Quick Answer: Should Software Engineering Teams Use ChatGPT to Generate Code?

Published 17 March 2023 - ID G00788409 - 4 min read

By Analyst(s): Van Baker, Arun Batchu

Initiatives: Software Engineering Technologies; Adopt Modern Architectures and Technologies; Evolve Technology and Process Capabilities to Support D&A

Interest in ChatGPT is high, especially for code generation. Software engineering leaders are faced with a decision on whether to allow software engineers to use ChatGPT to generate application code. The answer depends on your organization's current use of code-generating generative AI models.

More on This Topic

This is part of an in-depth collection of research. See the collection:

- ChatGPT Research Highlights

Quick Answer

Should the software engineering team use ChatGPT to generate code?

- Most enterprises should not use ChatGPT for code generation, with limited exceptions.
- The use of ChatGPT requires a number of trade-offs that most enterprises may find unacceptable.
- Other enterprise-friendly generative AI code generation tools, such as Codex, Copilot, CodeWhisperer or Project Wisdom, will be the better choice for almost all enterprises that are using or planning to use AI-enabled code generation tools.
- Inherent controls and data protection capabilities are key.

More Detail

GPT-3.5 (The Core Foundation Model)

OpenAI has created a lot of excitement in the market with the launch of ChatGPT. This generative AI service is based on a large language model (LLM) called GPT-3.5. The GPT-3.5 model incorporates:

- Training data that was used to build Codex
- Extensive reinforcement learning from human feedback (RLHF)

ChatGPT, the conversational application which uses the GPT model:

- Is a closed service that allows users to generate text and programming code in response to prompts.
- Has proven to be very proficient at generating code for the languages that are supported by the Codex model.
- Has received largely positive comments from developers that have used ChatGPT to generate code.
- Has generated a lot of interest in the capabilities of the application, resulting in interest and questions around the use of ChatGPT by software engineering teams to generate code. This has led to (unofficial) tools to integrate ChatGPT into developer environments being created with a large number of installs. [2]

Codex, a derivative model of GPT-3 specifically focused on code generation:

- Is a code generation model which is derivative of GPT-3 models.
- Is the model that powers the Copilot code generation service from Github.
- Has the same training data for application code that GPT-3.5 includes.
- Has regular data curation and prompt engineering.

Why Most Enterprises Shouldn't Use ChatGPT

There are a number of policies and limitations regarding the use of the ChatGPT application which should cause software engineering teams to reconsider using it to generate code:

- Developers implementing enterprise systems may need to provide intellectual property as a prompt to ChatGPT.
- Your prompts and the code generated may be included in future updates to OpenAI products.
- This may put you in breach of data privacy regulations.
- There is no ability for enterprises to opt out of this policy and retain the information that they enter.
- Any personally identifiable information will also be added to the data used to train the model. (OpenAI recommends against input of sensitive information, including enterprise intellectual property.)

Most enterprise software engineering teams that see the benefit of using generative AI-based code creation tools will be better off using either the Codex model or the Copilot service, which is based on the Codex model. The Codex model can be customized using one-shot or few-shot learning techniques and small datasets. These custom models are under the control of the enterprise and all of these services opt out of data retention for training of the Codex model.

We believe that improvements in the code generation capabilities evidenced in GPT-3.5 have been incorporated into the recently released next generation Codex model and thereby the Copilot service. Github has already announced that it is offering an updated Codex model in the Copilot service, which is likely equivalent to the capability shown by ChatGPT with this release.

Appropriate Exceptions

The only situation where it would be appropriate for enterprise software engineering teams to use ChatGPT is when they are just beginning to explore the use of generative AI-based code generation models for:

- Stub generation
- Prototyping
- Evaluation of English prompt tools versus the IDE plugin

Individuals can use the ChatGPT service for free if it is not too busy, and the company has launched a paid service for $20 per user per month that will give paid users priority. Software engineering teams that are just beginning to explore generative AI-based code generation can use the ChatGPT service as long as they:

- Avoid entering enterprise intellectual property into the prompt for the service.
- Avoid entry of any PII or other sensitive information in the model.
- Thoroughly examine the generated code prior to use.
- Go through the usual code review, testing and quality assurance processes.

View ChatGPT as a temporary solution. Prepare to switch to generative AI-based code generation integrated into your IDEs or other developer focused tools.

Recommended by the Authors

Tool: Enterprise Use Cases for ChatGPT

Quick Answer: How Can Enterprises Leverage ChatGPT and GPT3?

Gartner Addresses Frequently Asked Questions on ChatGPT

Evidence

[1] ChatGPT FAQ, OpenAi.

[2] ChatGPT Marketplace.

Communications from Microsoft regarding the addition of ChatGPT to the Azure OpenAI service.