What is Prompt Engineering?

Prompt engineering is the process of designing, refining, and optimizing prompts for artificial intelligence (AI) language models, such as GPT-3. Prompt engineering involves crafting prompts that provide clear and concise instructions for the model to generate the desired output.

The goal of prompt engineering is to improve the quality and efficiency of AI language models by guiding them towards more accurate and relevant responses. This involves carefully selecting and structuring the language and formatting of the prompts, as well as determining the most effective input-output pairs to train the model on.

Effective prompt engineering can lead to significant improvements in the performance of AI language models, enabling them to generate more human-like and accurate responses to a wide range of tasks and applications.

Types of Prompts: Zero Shot Prompts and One-Shot Prompts

Zero-shot prompts, and one-shot prompts are both techniques used in prompt engineering for training and fine-tuning AI language models, but they differ in their approach and application.

Zero-shot prompting, also known as zero-shot inference, involves providing instructions to an AI language model without providing any explicit training data or examples. This approach tests the model's ability to generalize and perform the task based on its understanding of the given instructions. If the model can handle the task, it can generate an output, but if it is not, the results may be inaccurate or irrelevant. Therefore, while zero-shot prompting allows for efficient and versatile use of AI language models, it is important to carefully design and test the prompts to ensure their effectiveness and accuracy.

Example: Asking a question to ChatGPT without context

On the other hand, one-shot prompts are a technique used to fine-tune AI language models for a specific task using a limited amount of input data, such as a single example or template. By providing the model with a small set of input-output pairs that demonstrate the desired behavior or response for the given task, the model can be trained to generate natural language text with a higher degree of accuracy and precision. One-shot prompting can be particularly useful in scenarios where training data is limited or when a specific task requires a high level of performance. However, the effectiveness of the fine-tuning process depends on the quality of the input data and the design of the prompts used to train the model.

Example: Providing context and/or Template as Input for better results in the output of the language model (ChatGPT)

In summary, zero-shot prompts are used to enable an AI language model to generate outputs for tasks it has not been specifically trained on, while one-shot prompts are used to fine-tune an AI language model for a specific task using a limited amount of training data.

The Importance of Prompt Quality in AI Language Models (ChatGPT)

A prompt is a critical component of AI language models, such as ChatGPT, as it serves as the input for generating a response. The quality and context of the prompt significantly impact the accuracy and relevance of the model's response. A high-quality prompt is specific and clearly defines the problem or question to be answered, including relevant keywords or phrases to help the model focus on the most important information. Crafting a well-crafted prompt is essential to achieve the best possible results from ChatGPT or any other language model. By providing clear and specific prompts, users can improve the accuracy and efficiency of AI language models, enabling them to generate more human-like and relevant responses to a wide range of tasks and applications.

How Can I Build Good Quality Prompts?

Using a Prompt Structure Template, we can create high-quality one-shot prompts.

Note: The only limit of this template is your imagination

Act as a [Expert] in the [Industry], I will give you [Relevant Context for the Action], and you will [Action]. [Optional Context for Action]. Do not start unless [Constraint].

- Once ChatGPT confirms that it understands the task at hand, enter as input the "specific information needed".
- The expected output of ChatGPT will be the "Action".

Prompt for Systems Analysts

Act as a [systems analyst] from a [pharmacy benefit management company], I will give you [requirements from a feature written by a product owner], and you will [analyze these requirements in order to develop a product backlog item containing a user story, reason for change, detailed technical description with the tasks in bullets, test scenarios and acceptance criteria]. Do not start unless [I give you the requirements].

- Once ChatGPT confirms that it understands the task at hand, enter as input the "Requirements from the Product Owner".
- The expected output of ChatGPT will be a "Product Backlog Item".

Prompt for Software Test Engineers

Act as a [software test engineer], I will give you [a product backlog item containing a user story, reason for change, detailed technical description with the tasks, test scenarios and acceptance criteria], and you will [develop detailed test cases with steps using the product backlog item]. Do not start unless [I give you the product backlog item].

- Once ChatGPT confirms that it understands the task at hand, enter as input the "Product Backlog Item".
- The expected output of ChatGPT will be "Detailed Test Cases".

Prompt for Software Developers P1

Act as a [software developer], I will give you [a technical description with tasks], and you will [analyze the technical description to provide me guidance on how to approach each of the steps described]. Do not start unless [I give you the technical description].

- Once ChatGPT confirms that it understands the task at hand, enter as input the "Technical Description".
- The expected output of ChatGPT will be "Guidance on how to approach each of the steps described".

Prompt for Software Developers P2

Act as a [software developer], I will give you [instructions of how to develop code], and you will [develop the code following the instructions]. Do not start unless [I give you the instructions].

- Once ChatGPT confirms that it understands the task at hand, enter as input the "Instructions".
- The expected output of ChatGPT will be "Code that was develop by following the instructions".

Prompt for Technical Writers

Act as a [technical writer], I will give you [information of a software system from a software developer], and you will [develop a technical specification document with a Business Purpose, Technical Overview, Scope, Validations to Consider, Dependencies and/or Impact, Risks, Assumptions, Security Considerations, and Error Handling]. Do not start unless [I give you the information of a software system]. The technical specification document must be accurate and complete, if the response generation stops unexpectedly, I will input the command "continue" and you will resume generating the response from where you left off.

- Once ChatGPT confirms that it understands the task at hand, enter as input the "information of a software system".
- The expected output of ChatGPT will be a "Technical Specification Document".