

 100 XP

Improve generative AI responses with prompt engineering

3 minutes

The quality of responses that a generative AI application returns not only depends on the model itself, but on the types of prompts it's given. The term *prompt engineering* describes the process of prompt improvement. Both developers who design applications and consumers who use those applications can improve the quality of responses from generative AI by considering prompt engineering.

Prompts are ways we tell an application what we want it to do. An engineer can add instructions for the program with prompts. For example, developers may build a generative AI application for teachers to create multiple-choice questions related to text students read. During the development of the application, developers can add other rules for what the program should do with the prompts it receives.

System messages

Prompt engineering techniques include defining a system message. The message sets the context for the model by describing expectations and constraints, for example, "You're a helpful assistant that responds in a cheerful, friendly manner". These system messages determine constraints and styles for the model's responses.

Writing good prompts

You can get the most useful completions by being explicit about the kind of response you want, for example, "Create a list of 10 things to do in Edinburgh during August". You can achieve better results when you submit clear, specific prompts.

Providing examples

LLMs generally support *zero-shot learning* in which responses can be generated without prior examples. However, you can also provide *one-shot* learning prompts that include one, or a few,

examples of the output you require such as, "Visit the castle in the morning before the crowds arrive". The model can then generate further responses in the same style as the examples provided in the prompt.

Grounding data

Prompts can include *grounding* data to provide context. You can use grounding data as a prompt engineering technique to gain many of the benefits of fine-tuning without having to train a custom model.

To apply this technique, include contextual data in the prompt so that the model can use it to generate an appropriate output. For example, suppose you want to use an LLM to generate a summary of an email. You can include the email text in the prompt with an instruction to summarize it.

Next unit: Exercise - Explore generative AI with Bing Copilot

[Continue >](#)