

Chapter 6. Prompt Engineering

These advanced components can quickly make a prompt quite complex. Some common components are:

Persona

Describe what role the LLM should take on. For example, use "You are an expert in astrophysics" if you want to ask a question about astrophysics.

Instruction

The task itself. Make sure this is as specific as possible. We do not want to leave much room for interpretation.

Context

Additional information describing the context of the problem or task. It answers questions like "What is the reason for the instruction?"

Format

The format the LLM should use to output the generated text. Without it, the LLM will come up with a format itself, which is troublesome in automated systems.

Audience

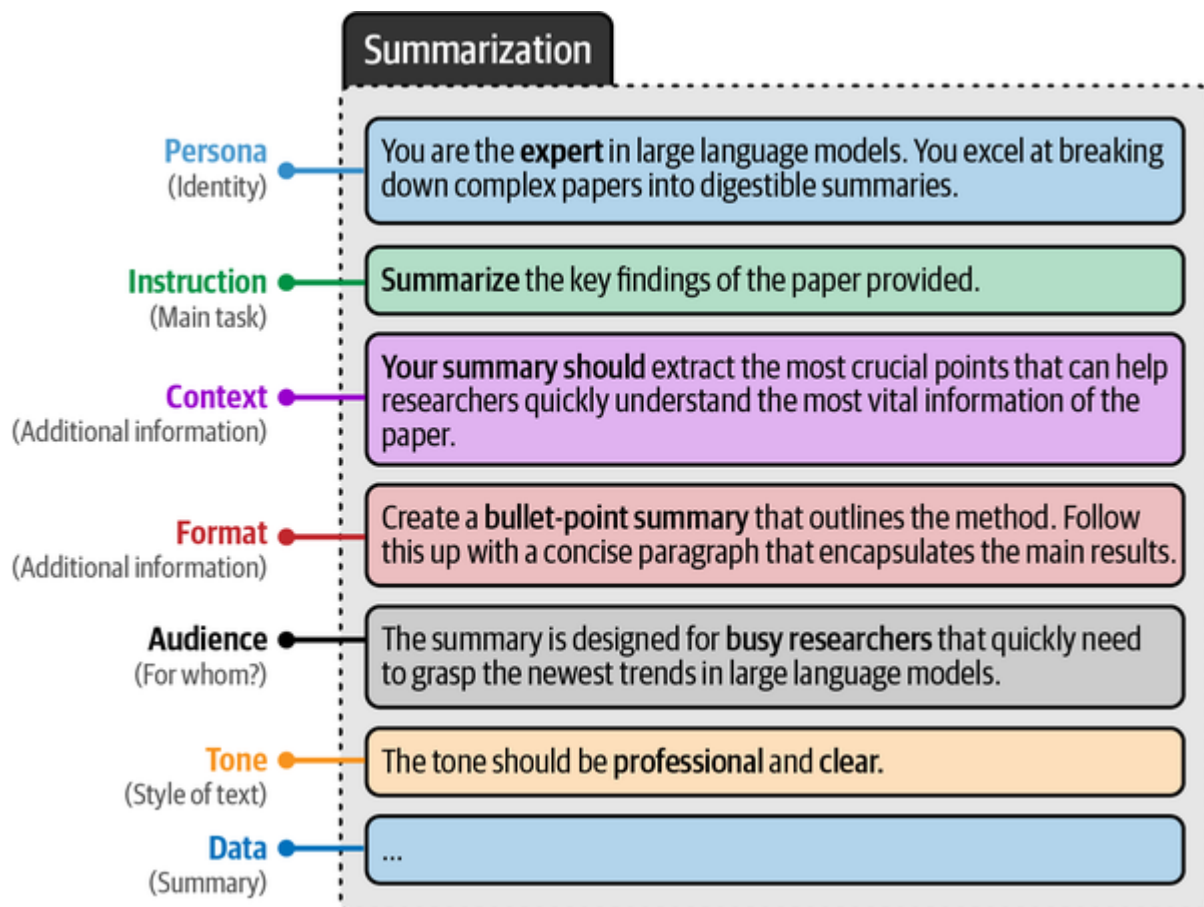
The target of the generated text. This also describes the level of the generated output. For education purposes, it is often helpful to use ELI5 ("Explain it like I'm 5").

Tone

The tone of voice the LLM should use in the generated text. If you are writing a formal email to your boss, you might not want to use an informal tone of voice.

Data

The main data related to the task itself.



Prompt components

persona = "You are an expert in Large Language models. You excel at breaking down complex papers into digestible summaries.\n"

instruction = "Summarize the key findings of the paper provided.\n"

context = "Your summary should extract the most crucial points that can help researchers quickly understand the most vital information of the paper.\n"

data_format = "Create a bullet-point summary that outlines the method. Follow this up with a concise paragraph that encapsulates the main results.\n"

audience = "The summary is designed for busy researchers that quickly need to grasp the newest trends in Large Language Models.\n"

tone = "The tone should be professional and clear.\n"

text = "MY TEXT TO SUMMARIZE"

data = f"Text to summarize: {text}"

#The full prompt - remove and add pieces to view its impact on the generated output

query = persona + instruction + context + data_format + audience + tone + data

Tree-of-Thought: Exploring Intermediate Steps

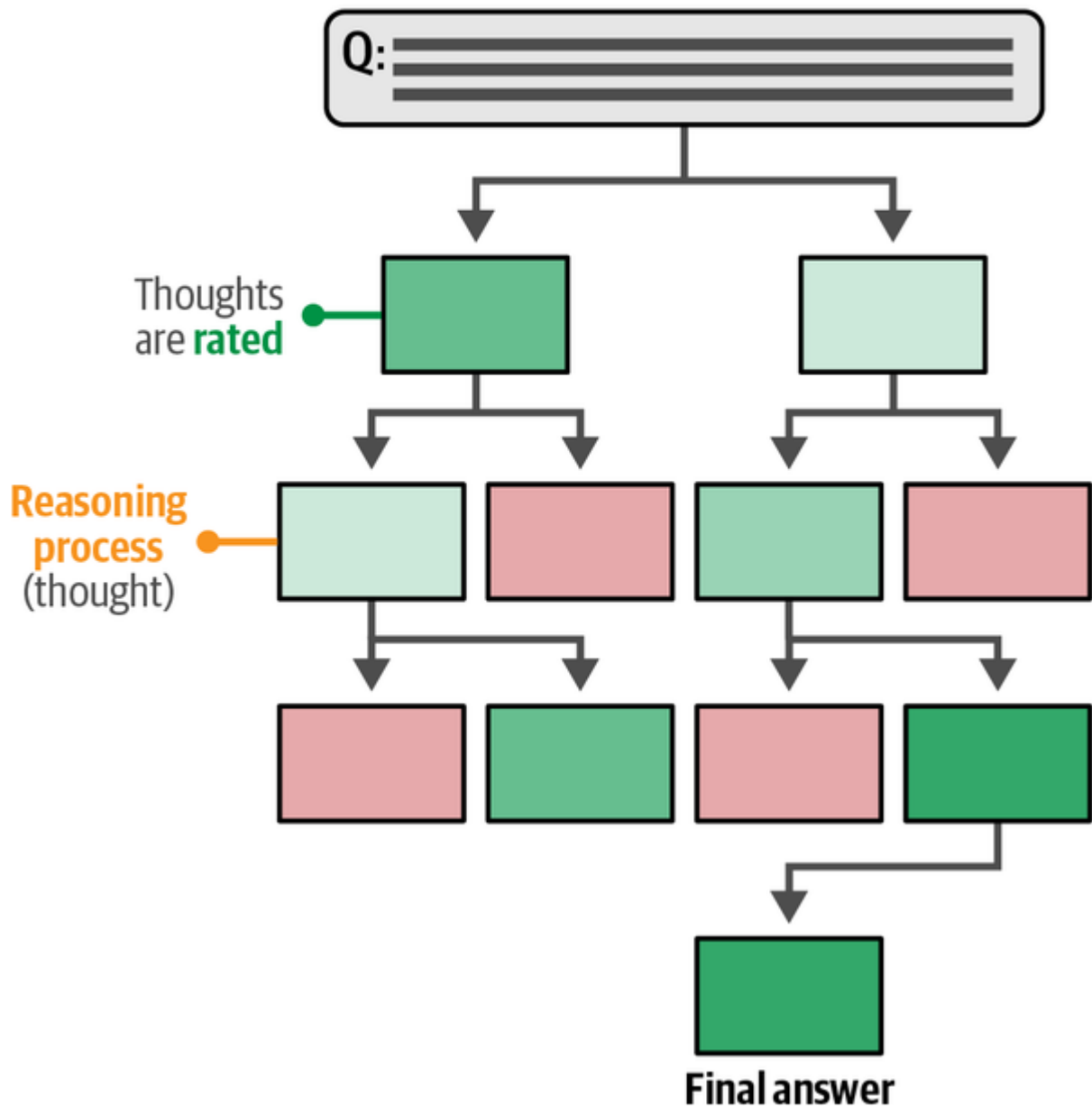
The ideas of chain-of-thought and self-consistency are meant to enable more complex reasoning. By sampling from multiple “thoughts” and making them more thoughtful, we aim to improve the output of generative models.

These techniques only scratch the surface of what is currently being done to mimic complex reasoning. An improvement to these approaches can be found in tree-of-thought, which allows for an in-depth exploration of several ideas.

The method works as follows. When faced with a problem that requires multiple reasoning steps, it often helps to break it down into pieces. At each step, and as illustrated in Figure 6-18, the generative model is prompted to explore different solutions to the problem at hand. It then votes for the best solution and continues to the next step.

Tree-of-thought

Exploring multiple paths



This method is tremendously helpful when needing to consider multiple paths, like when writing a story or coming up with creative ideas. A disadvantage of this method is that it requires many calls to the generative models, which slows the application significantly. Fortunately, there has been a successful attempt to convert the tree-of-thought framework into a simple prompting technique.¹¹

Instead of calling the generative model multiple times, we ask the model to mimic that behavior by emulating a conversation between multiple experts. These experts will question each other until they reach a consensus. An example of a tree-of-thought prompt is:

```
# Zero-shot tree-of-thought
zeroshot_tot_prompt=[{"role":"user","content":"Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realizes they're wrong at any point then they leave. The question is 'The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?' Make sure to discuss the results."}]
```

Expert 1: Step 1 - Start with the initial number of apples: 23 apples.

Expert 2: Step 1 - Subtract the apples used for lunch: $23 - 20 = 3$ apples remaining.

Expert 3: Step 1 - Add the newly bought apples: $3 + 6 = 9$ apples.

Expert 1: Step 2 - Confirm the final count: The cafeteria has 9 apples.

Expert 2: Step 2 - Review the calculations: $23 - 20 = 3$, then $3 + 6 = 9$. The calculations are correct.

Expert 3: Step 2 - Agree with the result: The cafeteria indeed has 9 apples.

All experts agree on the final count: The cafeteria has 9 apples.