Name:Chintha Nagajyothi

Reg No:212221230015

EXP 2: Comparative Analysis of Naïve Prompting versus Basic Prompting Using ChatGPT Across Various Test Scenarios

Aim:

- To test how ChatGPT responds to naïve prompts (broad or unstructured) versus basic prompts (clearer and more refined) across multiple scenarios, analyzing the quality, accuracy, and depth of the generated responses.

Procedure:

1. **Define the Two Prompt Types:**

   Naïve Prompts: A prompt that may lack depth or context, often based on incomplete or overly simplistic assumptions. It might be too broad, unclear, or not provide enough information for a detailed or accurate response.

   Basic Prompts: A straightforward and clear instruction given to an AI or model that contains the essential details required to generate a response. It is usually simple, focusing on a specific question or task.

2. **Prepare Multiple Test Scenarios:**

   Scenario 1: Code Generation

   - **Naïve Prompt:** "write a python code for even or odd."

   - **Basic Prompt:** "write a python code for even or odd without using %."

   Scenario 2: Language Translation

   - **Naïve Prompt:** "Translate this sentence"je m`appelle nagajyothi"."

   - **Basic Prompt:** "Translate this sentence"j'mapalle nagajyothi" into hindi."

   Scenario 3: Email Writing

   - **Naïve Prompt:** "Write an email to professor for extension of assessment"

   - **Basic Prompt:** "Write a formal email to your professor requesting an extension on your assignment."

3. Responses from chat GPT:

Scenario 1:

Naïve Prompt:

```python
# Function to check if a number is even or odd
def check_even_odd(num):
    if num % 2 == 0:
        print(f"{num} is even.")
    else:
        print(f"{num} is odd.")
# Input from the user
number = int(input("Enter a number: "))
check_even_odd(number)
```

Basic Prompt:

```python
# Function to check if a number is even or odd without using %
def check_even_odd(num):
    if (num // 2) * 2 == num:
        print(f"{num} is even.")
    else:
        print(f"{num} is odd.")

# Input from the user
number = int(input("Enter a number: "))
check_even_odd(number)
```

Scenario 2:

Naïve Prompt:

"Je m'appelle Nagajyothi," which means **"My name is Nagajyothi."**

Basic Prompt:

The sentence "Je m'appelle Nagajyothi" translates to Hindi as:

"मेरा नाम नागज्योति है।"

Scenario 3:

Naïve Prompt:

Here's a sample email you can use to request an extension for your assessment:

Subject: Request for Extension on Assessment Submission

Dear Professor [Professor's Last Name],

I hope this email finds you well. I am writing to kindly request an extension for the submission of the [assessment name/assignment]. Due to [briefly explain the reason for the request – e.g., personal circumstances, health issues, workload, etc.], I am finding it challenging to meet the original deadline.

I understand the importance of timely submissions, and I assure you that I am working diligently on the assessment. However, an extension of [specific number of days or new deadline] would greatly help me in completing the work to the best of my ability.

I sincerely apologize for any inconvenience this may cause, and I appreciate your understanding and consideration of my request. Please let me know if there are any additional steps I need to take or if any further documentation is required.

Thank you for your time and support.

Best regards,

Basic Prompt:

Here's a formal email template for requesting an assignment extension:

Subject: Request for Assignment Extension

Dear Professor [Professor's Last Name],

I hope you are doing well. I am writing to request an extension for the submission of my assignment for [course name]. Due to [briefly explain the reason, such as personal circumstances, health issues, or workload], I am unable to complete the assignment by the given deadline.

I would be grateful if you could grant me an extension until [desired date] to ensure that I can submit the assignment at my best quality. I sincerely apologize for any inconvenience this may cause and appreciate your understanding.

Please let me know if this extension is possible or if any additional steps are required.

Thank you for your consideration.

Best regards,

Conclusion:

Generally, ChatGPT tends to produce higher-quality, more accurate, and more in-depth responses when given in basic prompts**.** This is because basic prompts provide the model with clearer instructions and context, guiding it towards a more specific and relevant output. while basic prompts generally lead to better results, naïve prompts can be effective in certain scenarios, especially when creativity, open-endedness, or testing model capabilities are desired. It's often beneficial to experiment with both prompt types to determine the best approach for a given task.