

FBS- 3105: Prompts development reduce hallucination in response of LLMs.

Problem statement:

1. The response to be human-like, not the list, so the prompt should provide explanations and use the list as a guide.
2. We need to identify if things on the list are not what we have at hand so we don't mention them.

Prompts

- Raw Prompt : Zero shot
- Prompt 0: Memory
- Prompt 1: Memory + Context injection
- Prompt 1.1 : Memory + Context + Asking similarity
- Prompt 2: Memory + Context injection + Zero shot
- Prompt 3: Memory + knowledge base + chain of thoughts
- Prompt 4: Memory + knowledge base + chain of thoughts + Chaining of response

Chain of thoughts

Chain of Thought Prompting, instead of merely providing questions and answers, prompts the model to produce intermediate reasoning steps before delivering the final answer to a multi-step problem.

Outlines the process of implementing Chain of Thought Prompting effectively:

1. **Define the problem:** Start by clearly identifying the problem you want the language model to solve. The problem should be a multi-step reasoning task that requires the model to think through a series of intermediate steps before arriving at the final answer.
2. **Break down the problem:** Decompose the problem into smaller, more manageable steps. Identify the key components or intermediate steps necessary for solving the problem. Each step should represent a logical progression in the overall reasoning process. Each step should be
3. **Develop a chain of thought:** Create a structured sequence of prompts that correspond to the intermediate steps identified in the previous step. Each prompt should be phrased in a clear and concise manner to guide the language model through the thought process. Ensure that the prompts maintain a logical flow, moving from one step to the next.
4. **Prepare examples:** For few-shot prompting, prepare a set of examples that demonstrate the desired chain of thought for similar problems. These examples should showcase the intermediate steps and the final solution, providing the language model with a template to follow when solving the given problem. With most modern LLMs the number of examples necessary tends to be much smaller than those shown in the origin papers. Again, experiment with each LLM you work with to find that sweet spot.
5. **Incorporate the prompts into the model:** Input the chain of thought prompts and examples into the language model, either by concatenating them as part of the input text or by using any other method supported by the specific model you're working with. Ensure that the prompts are formatted consistently and accurately represent the intended intermediate steps.
6. **Test the model:** Evaluate the model's performance on the problem using the Chain of Thought Prompting approach. Analyze the intermediate steps produced by the model to ensure it follows the intended thought process. Assess the accuracy and efficiency of the final solution provided by the model.
7. **Iterate and refine:** Based on the evaluation, refine the chain of thought prompts and examples to improve the model's performance. This might involve rephrasing prompts, adjusting the order of steps, or providing additional examples to guide the model more effectively. Repeat the testing and refinement process until the desired performance level is achieved.

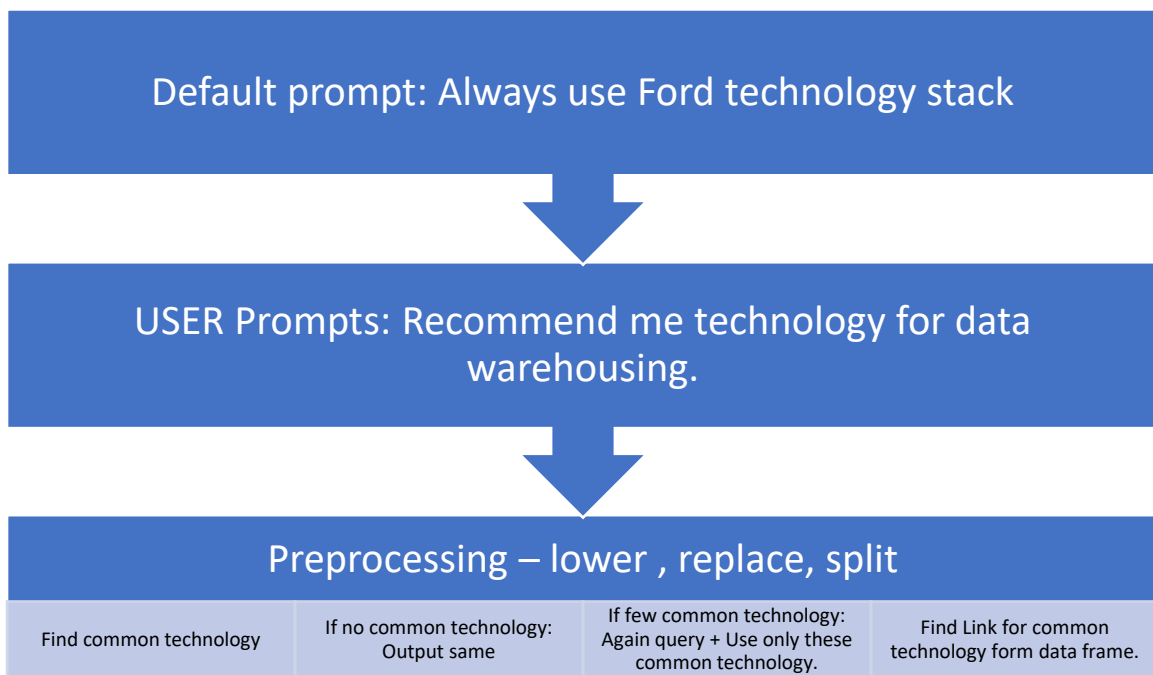
8. **Apply to new problems:** Once the Chain of Thought Prompting approach has been fine-tuned, apply the method to new problems within the same domain or related domains. The structured problem-solving approach can help the model tackle a wide range of multi-step reasoning tasks, enhancing its overall performance and versatility.

[\[Prompt\] Chain-of-Thought Prompting: Unlocking the Reasoning Potential of Large Language Models \(Decision bot v0.0.1\) \(linkedin.com\)](#)

Workflow

Filter

1. Default prompt: Always use Ford technology stack
2. USER Prompts: Recommend me technology for data warehousing.
3. Preprocessing – lower , replace, split ,
 - a. Find common technology
 - b. If no common technology: Output same
 - c. If few common technology: Again query + Use only these common technology.
 - d. Find Link for common technology form data frame.



User Interface

1. Default prompt: Always use Ford technology stack
2. USER Prompts:
 - a. Recommend me technology for data warehousing.
 - b. Description for project
 - c. Problem statement for project
3. if description: Again, query automatically
4. If problem: Again, query automatically
5. Pass USER Prompt to Function defined above.

Default prompt: Always use Ford technology stack



USER Prompts:

Recommend me technology for
data warehousing.

Description for project

Problem statement for project



if description: Again, query automatically



If problem: Again, query automatically



Pass USER Prompt to Function



Preprocessing – lower , replace, split ,

Find common technology

If no common technology:
Output same

If few common technology:
Again query + Use only these
common technology.

Find Link for common
technology form data
frame.