



Fine-Tuning for LLMs: from Beginner to Advanced

Prompting examples and use cases

397 9,765 |

Prompting Examples and Use-Cases

Introduction

Prompt engineering is an essential skill for anyone looking to leverage large language models (LLMs) like ChatGPT. By crafting effective prompts, you can influence the model's output to suit your needs, whether for generating code, creating content, or solving problems. This article introduces some fundamental prompt patterns, providing examples and code snippets to help beginners master prompt engineering.

Few-Shot Pattern

The few-shot pattern involves providing the LLM with a few examples to guide its response. This pattern is useful when you want the model to follow a specific format or style.

Example:

Prompt:

Translate the following English sentences into French:

Sentence: I love learning new languages.

Translation: J'aime apprendre de nouvelles langues.

Sentence: Can you help me with my homework?

Translation: Peux-tu m'aider avec mes devoirs?

Sentence: The weather is beautiful today.

Translation: Il fait beau aujourd'hui

Sentence: You look wonderful tonight



Response:

By giving a few examples, the model learns the desired output format and continues accordingly.

Cognitive Verifier Pattern

The cognitive verifier pattern is extremely useful when you need a reliable approach to a subject and aren't sure if all aspects are being tackled. It increases the reliability of the outputs by checking the information needed before providing a final response.

Example:

Prompt:

Whenever I ask a question, only ask me for additional information to clarify what I'm asking before providing a final answer. Please combine all my responses.

How do I set up a CI/CD pipeline for a Node.js application?

Response:

To clarify, do you have a preferred cloud provider or CI/CD tool, and are there specific stages or tasks you want to include in the pipeline (e.g., testing, deployment)?

Once you provide additional information, ChatGPT can give a more detailed and accurate response.

Question Refinement Pattern

The question refinement pattern helps improve the quality of questions posed to the LLM, ensuring more precise and relevant answers. It allows users to refine vague queries, making them as clear and detailed as possible.

Example:

Prompt:

When I ask a question, suggest a better question and ask me if I would like to use it instead.

How can I improve my Python skills?

By refining the question, the response becomes more targeted and useful.

Gameplay Pattern

The gameplay pattern encourages curiosity and motivation by creating interactive, game-like experiences with AI. It involves challenges, strategy, or playful engagement to improve at a certain topic or refine a creation.

Example:

Prompt:

Create a game for me around Python programming. Give fundamental rules of the game.

This pattern makes learning interactive and enjoyable.

Conclusion

Prompt engineering is a powerful tool for customizing the interaction with LLMs like ChatGPT. By using patterns such as few-shot, cognitive verifier, question refinement, and gameplay, beginners can enhance their prompt engineering skills and achieve better results. These patterns provide a structured approach to crafting prompts,

ensuring more accurate and relevant responses from the model. As you experiment with these techniques, you'll discover new ways to leverage the full potential of LLMs in your projects and daily tasks.

[Previous](#)[Next](#)