# The Future Engineer

## Redefining Software Engineering in the Age of AI

Annie Vella | 13 March 2025

# Who Am I?

Software Engineer | Technical Leader | Self Professed Nerd

Present:

- Distinguished Engineer at Westpac NZ
- Master’s of Engineering student at the University of Auckland

Previously:

- 2004 - 2014 - Software Engineer
- 2014 - Present - Technical Leader

# Why is this topic so important?

*Software development is the invisible hand behind this digital magic, transforming ideas into the tools that shape our lives.*

https://www.statista.com/outlook/tmo/software/worldwide

# 28.7 million software engineers worldwide

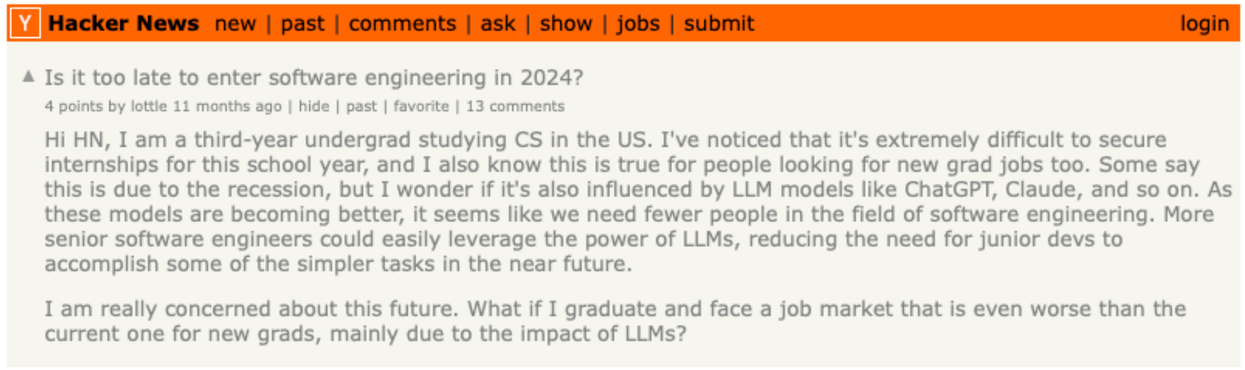
## US$3,567 billion

https://springsapps.com/knowledge/how-many-software-engineers-are-there-in-2024


https://x.com/KevinNaughtonJr/status/1796241418484334722

# Is this the End of Software Engineering?


https://www.reddit.com/r/WGU/comments/1b89euw/is_software_engineering_dying/




https://news.ycombinator.com/item?id=39664697


https://www.reddit.com/r/AskProgramming/comments/yqg0wk/my_dad_says_i_shouldnt_do_software_development/


COMMODORE 64
Manual del Usuario
C64 user's manual
commodore
commodore
commodore 64
POWER
PREGUNTON

# Study Design

- Longitudinal study on the **impact of AI on software engineering**
- Focusing on perceptions of professional software engineers using AI coding assistants
- Answering research questions on
  - Workflow and task focus
  - Productivity and developer experience
  - Skills acquisition and retention
  - Prompt engineering as a core competency
- Mixed methods approach
  - 2 x questionnaires
  - Semi-structured interviews

# Questionnaire #1

- Conducted online in October/November 2024 using Qualtrics
- Recruited participants through LinkedIn and other social and professional networks
- 33 organisations invited to participate worldwide, 9 accepted
- Ethics approval from the University of Auckland Human Participants Ethics Committee on 23/09/2024 for three years. Reference Number: UAHPEC27902.

- Key Stats:
  - Total Responses: 224
  - Completed Questionnaires: 192
  - Professional Software Engineers: 184
  - Professional Software Engineers Using AI: 168

# Insights from related research

# Productivity Improvements Vary

Developers using GitHub Copilot completed a JavaScript programming task **55.8% faster** than those without access to the tool

Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). The Impact of AI on Developer Productivity: Evidence from GitHub Copilot. arXiv:2302.06590.

When using GitHub Copilot, ANZ engineers saw a **42.36% improvement** in productivity on average

Chatterjee, S., Liu, C. L., Rowland, G., & Hogarth, T. (2024). The Impact of AI Tool on Engineering at ANZ Bank: An Empirical Study on GitHub Copilot within Corporate Environment. arXiv:2402.05636

Developers using GitHub Copilot reported up to a **50% reduction** in time spent on coding tasks, with an overall projected **productivity improvement of 33-36%** in cloud-first development

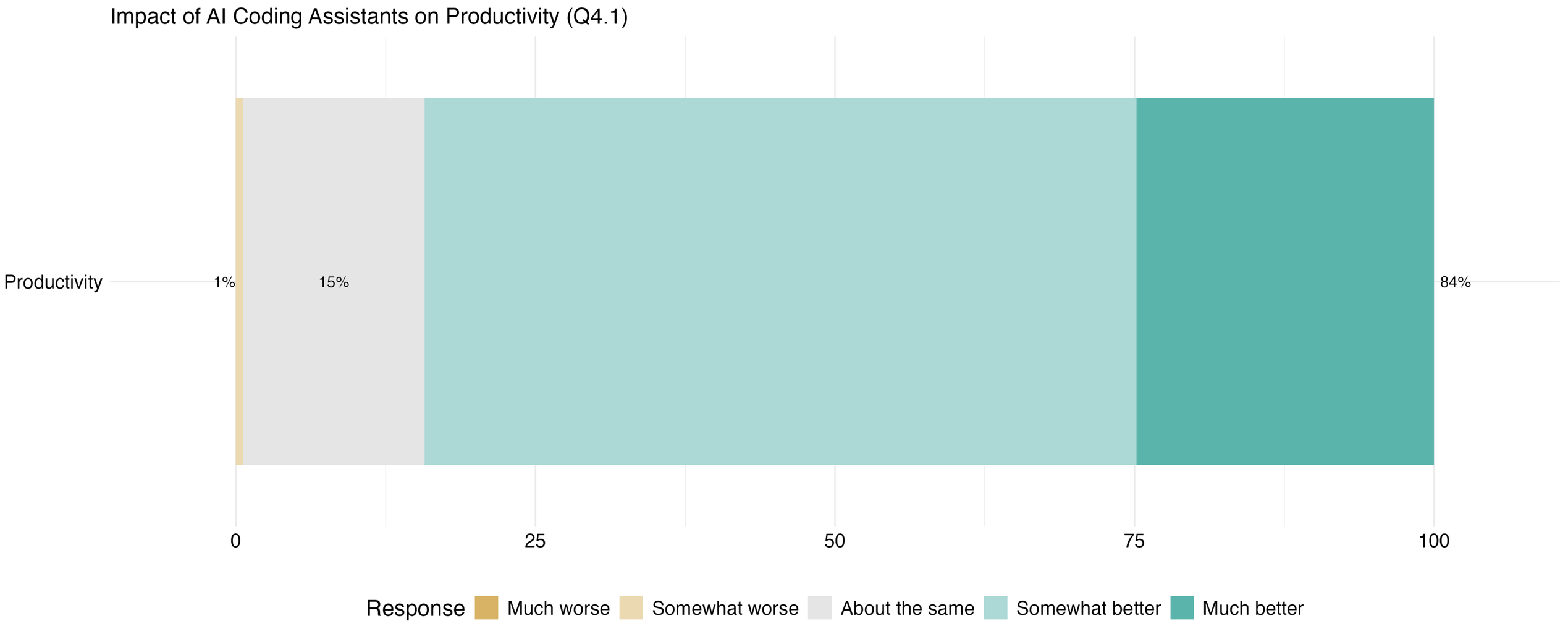
Pandey, R., Singh, P., Wei, R., & Shankar, S. (2024). Transforming Software Development: Evaluating the Efficiency and Challenges of GitHub Copilot in Real-World Projects. arXiv:2406.17910.

Productivity measures the extent to which an individual feels effective and efficient in their work, creating value and achieving tasks.

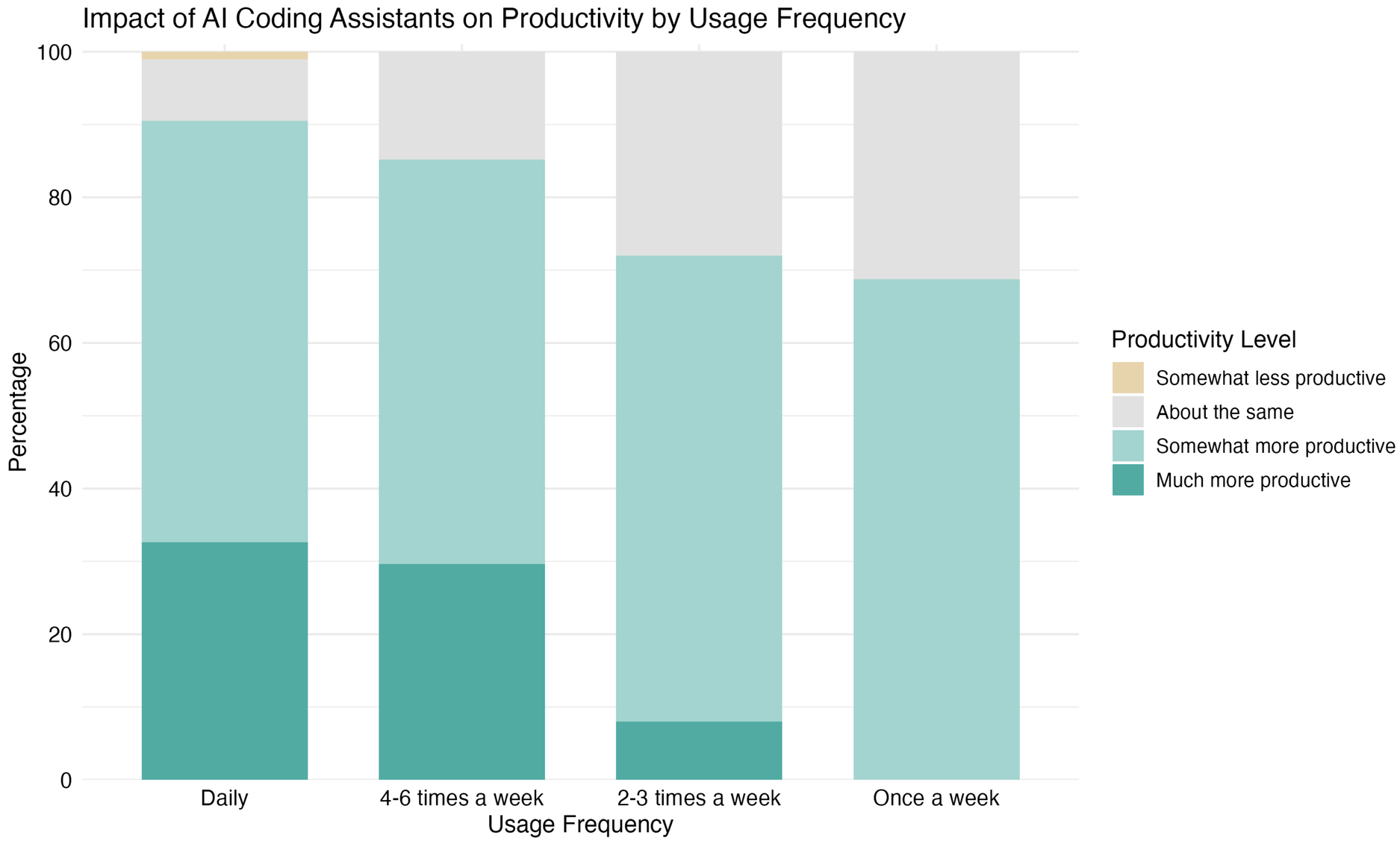
*Definition of productivity,*
*2024 Accelerate State of DevOps DORA Report*

# What does our research say?

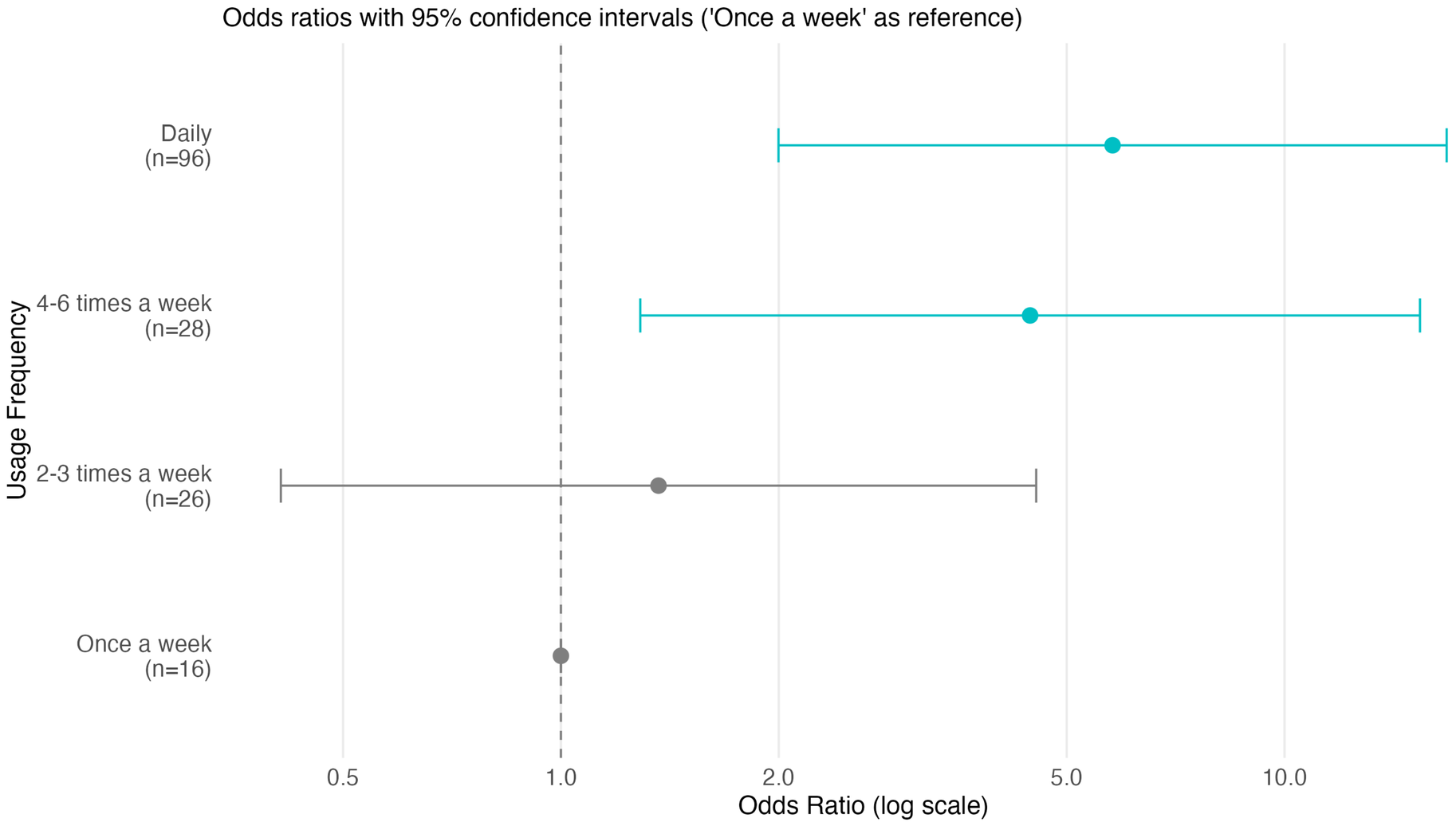
# 84% Report Improved Productivity



# How AI Usage Relates to Productivity



# Another View on Usage and Productivity




Note: Values > 1 indicate the frequency is more likely to report improved productivity with AI assistants
Values < 1 indicate the frequency is less likely to report improved productivity
Teal colors indicate statistically significant results ($p < 0.05$)

# More Usage = More Positive Sentiment

Frequent users mention:

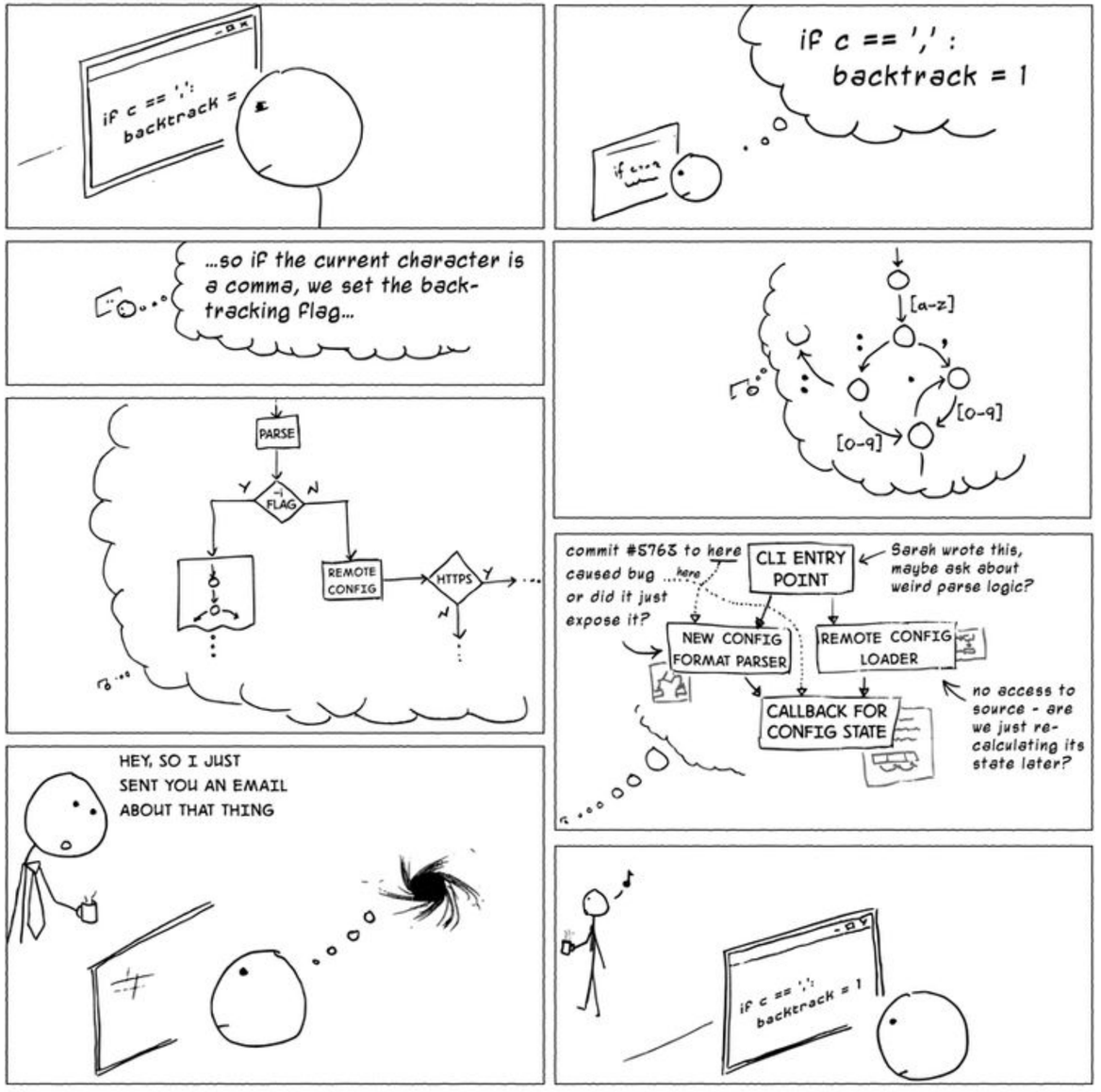
- ✅ Faster development
- ✅ Less boilerplate code
- ✅ Simpler refactoring
- ✅ Faster prototyping

Infrequent users mention:

- ❌ Manual reviews and adjustments
- ❌ Critical evaluation skills
- ✅ Technical exploration
- ✅ Faster development

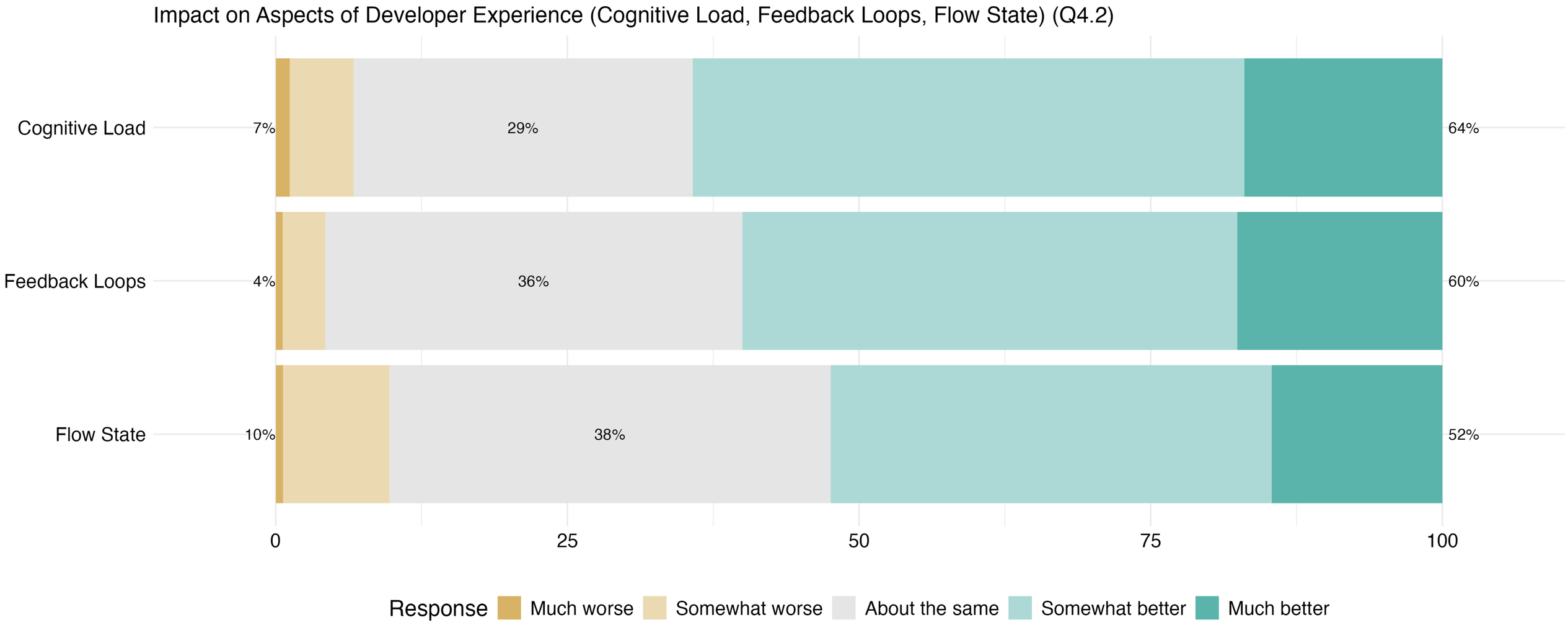
# How does AI make us more productive?

# THIS IS WHY YOU SHOULDN'T INTERRUPT A PROGRAMMER

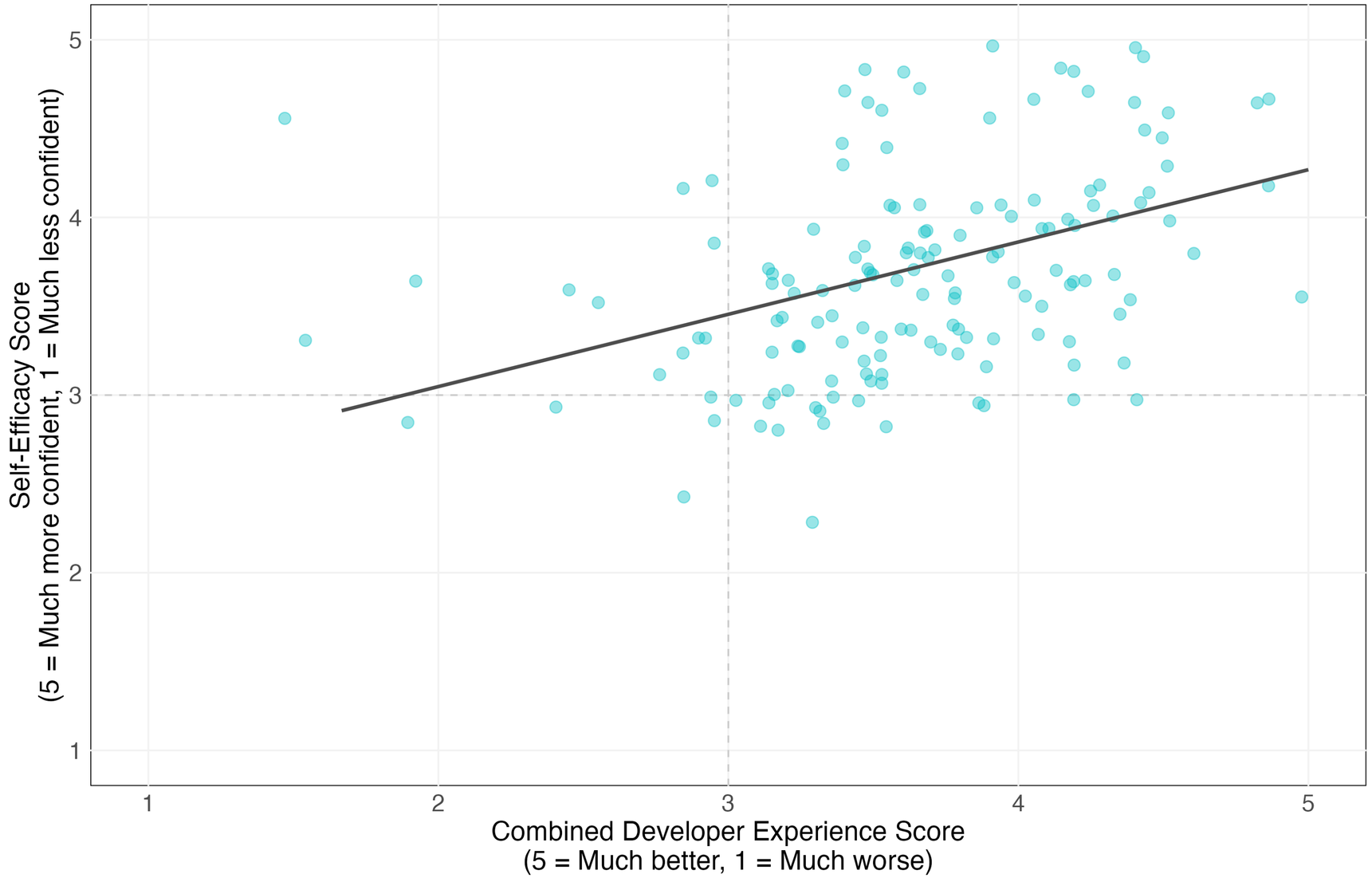



 heeris.id.au

# Improved Developer Experience

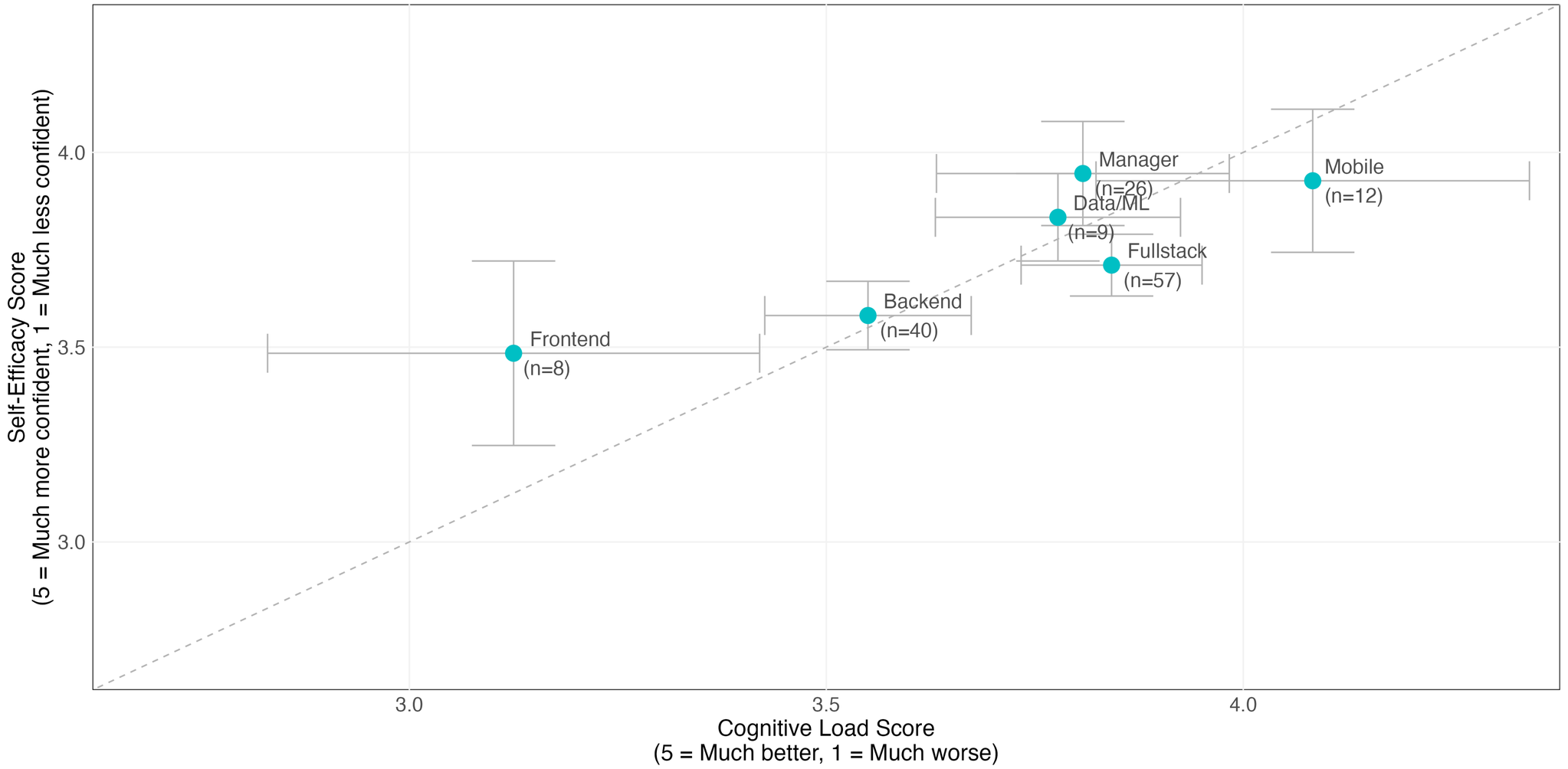


# Developer Experience & Self-Efficacy

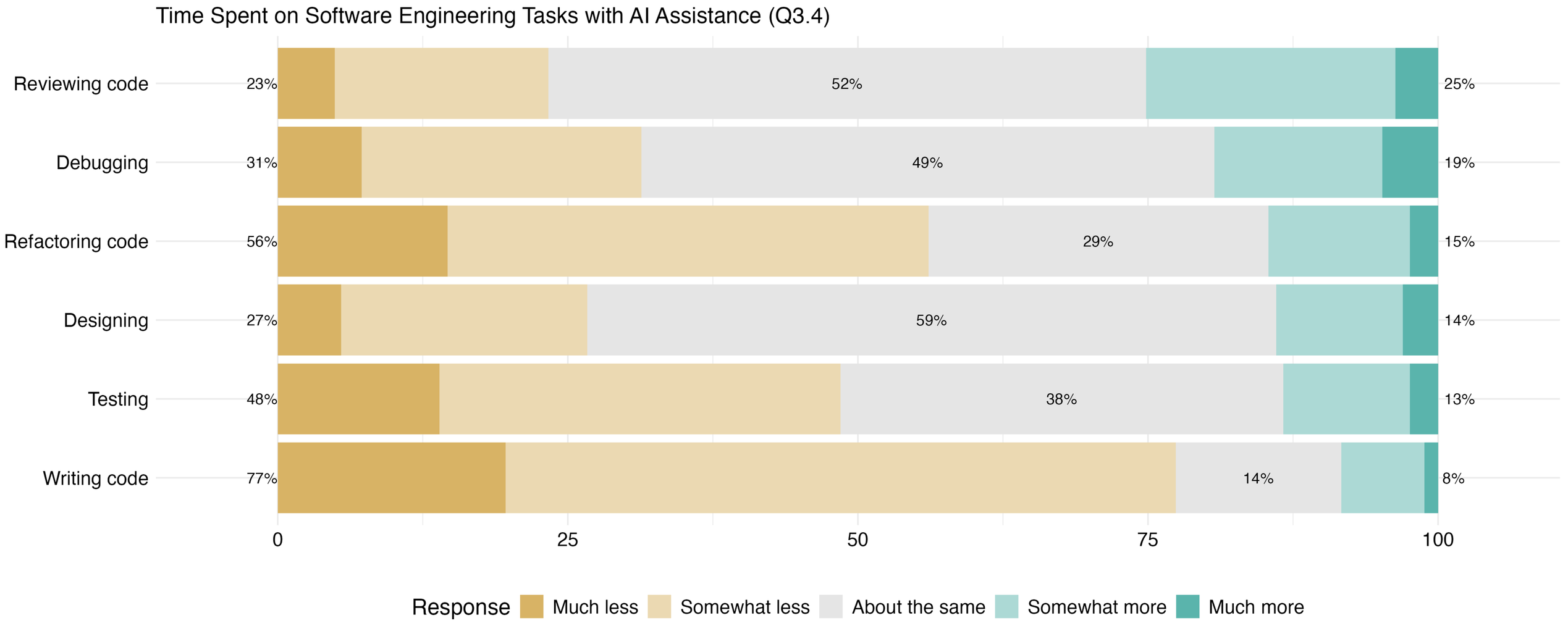


# Cognitive Load & Self-Efficacy by Role

Points above line: Greater improvement in self-efficacy
Points below line: Greater improvement in cognitive load



# Shifts in Task Focus



"When I am stuck this helps a lot. It's my coding buddy."

"Generating a bunch of boilerplate terraform is much easier now"

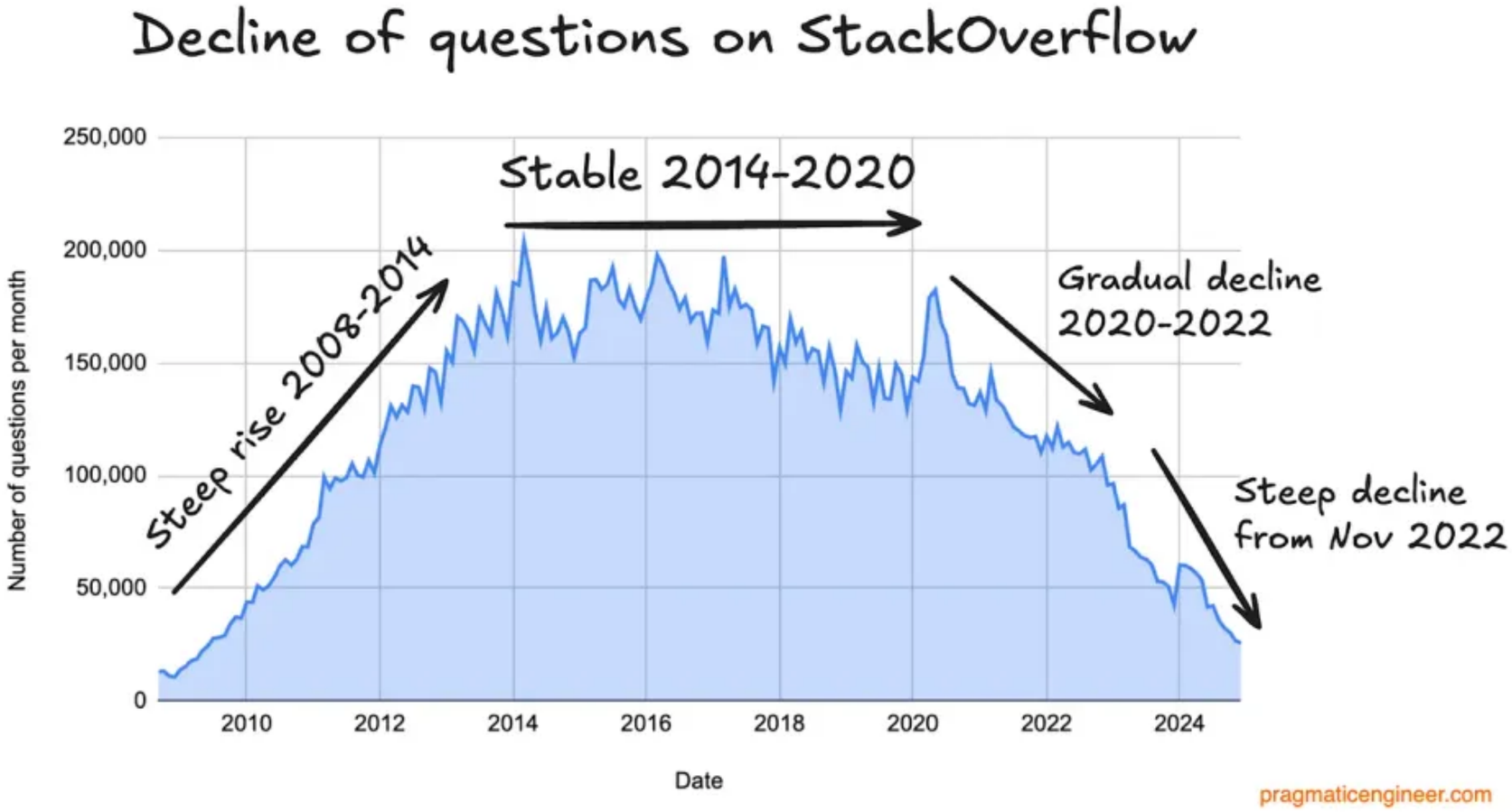
"It can find solutions which would otherwise require search and experimentation"

"I wouldn't have taken up a react front end role, AI assistant has enabled me to do it"

"Significantly helped me get new projects going"

"A situation whereby I had to integrate a technology I haven't used before, AI helped me effortlessly"

# Replacing Traditional Tools




https://blog.pragmaticengineer.com/are-llms-making-stackoverflow-irrelevant/

# Where AI Helps

## Time Efficiency

- Increased speed
- Faster boilerplate code
- Faster prototyping
- Easier test generation
- Get answers faster

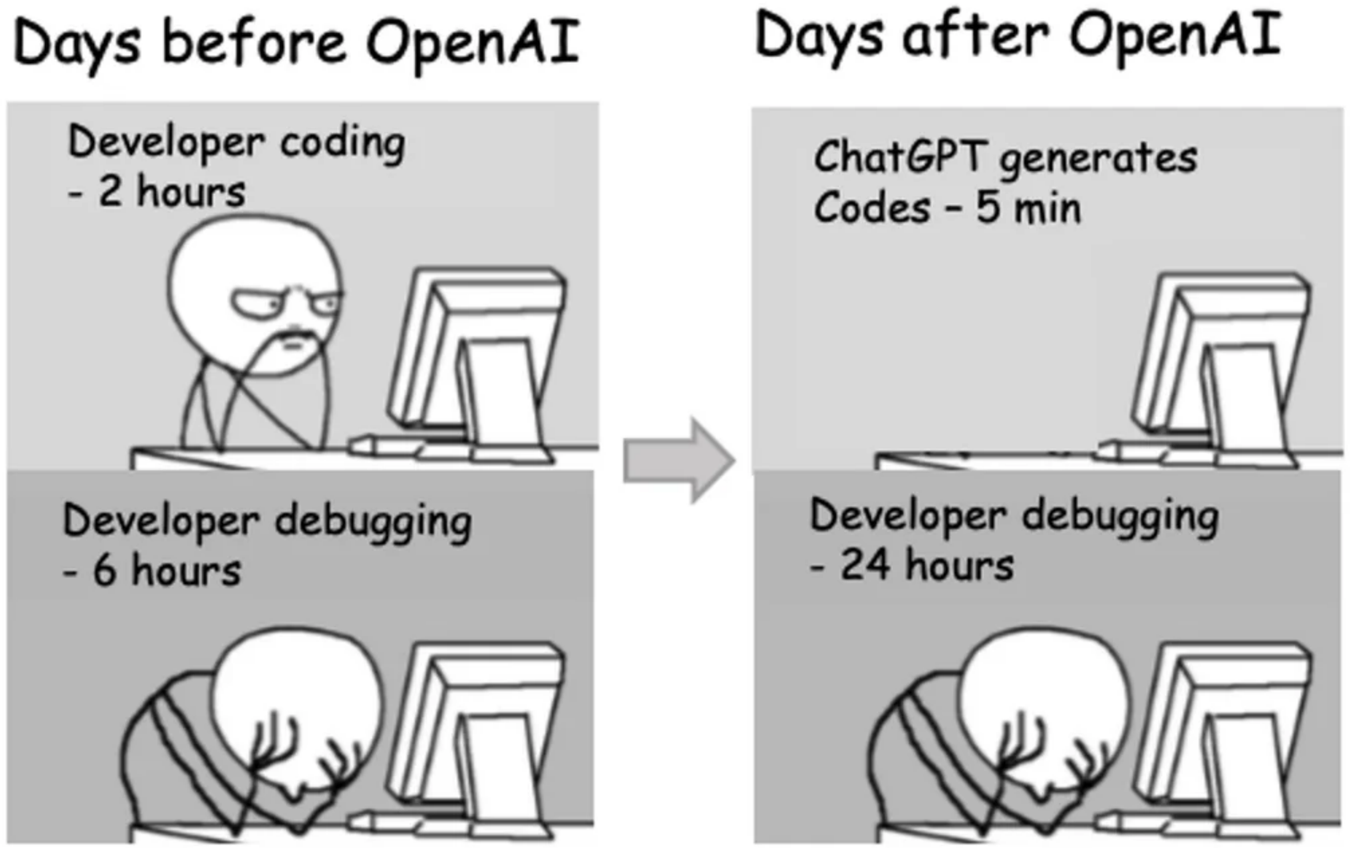
## Cognitive Support

- Enables technical exploration
- Explains legacy or complex code
- Makes a good ‘rubber ducky’
- Allows more focus on problem definition and system design

## Improved Flow State

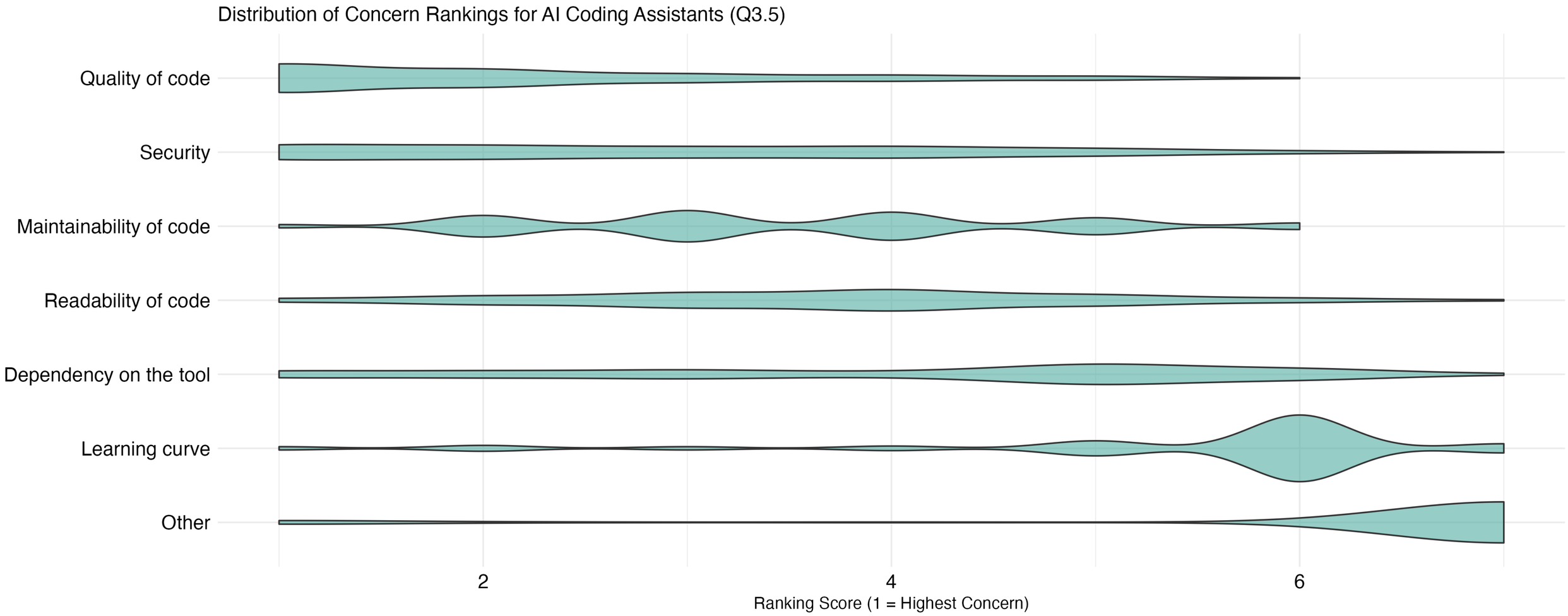
- Replaces traditional tools (e.g. Google, StackOverflow, documentation)
- Keeps software engineers in the IDE

# So where's the catch?




https://www.reddit.com/r/meme/comments/14pm80n/chatgpt_is_actually_a_problem_for_developers/

# Software Engineers Have Some Concerns



"I have found large proposals for whole objects generally poor quality in Swift, often including code which will not build, contains non-existents api functions or does build but is poorly implemented."

"The code it created was literally not functional code"

"Asking the AI to fix it, giving it the error messages, usually got me in a loop, where after some responses, it would give me the first version again..."

"Only time it has hindered me so far is when I was starting out when i didn't know how to construct prompts."

"Sometimes it is quite painful to make it do what I want"

"I also realised how dependent I became on it for writing tests at work"

# Where AI Hinders

## Code Quality & Manual Adjustment

- Low-quality output
- Non-existent APIs
- Outdated suggestions
- Effort to refine outputs

## Time Wasted & Frustration

- Prompt loops
- Ineffective solutions

## Job Displacement & Skill Erosion

- Dependency on AI
- Job security threat

# Related Research on Job Security Threat

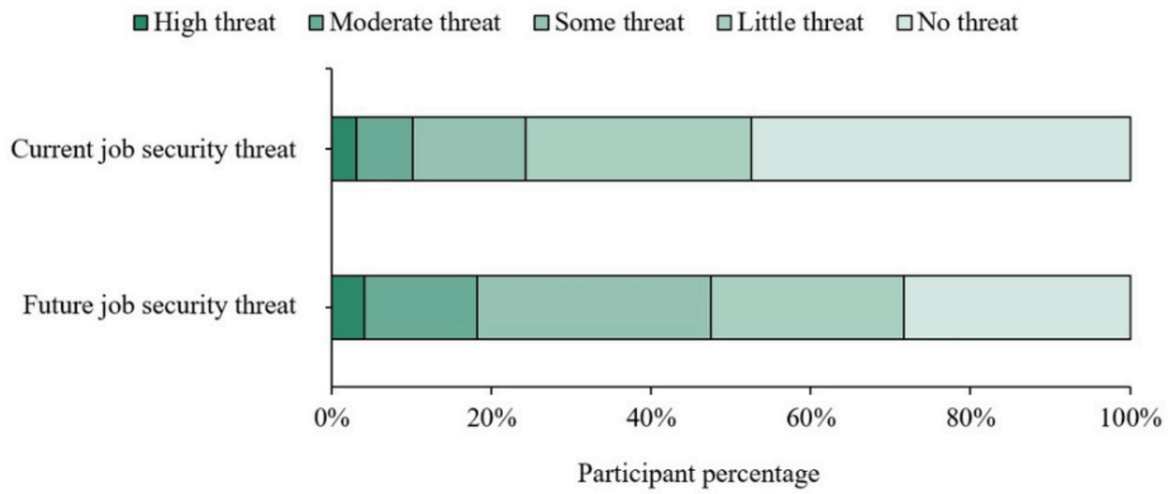



**Fig. 4.** Participants' perceived current and future job security threats.

Kuhaila, M. A., Mathew, S. S., Khalil, A., Berengueres, J., & Shah, S. J. H. (2024). "Will I be replaced?" Assessing ChatGPT's effect on software development and programmer perceptions of AI tools. arXiv:2406.07765.

## Pluralsight research finds 74% of software developers are planning to upskill in AI-assisted coding

Oct 18, 2023

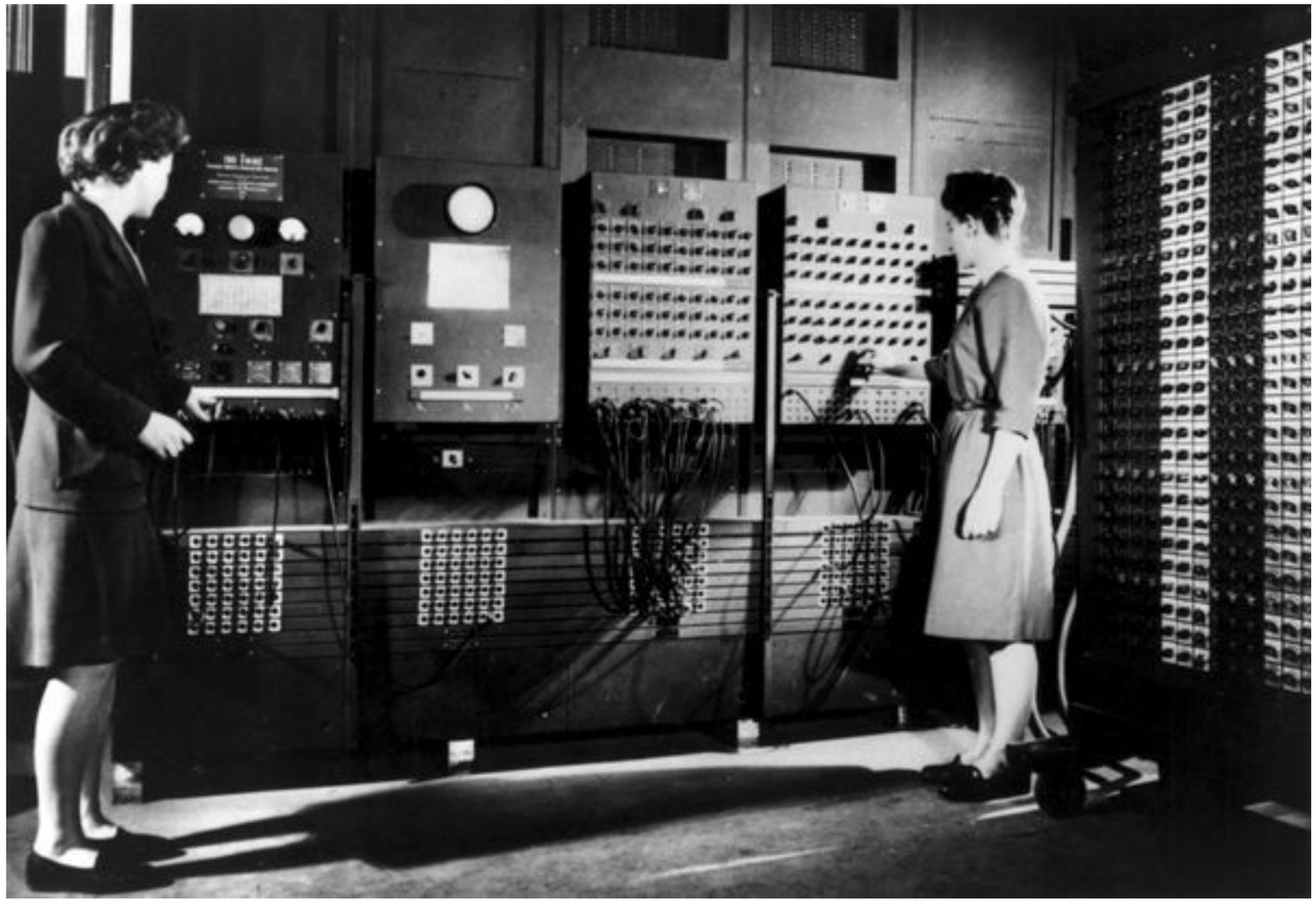
***Amid the rise of AI-assisted software development, the report also finds 45% of developers feel the value of their current skill sets are threatened.***

https://www.pluralsight.com/newsroom/press-releases/pluralsight-research-finds-74--of-software-developers-are-planni

# What does this mean for software engineers?

# The Role Isn't Disappearing, It's Evolving

- It's transforming into one focused on defining **intent** and **composing** solutions
- This isn't the first time:
  - Connecting physical circuits →
  - Writing binary code →
  - Writing assembly code →
  - Writing higher-level language code →
  - Writing specifications in natural language



Betty Jean Jennings and Frances Bilas program the ENIAC in 1946. Computer History Museum

# Core Principles Still Reign

## Software engineering fundamentals

- Foundational computer science concepts
- Strong software design principles
- Design patterns

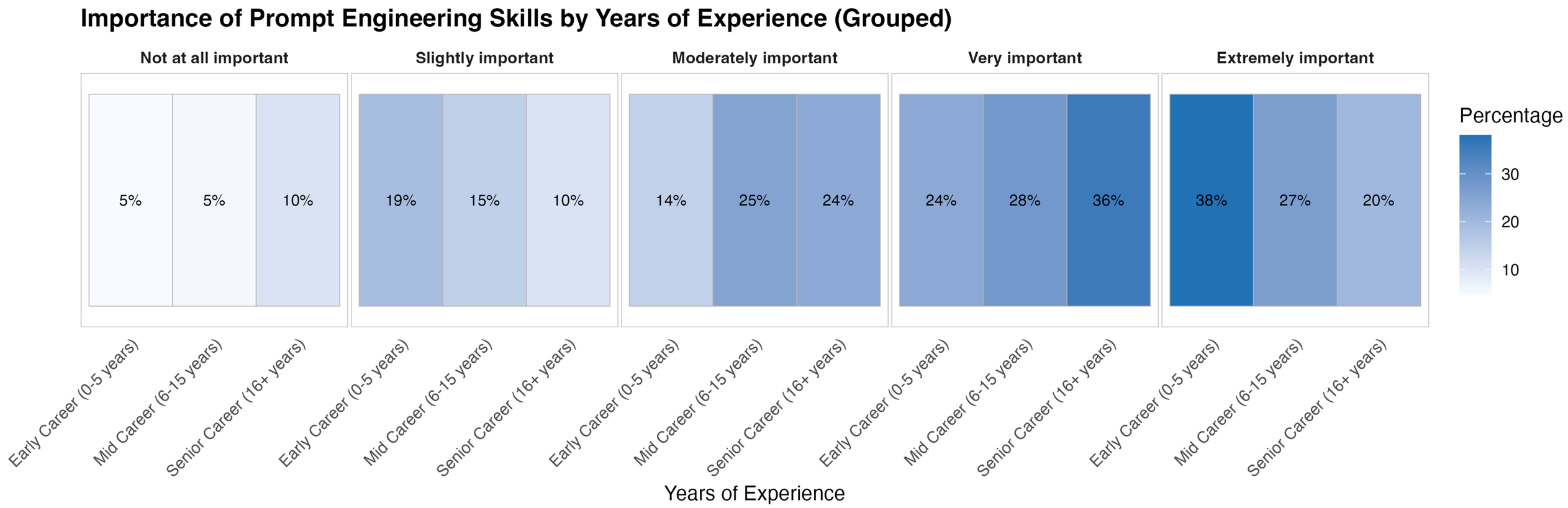
## Critical thinking & verification

- Clear requirements
- Testing and code reviews
- Security risks

## Soft skills, curiosity & learning

- Communication and collaboration
- Business acumen and problem context
- Human insight and creativity

# Prompt Engineering as a Natural Evolution

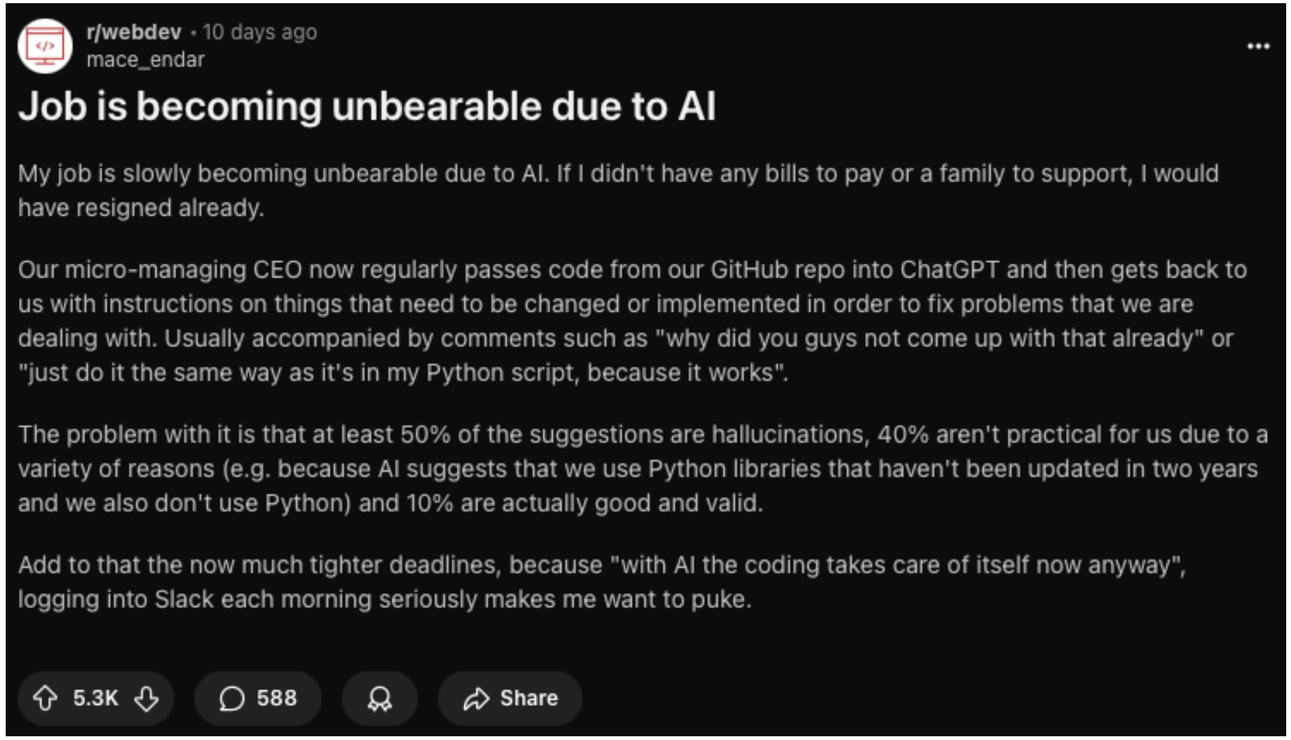



Those earlier in their careers value prompt engineering as a skill the most

# What does this mean for engineering leadership?

# AI Mismanagement: A Cautionary Tale

- Mismanagement of AI leads to engineers feeling replaced, devalued and frustrated
- This creates low morale, high churn, and poor adoption of AI tools




https://www.reddit.com/r/webdev/comments/1iixhhl/job_is_becoming_unbearable_due_to_ai/

# Human-AI Teaming - What Leaders Must Do

## Structured AI onboarding

- AI literacy training
- Build calibrated trust in AI by testing its limits in sandboxed environments
- Bridge AI acceleration with core engineering skills

## Evolving engineering roles

- Acknowledge AI changes how engineers work
- Shift from implementation to intent
- Architecture & system design
- Critical evaluation of AI outputs

## Redefining success metrics

- Traditional metrics don’t reflect AI’s impact (e.g LOC, velocity)
- Track AI’s impact on quality and maintainability
- Measure how well engineers collaborate with AI tools

# The future of software engineering

# It’s Time To Shift Software Engineering Left

- An opportunity to redefine the craft
- It’s time for software engineering to **shift left**, designing systems and orchestrating the build
- Focus on making great products that solve real problems and makes customer happy
- Adapt the whole SDLC

“AI won't take your job. It's somebody using AI that will take your job.”

*Richard Baldwin, 2023 World Economic Forum Growth Summit*

# The Future of AI-Driven Software Engineering

Valerio Terragni, Annie Vella, Partha Roop, and Kelly Blincoe. 2025. The Future of AI-Driven Software Engineering. ACM Transactions on Software Engineering and Methodology. Just Accepted (January 2025). https://doi.org/10.1145/3715003

# The Future Engineer: Adapting, Evolving, Thriving

- AI isn’t replacing software engineers - it’s redefining what software engineering is
- Our role is evolving - we’re becoming system thinkers, AI collaborators, and problem solvers at a higher level
- In summary, **yes**, AI is having a large impact on software engineering and we must **adapt**

# Contact Details

Website: https://annievella.com

LinkedIn: https://www.linkedin.com/in/annievella/

Twitter/X: https://x.com/codefrenzy