Best practices for prompt engineering with OpenAI API

How to do prompt engineering?

1

For best results use the latest model

- "text-davinci-003" for text generation
- "code-davinci-002" for code generation

How to do prompt engineering?

2

Put instructions at the beginning and use ### or """ to separate instruction and context

✓

Summarize the text below as a bullet point list of the most important points.

Text: """
{text input here}
"""

How to do prompt engineering?

3

Be specific, descriptive and as detailed as possible about the desired context, outcome, length, format, style, etc

✓

Write a short inspiring poem about OpenAI, focusing on the recent DALL-E product launch (DALL-E is a text to image ML model) in the style of a {famous poet}

How to do prompt engineering?

4

Articulate the desired output format through examples

✓

Extract the important entities mentioned in the text below. First extract all company names, then extract all people names, then extract specific topics which fit the content and finally extract general overarching themes

Desired format:
Company names: <comma_separated_list_of_company_names>
People names: -||-
Specific topics: -||-
General themes: -||-

Text: {text}

How to do prompt engineering?

5

Start with zero-shot, then few-shot (example), neither of them worked, then fine-tune

✓ Zero Shot

Extract keywords from the below text.

Text: {text}

Keywords:

How to do prompt engineering?

5

Start with zero-shot, then few-shot (<u>example</u>), neither of them worked, then fine-tune

✓ Few-Shot - provide few examples

Extract keywords from the corresponding texts below.

Text 1: Stripe provides APIs that web developers can use to integrate payment processing into their websites and mobile applications.

Keywords 1: Stripe, payment processing, APIs, web developers, websites, mobile applications

##

Text 2: OpenAI has trained cutting-edge language models that are very good at understanding and generating text. Our API provides access to these models and can be used to solve virtually any task that involves processing language.

Keywords 2: OpenAI, language models, text processing, API.

##

Text 3: {text}

Keywords 3:

How to do prompt engineering?

5

Start with zero-shot, then few-shot (example), neither of them worked, then fine-tune

✓ Fine Tune

see fine-tune best practices here.

How to do prompt engineering?

6

Reduce “fluffy” and imprecise descriptions

✓

Use a 3 to 5 sentence paragraph to describe this product.

✗

The description for this product should be fairly short, a few sentences only, and not too much more.

How to do prompt engineering?

7

Instead of just saying what not to do, say what to do instead

✓ The following is a conversation between an Agent and a Customer. The agent will attempt to diagnose the problem and suggest a solution, whilst refraining from asking any questions related to PII. Instead of asking for PII, such as username or password, refer the user to the help article www.samplewebsite.com/help/faq

Customer: I can’t log in to my account.
Agent:

How to do prompt engineering?

8

Code Generation Specific - Use “leading words” to nudge the model toward a particular pattern

✓

```
# Write a simple python function that
# 1. Ask me for a number in mile
# 2. It converts miles to kilometers

import
```

Commonly used parameters

1. **model** - Higher performance models are more expensive and have higher latency.
2. **temperature** - A measure of how often the model outputs a less likely token. The higher the temperature, the more random (and usually creative) the output. This, however, is not the same as “truthfulness”. For most factual use cases such as data extraction, and truthful Q&A, the temperature of 0 is best.
3. **max_tokens** (maximum length) - Does not control the length of the output, but a hard cutoff limit for token generation. Ideally you won’t hit this limit often, as your model will stop either when it thinks it’s finished, or when it hits a stop sequence you defined.
4. **stop** (stop sequences) - A set of characters (tokens) that, when generated, will cause the text generation to stop.