CWB-Module-3

Prompting, Variants in Prompting,
Chain of Thought Prompting, Prompt Templates,
Integrate LLM

Prompt

A prompt for a language model is a **set of instructions** or input provided by a user to guide the model's response, helping it understand the context and generate relevant and coherent language-based output, such as answering questions, completing sentences, or engaging in a conversation.

Variants in Prompting

- **Zero shot prompting -** no examples are shown to the model.

- **One shot prompting -** 1 example is shown to the model.

- **Few shot prompting** - 2+ examples are shown to the model.

Few Shot Prompting

Example 1 : Great Product → Positive
Example 2 : Didn't work well → negative

Your Input : My mobile is not working properly.
Model Output : Negative

Chain of Thought Prompting

It is a technique that guides LLMs to follow a reasoning process when dealing with hard problems.

This is done by showing the model a few examples where the step-by-step reasoning is clearly laid out.

Standard Prompting vs Chain of Thought Prompting

Standard Prompting

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

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. ✔

Prompt Template

Prompt templates are predefined recipes for generating prompts for language models.

A template may include instructions, few-shot examples, and specific context and questions appropriate for a given task.

```python
from langchain_core.prompts import ChatPromptTemplate

chat_template = ChatPromptTemplate.from_messages(
    [
        ("system", "You are a helpful AI bot. Your name is {name}."),
        ("human", "Hello, how are you doing?"),
        ("ai", "I'm doing well, thanks!"),
        ("human", "{user_input}"),
    ]
)

messages = chat_template.format_messages(name="Bob", user_input="What is your name
```

Today's Task