UNIVERSE'24

An inside look at how GitHub uses LLMs, fine-tuning, and prompt engineering in GitHub Copilot

Alireza Goudarzi

Senior Machine Learning Researcher

Copilot Prompt Team

Ex Copilot Model Team

Code completion with GitHub Copilot

fibonacci.py 1

Users > matthew-peter > Desktop > fibonacci.py

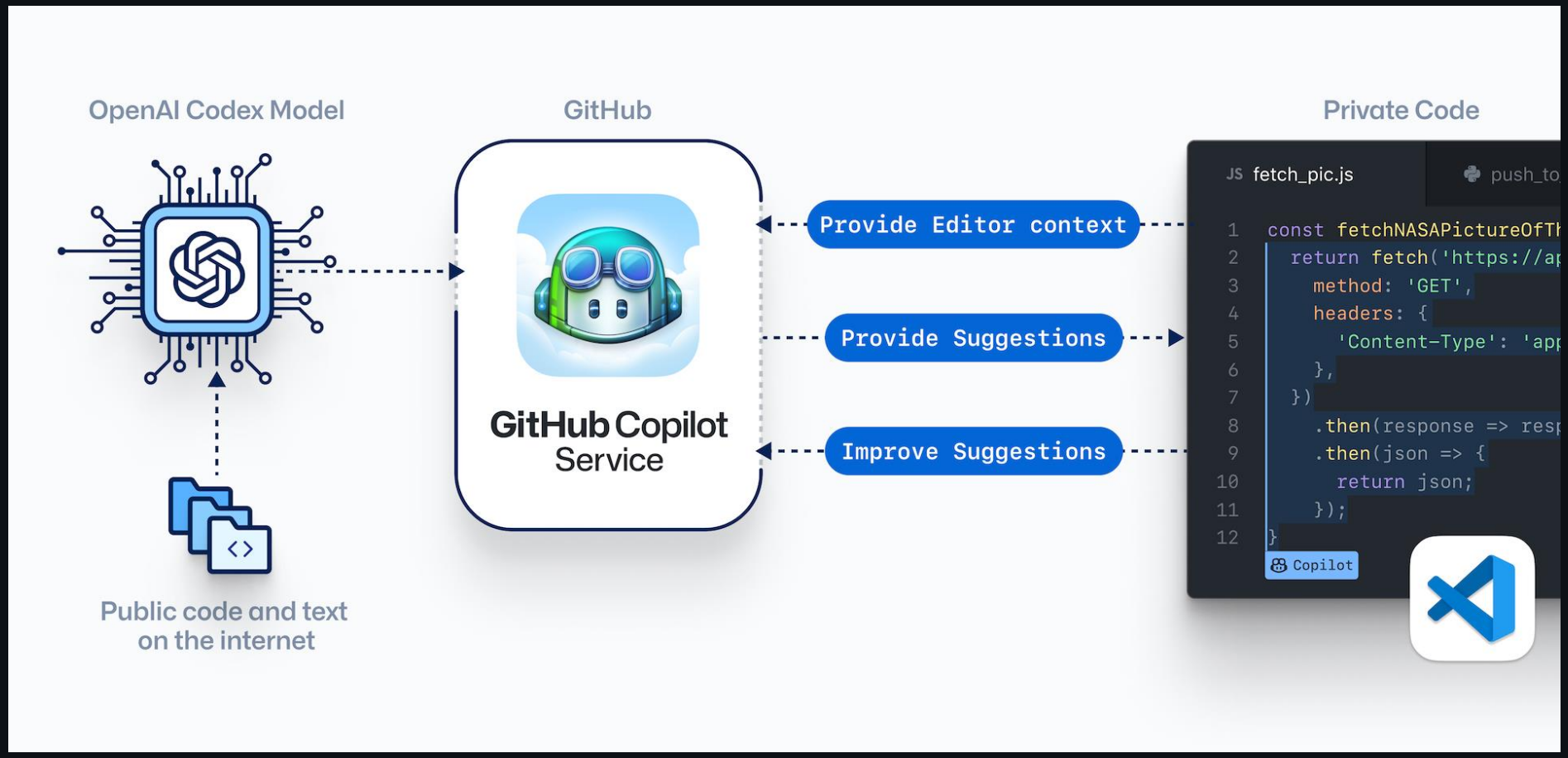
```
1  # write a function that returns the nth number in the fibonacci sequence
2  def
```

fibonacci.py 1

Users > matthew-peter > Desktop > fibonacci.py

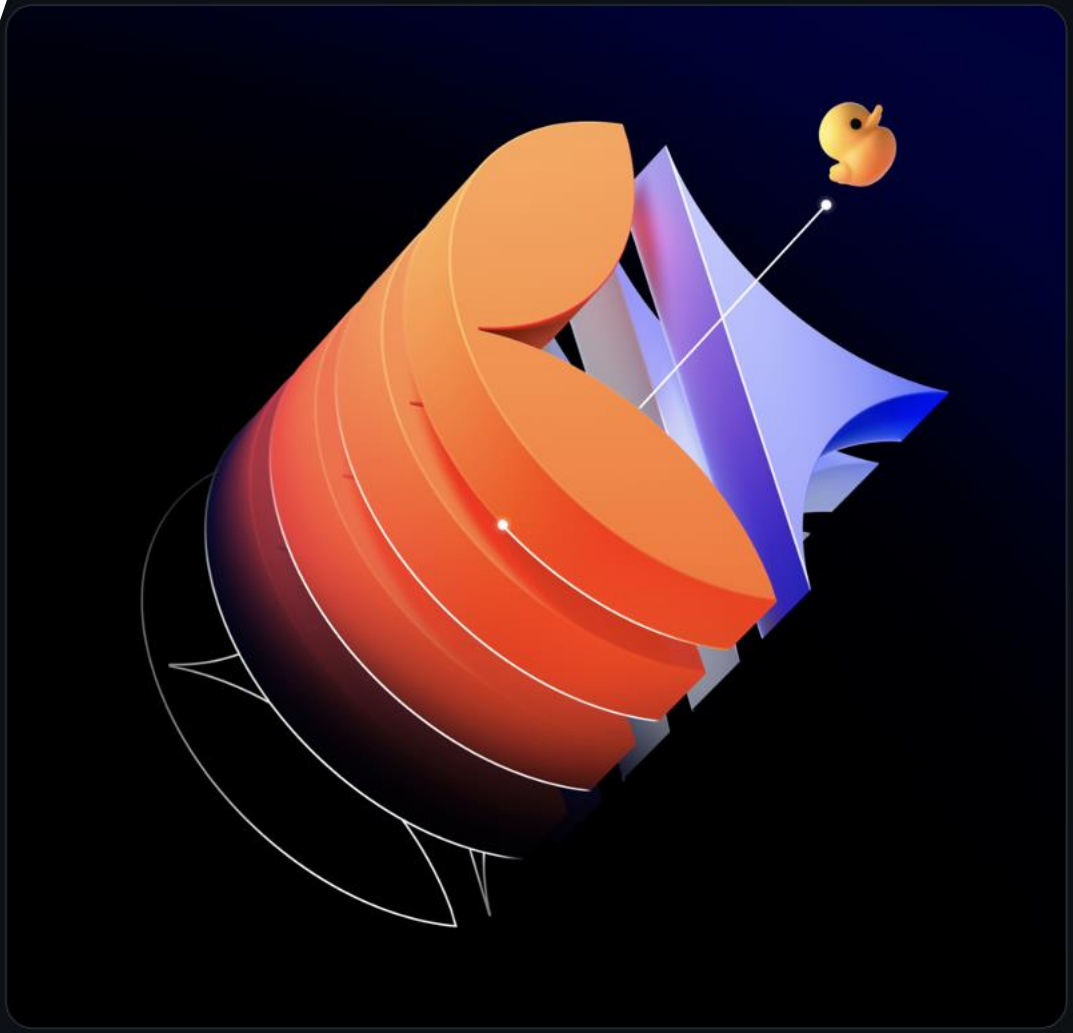
```
1  # write a function that returns the nth number in the fibonacci sequence
2  def fibonacci(n):
       if n == 0:
           return 0
       if n == 1:
           return 1
       return fibonacci(n - 1) + fibonacci(n - 2)
```

Copilot Experience Today

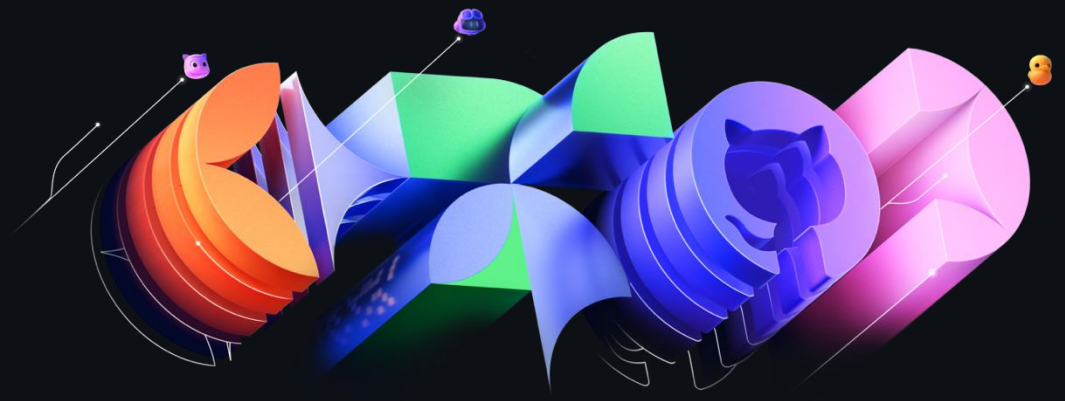


Copilot Experience Today

- Neighboring tabs
- Snipy
- Responsible AI
- Truncation



UNIVERSE'24



Let's get creative: Features, Improvements, and more!

The LLMs Behind Copilot

OpenAI

ANTHROP\C

Meta AI

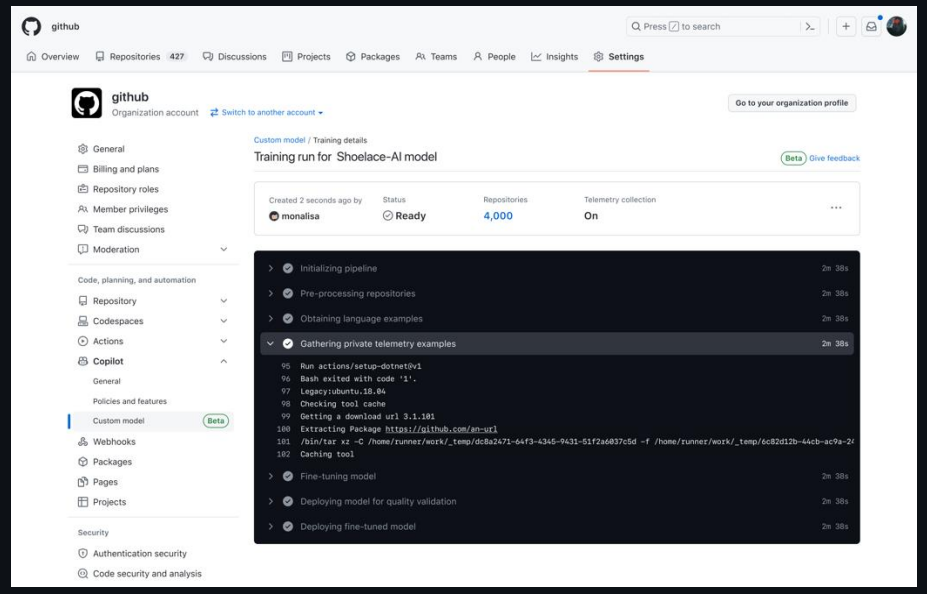
cohere

MISTRAL AI_

Fine-tuned Copilot Models for Enterprises

- Fine-tune GitHub Copilot to better understand and align with your organization's unique coding practices
  - Enhance library and API knowledge
  - Specialized languages
  - Adapt to evolving codebases

- Only accessible to your org

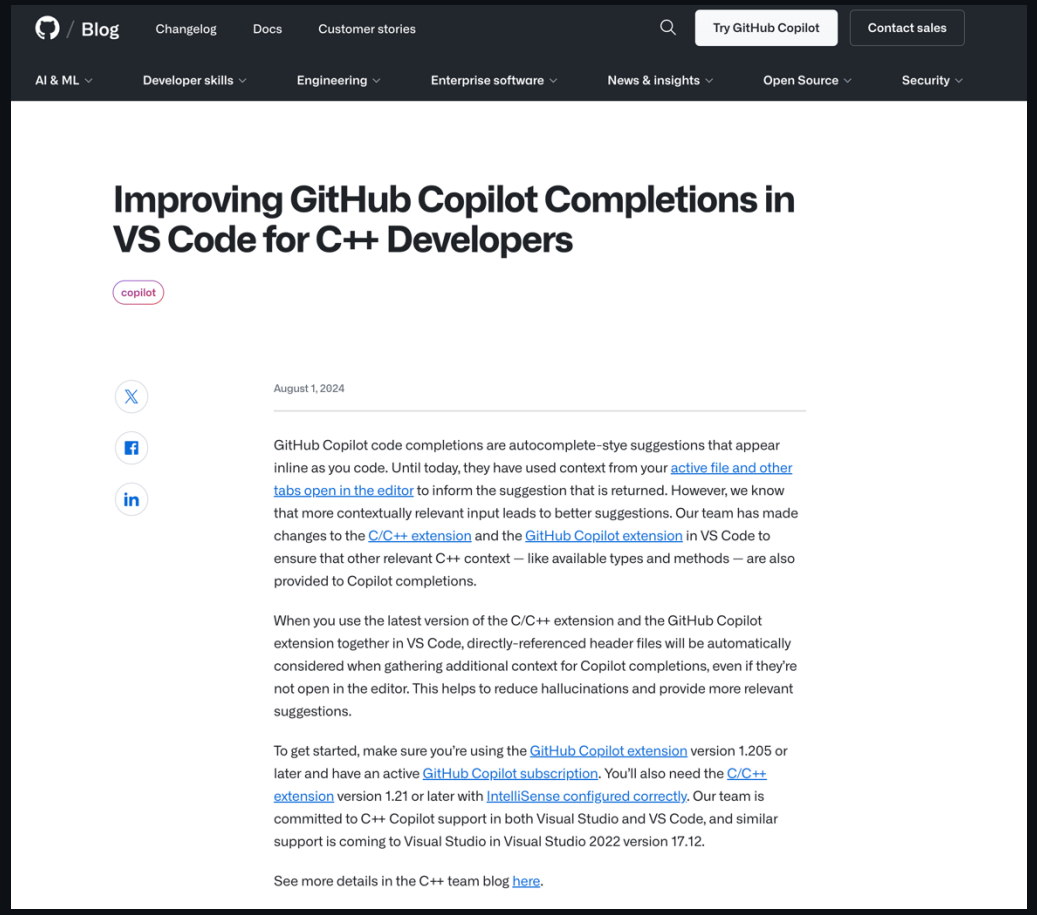


Language-specific LLM Fine-tuning

- Identify most impactful language opportunities to improve GitHub Copilot
  - Beginning with C# and C++
- How can we improve the underlying LLM to improve GitHub Copilot performance on a specific language?
  - Data engineering
  - Fine-tuning methodology

Contextualization

- Language-specific contextualization
- Project-wide RAG



Blog Changelog Docs Customer stories Try GitHub Copilot Contact sales

AI & ML Developer skills Engineering Enterprise software News & insights Open Source Security

Improving GitHub Copilot Completions in VS Code for C++ Developers

copilot

August 1, 2024

GitHub Copilot code completions are autocomplete-stye suggestions that appear inline as you code. Until today, they have used context from your active file and other tabs open in the editor to inform the suggestion that is returned. However, we know that more contextually relevant input leads to better suggestions. Our team has made changes to the C/C++ extension and the GitHub Copilot extension in VS Code to ensure that other relevant C++ context — like available types and methods — are also provided to Copilot completions.

When you use the latest version of the C/C++ extension and the GitHub Copilot extension together in VS Code, directly-referenced header files will be automatically considered when gathering additional context for Copilot completions, even if they're not open in the editor. This helps to reduce hallucinations and provide more relevant suggestions.

To get started, make sure you're using the GitHub Copilot extension version 1.205 or later and have an active GitHub Copilot subscription. You'll also need the C/C++ extension version 1.21 or later with IntelliSense configured correctly. Our team is committed to C++ Copilot support in both Visual Studio and VS Code, and similar support is coming to Visual Studio in Visual Studio 2022 version 17.12.

See more details in the C++ team blog here.

Next Edit Suggestions

- Can GitHub Copilot help make suggestions away from your cursor?

```
49
50  const enabled =
51    git.repositories.length > 0 &&
52    (store.enabled || git.repositories[0]?.state.HEAD?.name === EXTENSION_NAME);
53
54  updateContext(enabled, false);
```

```
const enabled =
  git.repositories.length > 0 &&
  (store.enabled || git.repositories[0]?.state.HEAD?.name === EXTENSION_NAME) &&
  !config.denyBranches.includes(git.repositories[0]?.state.HEAD?.name);
```

```
45  },
46  get filePattern() {
47    return config().get("filePattern", "**/*");
48  },
49  get pullOnOpen() {
50    return config().get("pullOnOpen", true);
```

```
get denyBranches(): string[] {
  return config().get("denyBranches", []);
},
get filePattern() {
```

Improvements to GitHub Copilot

- Offline evaluation is inherently difficult
- A/B testing: What do users prefer?

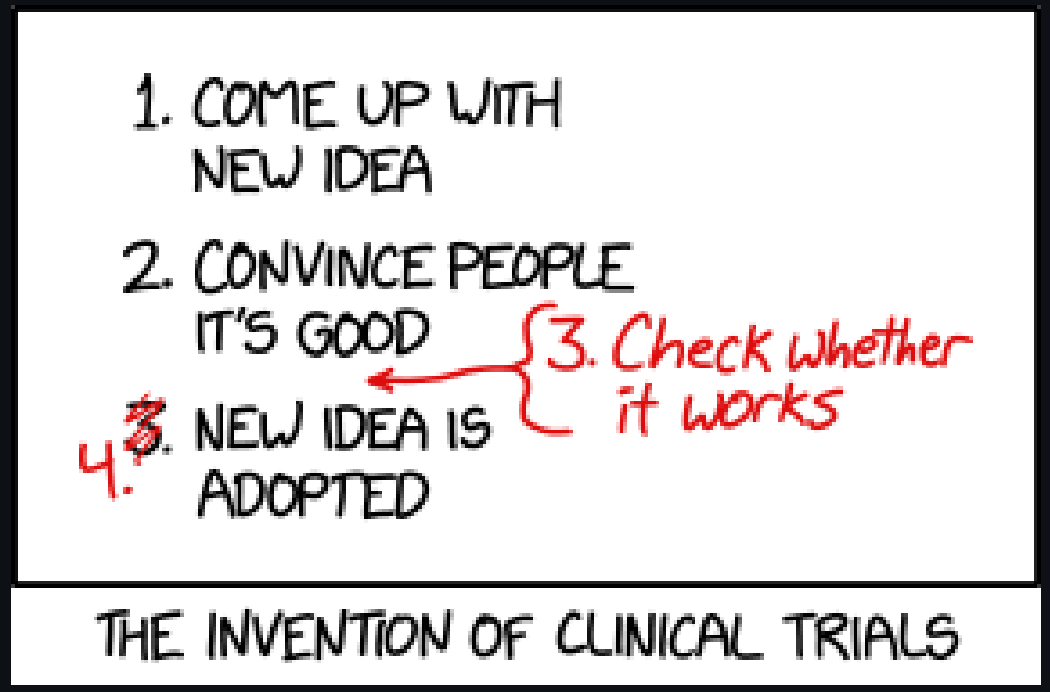


Image from https://xkcd.com/2530

Thank you!