



AI Toolkit for Developers

Speaker Rafał Dubowski



Who am I?

- Python developer
- Former IT recruiter
- Canary bird f(e)ather



Why?



In the beginning...



Then

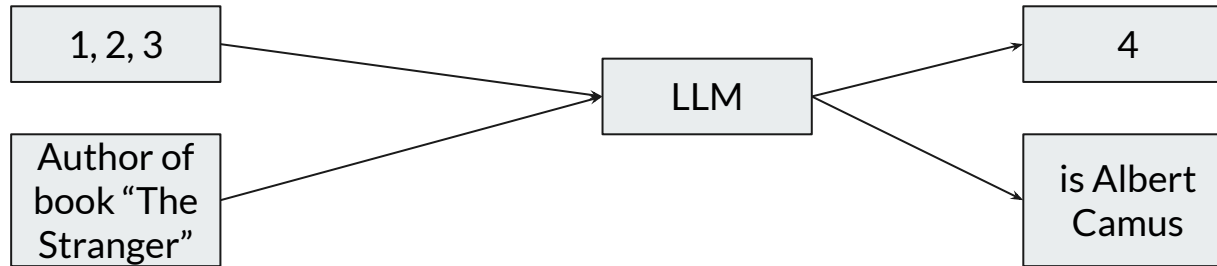


Agenda

1. Basic information about LLM
2. AI tools for developers
3. Custom tools
4. Security and other doubts
5. More tools
6. Q&A

Basic information

- a) Given previous tokens (letters, morphemes, words,...), predict the next token



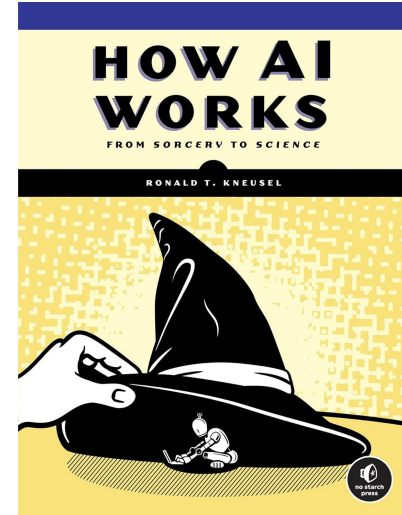
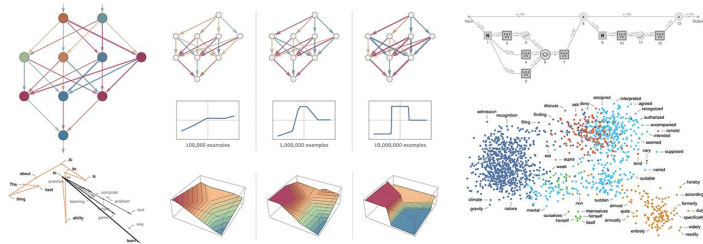
- b) Large language models (LLMs) are trained on massive datasets of text and code, which may or may not be publicly available

Basic information

Stephen Wolfram Article

What Is ChatGPT Doing ... and Why Does It Work?

February 14, 2023





Basic information - chat

Send a message



Zadaj pytanie





API

Step 1: Setup Python

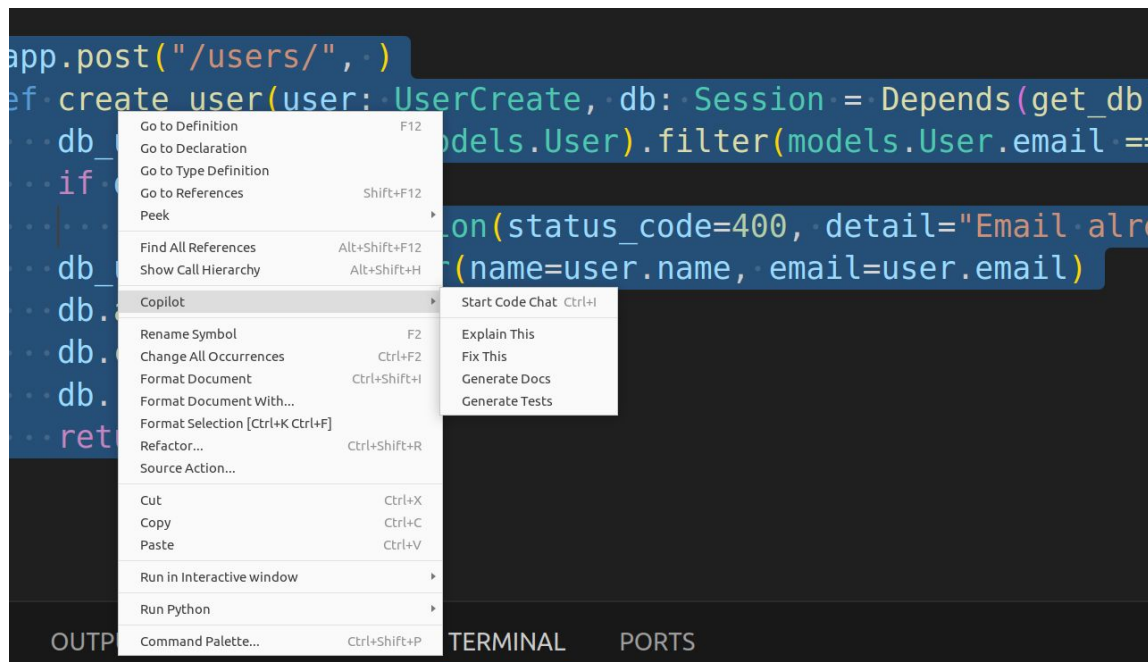
- Install Python
-
- Setup a virtual environment (optional)
-
- Install the OpenAI Python library
-

Step 2: Setup your API key

- Setup your API key for all projects (recommended)
-
- Setup your API key for a single project
-

Step 3: Sending your first API request

IDE / In IDE extensions



Chats



free

Results - one
step, 3
options to
choose

Show web
search
results



free/
20\$

Results - step
by step

Web search
only for 20\$

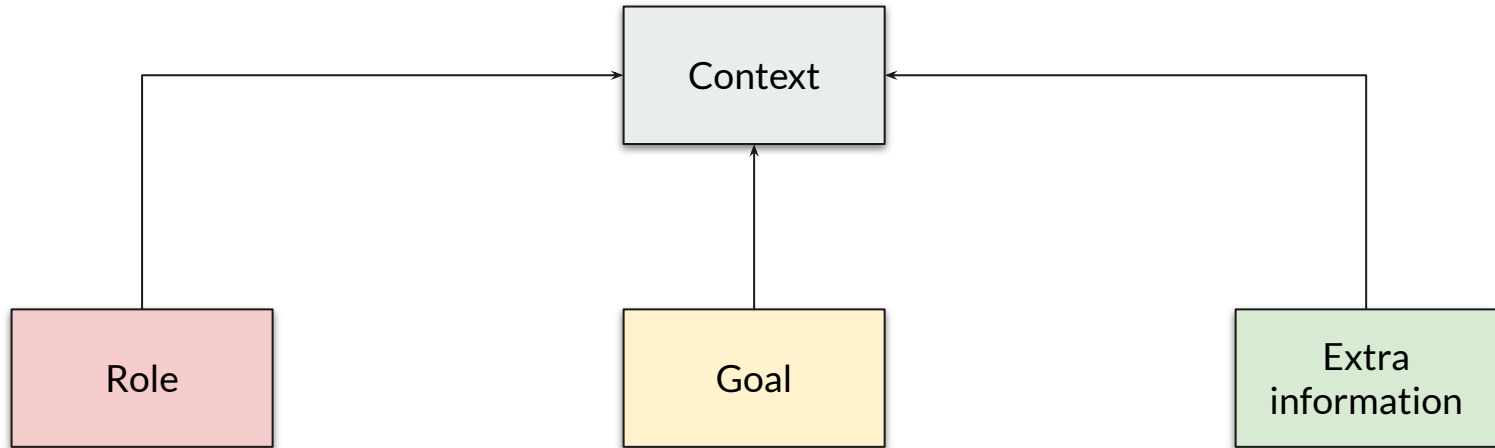


free

Results - step
by step

Show web
search
results

Prompts - it's about context





Prompts - it's about context

I want you to act as a software developer.

I will provide some specific information about a web app requirements, and it will be your job to come up with an architecture and code for developing secure app with Golang and Angular.

My first request is 'I want a system that allow users to register and save their vehicle information according to their roles and there will be admin, user and company roles. I want the system to use JWT for security'.



Types of tasks

1. Explain the code.
2. Refactor the code. Prompt: “Analyze the code and create a list of improvement suggestions”.
3. Getting dummy data.
4. Tests.
5. Building blocks of code for app (but ...).



My thoughts

1. Only for popular programming language.
2. Hard to develop according to the newest standards.
3. Hard to develop with tools from last 6 months. You can ask (if using bard or gpt-4) about using internet for search.
4. It's very slow (chat gpt).
5. Good for: "do according to that convention".
6. Setup project from scratch is very difficult.



IDE / In IDE extensions

Cursor
IDE









Cursor IDE

1. Built-in chat.
2. Make code changes.
3. Spot and fix bugs: Cursor.
4. Generate documentation and tests.
5. Free.

CHAT

MORE



2,12,61,2021-02-15,5,61.8

3,78,45,2020-12-09,1,23.5.

4,34,89,2021-01-30,3,78.9.

5,23,56,2020-11-20,2,45.6

6,67,34,2021-03-15,4,89.3.

7,89,12,2020-12-25,1,56.7.

8,56,78,2021-02-28,3,67.8

9,34,90,2020-11-29,2,78.9.

10,12,34,2021-03-20,5,89..

11,78,56,2020-12-15,1,90..

12,90,23,2021-01-10,3,67..

▼

How can I clean the data?

@analyze.py @sales_data.csv

@Pandas

↩ chat ⌘ ↩ with codebase

Debug errors

Cursor automatically investigates linter errors and stack traces to figure out the root cause of your bug

```
6
7 export const InjectableInterfaceAgentService =
8   createDecorator<InterfaceAgentService> ('InterfaceAgentService');
9
10
11 // Fixing Errors...
12 The 'InterfaceAgentService' decorator is created before its corresponding interface, potentially
13 causing unexpected behavior.
14 Suggested fix:
15
16 export interface InterfaceAgentService { /*methods*/ }
17 export interface InjectableInterfaceAgentService = createDecorator<I
18
19
20 gpt-3.5 · gpt-4.5
21
```

Apply Fix ⌘ D

Dismiss ⌘ D

Ban similar ⌘ D

Clarify in chat ⌘ E



Github copilot

1. Built in chat.
2. Writing tests work well.
3. Copilot is good with CSV and data.
4. Fast, smart code completions.
5. Copilot sometimes trips on explain.
6. Setup project from scratch is very difficult, better to use documentation than use Copilot.
7. Paid 10\$ (free for students).



Copilot		Start Code Chat Ctrl+I
Rename Symbol	F2	Explain This
Change All Occurrences	Ctrl+F2	Fix This
Format Document	Ctrl+Shift+I	Generate Docs
Format Document With...		Generate Tests

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
V2:
* 'orm_mode' has been renamed to 'from_attributes'
warnings.warn(message, UserWarning)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html

===== short test summary info =====
FAILED tests/test_main.py::test_authenticate_user_invalid_user - AttributeError: 'NoneType' object has no attribute 'dict'
FAILED tests/test_main.py::test_authenticate_user_invalid_password - sqlalchemy.exc.IntegrityError: (psycopg2.errors.UniqueViolation) duplicate key
value violates unique constraint "users_email_key"
===== 2 failed, 2 passed, 4 warnings in 1.26s =====
(venv) pyfal@pyfal-IdeaPad-5-14ARE05:~/Downloads/my_app$
```



Codeium

1. Similar to Github Copilot.
2. It doesn't work as well as Copilot.
3. But it's free.



Amazon Whisperer

1. Is able to generate code from TODO, but has problem with tests.
2. Very simple answers.
3. Writing from TODO does not always work.
4. Performance issue (vscode).
5. It doesn't work well.



Custom tools

Code Reviewer GPT

- a) build your own code reviewer
- b) use created CI/CD tool (f.e <https://github.com/mattzcarey/code-review-gpt>)



Security



Security - company perspective

Whoops, Samsung workers accidentally leaked trade secrets via ChatGPT

ChatGPT bug leaked users' conversation histories

🕒 23 March

Massive Leak Of ChatGPT Credentials: Over 100,000 Accounts Affected



Security - company perspective

<https://capeprivacy.com/>



Security - code perspective

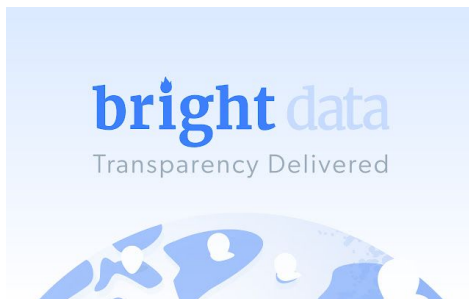
Table 7: The warning messages of the scan results by CodeQL, ESLint, and FindBugs manually matched with corresponding CWE IDs

Tool	Warning Message	CWE-ID
CodeQL	Reflected server-side cross-site scripting	CWE-79
CodeQL	Flask app is run in debug mode	CWE-215
CodeQL	Clear-text logging of sensitive information	CWE-532
CodeQL	Clear-text storage of sensitive information	CWE-312
CodeQL	Information exposure through an exception	CWE-209
CodeQL	Request without certificate validation	CWE-295
CodeQL	Assignment to constant	CWE-682
CodeQL	Log injection	CWE-117
CodeQL	Identical operands	CWE-570
CodeQL	Incomplete string escaping or encoding	CWE-176

part of publicly available projects hosted on GitHub. The goal is to investigate the types of security issues and their scale in real-world scenarios (rather than crafted scenarios). To this end, we identified 435 code snippets generated by GitHub Copilot from publicly available projects. We then conducted extensive security analysis to identify Common Weakness Enumeration (CWE) instances in these code snippets. The results show that (1) 35.8% of Copilot generated code snippets contain CWEs, and those issues are spread across multiple languages, (2) the security weaknesses are diverse and related to 42 different CWEs, in which *CWE-78: OS Command Injection*, *CWE-330: Use of Insufficiently Random Values*, and *CWE-703: Improper Check or Handling of Exceptional Conditions* occurred



More tools





Q&A