

01 Prompting Techniques - KirkYagami

1. Zero-shot Prompting

Large language models (LLMs) today, such as GPT-3.5 Turbo, GPT-4, and Claude 3, are tuned to follow instructions and are trained on large amounts of data. Large-scale training makes these models capable of performing some tasks in a "zero-shot" manner. Zero-shot prompting means that the prompt used to interact with the model won't contain examples or demonstrations. The zero-shot prompt directly instructs the model to perform a task without any additional examples to steer it.

Prompt:

```
Classify the text into neutral, negative or positive.
```

```
Text: I think the vacation is okay.
```

```
Sentiment:
```

Output:

```
Neutral
```

Note that in the prompt above we didn't provide the model with any examples of text alongside their classifications, the LLM already understands "sentiment" -- that's the zero-shot capabilities at work.

FINETUNED LANGUAGE MODELS ARE ZERO-SHOT LEARNERS

Instruction tuning has been shown to improve zero-shot learning. Instruction tuning is essentially the concept of finetuning models on datasets described via instructions. Furthermore, [RLHF](#) (reinforcement learning from human feedback) has been adopted to scale instruction tuning wherein the model is aligned to better fit human preferences.

2. Few-shot Prompting

Suggested Reading: <https://www.datacamp.com/tutorial/few-shot-prompting>

While large-language models demonstrate remarkable zero-shot capabilities, they still fall short on more complex tasks when using the zero-shot setting. Few-shot prompting can be used as a technique to enable in-context learning where we provide demonstrations in the prompt to steer the model to better performance. The demonstrations serve as conditioning for subsequent examples where we would like the model to generate a response.

1. Prompt for Sentiment Analysis:

YAML

Input: "The new restaurant has excellent service, and the food was absolutely delicious!"

Sentiment: Positive

Input: "The movie was boring and way too long. I wouldn't recommend it to anyone."

Sentiment: Negative

Input: "The product arrived on time, but it didn't meet my expectations. It was just okay."

Sentiment: Neutral

Now analyze the sentiment of the following sentence:

Input: "The vacation spot was beautiful, but the hotel service was terrible."

Sentiment:

Output:

YAML

Sentiment: Mixed

This sentence expresses both positive ("The vacation spot was beautiful") and negative ("the hotel service was terrible") sentiments, so the overall sentiment is mixed.

2. **Prompt:**

YAML

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:

We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:

Output:

YAML

When we won the game, we all started to farduddle in celebration.

We can observe that the model has somehow learned how to perform the task by providing it with just one example (i.e., 1-shot). For more difficult tasks, we can experiment with increasing the demonstrations (e.g., 3-shot, 5-shot, 10-shot, etc.).

How Few-Shot Prompting Works

The core of few-shot prompting lies in providing the LLM with a small set of relevant examples or demonstrations within the prompt itself. These examples guide the model, illustrating how to approach and respond to a particular type of task or question. Demonstrations are typically structured as follows:

- Input-output pairs: Each demonstration usually consists of an input (e.g., a question or a piece of text) and its corresponding output (the desired response or solution).
- Format consistency: The demonstrations maintain a consistent format, which helps the model recognize the pattern it should follow.
- Task relevance: The examples provided are directly relevant to the task at hand, showcasing the specific skill or knowledge required.

For instance, if we wanted the model to generate rhyming couplets, a few-shot prompt might look like this:

Generate a rhyming couplet about a tree:

Input: "cat"

Output: The curious cat, so sleek and fat, Curled up cozy on the welcome mat.

Input: "sky"

Output: Look up high into the endless sky, Where birds and clouds go drifting by.

Input: "tree"

Output:

When presented with these demonstrations, the language model engages in a process often called "in-context learning" or "learning by example." It works like this:

1. Pattern recognition: The model analyzes the provided examples, identifying patterns in how inputs are transformed into outputs.
2. Task inference: From these patterns, the model infers the nature of the task it's being asked to perform.
3. Generalization: The model then attempts to generalize from the given examples to new, unseen inputs.
4. Application: Finally, the model applies this learned pattern to the new input provided at the end of the prompt.

3. Chain-of-Thought (CoT) Prompting

Suggested Reading: <https://www.prompthub.us/blog/chain-of-thought-prompting-guide>

Chain of Thought prompting is a prompt engineering method that enhances the reasoning capabilities of large language models (LLMs) by encouraging them to break down their reasoning into a series of intermediate steps.

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Introduced in [Wei et al. \(2022\)](#), chain-of-thought (CoT) prompting enables complex reasoning capabilities through intermediate reasoning steps. You can combine it with few-shot prompting to get better results on more complex tasks that require reasoning before responding.

Prompt:

YAML

The odd numbers in this group add up to an even number: 4, 8, 9, 15, 12, 2, 1.

A: Adding all the odd numbers (9, 15, 1) gives 25. The answer is False.

The odd numbers in this group add up to an even number: 17, 10, 19, 4, 8, 12, 24.

A: Adding all the odd numbers (17, 19) gives 36. The answer is True.

The odd numbers in this group add up to an even number: 16, 11, 14, 4, 8, 13, 24.

A: Adding all the odd numbers (11, 13) gives 24. The answer is True.

The odd numbers in this group add up to an even number: 17, 9, 10, 12, 13, 4, 2.

A: Adding all the odd numbers (17, 9, 13) gives 39. The answer is False.

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.

A:

Output:

YAML

Adding all the odd numbers (15, 5, 13, 7, 1) gives 41. The answer is False.

Wow! We can see a perfect result when we provided the reasoning step. In fact, we can solve this task by providing even fewer examples, i.e., just one example seems enough:

<https://www.promptingguide.ai/techniques/cot>