The Rise of Stochastic Parrots for Developers

Marco D'Ambros, Andrea Mocci

codelounge

codelounge

Marco D'Ambros
Director

Michele Lanza
Co-Director

Andrea Mocci
Junior Group Leader

Davide Paolo Tua
Senior R&D Engineer

Jesper Findahl
Senior R&D Engineer

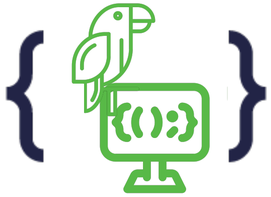
Valerie Burgener
Junior R&D Engineer

Aron Fiechter
Junior R&D Engineer

Francesco Bresciani
Junior Developer

Roberto Minelli
Ambassador

Mauro Prevostini
Ambassador



AI/LLM
Hype

Supporting
Developers

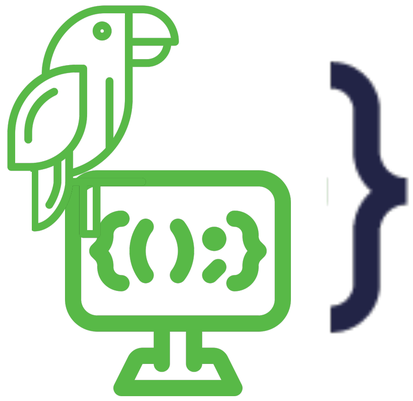
Parrots as
Companions

Parrots as
Task Solvers

Conclusion

The rise of
Stochastic Parrots for Developers

AI/LLM
Hype



Supporting
Developers

Parrots as
Companions

hype[1] | hʌɪp | *informal*

noun *[mass noun]*

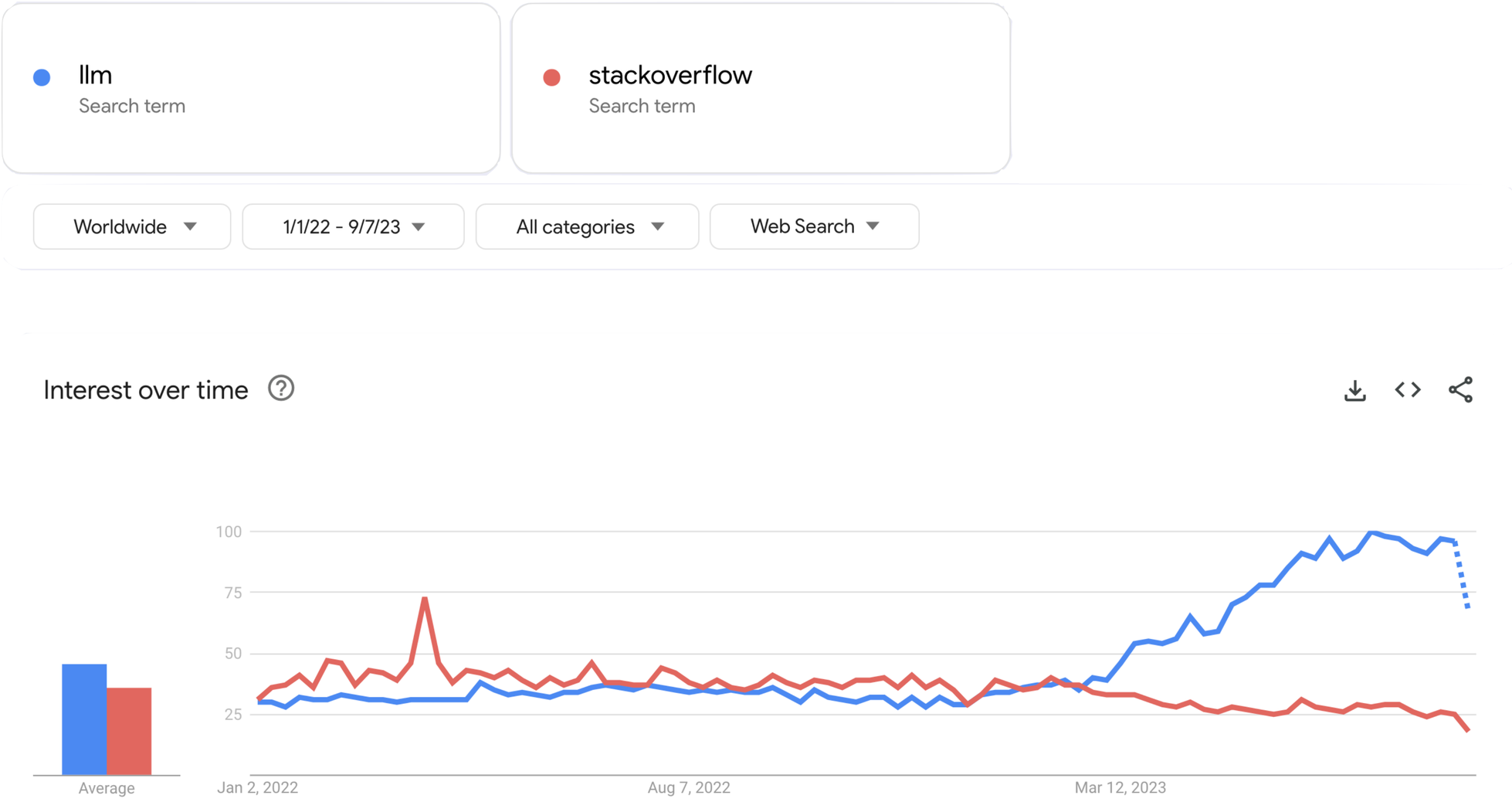
extravagant or intensive publicity or promotion: *his first album hit the stores amid a storm of hype*.

- *[count noun]* **a deception carried out for the sake of publicity:** *is his comeback a hype?*

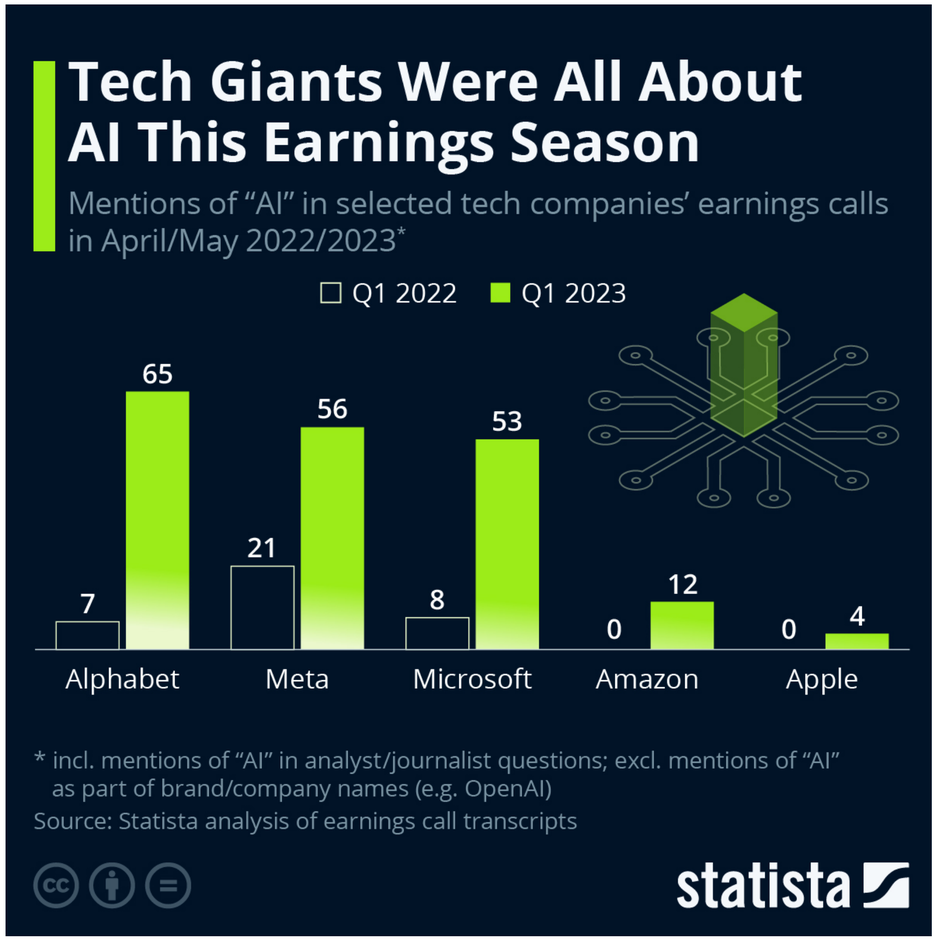
verb *[with object]*

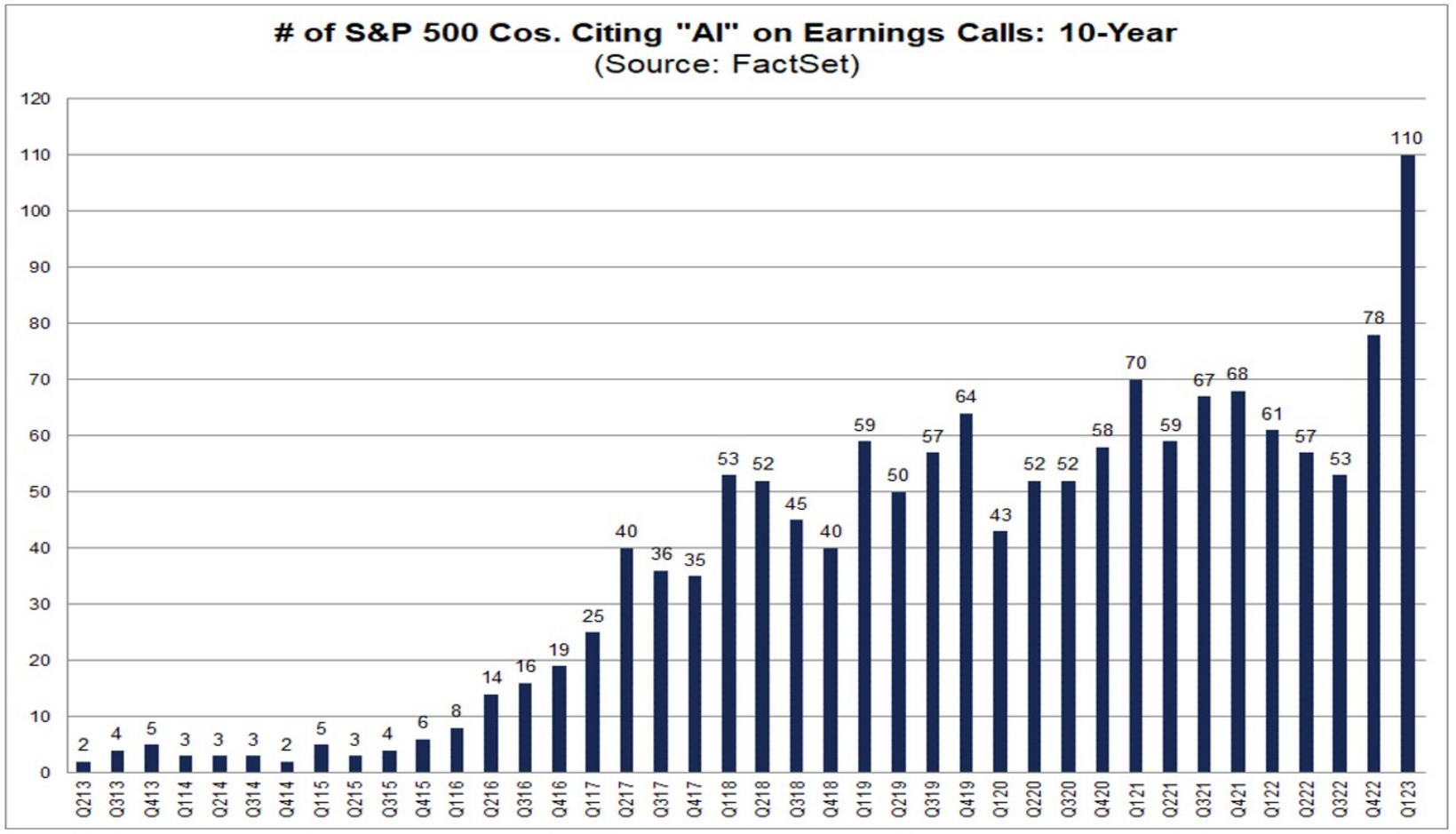
promote or publicize (a product or idea) intensively, often exaggerating its benefits: *an industry quick to hype its products* | *they were* ***hyping up*** *a new anti-poverty idea*.

Is there an AI/LLM hype going on?

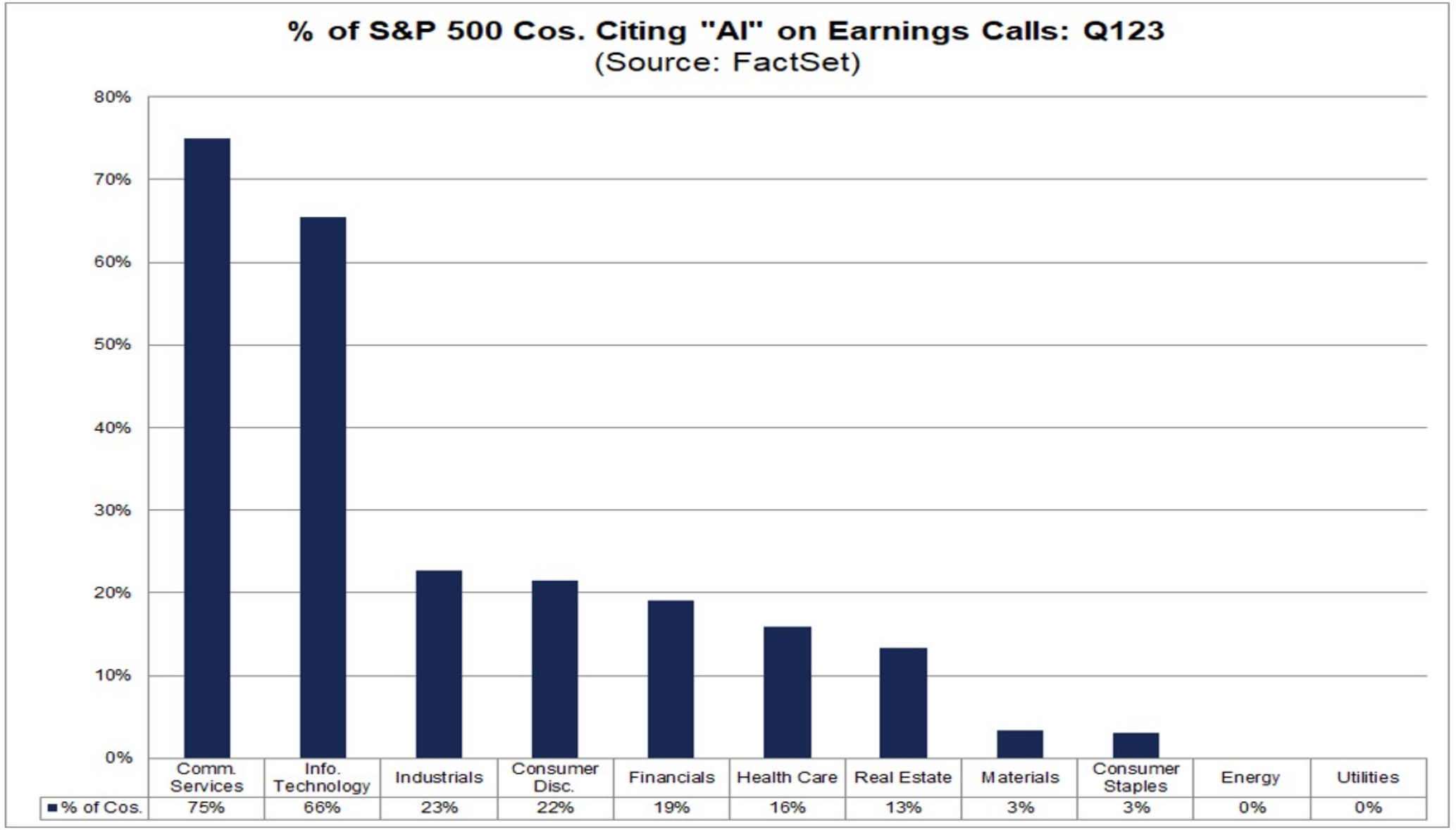


Is there an AI/LLM hype going on?





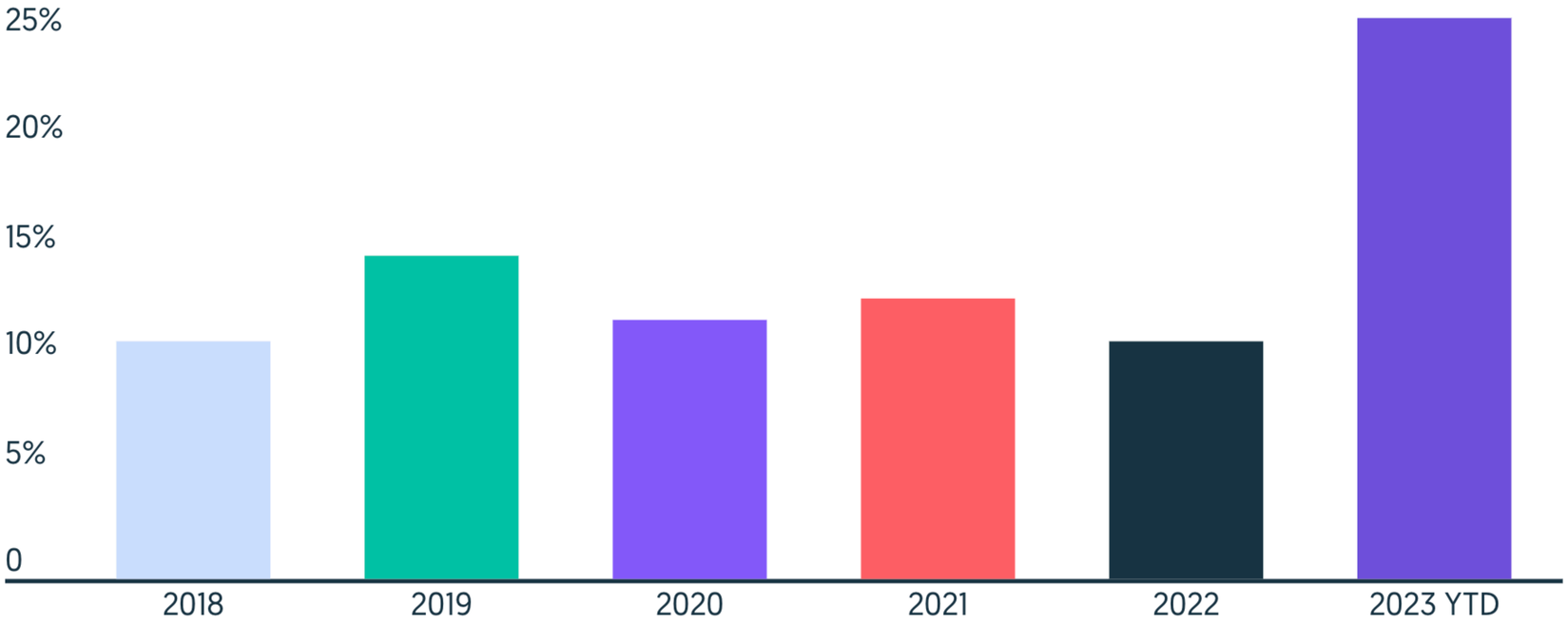
Is there an AI/LLM hype going on?
Existing companies



Is there an AI/LLM hype going on?
Existing companies

Percentage of US Venture Funding Going To AI-Related Startups

Includes seed through growth-stage rounds.

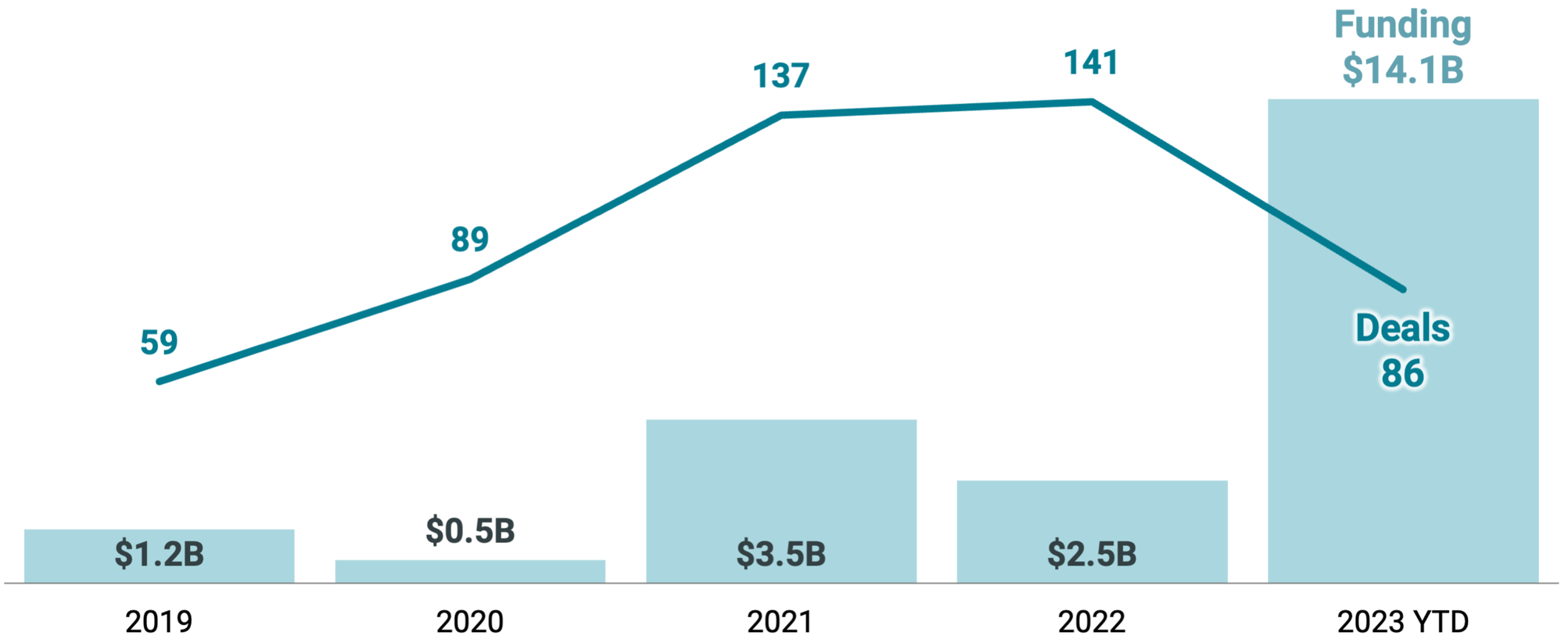


crunchbase

Is there an AI/LLM hype going on?
Venture funding

Investor interest in generative AI soars in 2023

Disclosed equity funding & deals (as of 06/30/2023)



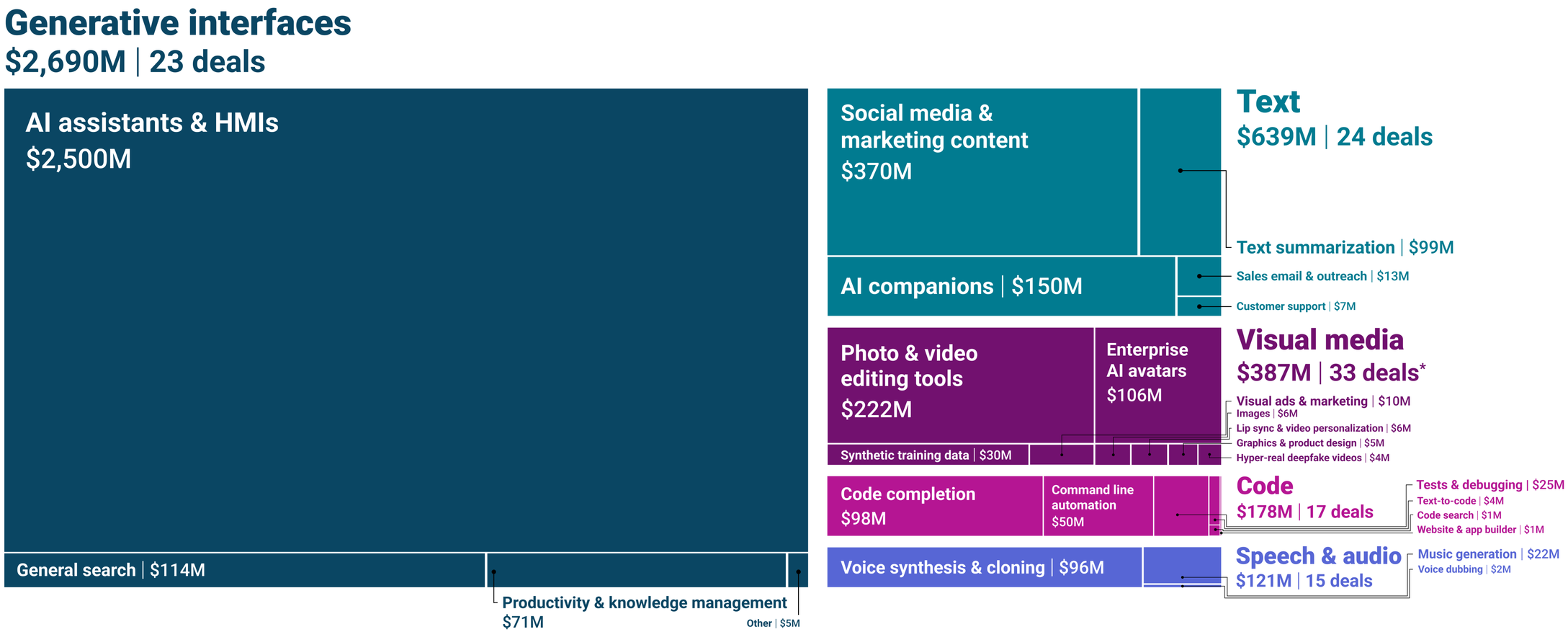
Source: CB Insights

CBINSIGHTS

Is there an AI/LLM hype going on?
Venture funding

Where is all the money going in generative AI?

Distribution of generative AI funding, Q3'22 – Q2'23



Source: CB Insights. Based on an analysis of 210+ generative AI companies building cross-industry enterprise solutions; excludes deals to industry-specific companies and model developers such as OpenAI.
*Includes 1 deal in motion capture animation and 1 deal in synthetic anonymization with undisclosed funding.

CB INSIGHTS

Is there an AI/LLM hype going on?
Venture funding

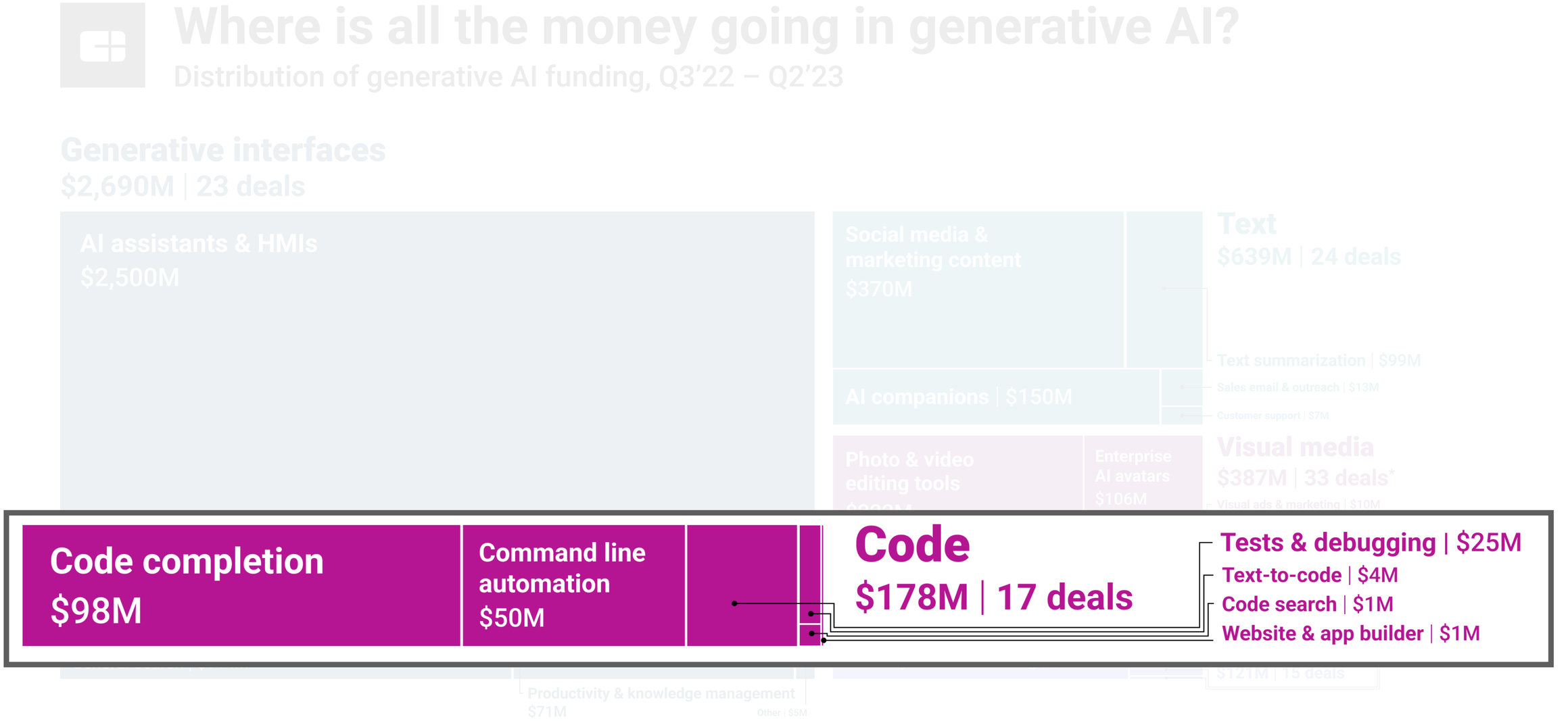
Where is all the money going in generative AI?

Distribution of generative AI funding, Q3'22 – Q2'23

Source: CB Insights. Based on an analysis of 210+ generative AI companies building cross-industry enterprise solutions; excludes deals to industry-specific companies and model developers such as OpenAI.
*Includes 1 deal in motion capture animation and 1 deal in synthetic anonymization with undisclosed funding.

CBINSIGHTS

Is there an AI/LLM hype going on?
Venture funding



Is there an AI/LLM hype going on?
Venture funding

ICSE 2024: out of 68 papers in the main track, 24 (35%) are related to LLM/DL

CoderEval: A Benchmark of Pragmatic Code Generation with Generative Pre-trained Models

Hao Yu [1] Bo Shen [2] Dezhi Ran [1] Jiaxin Zhang [2] Qi Zhang [2] Yuchi Ma [2]
Guangtai Liang [2] Ying Li [1] Tao Xie [1] Qianxiang Wang [2]

Abstract

Code generation models based on the pre-training and fine-tuning paradigm have been increasingly attempted by both academia and industry, resulting in well-known industrial models such as Codex, CodeGen, and PanGu-Coder. To validate the performance of these models, multiple existing benchmarks (e.g., AiXBench and HumanEval) are proposed, including only cases of generating a standalone function, i.e., a function that invokes or accesses only built-in functions and standard libraries. However, standalone functions constitute only about 30% of functions from real open-source projects. To assess a model's performance for pragmatic code generation (i.e., code generation for real settings of open source or proprietary code), in this paper, we propose a benchmark named CoderEval of pragmatic code generation with generative pre-trained models. Compared with the widely-used HumanEval benchmark from OpenAI, CoderEval can be used to assess the performance of models against pragmatic code generation beyond just generating standalone functions. Through the evaluation of three public available models (CodeGen, PanGu-Coder, and Codex) on CoderEval, we analyze and discuss the current progress and future directions of pragmatic code generation with a generative pre-trained model.

1. Introduction

Recent years have seen a trend to tackle open-domain code generation tasks with machine learning techniques, especially large generative pre-trained language models (Black et al., 2021; Brown et al., 2020; Mike Lewis, 2019; Radford et al., 2018) based on Transformer (Vaswani et al., 2017), such as Codex (Chen et al., 2021), AlphaCode (Li et al., 2022), In-Coder (Fried et al., 2022), CodeGen (Nijkamp et al., 2022), and PanGu-Coder (Christopoulou et al., 2022). These models can generate both standalone functions (i.e., a function that invokes or accesses only Python built-in functions and standard libraries) and non-standalone functions.

These recent efforts typically assess the performance of their models with these existing benchmarks such as HumanEval (Chen et al., 2021) for Python, MultiPL-E (Cassano et al., 2022) (which extends HumanEval from Python to 18 programming languages), and AiXBench (Hao et al., 2022) for Java. Released alongside Codex (Chen et al., 2021), HumanEval is a benchmark for Python to measure code generation models on the functional correctness of programs synthesized from docstrings. It consists of 164 samples (as hand-written programming problems and solutions in Python), each of which includes a function signature, docstring, function body, and multiple unit tests (7.7 tests per problem on average). As a multi-language parallel benchmark, MultiPL-E (Cassano et al., 2022) is resulted from extending HumanEval to support 18 programming languages. More recently, AiXBench (Hao et al., 2022) is proposed for Java, containing 175 samples. It also includes 161 additional samples, each of which however does not include unit tests, and thus requires manual evaluation when being used to assess model performance.

However, these existing benchmarks only contains standalone functions, they are limited to assessing the performance of models on pragmatic code generation. After analyzing the 100 most popular open-source projects on GitHub, we find that standalone functions accounted for less than 30% of open-source projects, and most functions either reference third-party lib APIs or variables/constants defined in the current project.

To address this limitation, we propose a benchmark named CoderEval to assess code generation models on pragmatic code generation. CoderEval now supports Python and Java, with 230 functions from 43 Python projects and 230 methods from 10 Java projects. For each function/method, we extract the original docstring/comment, the signature, the code implementation, and the corresponding test code (if

[1]Peking University [2]Huawei Cloud. Correspondence to: Ying Li <li.ying@pku.edu.cn>, Tao Xie <taoxie@pku.edu.cn>, Qianxiang Wang <wangqianxiang@huawei.com>.

Understanding the Usability of AI Programming Assistants

Jenny T. Liang
Carnegie Mellon University
Pittsburgh, PA, USA
jtliang@cs.cmu.edu

Chenyang Yang
Carnegie Mellon University
Pittsburgh, PA, USA
cyang3@cs.cmu.edu

Brad A. Myers
Carnegie Mellon University
Pittsburgh, PA, USA
bam@cs.cmu.edu

ABSTRACT

The software engineering community recently has witnessed widespread deployment of AI programming assistants, such as GitHub Copilot. However, in practice, developers do not accept AI programming assistants' initial suggestions at a high frequency. This leaves a number of open questions related to the usability of these tools. To understand developers' practices while using these tools and the important usability challenges they face, we administered a survey to a large population of developers and received responses from a diverse set of 410 developers. Through a mix of qualitative and quantitative analyses, we found that developers are most motivated to use AI programming assistants because they help developers reduce key-strokes, finish programming tasks quickly, and recall syntax, but resonate less with using them to help brainstorm potential solutions. We also found the most important reasons why developers do *not* use these tools are because these tools do not output code that addresses certain functional or non-functional requirements and because developers have trouble controlling the tool to generate the desired output. Our findings have implications for both creators and users of AI programming assistants, such as designing minimal cognitive effort interactions with these tools to reduce distractions for users while they are programming.

1 INTRODUCTION

The recent widespread deployment of AI programming assistants, such as GitHub Copilot [6] and ChatGPT [1], has introduced a new paradigm to building software that has taken the software engineering community by storm. Some current publications report that AI programming assistants are powerful enough to produce high-quality code suggestions for developers [55, 57]. While some recent studies do not find any significant difference in using AI programming assistants in terms of task completion [52, 56] and code quality [27], other studies find these tools are positively associated with developers' self-perceived productivity [58].

However, in practice, prior literature indicates that developers do not accept AI programming assistants' initial suggestions at a high frequency. Ziegler et al. [58] found that developers accepted 23.3%, 27.9%, and 28.8% of GitHub Copilot's suggestions for TypeScript, JavaScript, and Python respectively. There are many potential reasons for the lack of adoption of AI programming assistants' suggestions. One study shows that developers feel concerned that the generated code may contain defects, may not adhere to the project's coding style, or may be difficult to understand [52]. Other studies report that software developers face barriers in comprehending and debugging generated code to fit their use cases, because they need to have prior knowledge of the underlying programming principles, frameworks, or APIs [12, 56].

While prior work has surfaced initial results about the usability of state-of-the-art AI programming assistants, to our knowledge, they have not systematically investigated the prevalence of usability factors related to these tools. Quantifying the usability of AI programming assistants could help tool creators understand what usability aspects are currently successful in practice. Further, it could help tool creators prioritize features and improvements to the modeling and user interface of these tools in the future, potentially increasing the adoption of these tools and improving the productivity of developers. Usability is an important factor to study in AI programming assistants, since modeling improvements may not necessarily address the needs of developers, rendering these tools hard-to-use or even useless [42].

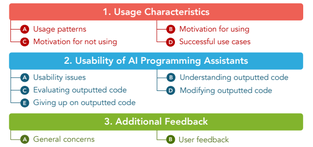


Figure 1: An overview of the topics covered in our usability study of AI programming assistants.

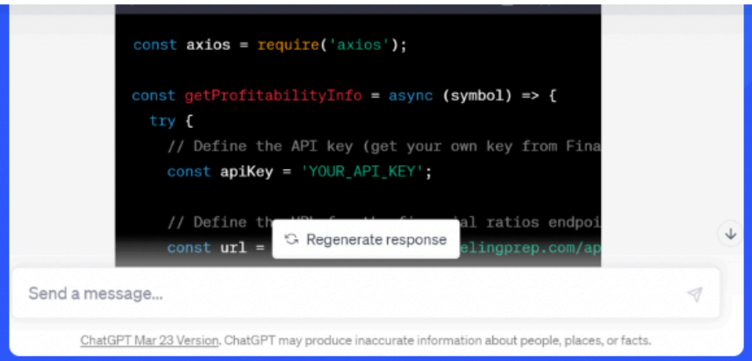
We performed an exploratory qualitative study in January 2023 to understand developers' practices when using AI programming assistants and the importance of the usability challenges that they face. We used a survey as a research instrument to collect large-scale data on these phenomena to understand their importance to the usability of AI programming assistants (see Figure 1).

In the end, we collected and analyzed responses from 410 developers who were recruited from GitHub repositories related to AI programming assistants, such as GitHub Copilot and Tabnine [2]. In summary, we find that:

Usage characteristics of AI programming assistants (Section 4)

(1) Developers who use GitHub Copilot report a median of 30.5% of their code being written with help from the tool.
(2) Developers report the most important reasons why they use AI programming assistants is because of the tools' ability to help developers reduce key-strokes, finish programming tasks quickly, and recall syntax.
(3) The most important reasons why developers do *not* use these tools at all are that the tools generate code does not meet certain functional or non-functional requirements and that it is difficult to control these tools to generate the desired output.

Is there an AI/LLM hype going on?
Research



Stack Overflow is ChatGPT Casualty: Traffic Down 14% in March

by David F. Carr • 5 Min.

April 19, 2023 | Updated June 21, 2023

Table of Contents

Developers increasingly get advice from AI chatbots and GitHub CoPilot rather than Stack Overflow message boards

While traffic to OpenAI's ChatGPT has been growing exponentially, Stack Overflow has been experiencing a steady decline – losing some of its standings as the go-to source developers turn to for answers to coding challenges.

Actually, traffic to Stack Overflow's community website has been dropping since the beginning of 2022. That may be in part because of a related development, the introduction of the CoPilot coding assistant from Microsoft's GitHub business.

Is there an AI/LLM hype going on?
Impact on consolidated practices?

Table of Contents

Developers increasingly get advice from AI chatbots and GitHub CoPilot rather than Stack Overflow message boards

While traffic to OpenAI's ChatGPT has been growing exponentially, Stack Overflow has been experiencing a steady decline – losing some of its standings as the go-to source developers turn to for answers to coding challenges.

Actually, traffic to Stack Overflow's community website has been dropping since the beginning of 2022. That may be in part because of a related development, the introduction of the CoPilot coding assistant from Microsoft's GitHub business. CoPilot is built on top of the same OpenAI large language model as ChatGPT, capable of processing both human language and programming language. A plugin to the widely used Microsoft Visual Studio Code allows developers to have CoPilot write entire functions on their behalf, rather than going to Stack Overflow in search of something to copy and paste. CoPilot now incorporates the latest GPT-4 version of OpenAI's platform.

Key takeaways

- On a year-over-year basis, traffic to Stack Overflow (stackoverflow.com) has been down by an average of 6% every month since January 2022 and was down 13.9% in March.
- ChatGPT doesn't have a year-over-year track record, having only launched at the end of November, but its website (chat.openai.com) has become one of the world's hottest digital properties in that short time, bigger than Microsoft's Bing search engine for worldwide traffic. It attracted 1.6 billion visits in March and another 920.7 million in the first half of April.

https://www.similarweb.com/blog/insights/ai-news/stack-overflow-chatgpt/#toc_H2_0

Is there an AI/LLM hype going on?
Impact on consolidated practices?

Insights into Stack Overflow's t

stackoverflow.blog/2023/08/08/insights-into-stack-overflows-traffic/

Update

Blog

Find something...

Log in Sign up

- Everything
- Productivity
- Career Advice
- AI/ML
- Open Source
- Company
- Releases
- Podcast
- Newsletter

PRODUCTS

Stack Overflow for Teams
Capture, share, & collaborate on knowledge internally.

Advertising
Promote your product or service to developers and technologists.

Talent
Engage the world's technology talent with your employer brand.

AUGUST 8, 2023

Insights into Stack Overflow's traffic

We're setting the record straight.

Over the last few weeks, we've seen inaccurate data and graphs circulating on social media channels

AUTHORS

Des Darilek
Staff

Announcements traffic

RECENT ARTICLES

SEPTEMBER 7, 2023
Computers are learning to decode the language of our minds

SEPTEMBER 5, 2023
Journey to the cloud part II: Migrating Stack Overflow for Teams to Azure

AUGUST 30, 2023
A new look and feel for the Stack Overflow blog

AUGUST 30, 2023

Is there an AI/LLM hype going on?
Impact on consolidated practices?

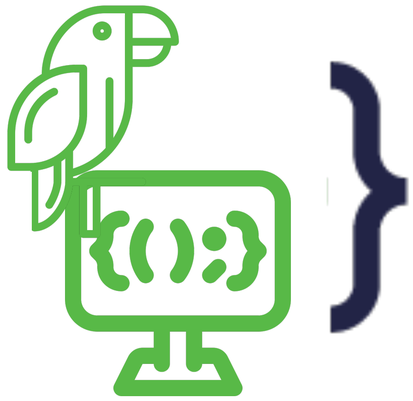
Although we have seen a small decline in traffic, in no way is it what the graph is showing (which some have incorrectly interpreted to be a 50% or 35% decrease). This year, overall, we're seeing an average of ~5% less traffic compared to 2022. Stack Overflow remains a trusted resource for millions of developers and technologists.

As anyone who has worked in the digital space knows, a website's traffic and engagement can be influenced by a variety of factors. Stack Overflow is no different, particularly given the rapid innovation happening as a result of new technologies. In 2020, as tech workers responded to the needs of a remote workforce, we shared information about the “pandemic spikes” we saw in new questions asked around cloud and security. Conversely, in April of this year, we saw an above average traffic decrease (~14%), which we can likely attribute to developers trying GPT-4 after it was released in March. Our traffic also changes based on search algorithms, which have a big influence on how our content is discovered.

The surge in generative AI, like the rise of any other disruptive technology, should cause us to reflect, challenge, and question how we measure success. The future of the internet and the modern tech landscape isn’t going to be measured by web traffic alone—it's about the quality of content, trust in the content, and the communities of experts and human beings curating the content.

Is there an AI/LLM hype going on?
Impact on consolidated practices?

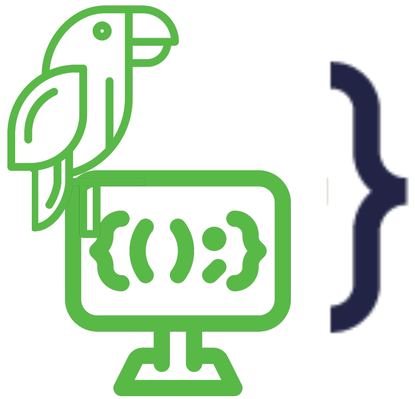
AI/LLM
Hype



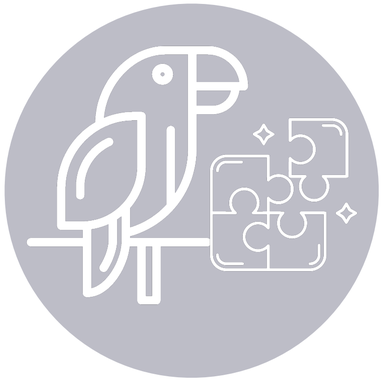
Supporting
Developers

Parrots as
Companions

Supporting
Developers



Parrots as
Companions



Parrots as
Task Solvers

Not just about

GitHub
copilot

codeium

tabnine

Cursor

Not just about

GitHub
copilot

CodeGeeX

BLACKBOX AI

CodeWhisperer

Copy participation link

1 Go to wooclap.com

2 Enter the event code in the top banner

Event code
EMRNUZ

Supporting Developers
What about you?

stackoverflow.com

stackoverflow About Products For Teams Search... Log in Sign up

Home

PUBLIC

Questions

Tags

Users

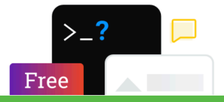
Companies

COLLECTIVES

Explore Collectives

TEAMS

Stack Overflow for Teams – Start collaborating and sharing organizational knowledge.



All Questions

Ask Question

7,323,969 questions with no upvoted or accepted answers

Newest | Active | Bountied 200 | Unanswered | More | Filter

90 votes
2 answers
2k views

android ndk gdb loaded sharedlibraries missing *.oat

Both gdb 7.7 and gbd 7.11 missed some shared libraries when debugging my device (oppo r7s). I've pulled all libraries to local. Here is a complete list of libraries shown...

android android-ndk gdb gdbserver

Joey.Z **4,492** asked Feb 5, 2018 at 3:03

72 votes
1 answer
1k views

Was the origin positioning 'box' removed from Xcode 6?

I don't see this tool in Xcode 6. Did they take it out? Can I re-enable it? I use it all the time.

xcode interface-builder xcode6

dMurdZ **979** asked Sep 16, 2014 at 13:32

69 votes

ReferenceError getValuesOfAutofillInputs, Can't find variable:

Featured on Meta

- Our Design Vision for Stack Overflow and the Stack Exchange network
- Call for volunteer reviewers for an updated search experience: OverflowAI Search
- Discussions experiment launching on NLP Collective
- Temporary policy: Generative AI (e.g., ChatGPT) is banned

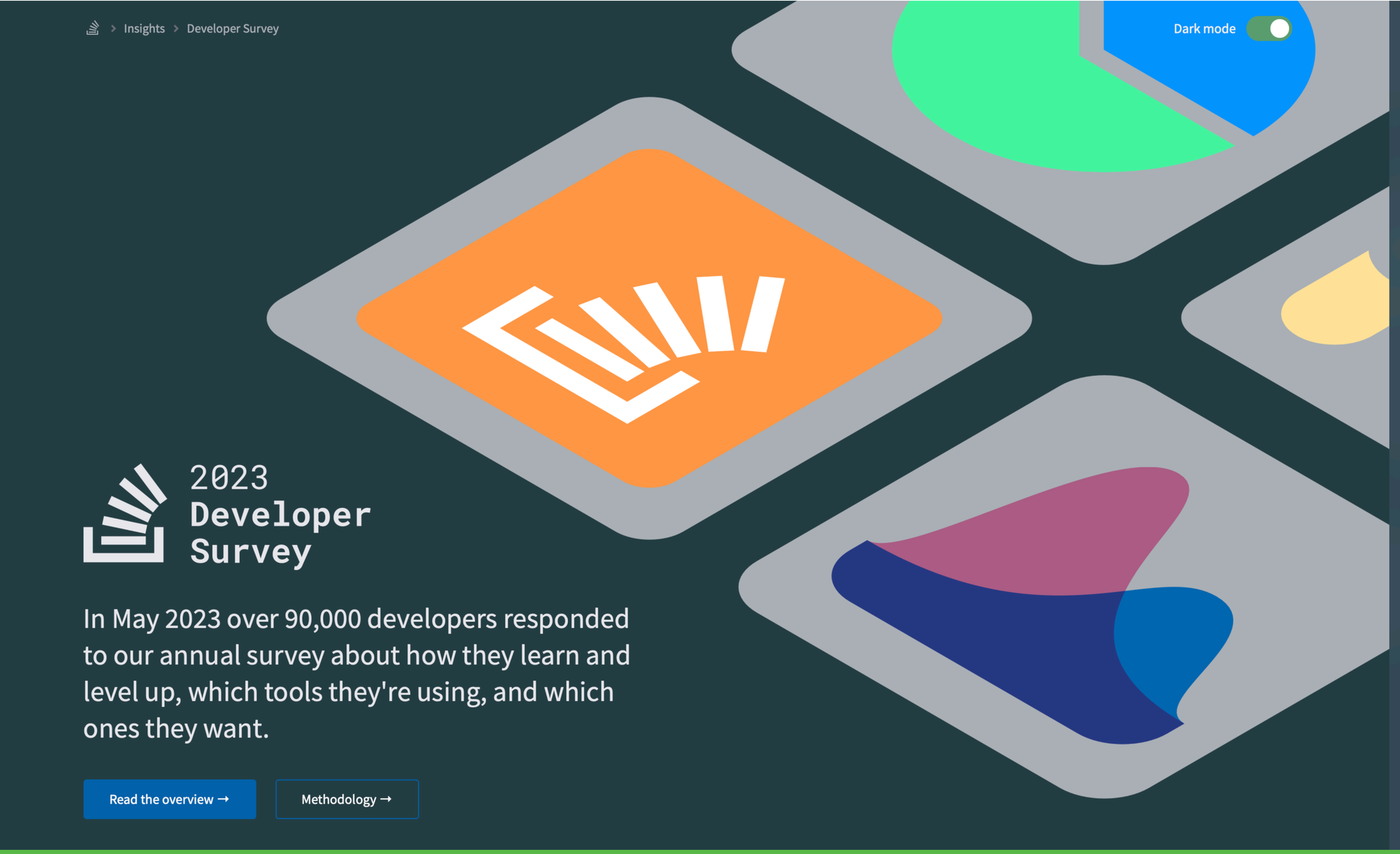
Collectives see all

Microsoft Azure
12k Members
Join

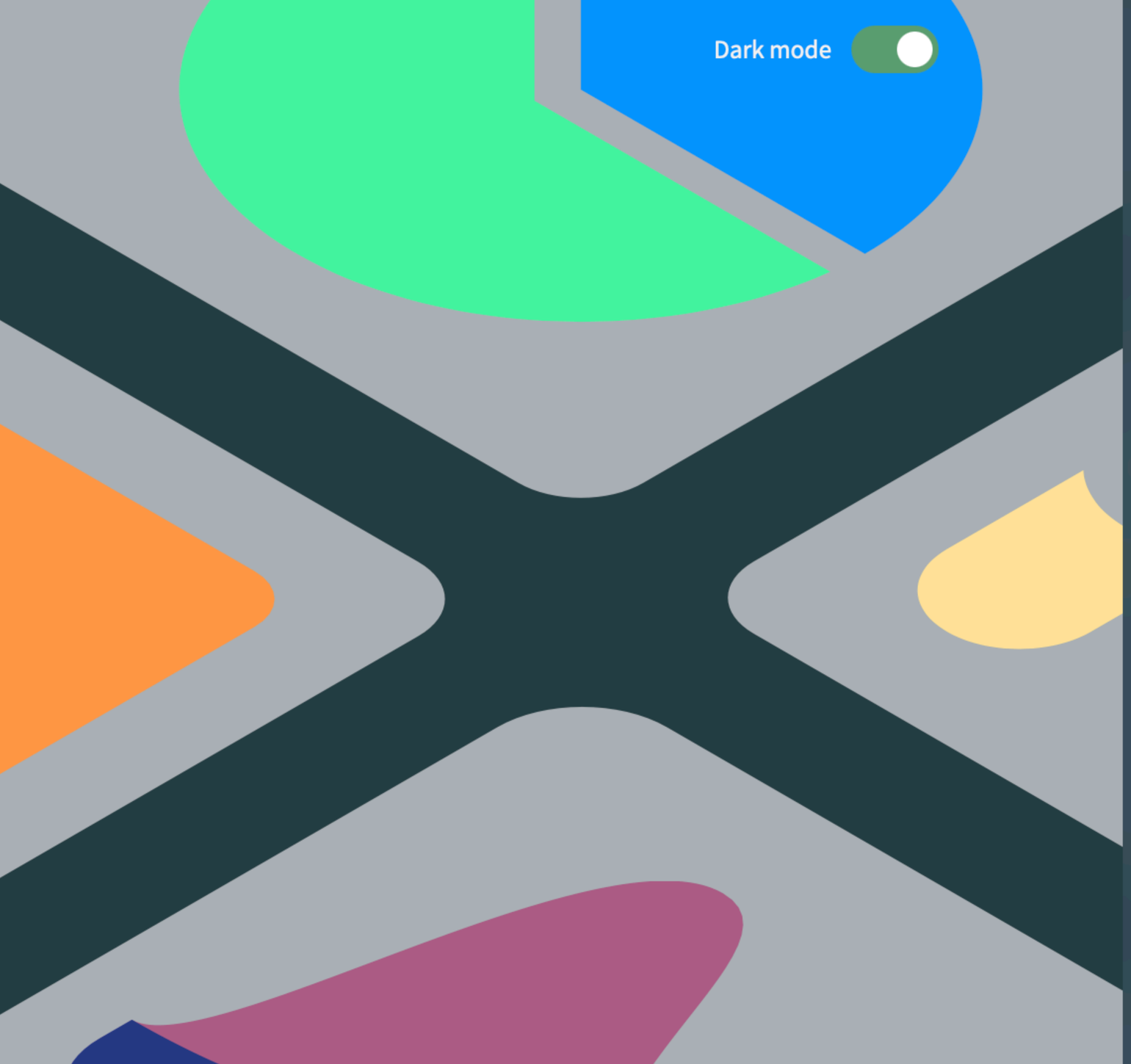
On-premises, hybrid, multicloud, or at the edge—build on your terms with best-in-class...

Twilio
10k Members
Join

Supporting Developers
Stack Overflow



The rise of Stochastic Parrots for Developers



Overview

Developer Profile

Technology

AI

Work

Community

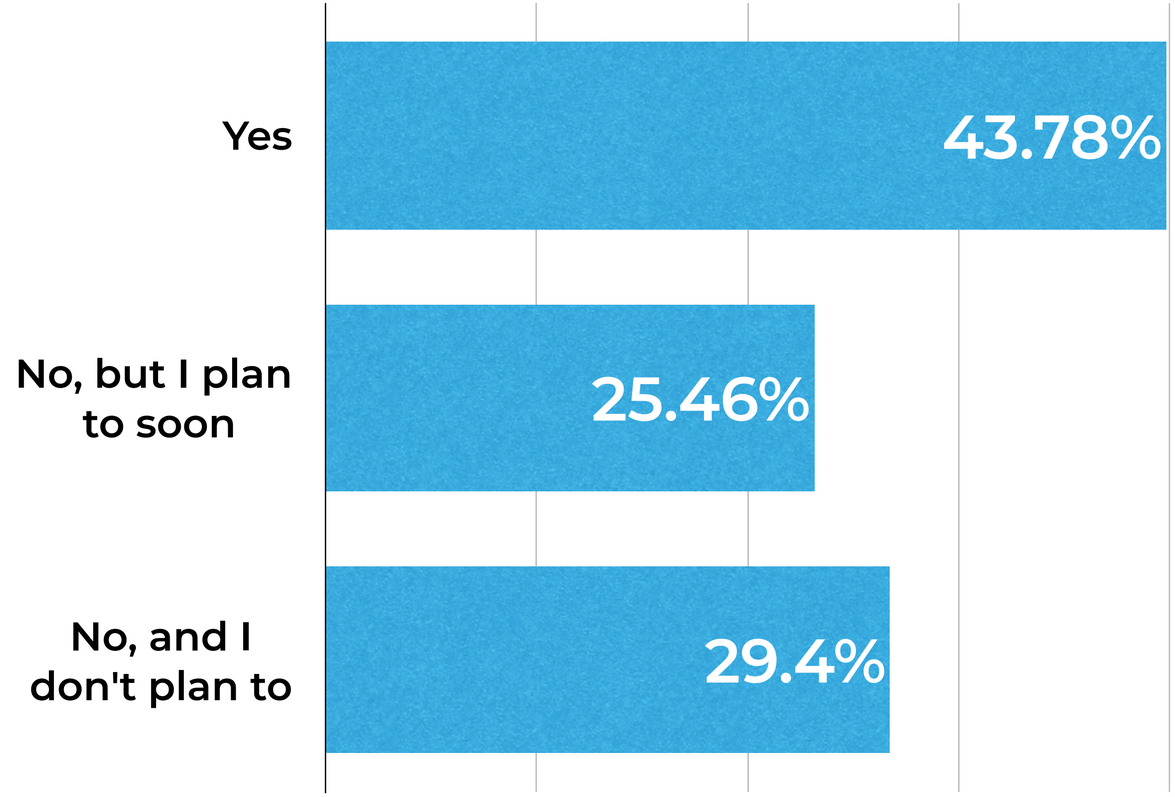
Professional Developers

Methodology

LABS

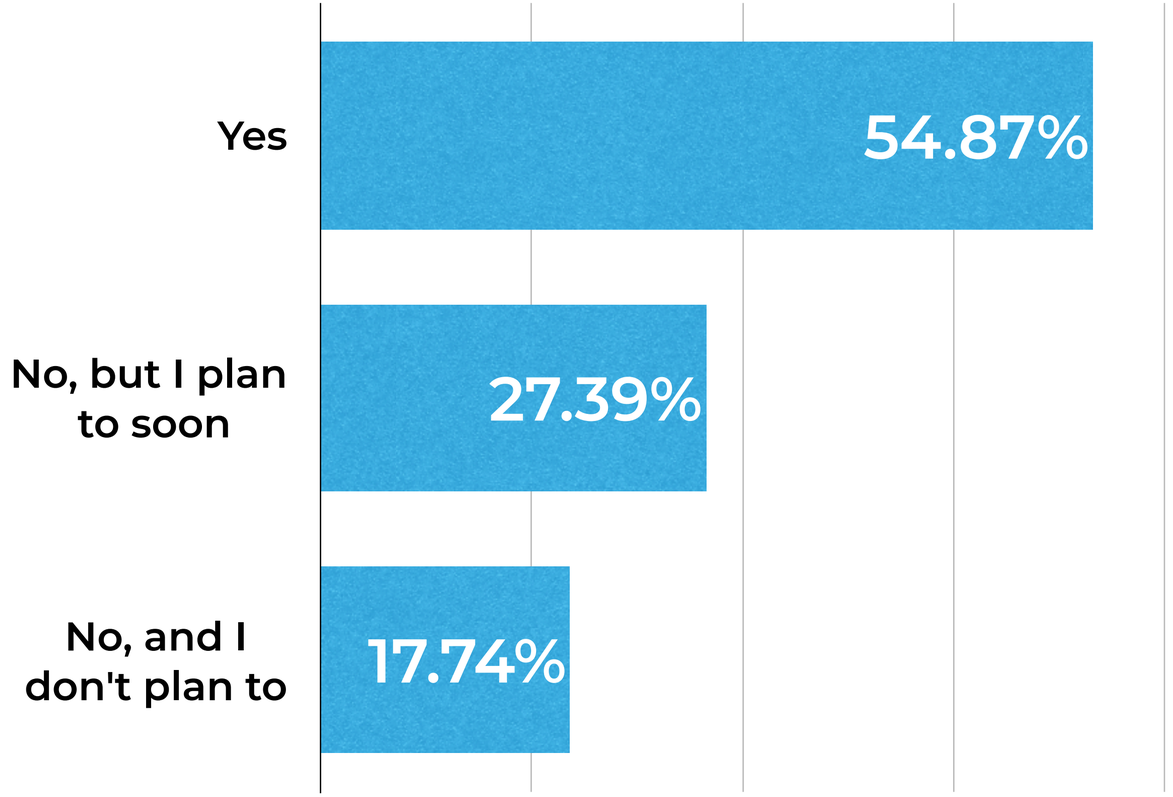
70% of all respondents **are using or are planning to use AI tools in their development process** this year.

Do you currently use AI tools in your development process?



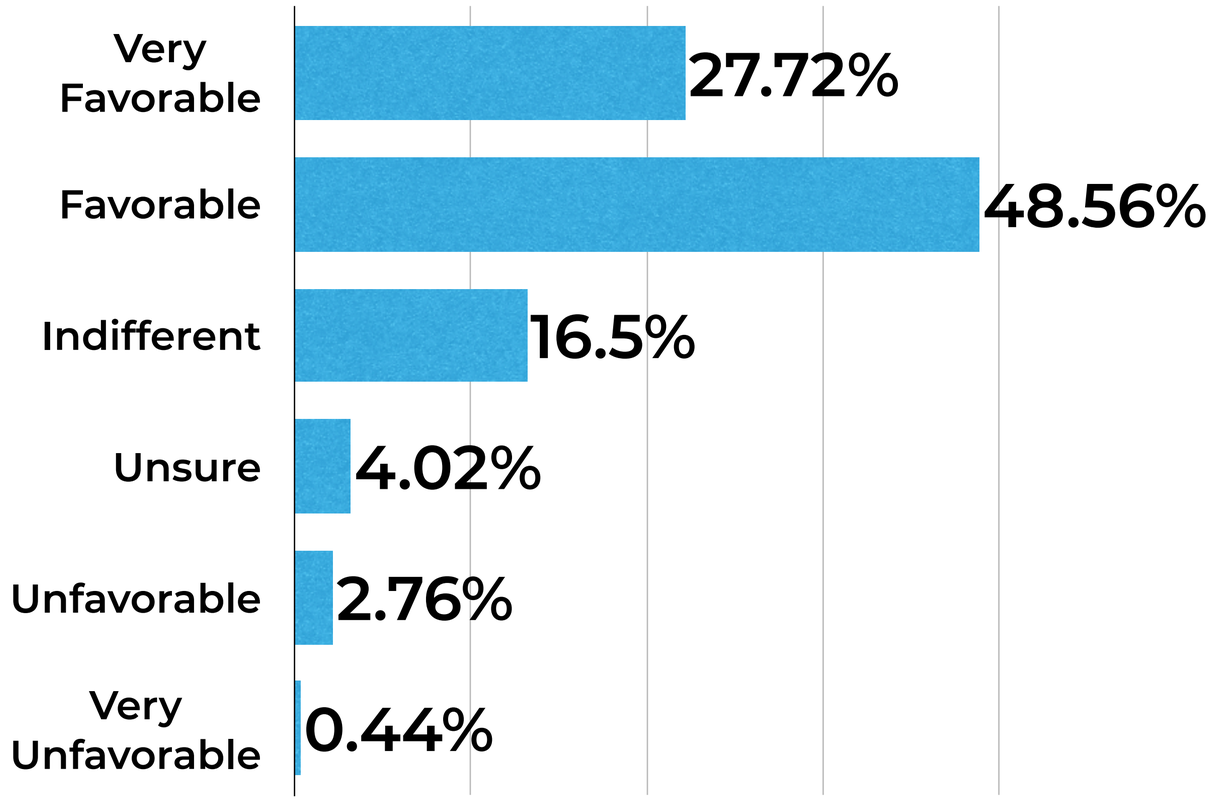
70% of all respondents are using or are planning to use AI tools in their development process this year. Those **learning to code** are **more likely than professional developers** to be using or use AI tools (**82% vs. 70%**).

Results from people learning to code



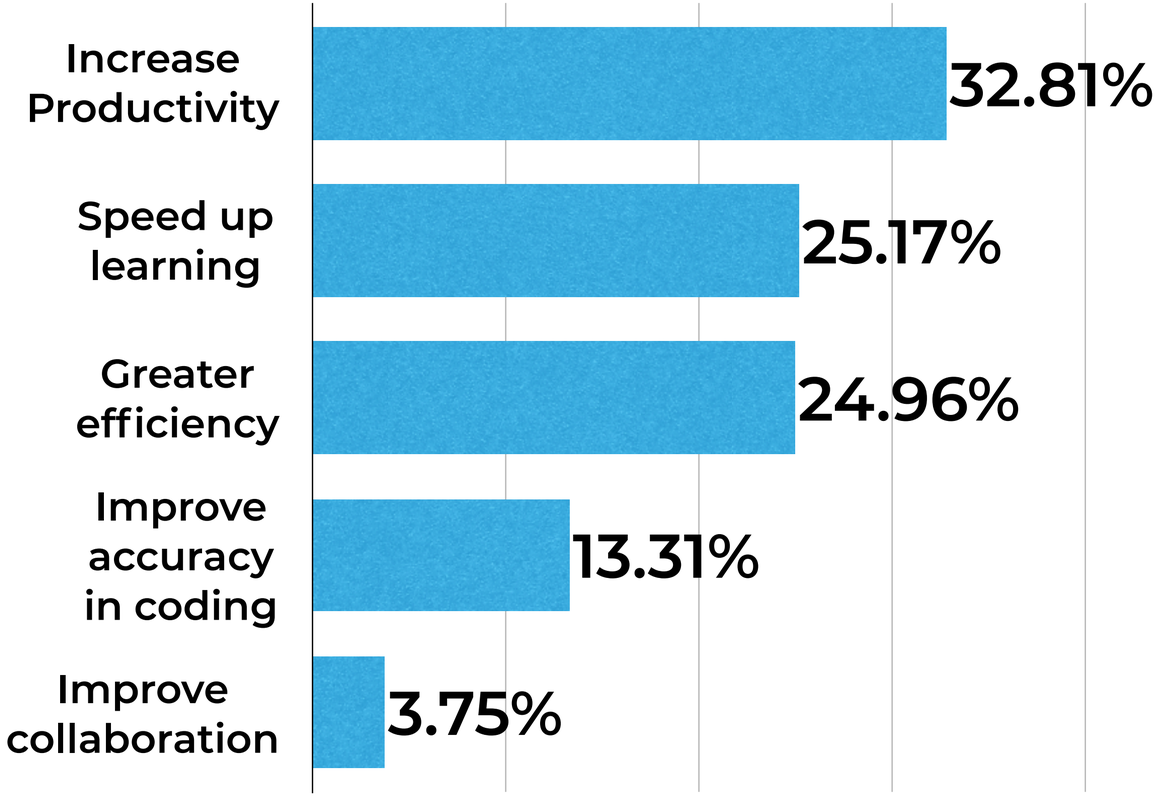
77% of all respondents are favorable or very favorable of AI tools for development. Professional developers are more likely to be indifferent than those learning to code (17% vs. 15%).

How favorable is your stance on using AI tools as part of your development workflow?



Increasing productivity is the biggest benefit that developers see from AI tools. **Speeding up learning and greater efficiency** are tied for secondary benefits.

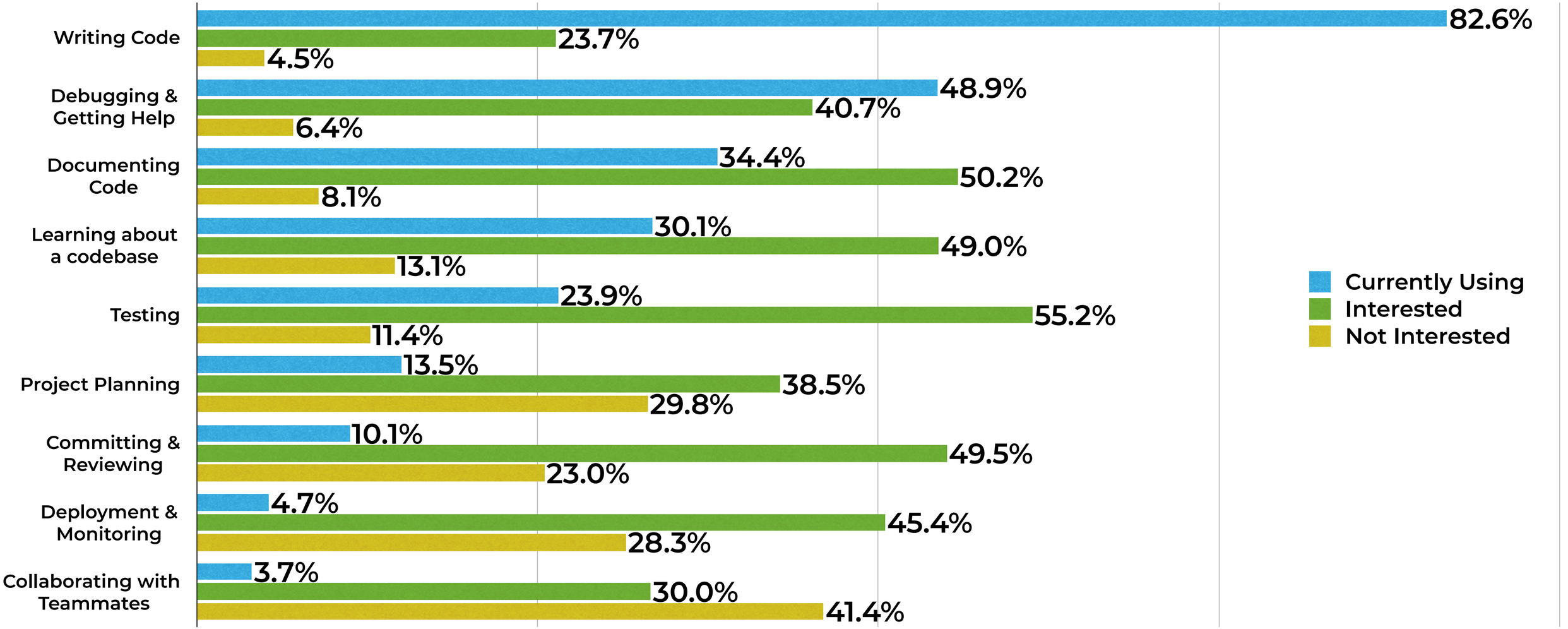
For the AI tools you use as part of your development workflow, what are the most important benefits you are hoping to achieve?



Stack Overflow Developer Survey 2023

Perceived Benefits of AI Tools

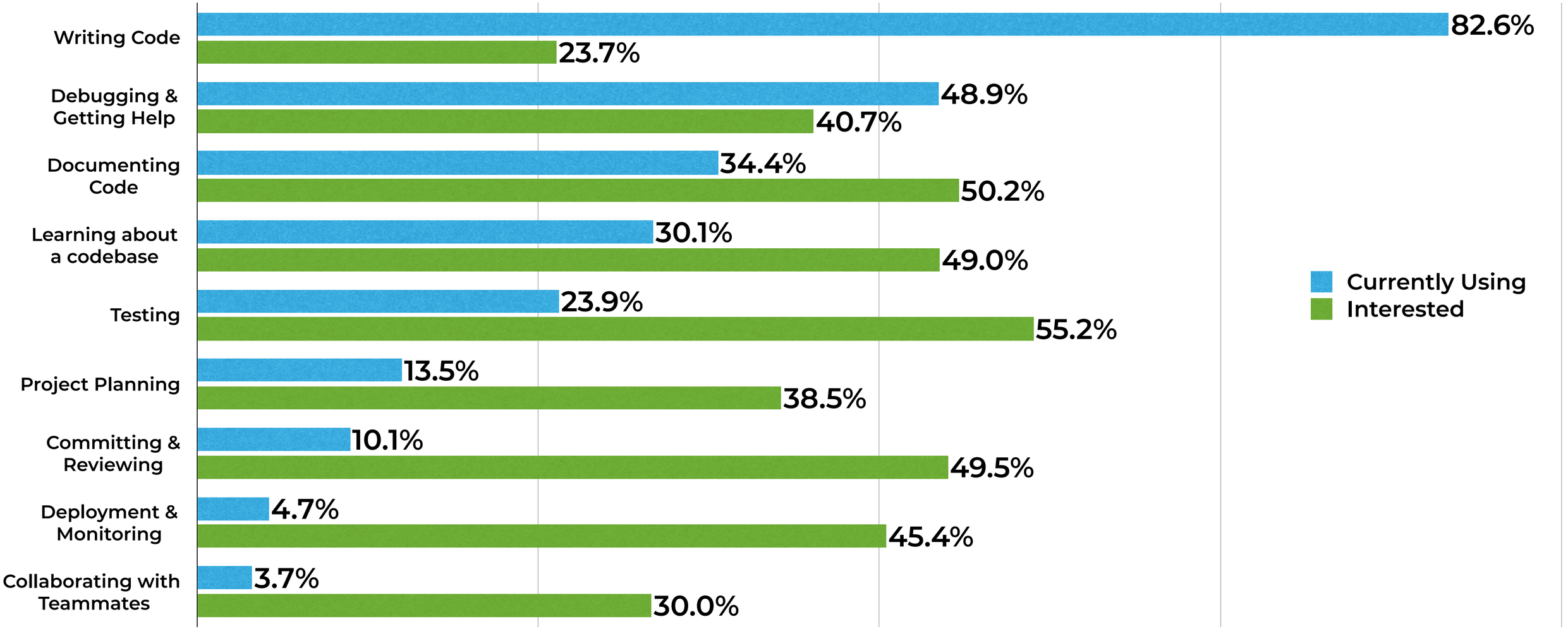
Which **parts of your development workflow** are you currently using AI tools for and **which are you interested in using AI tools** for over the next year? Please select all that apply.



Stack Overflow Developer Survey 2023

AI in the Development Workflow

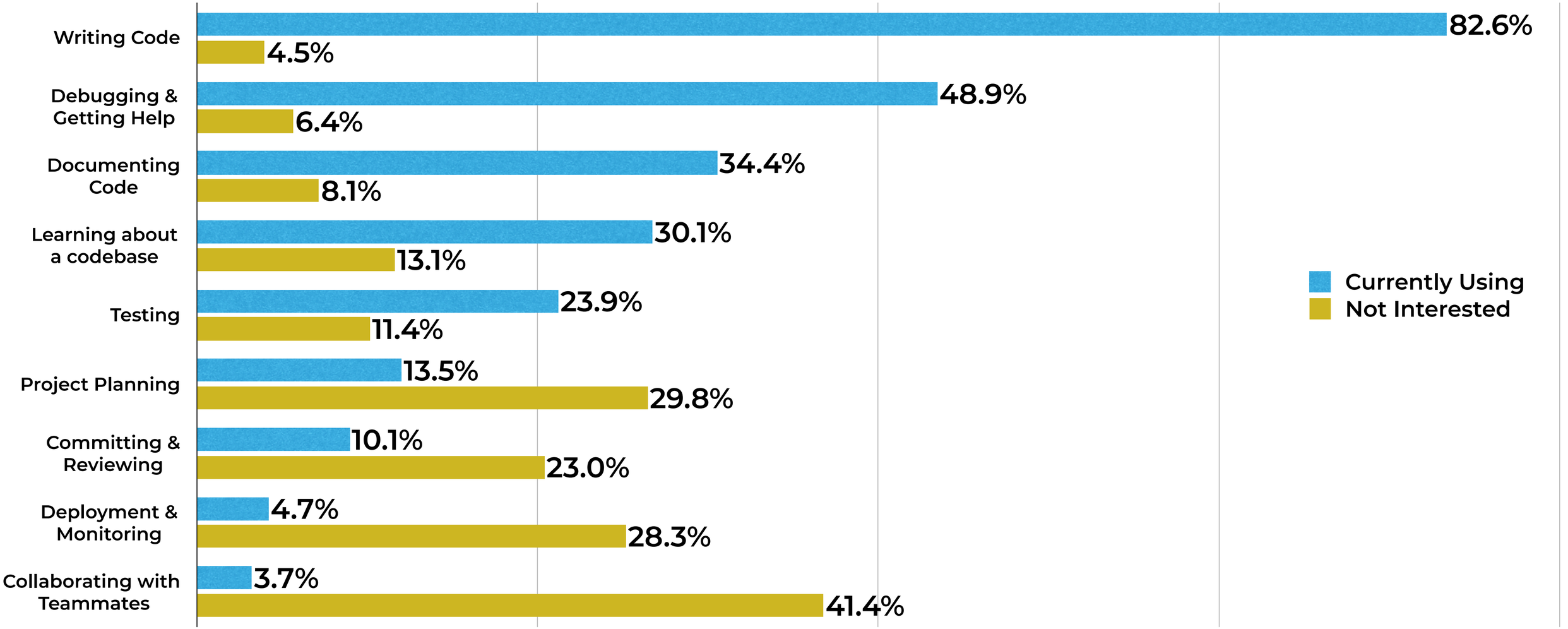
Which **parts of your development workflow** are you currently using AI tools for and **which are you interested in using AI tools** for over the next year? Please select all that apply.



Stack Overflow Developer Survey 2023

AI in the Development Workflow

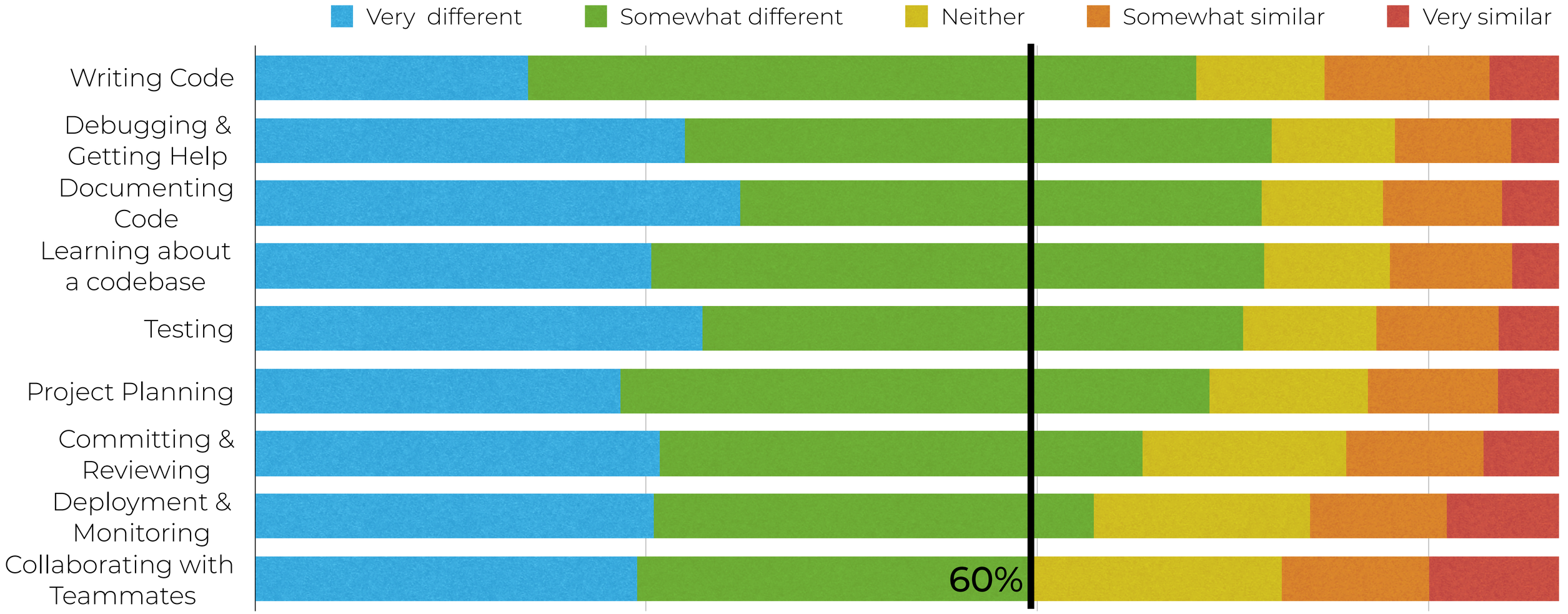
Which **parts of your development workflow** are you currently using AI tools for and **which are you interested in using AI tools** for over the next year? Please select all that apply.



Stack Overflow Developer Survey 2023

AI in the Development Workflow

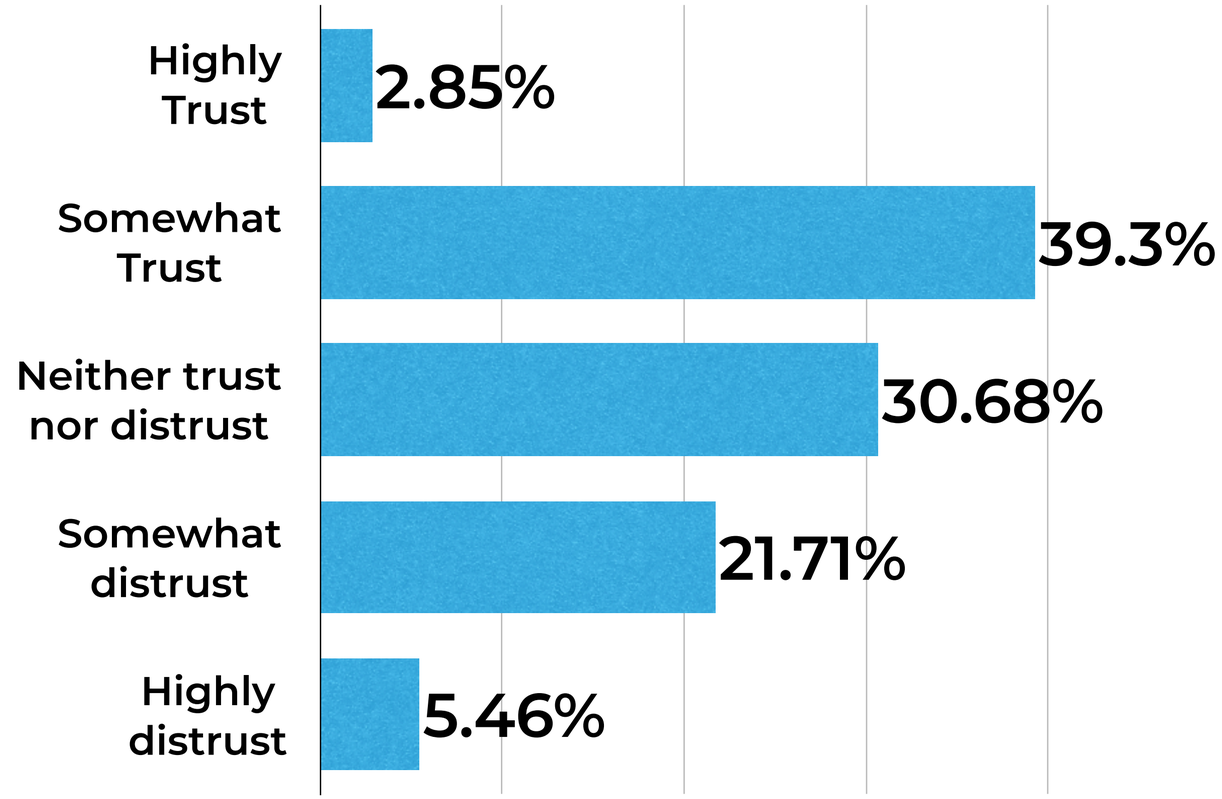
Thinking about how your **workflow** and process changes over time, **how similar or different do you anticipate your workflow to be 1 year from now as a result of AI tools you are currently using**?



Stack Overflow Developer Survey 2023
AI tools next year

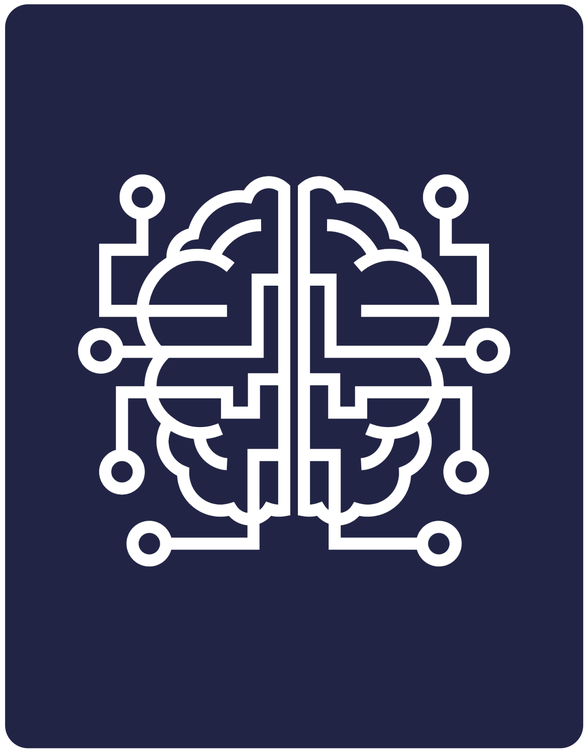
We see developers **split on their trust in the accuracy** of the AI output from tools. About 42% trust the accuracy of the output, while 31% are on the fence.

How much do you trust the accuracy of the output from AI tools as part of your development workflow?



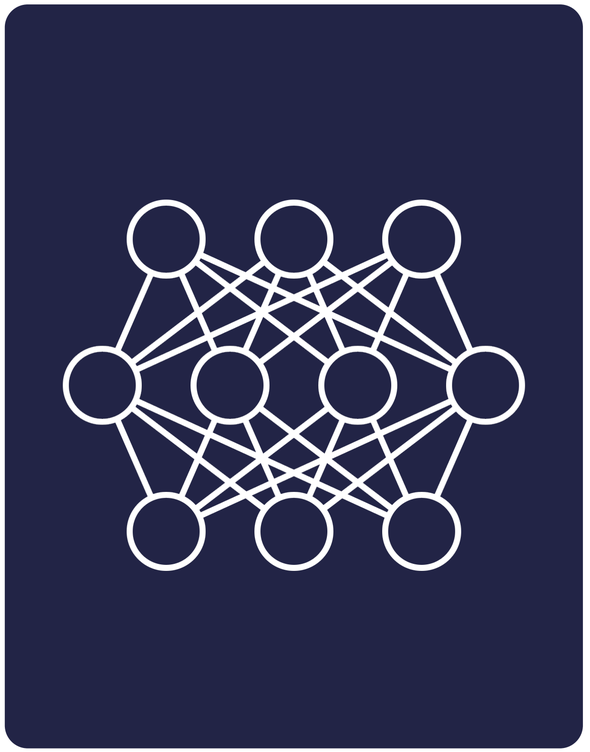
Stack Overflow Developer Survey 2023

Trust in Accuracy of AI Tools



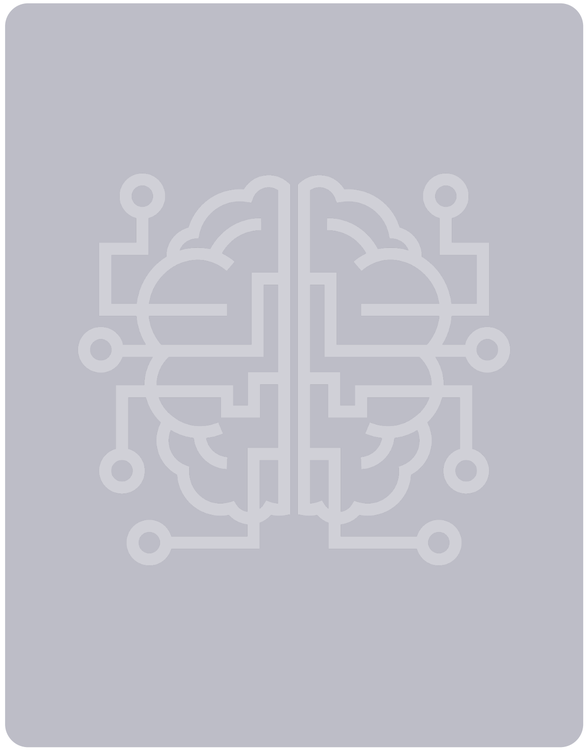
Artificial Intelligence

Machine Learning



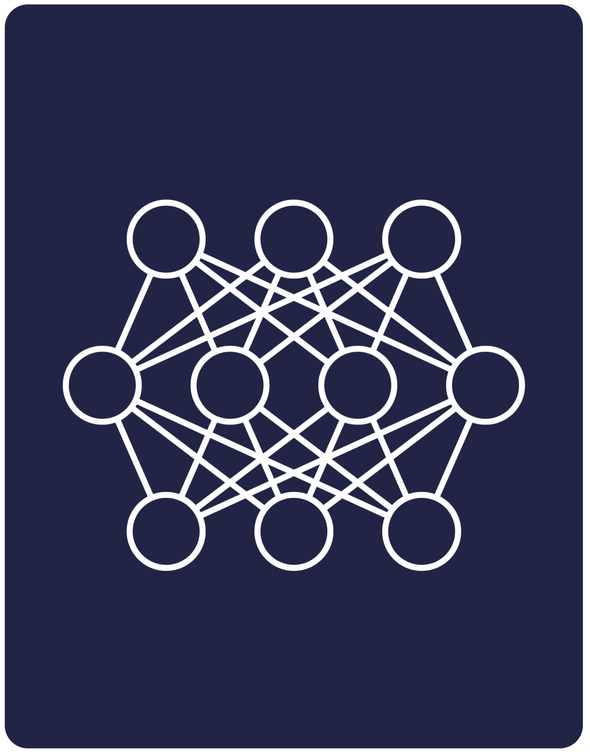
Large Language Models

What do we mean by **Artificial Intelligence?**



Artificial Intelligence

Machine Learning



Large Language Models

What do we mean by **Artificial Intelligence?**

On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜

Emily M. Bender*
ebender@uw.edu
University of Washington
Seattle, WA, USA

Timnit Gebru*
timnit@blackinai.org
Black in AI
Palo Alto, CA, USA

Angelina McMillan-Major
aymm@uw.edu
University of Washington
Seattle, WA, USA

Shmargaret Shmitchell
shmargaret.shmitchell@gmail.com
The Aether

ABSTRACT

The past 3 years of work in NLP have been characterized by the development and deployment of ever larger language models, especially for English. BERT, its variants, GPT-2/3, and others, most recently Switch-C, have pushed the boundaries of the possible both

alone, we have seen the emergence of BERT and its variants [39, 70, 74, 113, 146], GPT-2 [106], T-NLG [112], GPT-3 [25], and most recently Switch-C [43], with institutions seemingly competing to produce ever larger LMs. While investigating properties of LMs and how they change with size holds scientific interest, and large LMs

Large Language Models
'Intelligent' or just Stochastic Parrots?

*"Text generated by an LM is **not grounded in communicative intent**, **any model of the world**, or **any model of the reader's state of mind**. It can't have been, because the training data never included **sharing thoughts with a listener, nor does the machine have the ability to do that**. This can seem counter-intuitive given the **increasingly fluent qualities of automatically generated text**, but we have to account for the fact that our **perception of natural language** text, regardless of how it was generated, **is mediated by our own linguistic competence and our predisposition to interpret communicative acts as conveying coherent meaning and intent**, whether or not they do."*

"Contrary to how it may seem when we observe its output, an ***LM is a system for haphazardly stitching together sequences of linguistic forms it has observed in its vast training data****, according to* ***probabilistic information*** *about how they combine, but without any* ***reference to meaning: a stochastic parrot****."*

"It is well established by now that(L)LMs exhibit ***various kinds of bias****, like* ***stereotypical associations*** *or* ***negative sentiments*** *towards specific groups"*

Christine Basta, Marta R Costa-jussà, and Noe Casas. 2019. ***Evaluating the Underlying Gender Bias in Contextualized Word Embeddings****.*
In Proceedings of the 1st Workshop on Gender Bias in NLP. 33–39.

Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. 2019. ***Measuring Bias in Contextualized Word Representations****.*
In Proceedings of the 1st Workshop on Gender Bias in NLP. 166–172.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. ***The Woman Worked as a Babysitter: On Biases in Language Generation.***
In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 3407–3412.

Haoran Zhang, Amy X Lu, Mohamed Abdalla, Matthew McDermott, and Marzyeh Ghassemi. 2020. ***Hurtful words: quantifying biases in clinical contextual word embeddings****.*
In Proceedings of the ACM Conference on Health, Inference, and Learning. 110–120.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. ***Gender Bias in Contextualized Word Embeddings****. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 629–634.*

In Need of 'Pair' Review: Vulnerable Code Contributions by GitHub Copilot

Hammond Pearce | Research Assistant Professor, New York University
Benjamin Tan | Assistant Professor, University of Calgary
Brendan Dolan-Gavitt | Assistant Professor, New York University
Baleegh Ahmad | PhD Candidate, New York University

Date: Wednesday, August 10 | 1:30pm–2:10pm (South Seas CD (Level 3))
Format: 40-Minute Briefings
Track: AI, ML, & Data Science

On June 29 in 2021 GitHub announced and released their newest tool, 'Copilot' – an 'AI-based Pair Programmer', a deep learning model trained over vast quantities of open-source GitHub code. However, we humans wrote most of that code. And much of it isn't great. It has bugs, it contains dated coding practices, and many repositories even contain dangerously insecure code. Given the vast quantity of garbage code that Copilot has learned from, is it reasonable to trust the code suggestions that it generates?

In this talk, we demonstrate that GitHub Copilot is susceptible to writing vulnerabilities in multiple axis, from SQL injections to buffer overflows, use-after-free to cryptographic issues. We try different languages – C, Python, and even Verilog, where we show it also generates hardware bugs (when it can generate hardware at all).

Overall, we tried 89 different scenarios for Copilot, generating 1,689 suggestions, and found approximately 40% to be vulnerable.

Large Language Models Issues
Vulnerable Code

In Need of 'Pair' Review: Vulnerable Code Contributions by GitHub Copilot

Hammond Pearce | Research Assistant Professor, New York University
Benjamin Tan | Assistant Professor, University of Calgary
Brendan Dolan-Gavitt | Assistant Professor, New York University
Baleegh Ahmad | PhD Candidate, New York University
Date: Wednesday, August 10 | 1:30pm-2:10pm (South Seas CD (Level 3))
Format: 40-Minute Briefings
Track: AI, ML, & Data Science

On June 29 in 2021 GitHub announced and released their newest tool, 'Copilot' - an 'AI-based Pair Programmer', a deep learning model
rote most of that code. And much of it isn't great. It has
gerously insecure code. Given the vast quantity of garbage
that it generates?

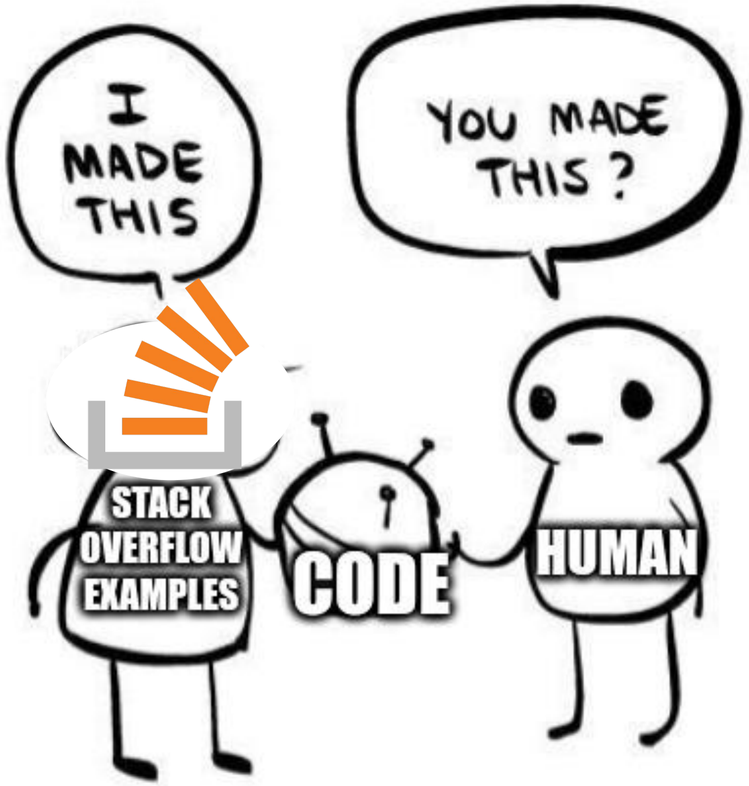
In this talk, we demonstrat ulnerabilities in multiple axis, from SQL injections to buffer
overflows, use-after-free to cryptographic i w it also generates
hardware bugs (when it can generate hardw

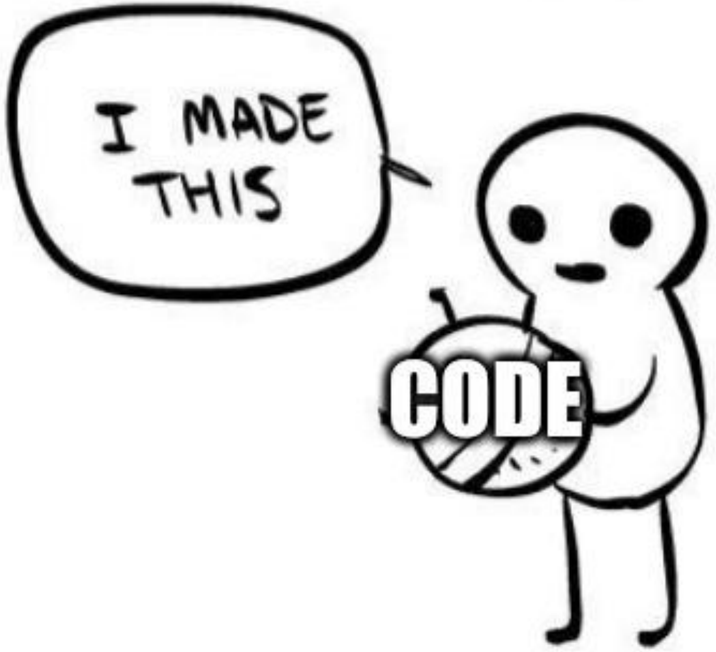
Overall, we tried 89 different scenarios for Copilot,
generating 1,689 suggestions,
and found approximately 40% to be vulnerable.

Overall, we tried 89 different scenarios for Copilot, generating 1,689 suggestions, and found approximately 40% to be vulnerable.

Large Language Models Issues
Vulnerable Code

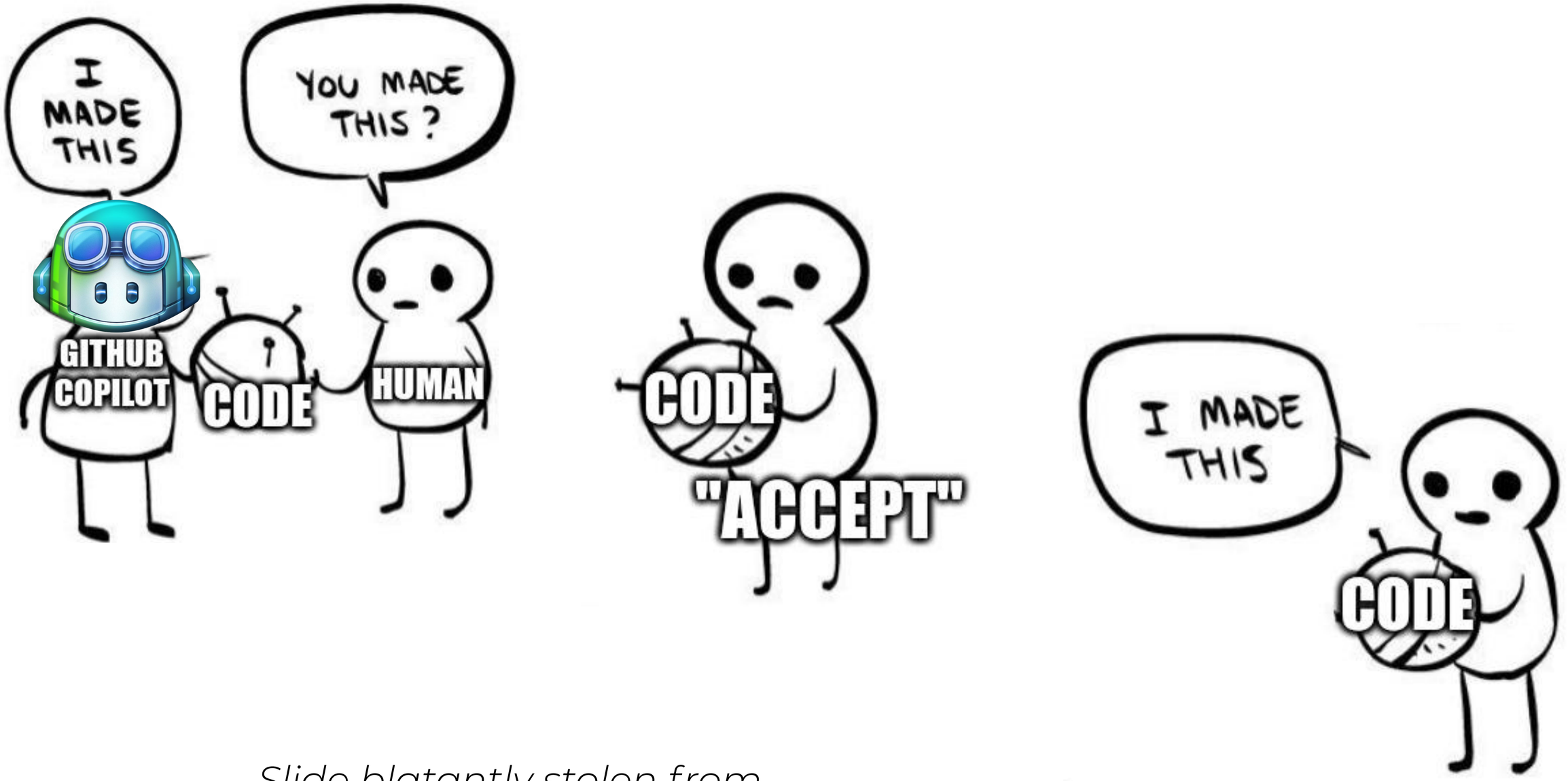
Naïve software development





Slide blatantly stolen from

A brave new world?



Slide blatantly stolen from

Copilot should remain a *Co*-pilot

Slide blatantly stolen from

Copilot should remain a *Co*-pilot

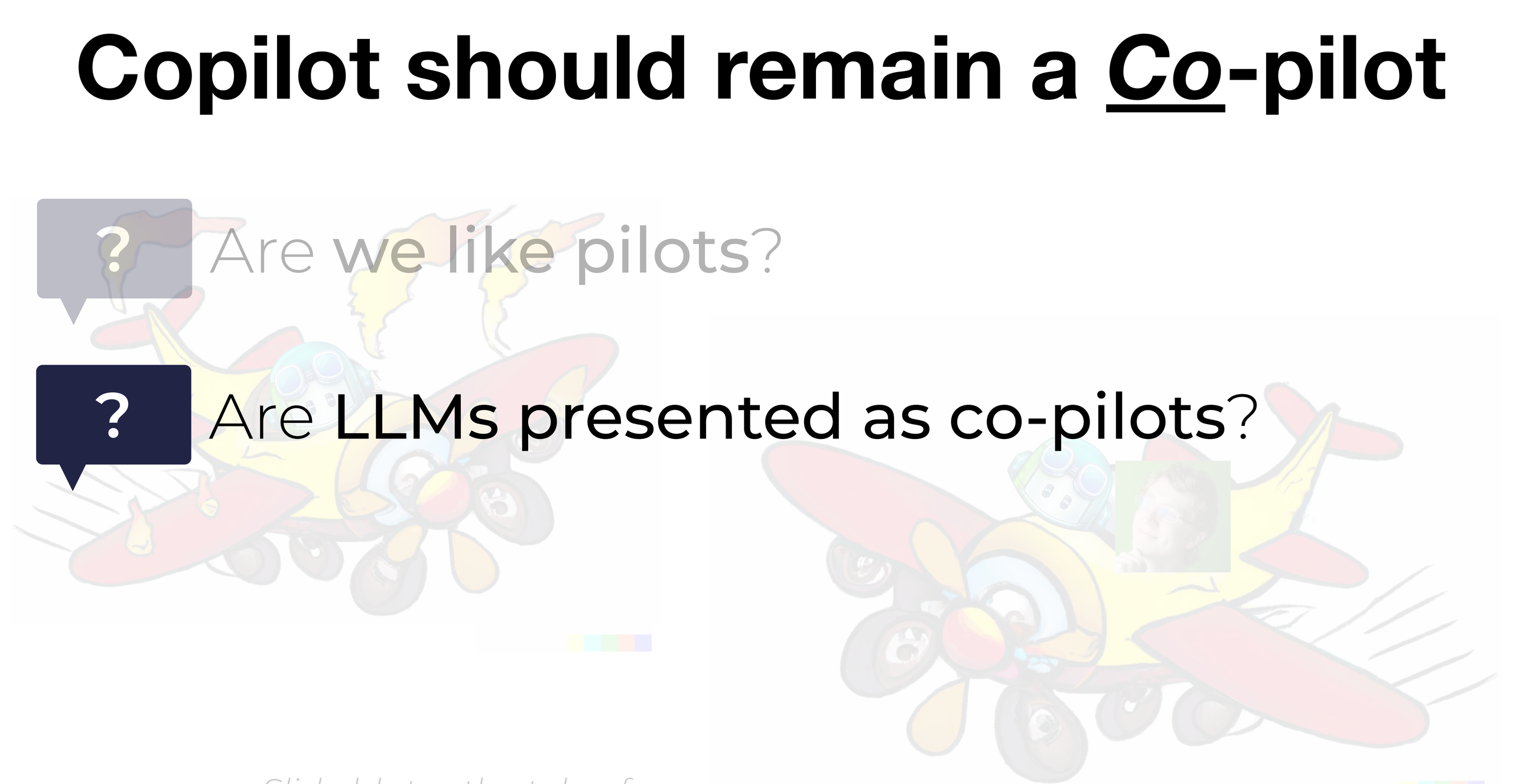
? Are we like pilots?

Slide blatantly stolen from

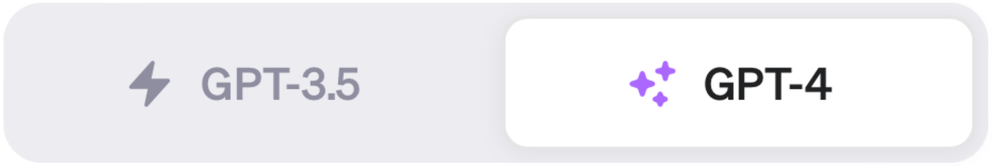
Copilot should remain a *Co*-pilot

? Are we like pilots?

? Are LLMs presented as co-pilots?



Slide blatantly stolen from



ChatGPT PLUS

Brainstorm incentives
for a customer loyalty program in a small book...

Help me pick
an outfit that will look good on camera

Compare storytelling techniques
in novels and in films

Recommend a dish
to impress a date who's a picky eater

Send a message

ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT August 3 Version

GPT-3.5 GPT-4

ChatGPT PLUS

Brainstorm incentives
for a customer loyalty program in a small book...

Help me pick
an outfit that will look good on camera

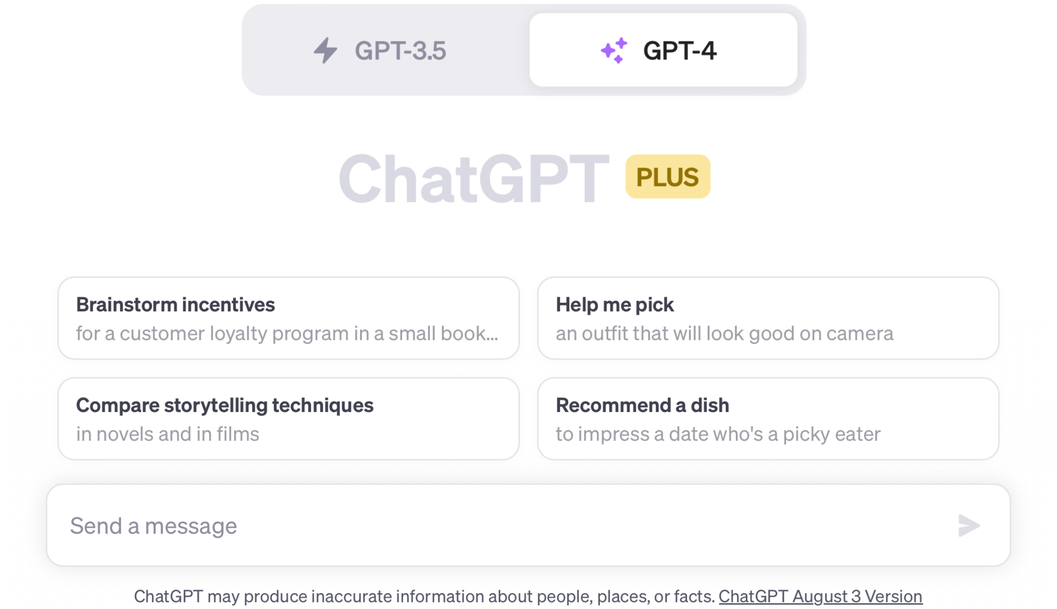
Compare storytelling techniques
in novels and in films

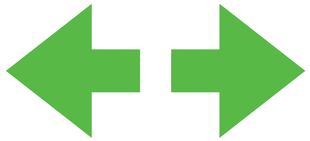
Recommend a dish
to impress a date who's a picky eater

Send a message

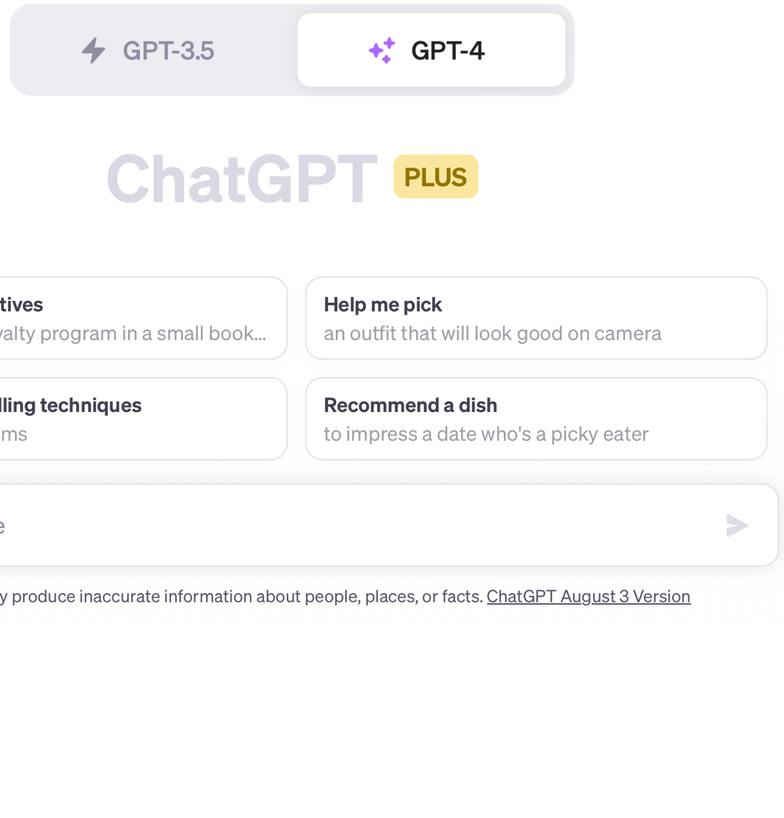
ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT August 3 Version

Large Language Models Issues
Presentation

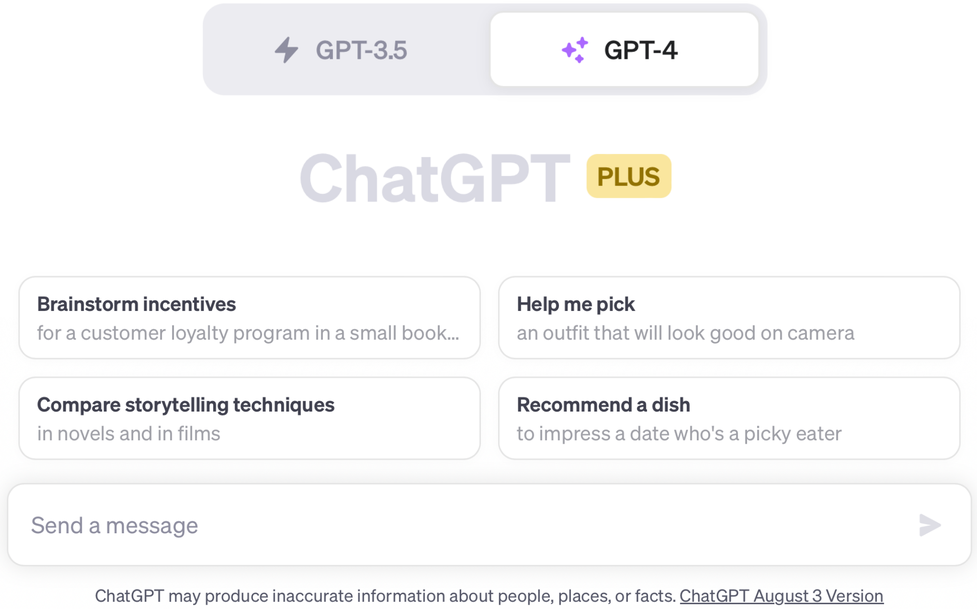




Large Language Models Issues
Presentation

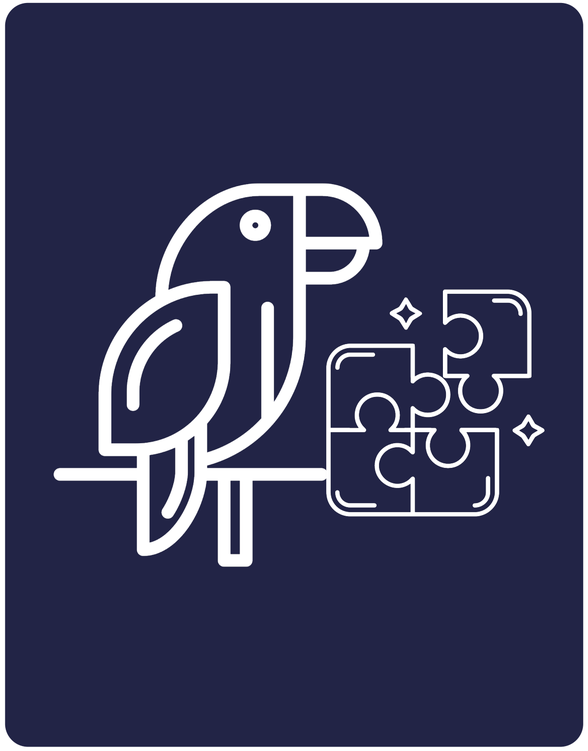


Large Language Models Issues
Presentation



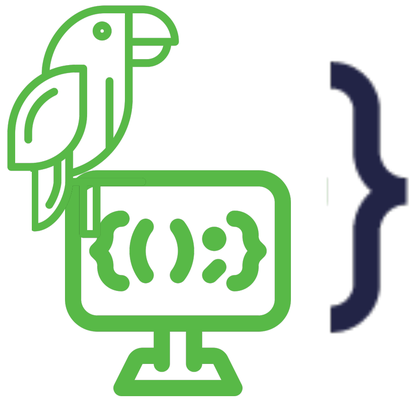
We have **no clear answer** to the issue of "how good" these LLMs are, but we want to highlight, with some examples, **when they work well and when they do not**, giving you interesting **research ideas**

as Companions



as Task Solvers

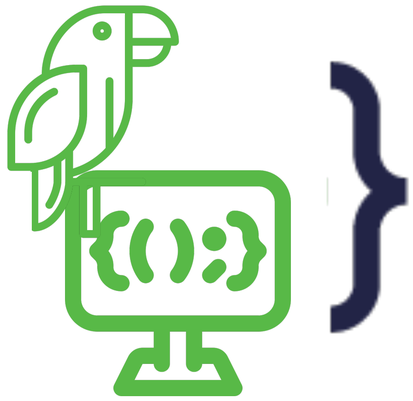
Supporting
Developers



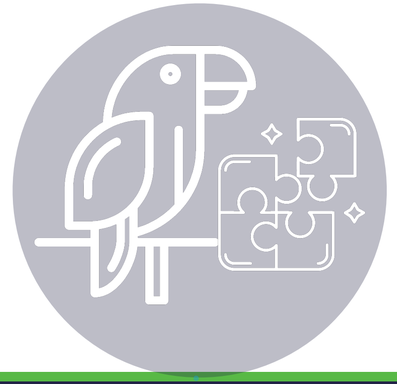
Parrots as
Companions

Parrots as
Task Solvers

Supporting
Developers



Parrots as
Companions



Parrots as
Task Solvers

Large Language Models
Bias Live Demo

marketplace.visualstudio.com

Visual Studio | Marketplace

Sign in

Visual Studio Code > Other > Tako

New to Visual Studio Code? Get it now.

Tako

codelounge | 99 installs | ★★★★★ (2) | Free

Tako (タコ) is an IDE interaction monitor for Visual Studio Code

Install

Trouble Installing?

Overview | Version History | Q & A | Rating & Review

Tako for Visual Studio Code

Tako is an extension for Visual Studio Code (VS Code) that records many aspects of your programming activity and provides a digest to allow some self-reflection on what you have done, how you spent your time, and which entities you worked on.

Understanding Copilot Usage
Tako

Interaction-Aware Development Environments

Recording, Mining, and Leveraging IDE Interactions to Analyze and Support the Development Flow

Roberto Minelli

Research Advisor

Prof. Michele Lanza

Research Co-Advisor

Dr. Andrea Mocci

Understanding Copilot Usage

Tako

Interaction-Aware Development Environments

Recording, Mining, and Leveraging IDE Interactions to Analyze and Support the Development Flow

Roberto Minelli

codelounge

HASLERSTIFTUNG

marketplace.visualstudio.com

Visual Studio | Marketplace

Sign in

Visual Studio Code > Other > Tako

New to Visual Studio Code? Get it now.

Tako

codelounge | 99 installs | ★★★★★ (2) | Free

Tako (タコ) is an IDE interaction monitor for Visual Studio Code

Install

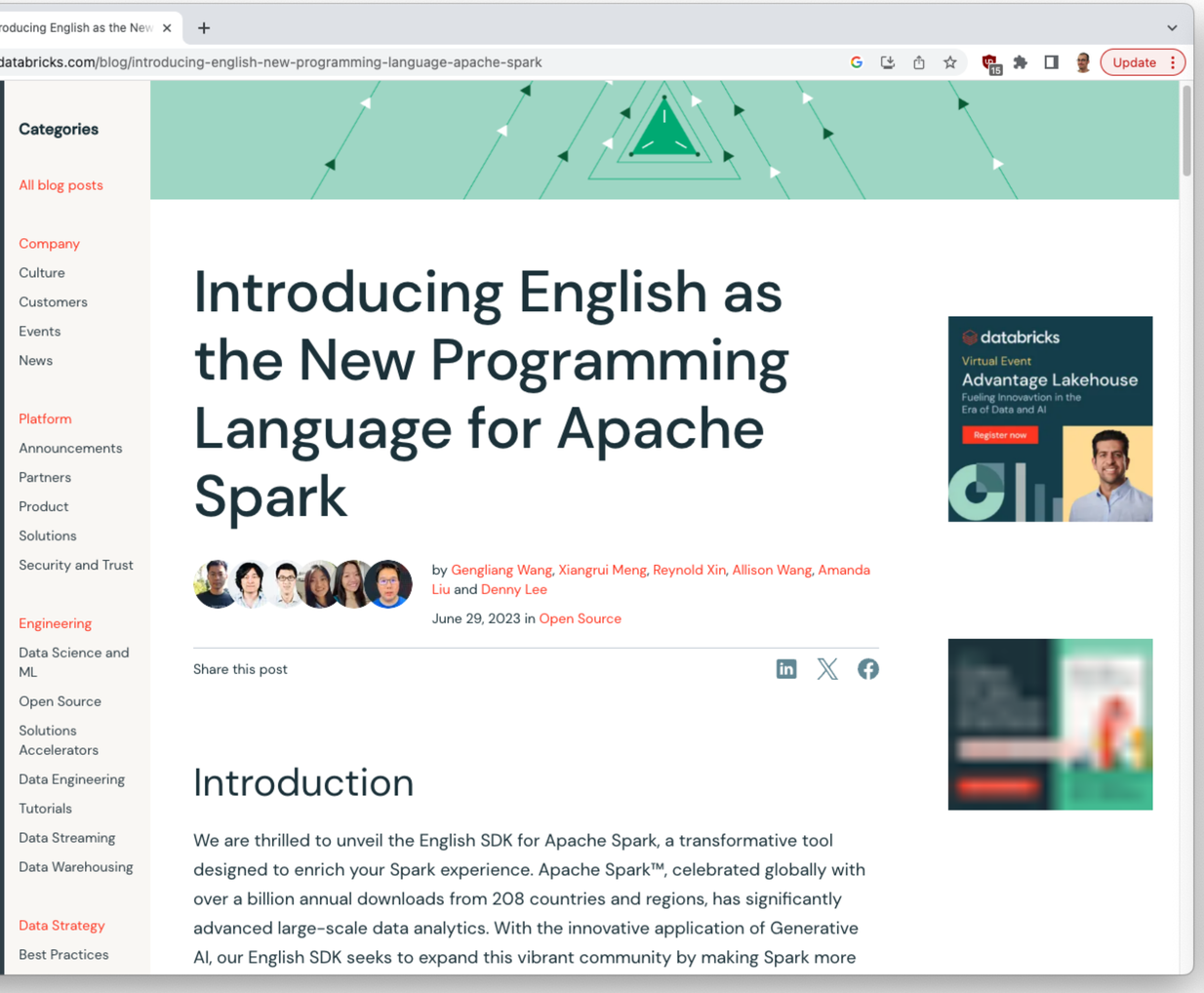
Trouble Installing?

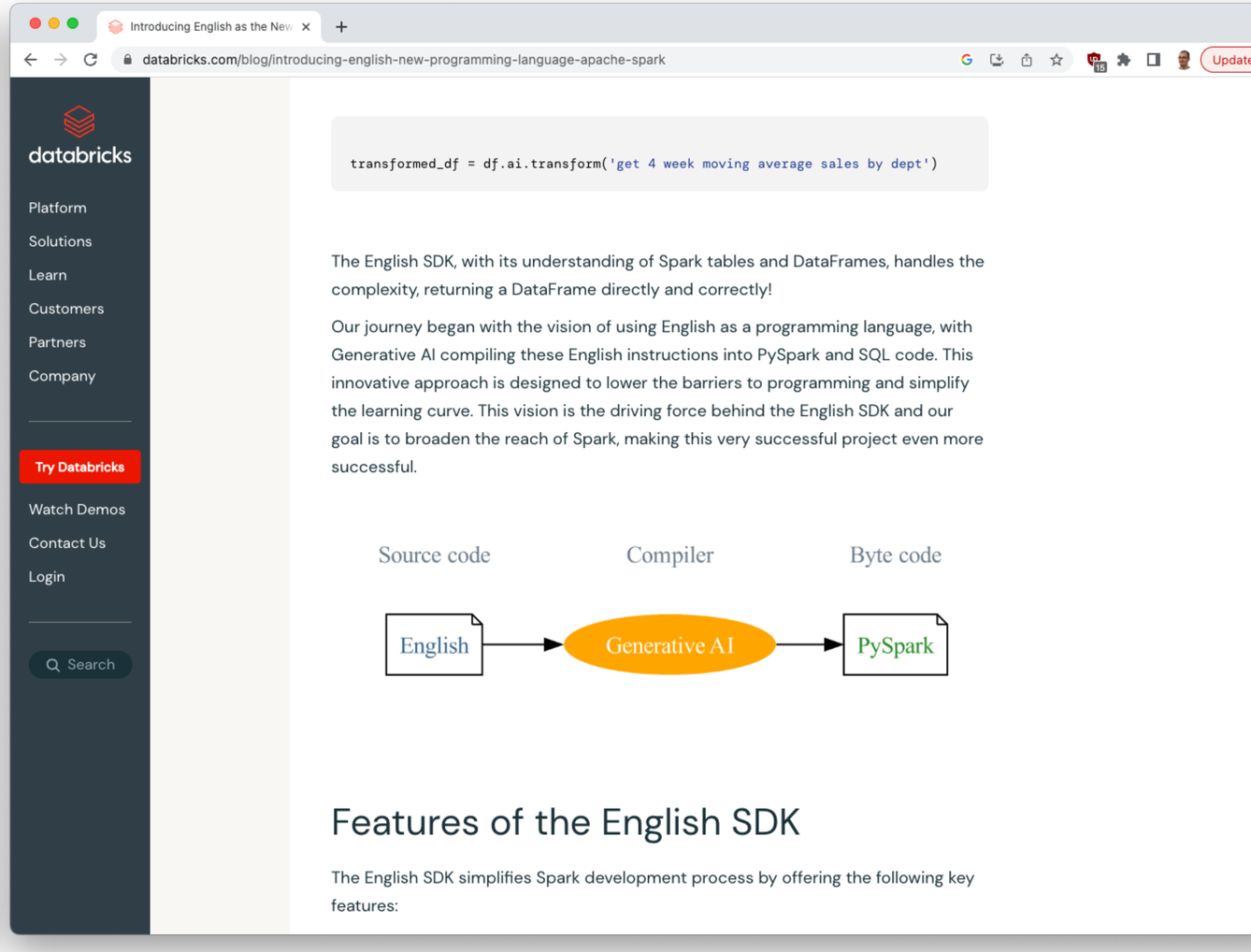
Overview | Version History | Q & A | Rating & Review

Tako for Visual Studio Code

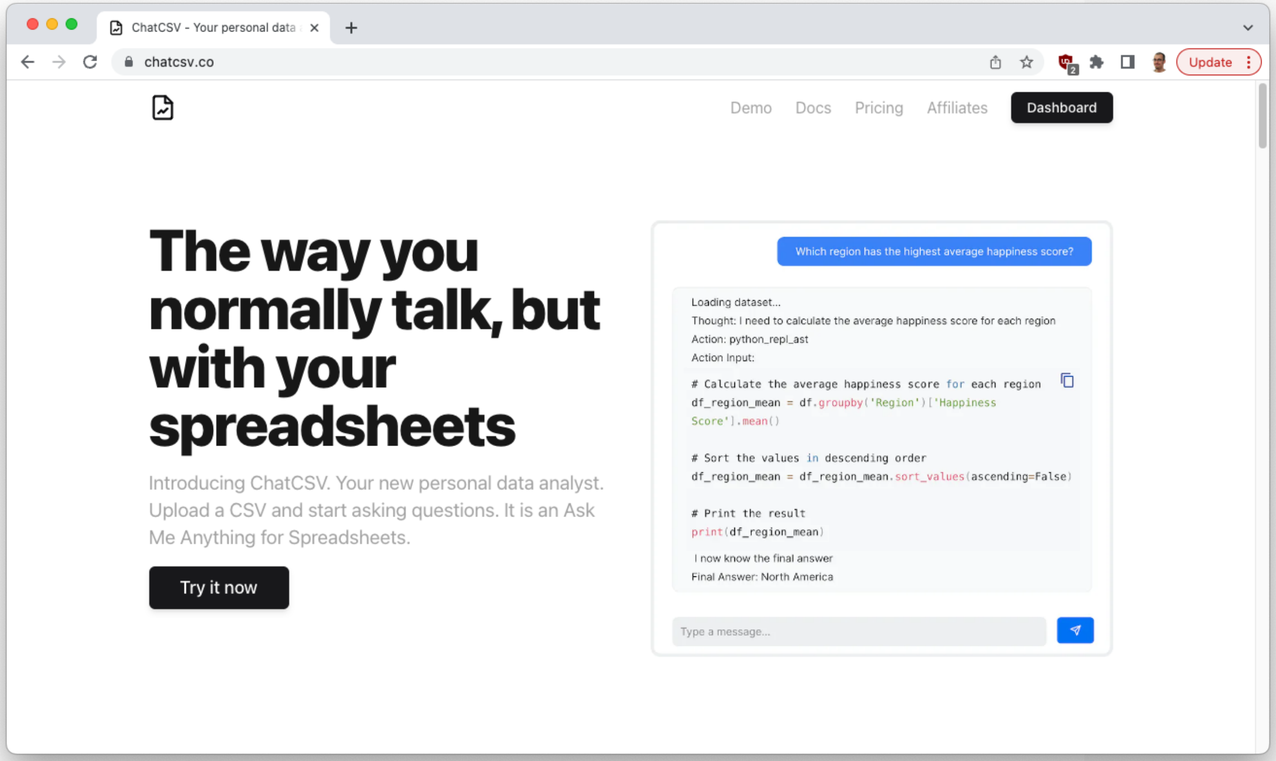
Tako is an extension for Visual Studio Code (VS Code) that records many aspects of your programming activity and provides a digest to allow some self-reflection on what you have done, how you spent your time, and which entities you worked on.

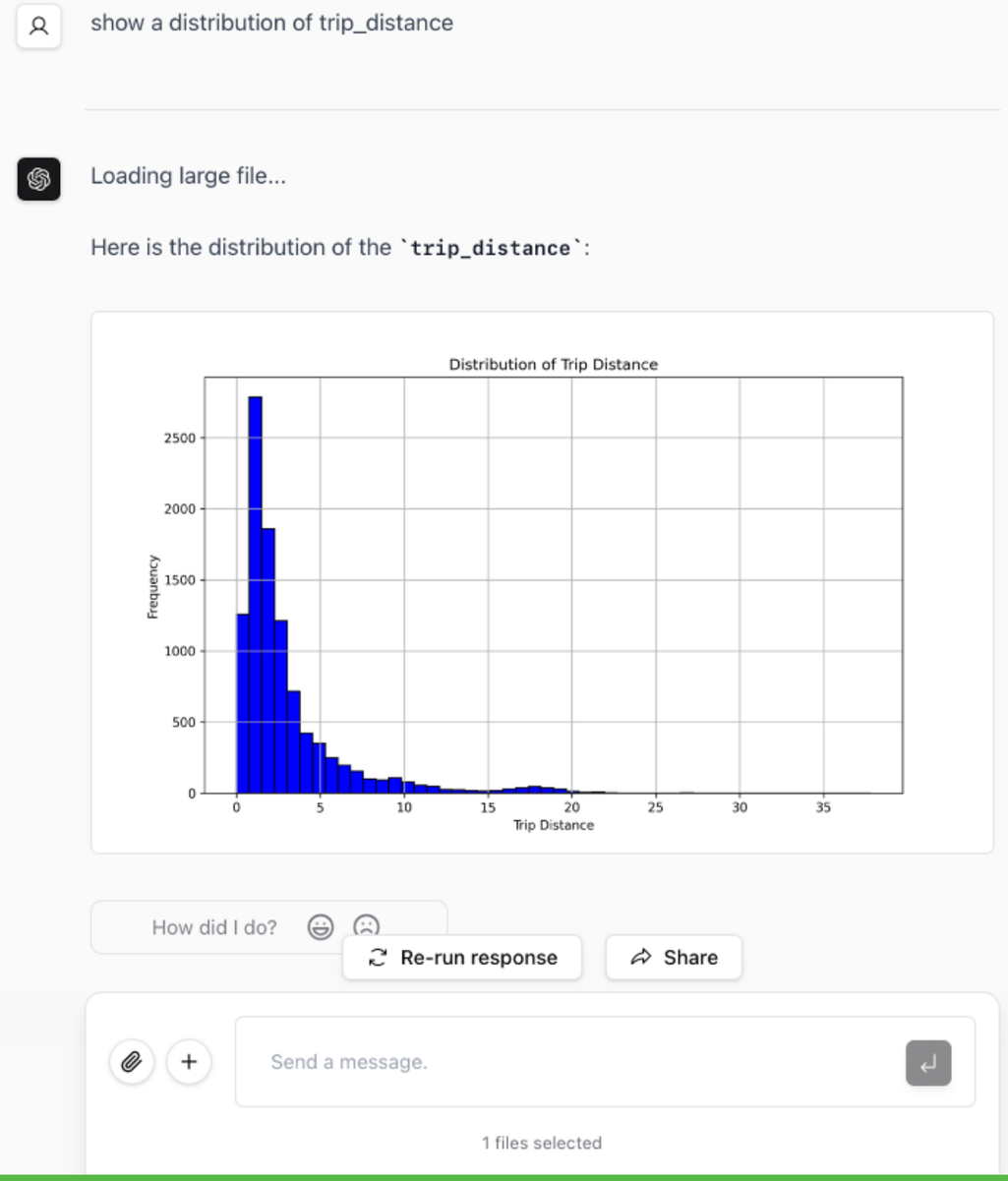
Understanding Copilot Usage
Copilot Usage in Tako (Demo)





The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis





The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis

Give me a quick rundown of this data

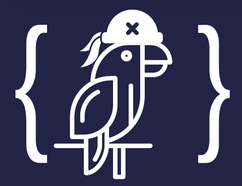
Loading large file...

The dataframe `df1` has 10,000 entries and 18 columns. There are no missing values in any of the columns. Here is a brief description of each column:

1. `index`: An integer index for the data.
2. `medallion`: An object type column, possibly a unique identifier for the taxi.
3. `hack_license`: An object type column, possibly a unique identifier for the taxi driver.
4. `pickup_datetime`: An object type column representing the date and time of the taxi pickup.
5. `dropoff_datetime`: An object type column representing the date and time of the taxi dropoff.
6. `passenger_count`: An integer column representing the number of passengers in the taxi.
7. `trip_time_in_secs`: An integer column representing the duration of the trip in seconds.
8. `trip_distance`: A float column representing the distance of the trip.
9. `pickup_longitude`: A float column representing the longitude of the pickup location.
10. `pickup_latitude`: A float column representing the latitude of the pickup location.
11. `dropoff_longitude`: A float column representing the longitude of the dropoff location.
12. `dropoff_latitude`: A float column representing the latitude of the dropoff location.
13. `fare_amount`: A float column representing the fare amount for the trip.
14. `surcharge`: A float column representing any additional surcharge for the trip.
15. `mta_tax`: A float column representing the MTA tax for the trip.
16. `tip_amount`: A float column representing the tip amount for the trip.
17. `tolls_amount` Re-run response Share mount for the trip.

how is the trip duration distributed?

1 files selected



The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis

13. `fare_amount`: A float column representing the fare amount for the trip.
14. `surcharge`: A float column representing any additional surcharge for the trip.
15. `mta_tax`: A float column representing the MTA tax for the trip.
16. `tip_amount`: A float column representing the tip amount for the trip.
17. `tolls_amount`: A float column representing the tolls amount for the trip.
18. `total_amount`: A float column representing the total amount for the trip.

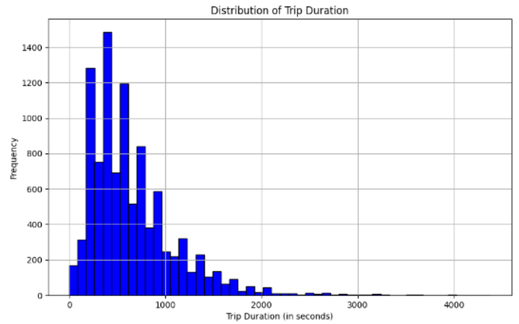
The dataframe is using approximately 1.4+ MB of memory.

How did I do?

how is the trip duration distributed?

Loading large file...

Here is the distribution of trip duration:



How did I do?

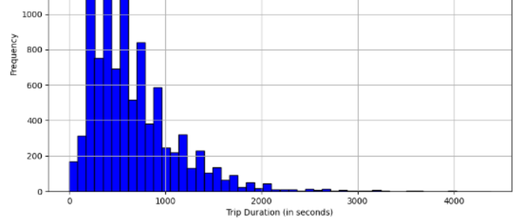
Re-run response

Share

Send a message.

1 files selected

The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis

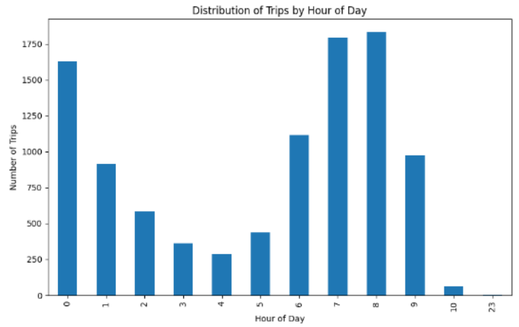


How did I do?

show the distribution of trips by hour of day

Loading large file...

Here is the distribution of trips by hour of day:



How did I do?

Re-run response

Share

Send a message.

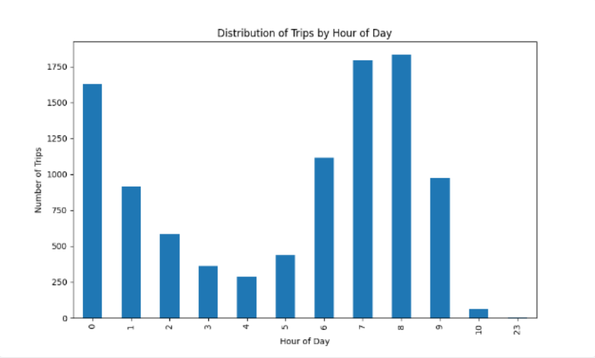
1 files selected

The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis

show the distribution of trips by hour of day

Loading large file...

Here is the distribution of trips by hour of day:



How did I do?

how many trips happened after 10:00?

Loading large file...

Apologies for the error. It seems I forgot to import the pandas library. Let me correct that and try again.There were 6 trips that happened after 10:00.

How did I do?

Re-run response Share

Send a message.

1 files selected

The rise of Stochastic Parrots for Developers
Parrots as Companions for Data Analysis

Can we use copilot to generate code for systems in production?

- Yes, Copilot can be used to generate code for systems in production
- Copilot is designed to assist developers in writing code efficiently and accurately.
- It supports a wide range of programming languages and frameworks commonly used in production environments.
- However, it's crucial to review and test the generated code thoroughly before deploying it in a production system.
- Copilot can significantly accelerate development processes while maintaining code quality and reliability.

MA

make a concise slide that answer the question "can copilot be used to generate code for systems in production?"

Slide Title: "Utilizing Copilot in Production Systems"

Slide Content:

- Bullet 1: "Yes, Copilot can be used to generate code for systems in production."
- Bullet 2: "Copilot is designed to assist developers in writing code efficiently and accurately."
- Bullet 3: "It supports a wide range of programming languages and frameworks commonly used in production environments."
- Bullet 4: "However, it's crucial to review and test the generated code thoroughly before deploying it in a production system."
- Bullet 5: "Copilot can significantly accelerate development processes while maintaining code quality and reliability."

Knowing that every delivery note has a reference to n boxes and every box has a weight, take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid). For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.

We have 3 tables: delivery_note, box, and delivery_notes_to_boxes.
Knowing that every delivery note has a reference to n boxes and every box has a weight, take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid). For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.
Note that the relationship between delivery notes and boxes is many to many, stored in the table delivery_notes_to_boxes. In this table the boxes are identified by the column box_uid and the delivery notes by delivery_note_uid. The joins should be inner joins.

We have 3 tables: delivery_note, box, and delivery_notes_to_boxes.
Knowing that every delivery note has a reference to n boxes and every box has a weight, **take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid).** For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.
Note that the relationship between delivery notes and boxes is many to many, stored in the table delivery_notes_to_boxes. In this table the boxes are identified by the column box_uid and the delivery notes by delivery_note_uid. The joins should be inner joins.

```
SELECT
    dn.finalization_date AS date,
    dn.source_location_uid AS location,
    COUNT(*) AS number_of_pickups,
    ARRAY_AGG(dn.delivery_note_uid) AS
delivery_notes_list,
    SUM(b.weight) AS total_weight,
    COUNT(DISTINCT db.box_uid) AS
total_number_of_boxes
FROM
    delivery_note dn
JOIN
    delivery_notes_to_boxes db ON
dn.delivery_note_uid = db.delivery_note_uid
JOIN
    box b ON db.box_uid = b.box_uid
WHERE
    dn.status = 'CLOSED'
GROUP BY
    dn.finalization_date, dn.source_location_uid;
```

We have 3 tables: delivery_note, box, and delivery_notes_to_boxes.
Knowing that every delivery note has a reference to n boxes and every box has a weight, take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid). **For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.**
Note that the relationship between delivery notes and boxes is many to many, stored in the table delivery_notes_to_boxes. In this table the boxes are identified by the column box_uid and the delivery notes by delivery_note_uid. The joins should be inner joins.

```
SELECT
    dn.finalization_date AS date,
    dn.source_location_uid AS location,
    COUNT(*) AS number_of_pickups,
    ARRAY_AGG(dn.delivery_note_uid) AS
delivery_notes_list,
    SUM(b.weight) AS total_weight,
    COUNT(DISTINCT db.box_uid) AS
total_number_of_boxes
FROM
    delivery_note dn
JOIN
    delivery_notes_to_boxes db ON
dn.delivery_note_uid = db.delivery_note_uid
JOIN
    box b ON db.box_uid = b.box_uid
WHERE
    dn.status = 'CLOSED'
GROUP BY
    dn.finalization_date, dn.source_location_uid;
```

We have 3 tables: delivery_note, box, and delivery_notes_to_boxes.
Knowing that every delivery note has a reference to n boxes and every box has a weight, take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid). For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.
Note that **the relationship between delivery notes and boxes is many to many, stored in the table delivery_notes_to_boxes**. In this table the boxes are identified by the column box_uid and the delivery notes by delivery_note_uid. The joins should be inner joins.

```
SELECT
    dn.finalization_date AS date,
    dn.source_location_uid AS location,
    COUNT(*) AS number_of_pickups,
    ARRAY_AGG(dn.delivery_note_uid) AS
delivery_notes_list,
    SUM(b.weight) AS total_weight,
    COUNT(DISTINCT db.box_uid) AS
total_number_of_boxes
FROM
    delivery_note dn
JOIN
    delivery_notes_to_boxes db ON
dn.delivery_note_uid = db.delivery_note_uid
JOIN
    box b ON db.box_uid = b.box_uid
WHERE
    dn.status = 'CLOSED'
GROUP BY
    dn.finalization_date, dn.source_location_uid;
```

We have 3 tables: delivery_note, box, and delivery_notes_to_boxes.
Knowing that every delivery note has a reference to n boxes and every box has a weight, take the delivery notes in status CLOSED, aggregate them by date (column finalization_date) and location of origin (column source_location_uid). For each date and location, show 1 as the number of pickups, the list of corresponding delivery notes in array form, the total weight of the corresponding boxes, and their total number.
Note that the relationship between delivery notes and boxes is many to many, stored in the table delivery_notes_to_boxes. In this table the boxes are identified by the column box_uid and the delivery notes by delivery_note_uid. The joins should be inner joins.

```
SELECT   DATE(dn.finalization_date)::TIMESTAMP
    dn.finalization_date AS date,
    dn.source_location_uid AS location,
  1 COUNT(*) AS number_of_pickups,
    ARRAY_AGG(dn.delivery_note_uid) AS
delivery_notes_list,
    SUM(b.weight) AS total_weight,
    COUNT(DISTINCT db.box_uid) AS
total_number_of_boxes
FROM
    delivery_note dn
JOIN
    delivery_notes_to_boxes db ON
dn.delivery_note_uid = db.delivery_note_uid
JOIN
    box b ON db.box_uid = b.box_uid
WHERE
    dn.status = 'CLOSED'
GROUP BY
    dn.finalization_date, dn.source_location_uid;
    DATE(dn.finalization_date)
```

So, we could use the parrot to generate the production SQL

Did we actually do it?

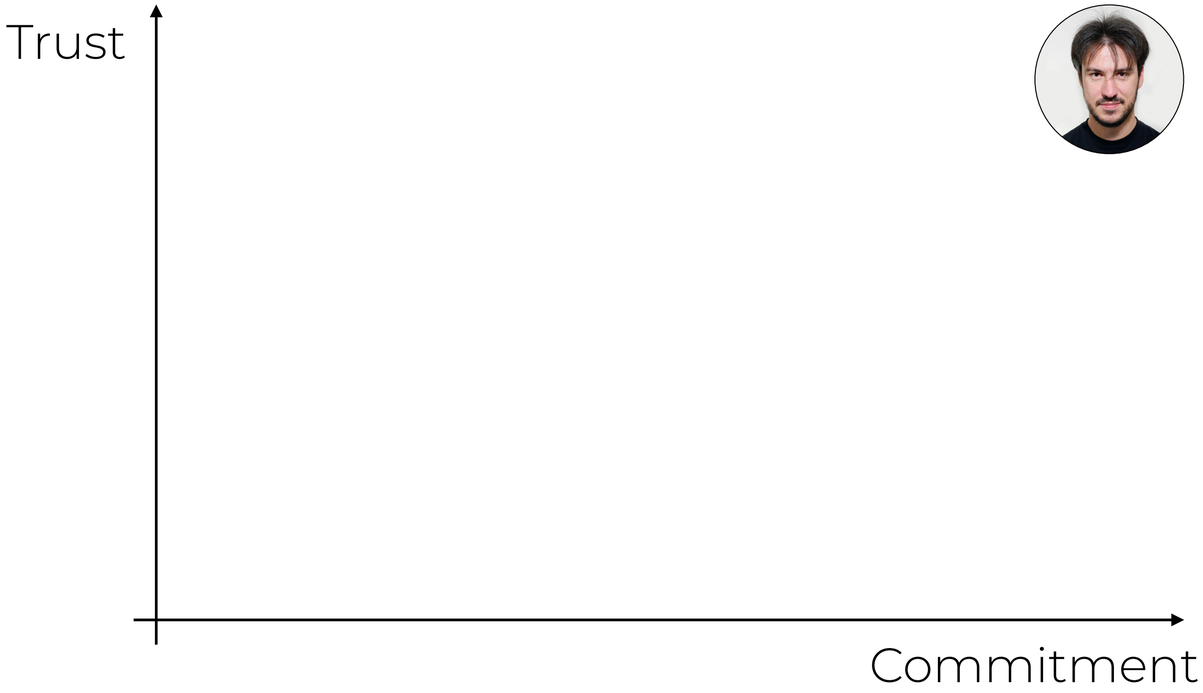
So, we could use the parrot to generate the production SQL

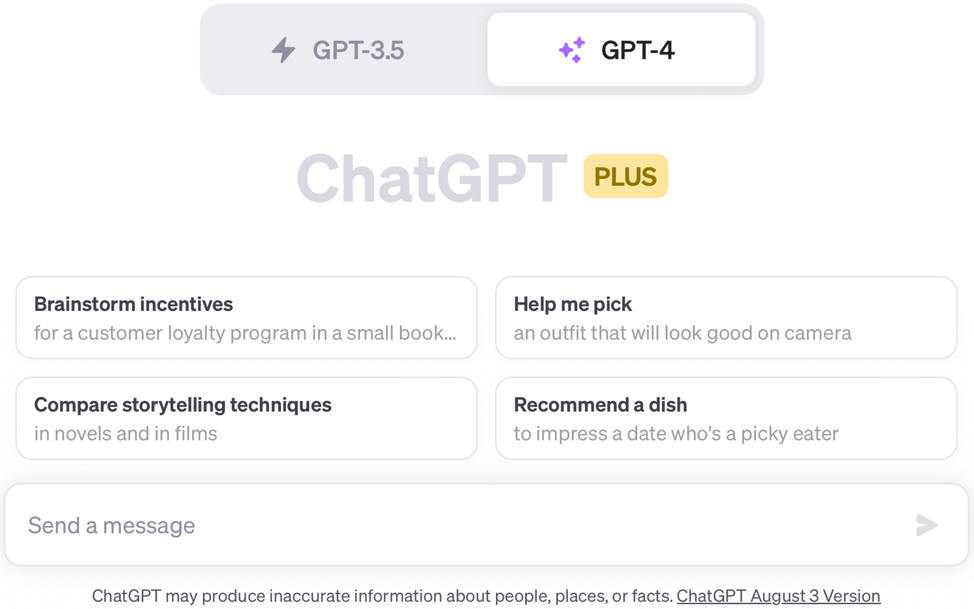
Did we actually do it?

No

Why?

- Time saving?
  - Time to write the description
  - Time to verify
- Trust
- Ownership





There are possible trust issues when using LLMs as companions in production. What about other contexts?

Parrots as Companions

Teaching Context

Consider the following snippet of Java code... It represents ***a method that checks that a given configuration of a TicTacToe board, represented as a char array of 9 elements, is valid*** *- i.e., it represents a board configuration that can be reached when following the rules of the game... The game assumes that X starts.*

```
public class TicTacToe {
  // ... missing irrelevant part ...
  static boolean isValid(char board[]) {

    // Count number of 'X' and 'O'
    // in the given board
    int xCount = 0, oCount = 0;
    for (int i = 0; i < 9; i++) {
        if (board[i] == 'X') {
            xCount++;
        }
        if (board[i] == 'O') {
            oCount++;
        }
    }
```

```java
public class TicTacToe {
  // ... missing irrelevant part ...
  static boolean isValid(char board[]) {

    // Count number of 'X' and 'O'
    // in the given board
    int xCount = 0, oCount = 0;
    for (int i = 0; i < 9; i++) {
        if (board[i] == 'X') {
            xCount++;
        }
        if (board[i] == 'O') {
            oCount++;
        }
    }
```

```java
    // Board can be valid only if either xCount and oCount
    // is same or count is one more than oCount
    if (xCount == oCount || xCount == oCount + 1) {
     if (isWinningConfig(board, 'O')) {
         // Check if 'X' is also winner, then return false
         if (isWinningConfig(board, 'X')) {
            return false;
         }
         // Else return true xCount and yCount are same
         return (xCount == oCount);
      }
      // If 'X' wins, then count of X must be greater
      if (isWinningConfig(board, 'X') && xCount != oCount + 1) {
         return false;
      }
      // If 'O' is not winner, then return true
      return true;
    }
    return false;
  }
```

Parrots as Companions

Teaching Context for Testing

```java
public class TicTacToe {
  // ... missing irrelevant part ...
  static boolean isValid(char board[]) {

    // Count number of 'X' and 'O'
    // in the given board
    int xCount = 0, oCount = 0;
    for (int i = 0; i < 9; i++) {
        if (board[i] == 'X') {
            xCount++;
        }
        if (board[i] == 'O') {
            oCount++;
        }
    }
```

There is a test case which will cover all branches but not all statements.

```java
    // Board can be valid only if either xCount and oCount
    // is same or count is one more than oCount
    if (xCount == oCount || xCount == oCount + 1) {
     if (isWinningConfig(board, 'O')) {
         // Check if 'X' is also winner, then return false
         if (isWinningConfig(board, 'X')) {
            return false;
         }
         // Else return true xCount and yCount are same
         return (xCount == oCount);
      }
      // If 'X' wins, then count of X must be greater
      if (isWinningConfig(board, 'X') && xCount != oCount + 1) {
         return false;
      }
      // If 'O' is not winner, then return true
      return true;
    }
    return false;
  }
```

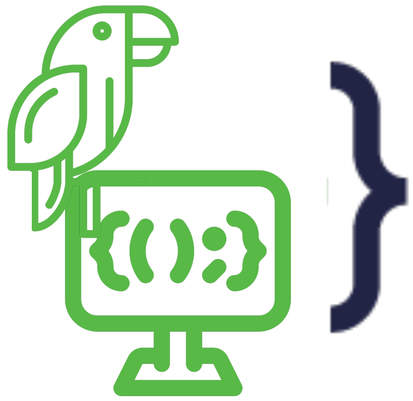
```java
public class TicTacToe {
  // ... missing irrelevant part ...
  static boolean isValid(char board[]) {

    // Count number of 'X' and 'O'
    // in the given board
    int xCount = 0, oCount = 0;
    for (int i = 0; i < 9; i++) {
        if (board[i] == 'X') {
            xCount++;
        }
        if (board[i] == 'O') {
            oCount++;
        }
    }
```

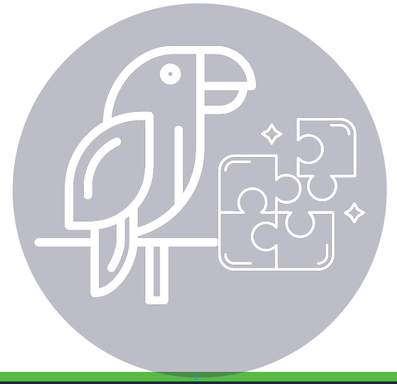
There is a test case which will cover all branches but not all statements.

```java
    // Board can be valid only if either xCount and oCount
    // is same or count is one more than oCount
    if (xCount == oCount || xCount == oCount + 1) {
     if (isWinningConfig(board, 'O')) {
         // Check if 'X' is also winner, then return false
         if (isWinningConfig(board, 'X')) {
            return false;
         }
         // Else return true
         return (xCount == o
     }
     // If 'X' wins, then c
     if (isWinningConfig(bo
         return false;
     }
     // If 'O' is not winne
     return true;
    }
    return false;
  }
```

Supporting
Developers

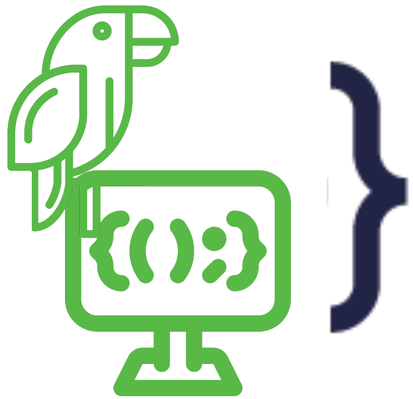


Parrots as
Companions



Parrots as
Task Solvers

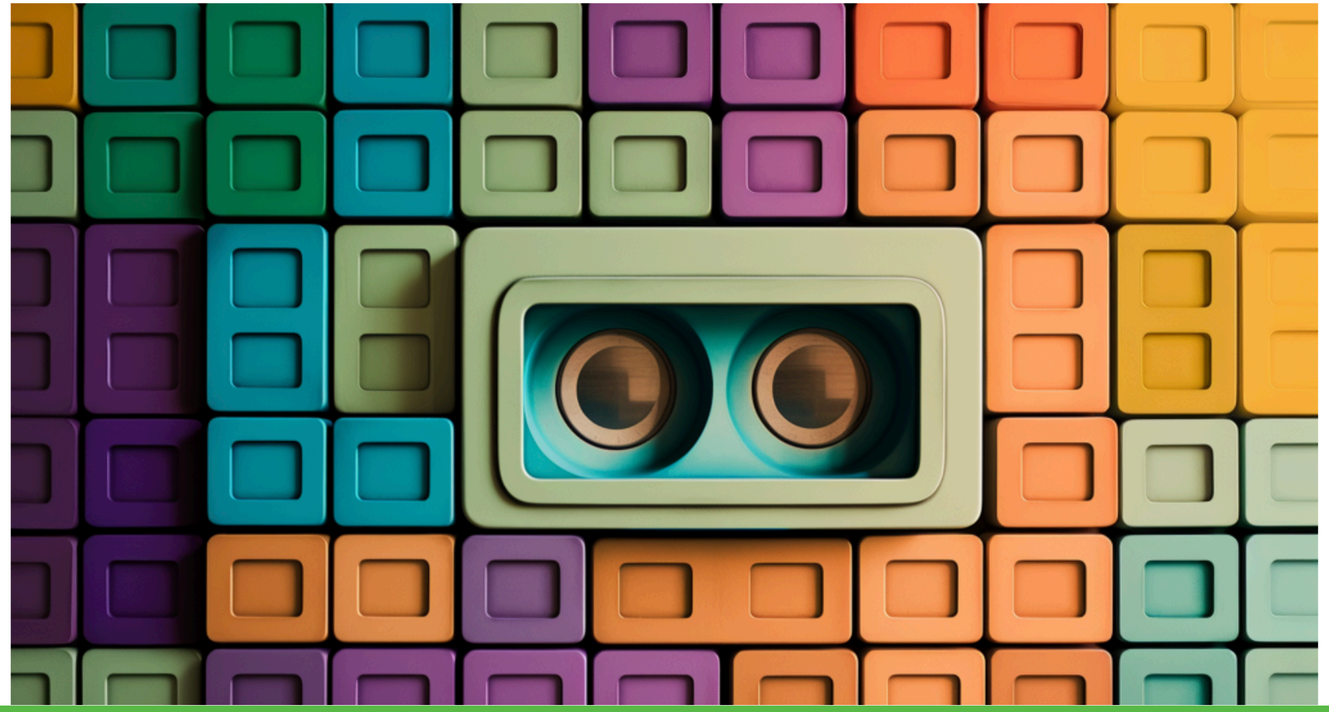
Parrots as
Companions



Parrots as
Task Solvers

Issues
& Next Steps

I replaced 50 lines of code with a single LLM prompt



REFLEX

REGISTRY OF FIRMS' LIFE AND EXIT

REFLEX Project &
Swiss Gazzette of Commerce

Business Founding, Liquidations

Acquisitions, Splittings

(Re)locations

...

Trademarks


№ I — II. Theil. 1883 IV. Partie — № I

Schweizerisches Handelsamtsblatt

Feuille officielle suisse du commerce – Foglio ufficiale svizzero di commercio

Organisation des Handelsregisters in den Kantonen.


Schweizerisches Handelsamtsblatt

Feuille officielle suisse du commerce – Foglio ufficiale svizzero di commercio


Schweizerisches Handelsamtsblatt

Feuille officielle suisse du commerce • Foglio ufficiale svizzero di commercio

Amtlicher Teil – Partie officielle – Parte ufficiale


Schweizerisches Handelsamtsblatt

Feuille officielle suisse du commerce

Foglio ufficiale svizzero di commercio

AMTLICHER TEIL – PARTIE OFFICIELLE – PARTE UFFICIALE


Schweizerisches Handelsamtsblatt

Feuille officielle suisse du commerce

Foglio ufficiale svizzero di commercio

AMTLICHER TEIL – PARTIE OFFICIELLE – PARTE UFFICIALE

1883

2001

REFLEX

REGISTRY OF FIRMS' LIFE AND EXIT

SOGC, FOSC, FUSC

Swiss Gazzette of Commerce

14 mai 1998
Restaurant-Pizzeria l'Altro Mondo Sàrl, à V e v e y, avenue Nestlé 11. Nouvelle société à responsabilité limitée. Statuts: 11 mai 1998. But: exploitation d'établissement publies, notamment d'un restaurant-pizzeria. Capital: fr. 20 000. Associés-gérants avec signature individuelle: Dino Addorisio, Gina Carosielli, tous deux d'Italie, à Vevey, chacun pour une part de fr. 10 000. Publications: FOSC.

New registrations

14 mai 1998
Fountain Léman, Claude Cegarra, à V e v e y, vente de distributeurs de boissons chaudes (FOSC du 9. 4. 1996, p. 1982). Nouvelle adresse: rue de la Madeleine 11.

Change of Address (Location)

26.05.1998 (6104)
Hilera SA, à G e n è v e ; prise et administration de participations à toutes entreprises, etc. (FOSC du 12.04.1989, p. 1467). Nouvcaux statuts du 20-05-1998. Mathier Jacques n'cst plus administratcur; ses pouvoirs sont radiés. Administration: Schmid Walter, de et à Genève, administratcur unique avec signature individuelle. Réviscur: "STG-Coopers & Lybrand SA", succursale à Genève. Nouvelle adresse: rue Verdaine 15, c/Walter Schmid.

Other mutations (people involved, capital, etc.)

Berichtigungen
RS Roland Selm AG, in L u z e r n (SHAB Nr. 96 vom 20. 5. 1998, S. 3431). Domizil: Lützelmattstrasse 18, 6006 Luzern (nicht Lugano).

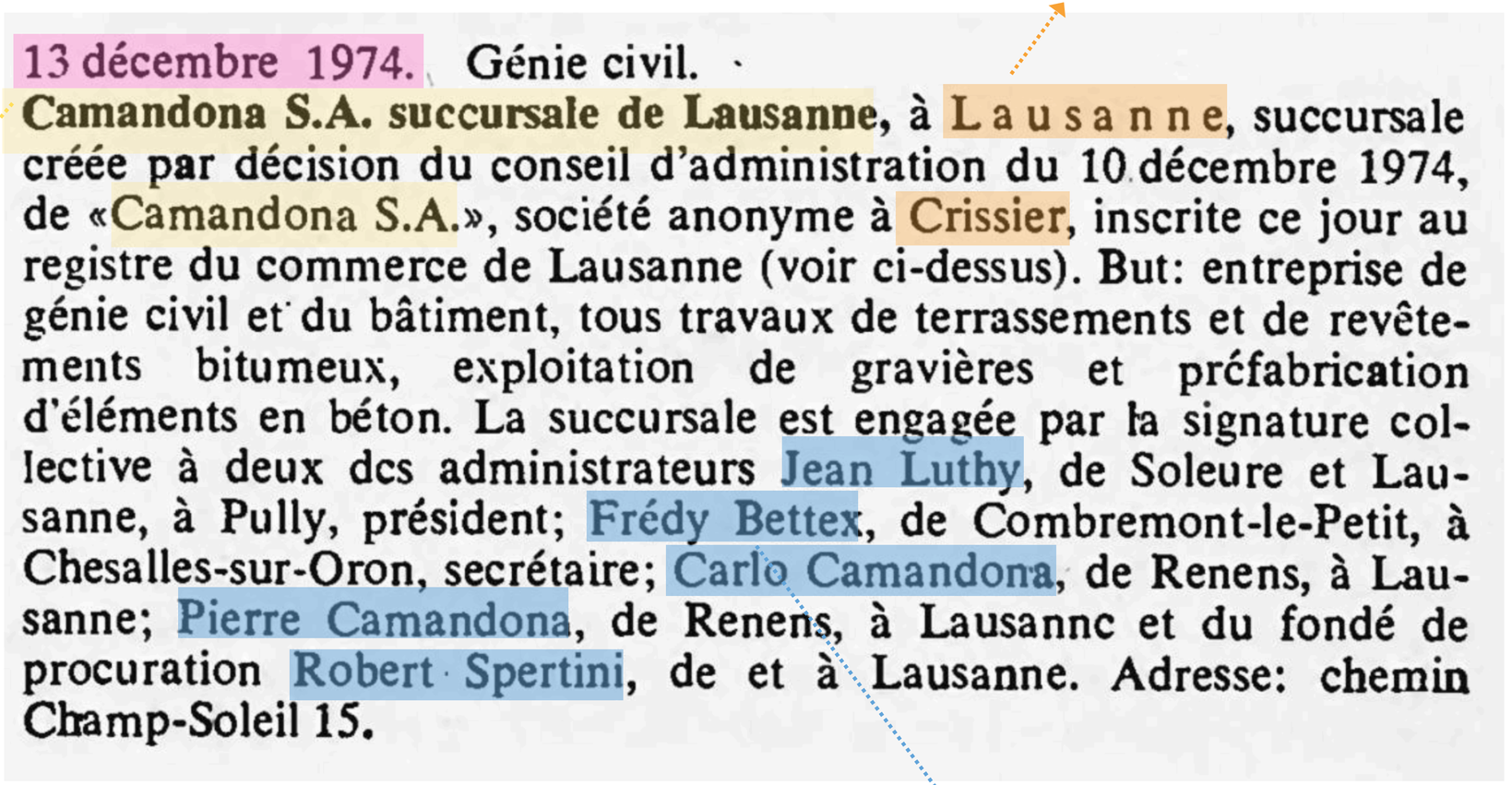
Getränkehandel und Transporte Feuchter, in M a l t e r s (SHAB Nr. 96 vom 20. 5. 1998, S. 3432). Domizil: Eistrasse 3 (nicht 2), 6102 Malters.

Corrections

SOGC
Commercial Registry Entries

Date

Location



13 décembre 1974. Génie civil.

Camandona S.A. succursale de Lausanne, à Lausanne, succursale créée par décision du conseil d'administration du 10 décembre 1974, de «Camandona S.A.», société anonyme à Crissier, inscrite ce jour au registre du commerce de Lausanne (voir ci-dessus). But: entreprise de génie civil et du bâtiment, tous travaux de terrassements et de revêtements bitumeux, exploitation de gravières et préfabrication d'éléments en béton. La succursale est engagée par la signature collective à deux des administrateurs Jean Luthy, de Soleure et Lausanne, à Pully, président; Frédy Bettex, de Combremont-le-Petit, à Chesalles-sur-Oron, secrétaire; Carlo Camandona, de Renens, à Lausanne; Pierre Camandona, de Renens, à Lausanne et du fondé de procuration Robert Spertini, de et à Lausanne. Adresse: chemin Champ-Soleil 15.

Company

People

SOGC

Commercial Registry Entries

Date

Location

Company

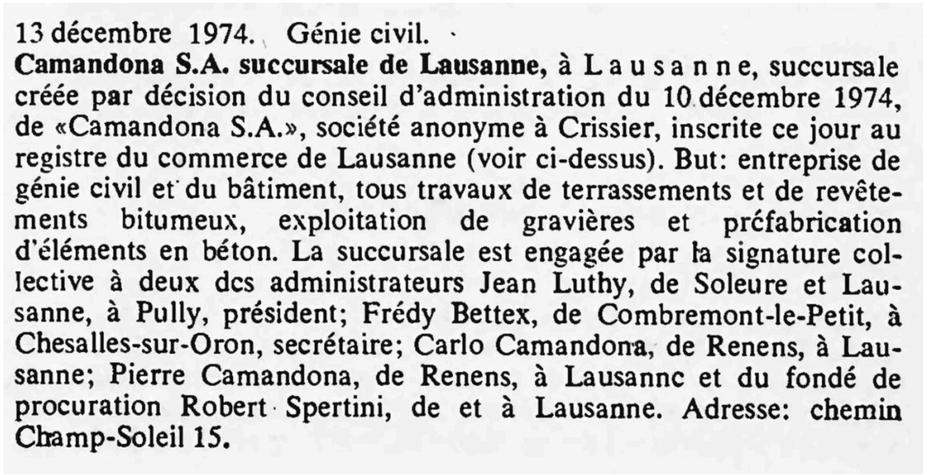
People

13 décembre 1974. Génie civil.

Camandona S.A. succursale de Lausanne, à Lausanne, succursale créée par décision du conseil d'administration du 10 décembre 1974, de «Camandona S.A.», société anonyme à Crissier, inscrite ce jour au registre du commerce de Lausanne (voir ci-dessus). But: entreprise de génie civil et du bâtiment, tous travaux de terrassements et de revêtements bitumeux, exploitation de gravières et préfabrication d'éléments en béton. La succursale est engagée par la signature collective à deux des administrateurs Jean Luthy, de Soleure et Lausanne, à Pully, président; Frédy Bettex, de Combremont-le-Petit, à Chesalles-sur-Oron, secrétaire; Carlo Camandona, de Renens, à Lausanne; Pierre Camandona, de Renens, à Lausanne et du fondé de procuration Robert Spertini, de et à Lausanne. Adresse: chemin Champ-Soleil 15.

SOGC

Named Entity Recognition



13 décembre 1974. Génie civil.
Camandona S.A. succursale de Lausanne, à Lausanne, succursale créée par décision du conseil d'administration du 10 décembre 1974, de «Camandona S.A.», société anonyme à Crissier, inscrite ce jour au registre du commerce de Lausanne (voir ci-dessus). But: entreprise de génie civil et du bâtiment, tous travaux de terrassements et de revêtements bitumeux, exploitation de gravières et préfabrication d'éléments en béton. La succursale est engagée par la signature collective à deux des administrateurs Jean Luthy, de Soleure et Lausanne, à Pully, président; Frédy Bettex, de Combremont-le-Petit, à Chesalles-sur-Oron, secrétaire; Carlo Camandona, de Renens, à Lausanne; Pierre Camandona, de Renens, à Lausanne et du fondé de procuration Robert Spertini, de et à Lausanne. Adresse: chemin Champ-Soleil 15.

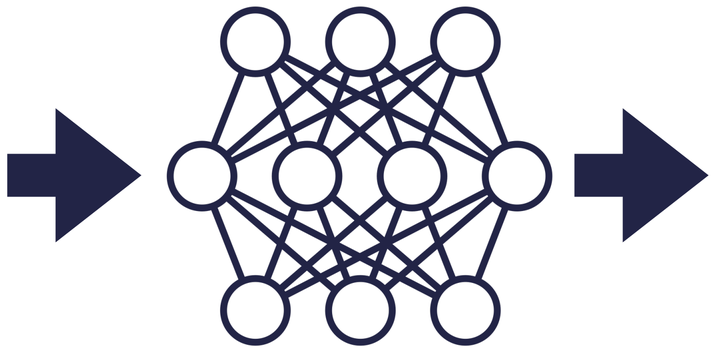
→ spaCy →


```
{
   "entities": [
     {
       "text": "Camandona...",
       "type": "ORG",
       "start": ...,
       "end": ...
     },
     {
       "text": "Lausanne",
       "type": "LOC",
       "start": ...,
       "end": ...
     },
     ...
]
```


SOGC
Named Entity Recognition

13 décembre 1974. Génie civil.

Camandona S.A. succursale de Lausanne, à Lausanne, succursale créée par décision du conseil d'administration du 10 décembre 1974, de «Camandona S.A.», société anonyme à Crissier, inscrite ce jour au registre du commerce de Lausanne (voir ci-dessus). But: entreprise de génie civil et du bâtiment, tous travaux de terrassements et de revêtements bitumeux, exploitation de gravières et préfabrication d'éléments en béton. La succursale est engagée par la signature collective à deux des administrateurs Jean Luthy, de Soleure et Lausanne, à Pully, président; Frédy Bettex, de Combremont-le-Petit, à Chesalles-sur-Oron, secrétaire; Carlo Camandona, de Renens, à Lausanne; Pierre Camandona, de Renens, à Lausannc et du fondé de procuration Robert Spertini, de et à Lausanne. Adresse: chemin Champ-Soleil 15.



```
{
   "entities": [
     {
       "text": "Camandona...",
       "type": "ORG",
       "start": ...,
       "end": ...
     },
     {
       "text": "Lausanne",
       "type": "LOC",
       "start": ...,
       "end": ...
     },
     ...
]
```

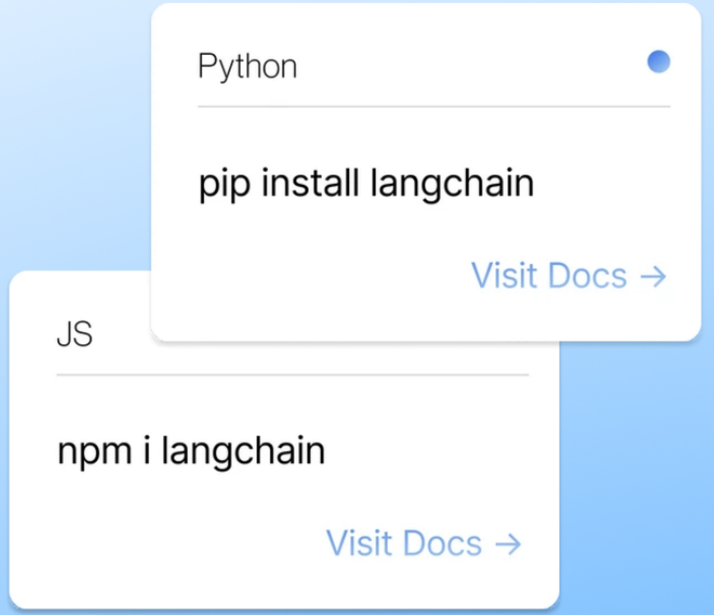
SOGC

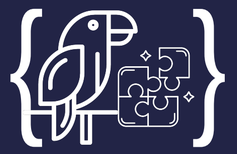
Named Entity Recognition

LangChain

Get your LLM application from prototype to production

Build context-aware, reasoning applications with LangChain's flexible abstractions and AI-first toolkit.





Stochastic Parrots as Task Solvers
Engineering LLM Solutions

<Modern AI> made easy.

XEF is a library to bring the power of modern AI to your application or service, in the form of LLM (Large Language Models), image generation, and many others.

What is xef | Quickstart

Stochastic Parrots as Task Solvers
Engineering LLM Solutions

<Modern AI> made easy.

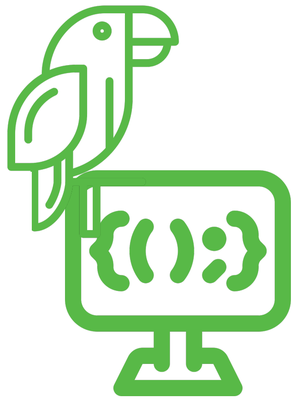
Demo

What is xef | Quickstart

Stochastic Parrots as Task Solvers

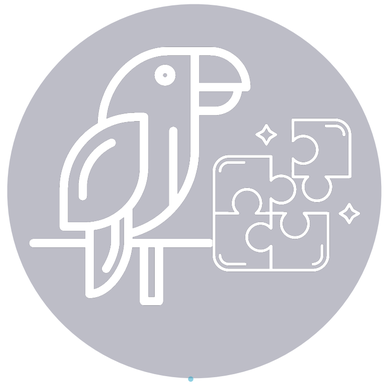
Engineering LLM Solutions: Demo

Parrots as
Companions



Parrots as
Task Solvers

Conclusion

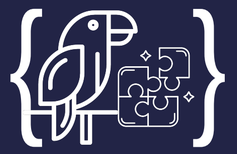


Parrots as
Task Solvers

Conclusion

AI is moving quick, impacting many sectors and industries

Studying the side-effects of certain practices is important and urgent



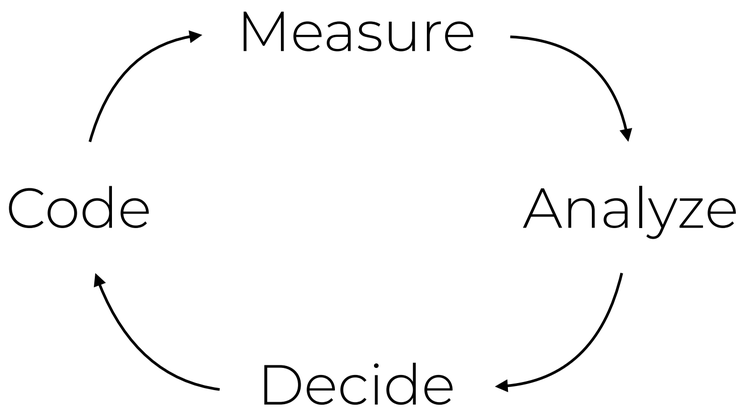
Conclusion
A real disruptive innovation

AI is moving quick, impacting many sectors and industries

Studying the side-effects of certain practices is important and urgent

Conclusion
A real disruptive innovation

Variation of DMAIC



received an email from the CEO of a company that is interested in measuring the impact that CoPilot has in practice. The company is quite well established They want to first use Copilot for a small project, and see whether they can generalize its use depending on the results.

Conclusion
Should I use an AI assistant?

Book.scala | bias.py | 2023-02-16T08:32:39.350Z.ipynb - Tako Copilot Digest

(2 hours, 4677 events)
(2 hours, 13508 events)
M (13 minutes, 3230 eve...
(3 hours, 16206 events)
(4 hours, 15950 events)
M (36 minutes, 6111 even...
(2 hours, 15095 events)
(an hour, 5617 events)
M (4 hours, 12163 events)
(3 hours, 11901 events)
(4 hours, 6520 events)
M (an hour, 4156 events)
(an hour, 21199 events)
(an hour, 4109 events)
(2 hours, 18000 events)
(an hour, 3131 events)
(40 minutes, 2162 events)
(3 hours, 8388 events)
(an hour, 5356 events)
(3 hours, 2011 events)

Session Statistics

Project	company-networks
Date	16/02/2023
Start-End	09:32 – 12:13

Languages

Copilot Statistics

Times Triggered	1940
Suggestions	251
Suggestion Rate	12.94%
Accepted Suggestions	30
Acceptance Rate	11.95%
Avg. Time to Acceptance	1.0 s

Copilot Metrics

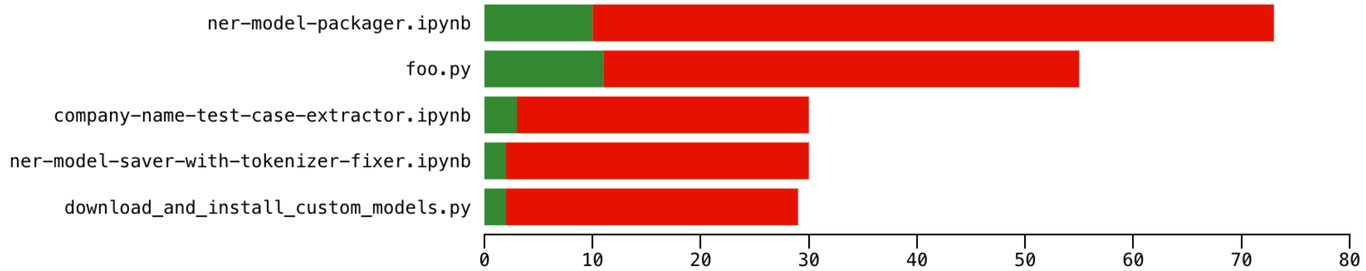
Lines of Code

Suggested	313
Accepted	43
% Accepted	13.74%

Characters

Suggested	11415
Accepted	1738
% Accepted	15.23%

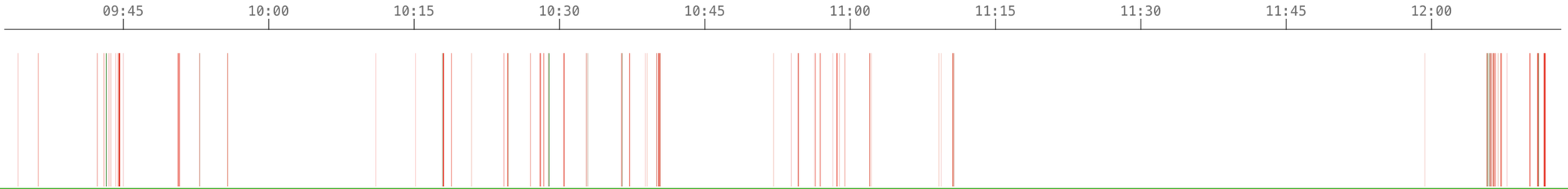
Hottest Documents

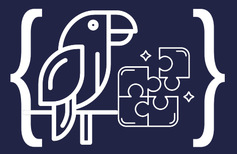


Legends

Copilot Event Types

accepted
rejected





Conclusion

Should I use an AI assistant?

The Rise of Stochastic Parrots for Developers

Marco D'Ambros, Andrea Mocci

codelounge