131

Valentina Lenarduzzi

“Unexpectedly, quality flaws measured by PMD turned out not to affect the acceptance of a pull request at all. As suggested by other works, other factors such as the reputation of the maintainer and the importance of the delivered feature might be more important than other qualities in terms of pull request acceptance.”

— Valentina Lenarduzzi, Vili Nikkola, Nyyti Saarimäki, and Davide Taibi. Does Code Quality Affect Pull Request Acceptance? An Empirical Study, 2021

7. Commit the code and its tests in different pull requests [Bugayenko, 2022].

Test first, fix next

```
// @todo #42 This test is disabled
// because the fibo() doesn't work
// correctly with this input, returning
// 17711 instead of 28657.  Fix it.
#[test]
#[ignore]
fn calculates_23rd_fibonacci_number() {
  let x = fibo(23);
  assert_eq!(28657, x);
}
fn fibo(x: i32) {
  0
}
```

```
// @todo #42 This test is disabled
// because the fibo() doesn't work
// correctly with this input, returning
// 17711 instead of 28657.  Fix it.
#[test]
#[ignore]
fn calculates_23rd_fibonacci_number() {
  let x = fibo(23);
  assert_eq!(28657, x);
}
fn fibo(x: i32) {
  if (x == 23) {
    return 28657;
  }
  0
}
```

References

Amiangshu Bosu, Michaela Greiler, and Christian Bird. Characteristics of Useful Code Reviews: An Empirical Study at Microsoft. In *Proceedings of the 12th Working Conference on Mining Software Repositories*, pages 146–156. IEEE, 2015. doi:10.1109/MSR.2015.21.

Yegor Bugayenko. Open Source Etiquette. `https://www.yegor256.com/200729.html`, jul 2020. [Online; accessed 07-02-2024].

Yegor Bugayenko. The Code and Its Tests in Different Pull Requests. `https://www.yegor256.com/220804.html`, aug 2022. [Online; accessed 08-02-2024].

Jacek Czerwonka, Michaela Greiler, and Jack Tilford. Code Reviews Do Not Find Bugs: How the Current Code Review Best Practice Slows Us Down. In *Proceedings of the 37th International Conference on Software Engineering*, volume 2, pages 27–28. IEEE, 2015. doi:10.1109/ICSE.2015.131.

Carolyn D. Egelman, Emerson Murphy-Hill, Elizabeth Kammer, Margaret Morrow Hodges, Collin Green, Ciera Jaspan, and James Lin. Predicting Developers' Negative Feelings About Code Review. In *Proceedings of the 42nd International Conference on Software Engineering*, pages 174–185, 2020. doi:10.1145/3377811.3380414.

Denae Ford, Mahnaz Behroozi, Alexander Serebrenik, and Chris Parnin. Beyond the Code Itself: How Programmers *Really* Look at Pull Requests. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society*, pages 51–60. IEEE, 2019. doi:10.1109/ICSE-SEIS.2019.00014.

Rahul N. Iyer, S. Alex Yun, Meiyappan Nagappan, and Jesse Hoey. Effects of Personality Traits on Pull Request Acceptance. *IEEE Transactions on Software Engineering*, 47(11):2632–2643, 2019. doi:10.1109/TSE.2019.2960357.

Yujuan Jiang, Bram Adams, and Daniel M. German. Will My Patch Make It? And How Fast? Case

Study on the Linux Kernel. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR)*, pages 101–110. IEEE, 2013. doi:10.1109/MSR.2013.6624016.

Valentina Lenarduzzi, Vili Nikkola, Nyyti Saarimäki, and Davide Taibi. Does Code Quality Affect Pull Request Acceptance? An Empirical Study, 2021.

Zhongxin Liu, Xin Xia, Christoph Treude, David Lo, and Shanping Li. Automatic Generation of Pull Request Descriptions. In *Proceedings of the 34th International Conference on Automated Software Engineering*, pages 176–188. IEEE, 2019. doi:10.1109/ase.2019.00026.

Marco Ortu, Giuseppe Destefanis, Daniel Graziotin, Michele Marchesi, and Roberto Tonelli. How Do You Propose Your Code Changes? Empirical Analysis of Affect Metrics of Pull Requests on GitHub. *IEEE Access*, 8(1):110897–110907, 2020. doi:10.1109/ACCESS.2020.3002663.

Caitlin Sadowski, Emma Söderberg, Luke Church, Michal Sipko, and Alberto Bacchelli. Modern Code Review: A Case Study at Google. In *Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice*, pages 181–190, 2018. doi:10.1145/3183519.3183525.

Jason Tsay, Laura Dabbish, and James Herbsleb. Influence of Social and Technical Factors for Evaluating Contribution in GitHub. In *Proceedings of the 36th International Conference on Software Engineering*, pages 356–366, 2014a. doi:10.1145/2568225.2568315.

Jason Tsay, Laura Dabbish, and James Herbsleb. Let's Talk About It: Evaluating Contributions Through Discussion in GitHub. In *Proceedings of the 22nd International Symposium on Foundations of Software Engineering*, pages 144–154, 2014b. doi:10.1145/2635868.2635882.

Yue Yu, Huaimin Wang, Vladimir Filkov, Premkumar Devanbu, and Bogdan Vasilescu. Wait for It: Determinants of Pull Request Evaluation Latency on GitHub. In *Proceedings of the 12th Working Conference on Mining Software Repositories*, pages 367–371. IEEE, 2015. doi:10.1109/MSR.2015.42.