Making Changes

Yegor Bugayenko

Lecture #3 out of 8
80 minutes

"We found that <u>33%</u> of the patches makes it into a Linux release, and that most of them need <u>3 to 6 months</u> 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

BRAM ADAMS

1. Make small pull requests[1]

---

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

Amiangshu Bosu

"We found that the <u>more files</u> that are in a change, the <u>lower</u> the proportion of <u>comments</u> 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

"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

CAITLIN SADOWSKI

2. Don't group your changes[2]

---

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

"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 <u>split</u> change requests for <u>easier</u> and <u>quicker</u> 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

CAROLYN D. EGELMAN

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

Marco Ortu

"Our results show that <u>valence</u> (expressed in comments received and posted by a reporter) and <u>joy</u> expressed in the comments written by a reporter are linked to a <u>higher likelihood</u> 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*, 8:110897–110907, 2020

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

Denae Ford

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

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

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 <u>politely</u> 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, 2014

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

disable comments for anonymous abstract objects at XMIR-to-EO generation #2877
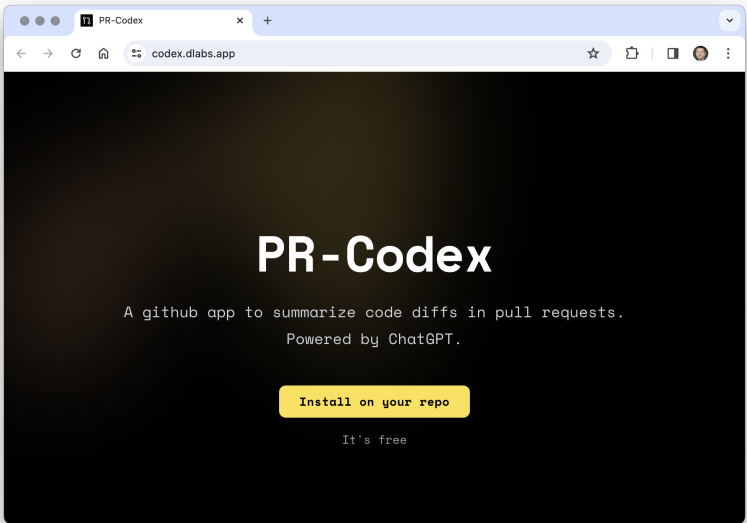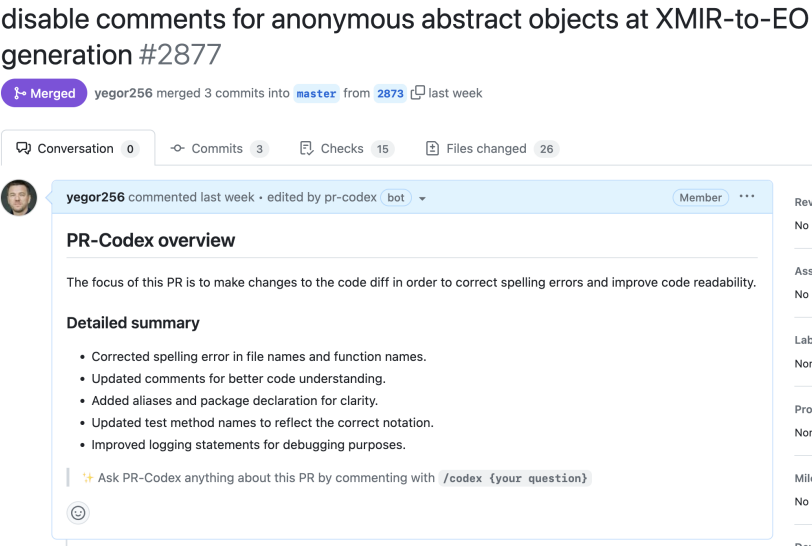
6. Be prepared for criticism about your style, not functionality

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 <u>maintainability</u> 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 IEEE International Conference on Software Engineering*, volume 2, pages 27–28. IEEE, 2015

"Unexpectedly, quality <u>flaws</u> 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 <u>reputation</u> of the maintainer and the <u>importance</u> 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. *Journal of Systems and Software*, 171:110806, 2021

Valentina Lenarduzzi

7. Commit the code and its tests in different pull requests[3]

---

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

Test first, fix next

```
1  // @todo #42 This test is disabled
2  // because the fibo() doesn't work
3  // correctly with this input, returning
4  // 17711 instead of 28657.  Fix it.
5  #[test]
6  #[ignore]
7  fn calculates_23rd_fibonacci_number() {
8    let x = fibo(23);
9    assert_eq!(28657, x);
10  }
11  fn fibo(x: i32) {
12    0
13  }
```

```
1  // @todo #42 This test is disabled
2  // because the fibo() doesn't work
3  // correctly with this input, returning
4  // 17711 instead of 28657.  Fix it.
5  #[test]
6  #[ignore]
7  fn calculates_23rd_fibonacci_number() {
8    let x = fibo(23);
9    assert_eq!(28657, x);
10  }
11  fn fibo(x: i32) {
12    if (x == 23) {
13      return 28657;
14    }
15    0
16  }
```

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.

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

Yegor Bugayenko. The Code and Its Tests in Different Pull Requests. `https://www.yegor256.com/220804.html`, August 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 IEEE International Conference on Software Engineering*, volume 2, pages 27–28. IEEE, 2015.

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.

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.

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.

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.

Valentina Lenarduzzi, Vili Nikkola, Nyyti Saarimäki, and Davide Taibi. Does code quality affect pull

request acceptance? An empirical study. *Journal of Systems and Software*, 171:110806, 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.

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*, 8:110897–110907, 2020.

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.

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, 2014.