Task 1: Advanced Prompt Engineering Techniques

WHAT IS PROMPT ENGINEERING?

- ❑ Prompt engineering is the process of creating effective prompts that enable AI models to generate responses based on given inputs.

- ❑ Prompt engineering essentially means writing prompts intelligently for text-based Artificial Intelligence tasks, more specifically, Natural Language Processing (NLP) tasks.

- ❑ In the case of such text-based tasks, these prompts help the user and the model generate a particular output as per the requirement.

WHAT ARE PROMPTS?

Prompts are short pieces of text that are used to provide context and guidance to machine learning models. When talking about the specific text AI tasks, also called NLP tasks, these prompts are useful in generating relevant outputs which are as close to the expected output itself. Precisely, these prompts help in generating accurate responses by:

- ➢ Adding on some additional guidance for the model.
- ➢ Not generalizing a prompt too much.
- ➢ Make sure the information added is not too much as that can confuse the model.
- ➢ Making the user intent and purpose clear for the model to generate content in the relevant context only.

ADVANCED TECHNIQUES

❑ 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.

We tried a few zero-shot examples in the previous section. Here is one of the examples (ie., text classification) we used:

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.

- When zero-shot doesn't work, it's recommended to provide demonstrations or examples in the prompt which leads to few-shot prompting. In the next section, we demonstrate few-shot prompting.

APPLICATIONS:

- **Simple Classification:** Sorting text into categories like spam/not spam or positive/negative sentiment.
- **Fact-based Q&A:** Answering straightforward questions that don't require complex reasoning.
- **Basic Text Generation:** Creating a short paragraph or summary on a well-known topic.

STRENGTH & LIMITATION

- **Strength:** Fast and flexible.
- **Limitation:** May lack accuracy on complex tasks.

❑ 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.

In the example, the task is to correctly use a new word in a sentence.

➢ **Prompt**

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**

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

APPLICATIONS:

- ❑ **Text Summarization:** Providing examples of long text and their concise summaries to guide the model's summarization style.
- ❑ **Content Creation:** Generating creative text in a specific style or format by showing a few examples of that style.
- ❑ **Code Generation:** Giving examples of a problem and its corresponding code solution to help the model generate new code snippets.

IMPACT

- ➢ Boosts accuracy significantly compared to zero-shot, especially for nuanced tasks.

❑ Chain-of-Thought Prompting

- ❑ 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.

EXAMPLE:

Prompt: "The user bought 3 bags of apples. Each bag had 5 apples. The user ate 2 apples. How many apples are left?"

Standard Model Response: "13" (without explanation).

CoT Prompt: "The user bought 3 bags of apples. Each bag had 5 apples. The user ate 2 apples. How many apples are left? Let's think step by step."

CoT Model Response: "First, calculate the total number of apples: 3 bags * 5 apples/bag = 15 apples. Then, subtract the apples eaten: 15 - 2 = 13.

The user has 13 apples left."

Applications:

- ❑ **Mathematical Problems:** Solving multi-step arithmetic, algebraic equations, or word problems.
- ❑ **Common Sense Reasoning:** Answering questions that require a logical deduction based on everyday knowledge.
- ❑ **Logical Puzzles:** Solving riddles or puzzles by breaking down the constraints and steps.
- ❑ **Debugging Code:** Asking the model to explain its thought process as it identifies and fixes errors in a code snippet.

- ➢ **Case study:** PaLM model’s GSM8K benchmark accuracy jumped from 17.9% to 58.1% with CoT.

REAL-WORLD IMPACT APPLICATIONS ACROSS INDUSTRIES

❑ **Customer Support**

Few-shot prompts tailor AI to brand tone and FAQs, enhancing customer interactions.

❑ **Education**

CoT prompting helps AI tutor students through problem-solving steps, fostering deeper understanding.

❑ **Finance**

Zero-shot CoT aids in complex report analysis and decision explanations, improving financial insights.

❑ **Research**

Enables AI to generate structured, logical scientific hypotheses, accelerating discovery.

CONCLUSION: MASTERING PROMPT ENGINEERING FOR AI EXCELLENCE

❑ **Unlock AI Potential**

Advanced prompting unlocks AI’s reasoning and problem-solving potential.

❑ **Foundational Techniques**

Zero-shot, few-shot, and chain-of-thought are foundational techniques.

❑ **Drive Superior Applications**

Combining these methods drives superior, trustworthy AI applications.

❑ **Self-Consistency:**
Aggregates multiple outputs to improve reliability.

Start experimenting today to harness AI’s full capabilities and gain competitive edge!