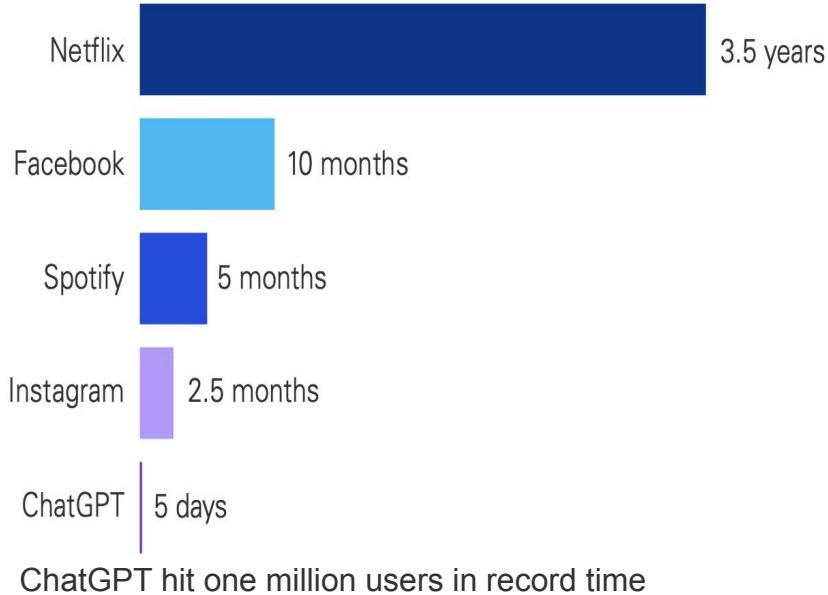


Coding with ChatGPT

Issac Lee

Why does it matter?



- ChatGPT is a tool that can assist you in coding, but it cannot completely replace the creativity, problem-solving skills, and critical thinking abilities of a human software engineer.
- Currently ChatGPT requires human supervision and direction to operate effectively

Demo

- Generate code from scratch.
- Instrument your code for monitoring.
- Document your code.
- Write unit tests.
- Create a Dockerfile

Follow me requirements

- A ChatGPT account
- Basic knowledge of Python programming language
- [Visual Studio Code](#) IDE
- Local development tools
 - [Pyenv](#)
 - [Jupyter](#)
 - [Postman](#)

How to ask ChatGPT for coding?

- Provide context
 - Pretend you are/Act as ...
- Express your intent
 - Create/Write/Make/Generate/Tell/Give/Show/Explain/Discover/Change/Optimize ...

Example prompts

- Pretend you are a software engineer and want to write a RESTful API service using FastAPI.
- Tell me about FastAPI. Do you recommend using it as a framework for a project?
- Please write me an example using FastAPI
- Where can I find the development guide and documentation for FastAPI?
- Explain what this OpenAI function does and its parameters:
`openai.Completion.create()`

Generate code from scratch

- Write me a FastAPI program using OpenAI SDK to perform text summarization. The program will get the OPENAI_API_KEY from the environment variable.

Write code snippets

- Write me the code snippet to enable the metrics collection using the Starlette Prometheus package in FastAPI

Write an unit test

- Write a Pytest for the following RESTFul API:

Write code documentation

- Add Python Docstrings to the function below:

```
@app.post("/summarize")
async def summarize_text(request: SummarizationRequest):
    response = openai.Completion.create(
        engine="text-davinci-002",
        prompt=f"Summarize '{request.text}' to {request.summary_length} sentences",
        temperature=0,
        max_tokens=60
    )

    summary = response.choices[0].text.strip()

    return {"summary": summary}
```

Create a Dockerfile

- Create a Dockerfile file for a FastAPI project with all packages dependency in the requirements.txt file. Expose port 8000

```
# Use an official Python runtime as a parent image
FROM python:3.8-slim-buster
WORKDIR /app
COPY requirements.txt /app
RUN pip install --trusted-host pypi.python.org -r requirements.txt
COPY . /app
EXPOSE 8000

CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8000"]
```

Build, Test and Run

```
pytest -s
```

```
uvicorn main:app --reload
```

```
docker build -t summarize .
```

```
docker run --rm -h localhost -e
```

```
OPENAI_API_KEY=$OPENAI_API_KEY --name summarize -p 8000:8000  
summarize
```

- <http://localhost:8000/docs>
- <http://localhost:8000/metrics>

Thank you!