



Prompt Engineering

29 April 2023 16:37

Find out more in <https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/>



Two Types of large language models (LLMs)

Base LLM

Predicts next word, based on text training data

Once upon a time, there was a unicorn that lived in a magical forest with all her unicorn friends

What is the capital of France?
What is France's largest city?
What is France's population?
What is the currency of France?

Instruction Tuned LLM

Tries to follow instructions

Fine-tune on instructions and good attempts at following those instructions.

RLHF: Reinforcement Learning with Human Feedback

Helpful, Honest, Harmless

What is the capital of France?
The capital of France is Paris.

Principle 1: Write clear and specific instructions

- Use delimiters
 - o Triple quotes: """,
 - o Triple backticks: ```,
 - o Triple dashes: ---,
 - o Angle brackets: <>,
 - o XML tags: <tag> </tag>
- Avoid prompt injections:
Conflicting texts will confuse the AI.
- Ask for structured output:
 - o HTML
 - o JSON

Principle 2: Give the model time to think

- Specify the steps to complete a task
 - Step 1...
 - Step 2...
 - ...
 - Step N...
- Instruct the model to work out its own solution before rushing to a conclusion

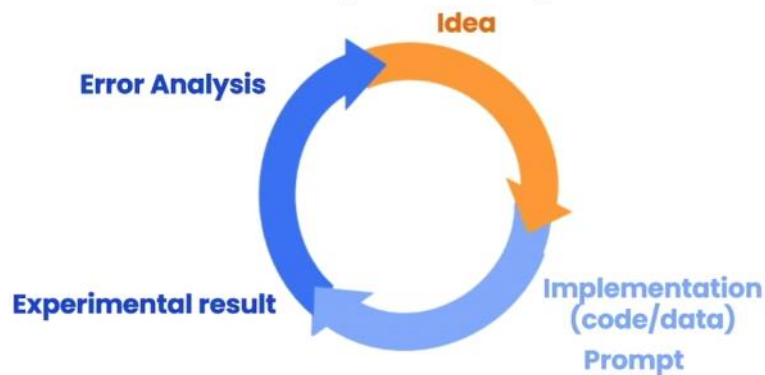
Model Limitations:

- Hallucination.
- Makes statements that sound plausible but are not true.
- Reducing Hallucinations:
 - o First find relevant information
 - o Then answer the question based on the relevant information.

DeepLearning.AI

OpenAI

Iterative Prompt Development



Iterative Process

- o Try something
- o Analyze where the result does not give what you want
- o Clarify instructions, give more time to think
- o Refine prompts with a batch of examples

Summarizing Large Texts

- With a word/sentence/character limit.
- With focus on shipping and delivery.
- With focus on price and value.
- Try "Extract" instead of "Summarize".
- Multiple product reviews using simple loops.
 - o Example:

```
reviews = [review_1, review_2, review_3, review_4]
```

```
for i in range(len(reviews)):
    prompt = f"""
    Your task is to generate a short summary of a product \
    review from an ecommerce site.

    Summarize the review below, delimited by triple \
    backticks in at most 20 words.

    Review: ```{reviews[i]}```
    """

    response = get_completion(prompt)
    print(i, response, "\n")
```

Identify Different Emotions

```
In [ ]: prompt = f"""
Identify a list of emotions that the writer of the \
following review is expressing. Include no more than \
five items in the list. Format your answer as a list of \
lower-case words separated by commas.

Review text: '{lamp_review}'
"""
response = get_completion(prompt)
print(response)
```

Identify anger

```
In [ ]: prompt = f"""
Is the writer of the following review expressing anger?\
The review is delimited with triple backticks. \
Give your answer as either yes or no.

Review text: '{lamp_review}'
"""
response = get_completion(prompt)
print(response)
```

Extract Product and Company Name from Customer Reviews

```
In [ ]: prompt = f"""
Identify the following items from the review text:
- Item purchased by reviewer
- Company that made the item

The review is delimited with triple backticks. \
Format your response as a JSON object with \
"Item" and "Brand" as the keys.
If the information isn't present, use "unknown" \
as the value.
Make your response as short as possible.

Review text: '{lamp_review}'
"""
response = get_completion(prompt)
print(response)
```