A Brief Introduction

2010-2018

Faculty at NCSU

Research creating developer tools that fit with the way developers work.

2018 - 2024

Research Scientist at Google

Worked on developer productivity and how to make internal developer systems more inclusive.

2024 - …

Research Scientist at Microsoft

Worked on Github Copilot, Excel Agent, and now internal developer productivity

Software Engineering in Practice

We’re in a fast-moving transition where workflows are changing quickly, but stable norms have not yet emerged

- but developer productivity improvements are incremental, because we’re hitting different bottlenecks
- and limits are unclear

My outlook:

- Software engineering is one of the best applications of generative AI
- Human communication between people remains one of the biggest challenges
- Don’t accept the narrative that “AI are like very productive junior engineers”; it’s disrespectful, and feeds into an unsubstantiated belief that we need fewer junior engineers

GenderMag Improves Discoverability in the Field, Especially for Women

Emerson Murphy-Hill

Work done while at Google

Murphy-Hill E, Elizondo A, Murillo A, Harbach M, Vasilescu B, Carlson D, Dessloch F. GenderMag Improves Discoverability in the Field, Especially for Women. In IEEE/ACM 46th International Conference on Software Engineering (ICSE). 2024.

Woman Spent Five Months in Jail After A.I. Linked Her to Bank Fraud Case

The police chief in Fargo, N.D., acknowledged "missteps" but stopped short of apologizing to Angela Lipps, a Tennessee resident who said she had never been to North Dakota before she was arrested.

Listen · 6:04 min

Share full article

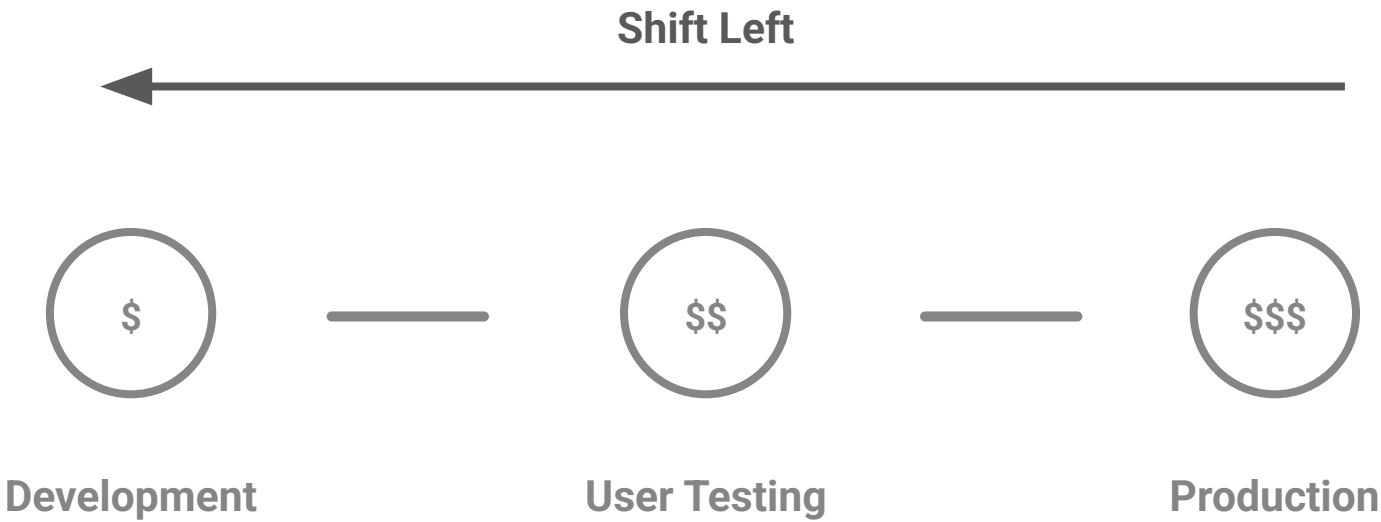
Angela Lipps outside her home in Elizabethton, Tenn., before she was arrested in July 2025. She spent five months in jail in Tennessee and North Dakota before she was released on Dec. 24. Angela Lipps

NIST consistently found systems had 2-5x more false negatives for women than men

Making systems work for everyone is hard

Grother, P.J., Grother, P.J., Ngan, M. and Hanaoka, K., 2014. *Face recognition vendor test (FRVT)*. US Department of Commerce, National Institute of Standards and Technology.

Development **User Testing** **Production**


Shift Left
$
$$
$$$
Development
User Testing
Production

Development

Abi (Abigail/Abishek)

- 28 Years Old
- Employed as an Accountant
- Lives in Cardiff, Wales

Abi has always liked music. When she is on her way to work in the morning, she listens to music that spans a wide variety of styles. But when she arrives at work, she turns it off, and begins her day by *scanning all her emails first to get an overall picture before answering any of them.* (This extra pass takes time but seems worth it.) Some nights she exercises or stretches, and sometimes she likes to play computer puzzle games like Sudoku

Background and Skills

Abi works as an accountant. She is comfortable with the technologies she uses regularly,but she just moved to this employer 1 week ago,and their software systems are new to her. Abi says she's a "numbers person", but she has never taken any computer programming or IT systems classes. She likes Math and knows how to think with numbers. She writes and edits spreadsheet formulas in her work. In her free time, she also enjoys working with numbers and logic. she especially likes working out puzzles and puzzle games, either on paper or on the computer.

Motivations and Attitudes

Motivations: Abi uses technologies to accomplish her tasks. She learns new technologies if and when she needs to, but prefers to use methods she is already familiar and comfortable with, to keep her focus on the tasks she cares about.

Computer Self-Efficacy: Abi has lower self confidence than her peers about doing unfamiliar computing tasks. If problems arise with her technology, she often blames herself for these problems. This affects whether and how she will persevere with a task if technology problems have arisen.

Attitude toward Risk: Abi's life is a little complicated and she rarely has spare time. So she is risk averse about using unfamiliar technologies that might need her to spend extra time on them, even if the new features might be relevant. She instead performs tasks using familiar features, because they're more predictable about what she will get from them and how much time they will take.

Attitude to Technology

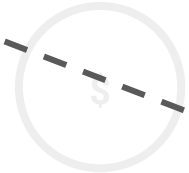
Information Processing Style: Abi tends towards a comprehensive information processing style when she needs to gather more information. So, instead of acting upon the first option that seems promising, she gathers information comprehensively to try to form a complete understanding of the problem before trying to solve it. Thus, her style is "burst-y"; first she reads a lot, then she acts on it in a batch of activity.

Learning: by Process vs. by Tinkering : When learning new technology, Abi leans toward process-oriented learning, e.g., tutorials, step-by-step processes, wizards, online how-to videos, etc. She doesn't particularly like learning by tinkering with software (i.e., just trying out new features or commands to see what they do), but when she does tinker, it has positive effects on her understanding of the software.

[1]Abi represents users with motivations/attitudes and information/learning styles similar to hers. For data on men and women similar to and different from Abi, see http://gendermag.org/foundations.php

Burnett, Margaret, Simone Stumpf, Jamie Macbeth, Stephann Makri, Laura Beckwith, Irwin Kwan, Anicia Peters, and William Jernigan. "GenderMag: A method for evaluating software's gender inclusiveness." *Interacting with Computers* 28, no. 6 (2016): 760-787.

Learning: by Process vs. by Tinkering : When learning new technology, Abi leans toward process-oriented learning, e.g., tutorials, step-by-step processes, wizards, online how-to videos, etc. She doesn't particularly like learning by tinkering with software (i.e., just trying out new features or commands to see what they do), but when she does tinker, it has positive effects on her understanding of the software.



Development

Motivations and Attitudes

Motivations: Abi uses technologies to accomplish her tasks. She learns new technologies if and when she needs to, but prefers to use methods she is already familiar and comfortable with, to keep her focus on the tasks she cares about.

Computer Self-Efficacy: Abi has lower self confidence than her peers about doing unfamiliar computing tasks. If problems arise with her technology, she often blames herself for these problems. This affects whether and how she will persevere with a task if technology problems have arisen.

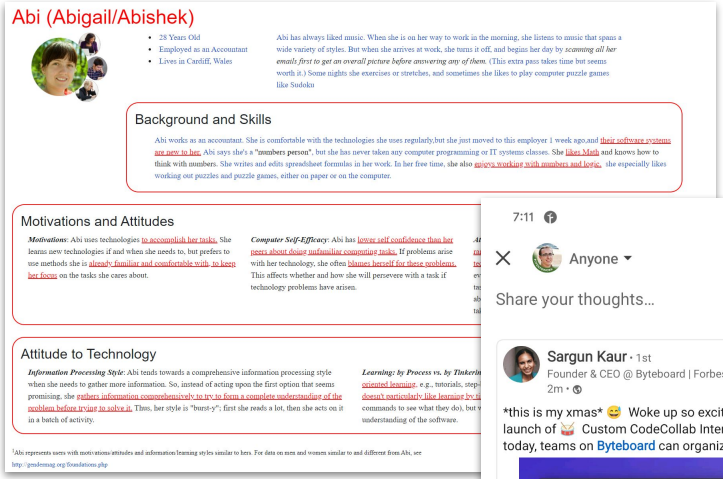
Attitude toward Risk: Abi's life is a little complicated and she rarely has spare time. So she is risk averse about using unfamiliar technologies that might need her to spend extra time on them, even if the new features might be relevant. She instead performs tasks using familiar features, because they're more predictable about what she will get from them and how much time they will take.

Attitude to Technology

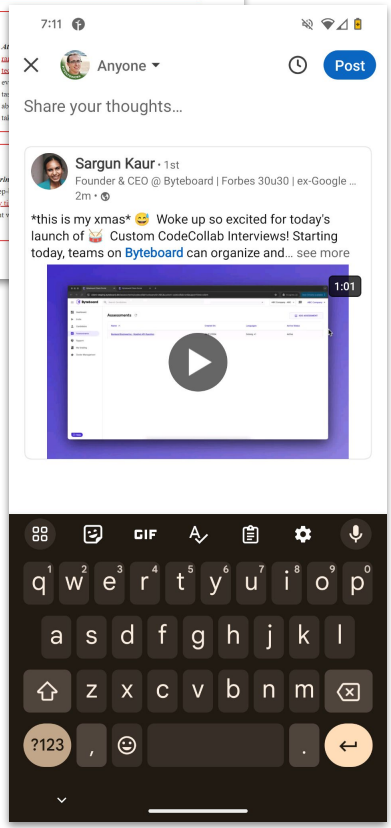
Information Processing Style: Abi [illegible] towards a comprehensive information processing style when she needs to gather more information. [illegible] kind of acting upon the first option that seems promising, she gathers information comprehensively to [illegible] form a complete understanding of the problem before trying to solve it. Thus, her style is "burst-y": first [illegible] reads a lot, then she acts on it in a batch of activity.

Learning: by Process vs. by Tinkering : When learning new technology, Abi leans toward process-oriented learning, e.g., tutorials, step-by-step processes, wizards, online how-to videos, etc. She doesn't particularly like learning by tinkering with software (i.e., just trying out new features or commands to see what they do), but when she does tinker, it has positive effects on her understanding of the software.

[1]Abi represents users with motivations/attitudes and information/learning styles similar to hers. For data on men and women similar [illegible]
http://gendermag.org/foundations.php



Development



Use Case: Share a post on LinkedIn with comments

1 **Scroll to the bottom of the post**

- Will ____ have thought of this as a step toward achieving the overall use case?
- <Before> Will ____ do this?
- <After> Will ____ know that they did the right thing and are making progress toward their goal?

2 **Tap "Repost with your thoughts"**

- Will ____ have thought of this as a step toward achieving the overall use case?
- <Before> Will ____ do this?
- <After> Will ____ know that they did the right thing and are making progress toward their goal?

3 **Type your comments**

- Will ____ have thought of this as a step toward achieving the overall use case?
- <Before> Will ____ do this?
- <After> Will ____ know that they did the right thing and are making progress toward their goal?

4 **Tap "Post"**

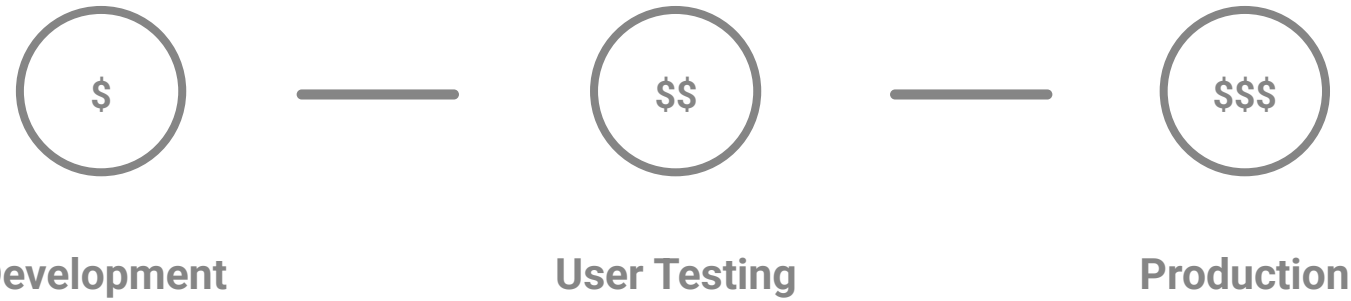
- Will ____ have thought of this as a step toward achieving the overall use case?
- <Before> Will ____ do this?
- <After> Will ____ know that they did the right thing and are making progress toward their goal?

Is GenderMag effective?

The most direct evidence from the literature:

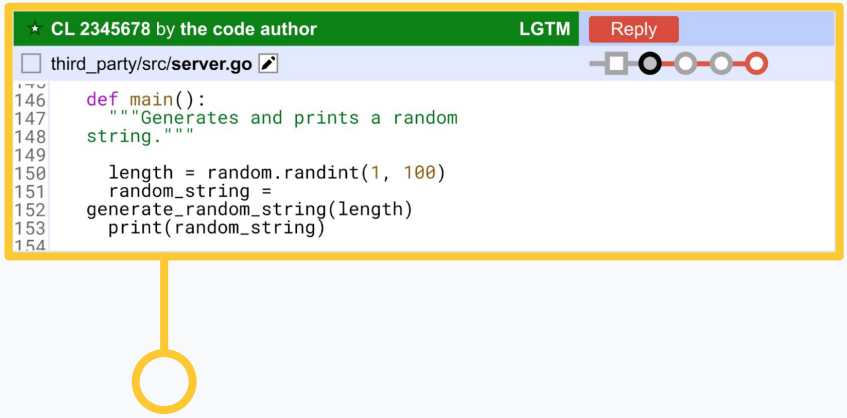
Method: Apply GenderMag to a search engine, count task failures in a laboratory study with 20 faculty members and students

Finding: Women had 2x as many task failures before GenderMag, but the same number in the GenderMag version



Vorvoreanu, M., Zhang, L., Huang, Y.H., Hilderbrand, C., Steine-Hanson, Z. and Burnett, M., 2019, May. From gender biases to gender-inclusive design: An empirical investigation. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (pp. 1-14).

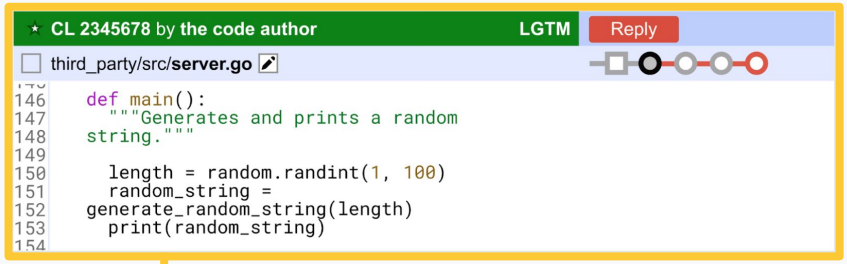
What We Did

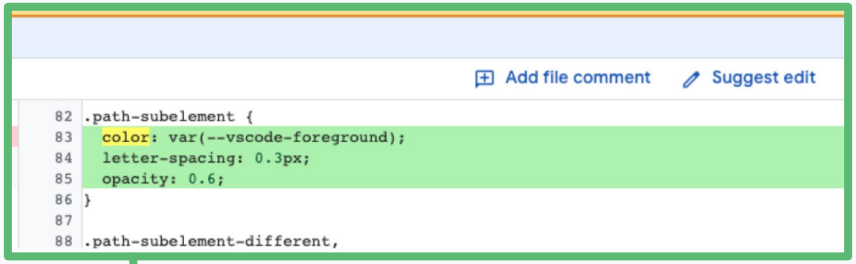


2014

Suggest Edit originally introduced into Critique

What We Did





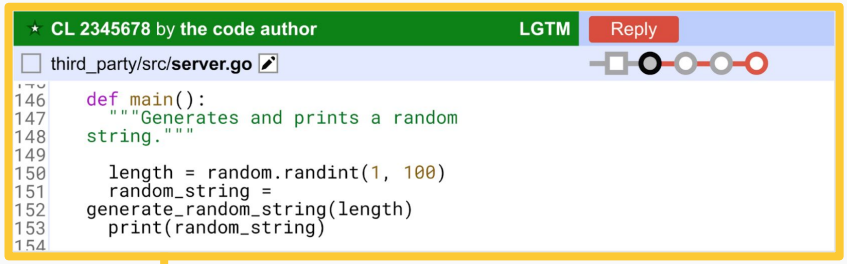
2014
Suggest Edit originally introduced into Critique

2019
In a sprint, we apply GenderMag to Suggest Edit on Critique, create mockups

2020
Mockups implemented and deployed

2023
Evaluation

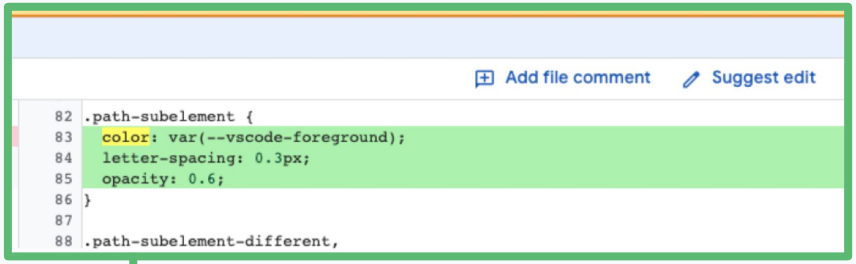
Example Finding from GenderMag Session



Before

Suggest Edit is hidden as a small, cryptic icon. Low self-efficacy/risk averse users may not try it.

The icon also only appears on a secondary page. Non-tinkering users may not discover it.



After

The Suggest Edit button now has equal prominence to the “Add file comment” button, and has text added next to the icon.

It appears in both the file view and the in-place file view

Method

How did our redesign of Suggest Edit affect how quickly Critique users *discover* Suggest Edit? Men vs. women?

Data from

- ~42,000 male and ~11,000 female Critique users
- ~3 years before and after redesign

Naive method: use data on time it takes users to discover Suggest Edit

- But that ignores data from users who *never* discovered Suggest Edit

Hypothetical data:

- Before: Median 6 months to discovery
- After: Median 3 months to discovery

But also:

- Before: 50% of people never discovered
- After: 80% of people never discovered

Method

How did our redesign of Suggest Edit affect how quickly Critique users *discover* Suggest Edit? Men vs. women?

Data from

- ~42,000 male and ~11,000 female Critique users
- ~3 years before and after redesign

Naive method: use data on time it takes users to discover Suggest Edit

- But that ignores data from users who *never* discovered Suggest Edit

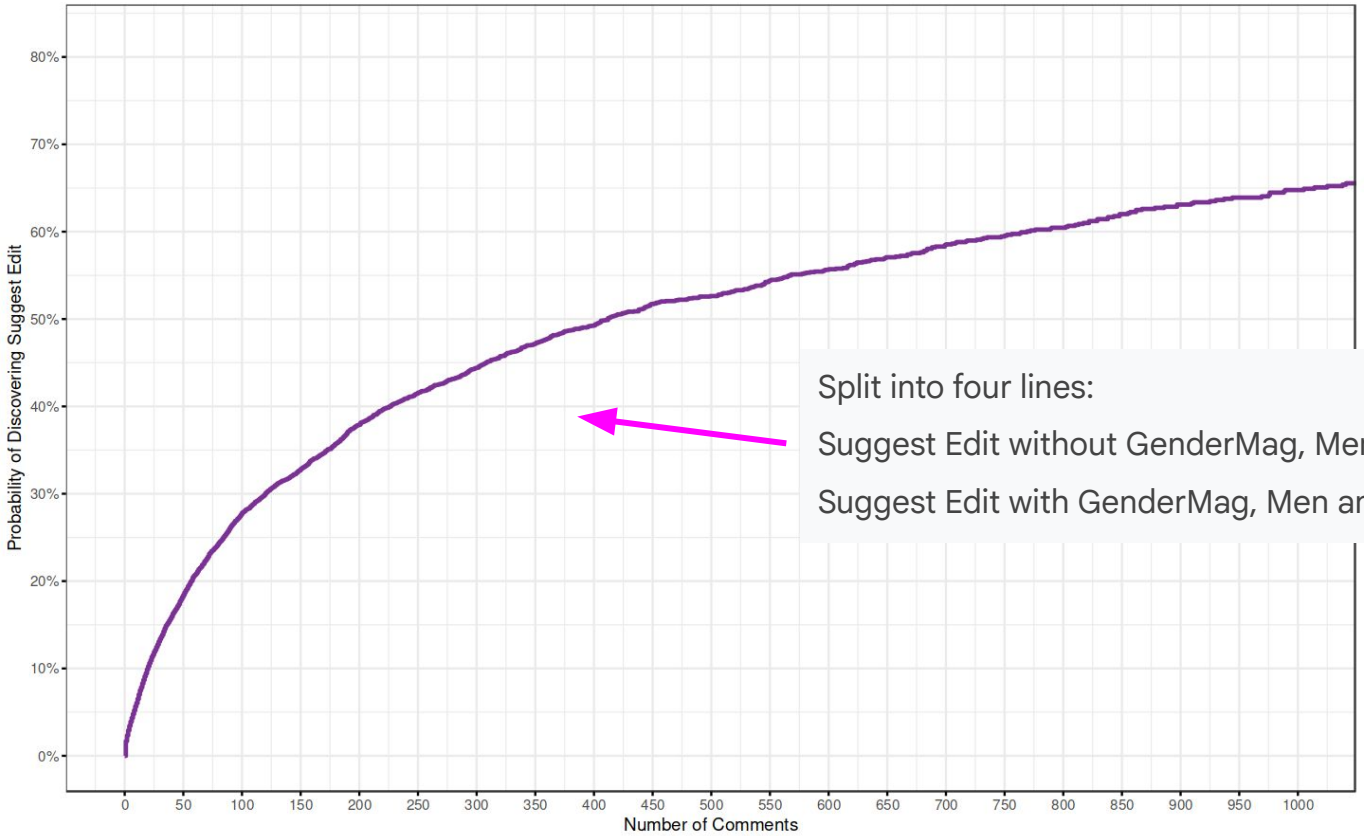
The key idea of our evaluation is that we can repurpose *survival analysis*, integrating both types of data

- Survival duration: number of comments until first Suggest Edit use
- Right censor: number of comments until end of data

Estimate differences

- Statistically, using a Cox regression, controlling for covariates
- Intuitively, using a Kaplan-Meier plots

Method

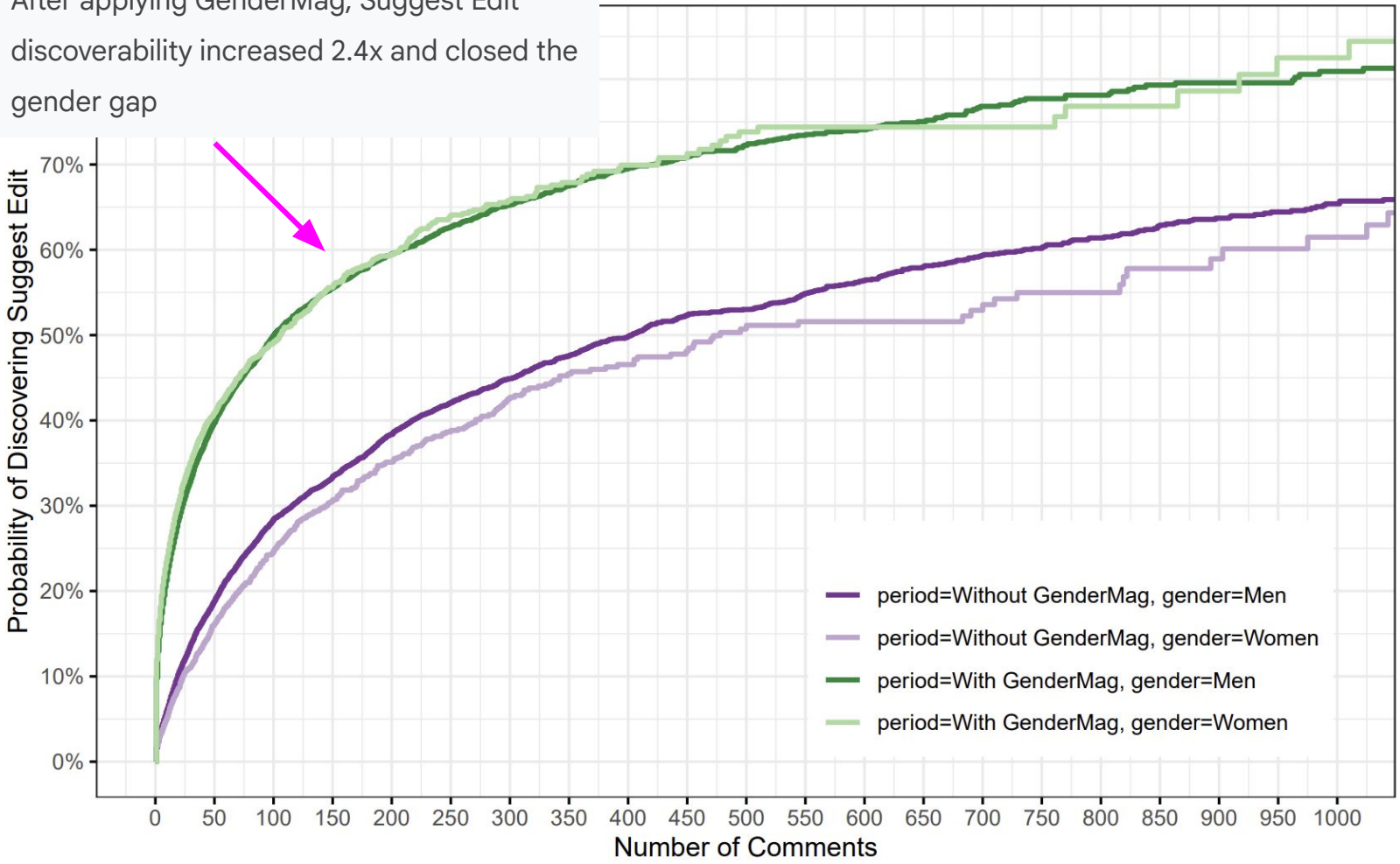


Split into four lines:

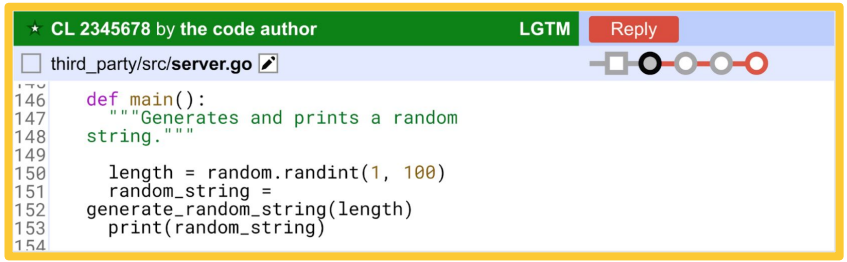
Suggest Edit without GenderMag, Men and Women

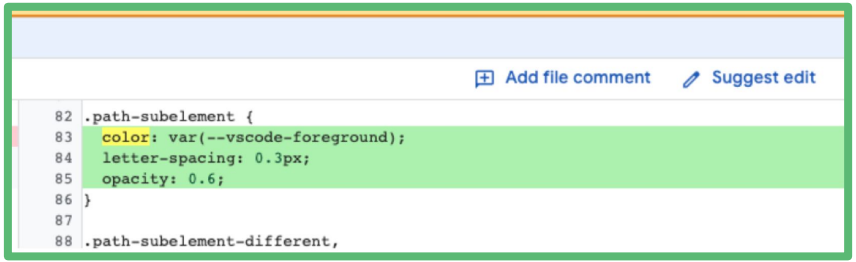
Suggest Edit with GenderMag, Men and Women

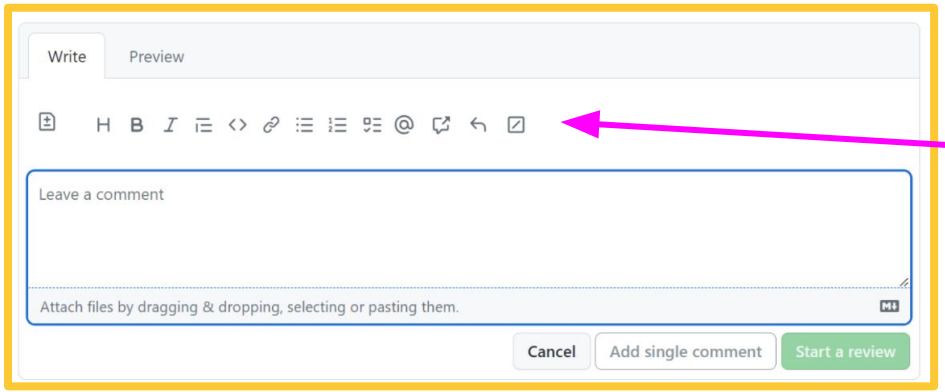
After applying GenderMag, Suggest Edit discoverability increased 2.4x and closed the gender gap



So you’re giving a talk about changing a button?







GitHub pull request reviews have a similar feature with a near-identical discoverability problem

▲ makerofthings 4 months ago | prev | next [–]

One thing I like about it is that reviewers can suggest changes and you can accept them inline. Makes it really easy to deal with nits.

▲ eftychis 4 months ago | parent | next [–]

You can do that in Github, but for some reason a lot of reviewers are not familiar or bother doing that. Fixing nits that way or giving a suggestion improves turnaround speed greatly and builds a relationship between reviewer and proposer and the final product/commit(s).

▲ yohannparis 4 months ago | root | parent | next [–]

Yes, I have been "suggesting" a lot of one liners, typo, rephrasing, or just simple clean up of code with it. Make it a breeze, it's like playing tidy-up without having to branch out or bother much the author.

▲ froh 4 months ago | root | parent | next [–]

I feel stupid. How do I do this?

▲ mplanchard 4 months ago | root | parent | next [–]

When writing an inline comment, there's a button to make a suggestion. It inserts a markdown code fence with the existing code on the line(s), which you can edit. When you submit the comment, it shows up with diff highlighting and can be applied by the author with one click.

Essentially the same thing exists in both GitHub and GitLab

▲ froh 4 months ago | root | parent | next [–]

ah yes! duh. and thanks. I need better glasses...

https://news.ycombinator.com/item?id=38518473

Discussion

Can we use survival analysis to model feature discovery in a variety of products?

Can we shift product inclusion left further than GenderMag?

Conclusion