Does Prompt Design Affect LLM Outputs? A Study Across Structured and Vague Inputs

Aakanksha Bomble[1], Prashant Kulkarni[2], Vishnu Potdar[3]
[1]*Student, Symbiosis Skills and Professional University, Pune*
[2]*Research Guide, Symbiosis Skills and Professional University, Pune*
[3] *Research Co-Guide, Symbiosis Skills and Professional University, Pune*

***Abstract*—Prompt engineering is essential for improving the performance of large language models (LLMs) like GPT-3.5 and GPT-4. This study investigates how various prompt structures—such as differences in wording, tone, specificity, and formatting—affect LLM performance in key natural language processing tasks, including sentiment analysis, summarization, and question answering. Multiple prompt styles were evaluated for each task, ranging from direct to descriptive and formal to informal. Both human evaluations and quantitative metrics, such as accuracy and clarity scores, were used for assessment. The findings indicate that even slight modifications in prompt wording can greatly impact the clarity, accuracy, and completeness of the model's outputs. These results highlight the significance of prompt design in practical LLM applications and suggest that how prompts are formulated is a vital element in achieving the best outcomes.**

I. INTRODUCTION

Large Language Models (LLMs) like GPT-3.5 and GPT-4 have transformed the field of natural language processing by demonstrating impressive generalization abilities across various tasks, such as summarization, sentiment analysis, and question answering. These models generate responses based on prompts—textual instructions from users—making the way prompts are structured a vital element in determining the quality and relevance of the outputs.

While many users concentrate on the questions they wish to ask, few take into account how the wording, tone, and specificity of a prompt can affect the model's response. In fact, even minor variations in prompt structure can lead to considerable differences in the accuracy, clarity, or correctness of the results. The emerging discipline of prompt engineering aims to systematically design and refine prompts to better direct the behavior of LLMs.

Despite the remarkable abilities of LLMs, research on how different prompt structures—such as direct versus vague or specific versus generic phrasing—affect model performance across NLP tasks remains limited. Understanding this influence is essential not only for developers and researchers but also for end users who depend on LLMs in practical applications.

This study explores how changes in prompt structure impact LLM performance. Using GPT-3.5 as the foundational model, we examine three types of prompt styles (Direct, Specific, and Vague) across three commonly used NLP tasks: summarization, sentiment analysis, and question answering. The quality of the model's outputs is assessed through both quantitative metrics (accuracy, clarity scoring) and human evaluation.

The findings of this study provide valuable insights into how minor adjustments in prompt design can greatly affect output quality, highlighting the significance of prompt engineering as a crucial aspect of effectively utilizing LLMs.

II. LITERATURE REVIEW

large language models LLMs such as gpt-3 gpt-35 gpt-4 and open-source alternatives like llama and bloom have significantly progressed the natural language processing NLP domain these models can handle a variety of language-related tasks including summarization translation question answering and sentiment analysis with little to no specific training for each task their adaptability stems from their capability to comprehend instructions given in natural language prompts 21 prompt engineering prompt engineering is the process of crafting input prompts to achieve specific responses from LLMs unlike earlier NLP systems that depended on structured inputs and fixed

formats contemporary LLMs can interpret unstructured conversational instructions this evolution necessitates an understanding of how the wording tone and format of prompts can impact the outputs generated by the models researchers and practitioners have noted that even minor adjustments in phrasing can significantly influence the quality accuracy or tone of the resulting output 22 prompting paradigms zero-shot few-shot and chain-of-thought the concepts of zero-shot and few-shot prompting were introduced by brown et al 2020 in the original gpt-3 paper zero-shot prompting enables the model to perform a task based solely on a general instruction without any examples in contrast few-shot prompting provides 13 examples within the prompt to assist the model chain-of-thought cot prompting proposed by wei et al 2022 encourages the model to outline intermediate reasoning steps before arriving at a final answer particularly for complex tasks like mathematics or logic while these paradigms enhance performance the focus has primarily been on the information provided such as examples or reasoning rather than the specific wording of the instructions.

2.3 Significance of Prompt Structure

Recent research indicates that the structure of prompts—regardless of few-shot or Chain of Thought (CoT) methods—can greatly affect the output of models. For instance:

Reynolds & McDonell (2021) highlighted that the specificity, clarity, and formatting of prompts influence the quality of responses.

Mishra et al. (2022) discovered that minor adjustments, such as including "please" or rephrasing commands as questions, can alter the tone and confidence of the answers.

OpenAI's documentation suggests experimenting with different tones (polite vs. direct), varying sentence lengths, and using clear instructional keywords (e.g., “explain in steps” versus “what is…”).

These insights reinforce the notion that the way prompts are phrased is not merely superficial; it plays a crucial role in shaping how models behave.

2.4 Prompting Across NLP Tasks

Research indicates that the effectiveness of prompts can differ depending on the task:

For summarization tasks, structural cues like “summarize in 2 lines” are beneficial.

In sentiment analysis, clear instructions are essential, such as directly asking for “Positive, Negative, or Neutral” responses.

Question answering yields the best results when questions are straightforward and clear.

However, most current studies assess these prompts informally or within proprietary frameworks. There is a lack of open research that systematically compares various prompt styles—such as Direct, Specific, and Vague—across different tasks.

III. METHODOLOGY

This section describes the experimental design used to assess how different prompt structures affect the outputs of large language models. The methodology encompasses the selection of natural language processing (NLP) tasks, classification of prompt types, evaluation methods, and analytical tools.

3.1 Research Design

The study employs an experimental approach, testing the large language model (GPT-3.5) with three types of prompt structures—Direct, Specific, and Vague—across three separate NLP tasks:

- Summarization
- Sentiment Analysis
- Question Answering

Each task consists of 10 unique input samples. All three prompt types are applied to each sample, leading to a total of 90 outputs (3 prompt types × 10 inputs × 3 tasks). These outputs are assessed through a combination of human evaluation and basic quantitative measures.

3.2 Prompt Structure Categories

The experiment categorizes prompts into three main styles:

- Direct Prompts: Clear and straightforward instructions (e.g., “Summarize the above text.”)
- Specific Prompts: More detailed instructions with specific formatting or constraints (e.g., “Summarize the passage in 2 lines.”)
- Vague Prompts: Open-ended or conversational instructions (e.g., “What do you think about this?”). This classification is applied consistently across all tasks.

3.3 NLP Tasks and Inputs

- Summarization: Participants received paragraphs on topics such as technology, environment, or

society, with the aim of producing concise summaries.
- Sentiment Analysis: Each input consisted of a sentence or review that the model needed to classify by sentiment.
- Question Answering: Factual questions were asked, requiring the model to provide accurate and complete responses.

Each task included 10 input samples, resulting in a total of 30 unique input.

3.4 Evaluation Strategy

Each output generated by the language model was evaluated using two key criteria: correctness and clarity. Correctness refers to whether the response was factually accurate, logically valid, or matched the expected sentiment or answer for the given input. This was assessed using a simple Yes/No binary label.

Clarity was rated on a scale of 1 to 5, with 5 being extremely clear, concise, and relevant to the prompt, and 1 being vague, confusing, or off-topic. This scoring was done manually based on human judgment.

In addition to correctness and clarity, evaluators also recorded qualitative comments for each response, especially when the output was ambiguous, partially correct, or notably strong or weak. This helped in understanding not just whether the model got it "right", but how well it articulated the answer.

3.5 Tools Used

- Model Interface: ChatGPT (GPT-3.5) via OpenAI platform
- Data Recording: Microsoft Excel (for recording scores and outputs)
- Analysis & Visualization: Python (Jupyter Notebook) using pandas, matplotlib, and seaborn.

3.6 Summary

This methodology allows for a controlled comparison of how different prompt structures affect the output of a state-of-the-art language model across various tasks. The consistent format, task design, and evaluation make the results reliable and interpretable.

IV. EXPERIMENTAL SETUP

This chapter explains how the practical phase of the study was carried out, including the model used, configuration settings, and the process of output generation and recording.

4.1 Language Model Used

The experiment was conducted using ChatGPT powered by OpenAI's GPT-3.5. This model is known for its strong generalization across diverse natural language tasks. All prompts were run through the standard web-based interface without any fine-tuning or API customizations. The temperature and other hyperparameters were left at their default values to ensure consistency and reproducibility across sessions.

4.2 Prompt Types

The three prompt styles used in this study—Direct, Specific, and Vague—were previously described in Section 3.2. These styles were consistently applied across all input samples to test their effect on output quality. Each prompt was entered manually for each input task.

4.3 Input Samples

Each task—summarization, sentiment analysis, and question answering—included 10 distinct input samples. These were selected to reflect realistic scenarios and language diversity. A total of 90 outputs were generated (3 prompt styles × 10 inputs × 3 tasks). The input samples have been described in detail in Section 3.3.

4.4 Output Collection Process

Each prompt-input pair was submitted manually through ChatGPT (GPT-3.5). The corresponding outputs were then recorded in an Excel spreadsheet immediately. Each entry was tagged with the task type, prompt style, and input number to maintain traceability. Care was taken to avoid repetition or caching effects by using a fresh session or clearing context between prompts. The outputs were later reviewed and scored based on correctness, clarity, and qualitative feedback as outlined in Section 3.4.

V. RESULTS AND DISCUSSION

This study investigates how the structure of prompts—Direct, Specific, and Vague—affects the quality of outputs generated by a large language model (GPT-3.5) across three NLP tasks: summarization, sentiment analysis, and question answering. The results are analysed using two main metrics: accuracy and clarity, both evaluated manually.

5.1 Clarity Score Comparison by Prompt Type

The clarity of outputs was rated on a scale of 1 to 5. The bar chart below illustrates that Direct and Specific prompts received the highest average clarity scores (5.0), while Vague prompts scored significantly lower (≈2.4).

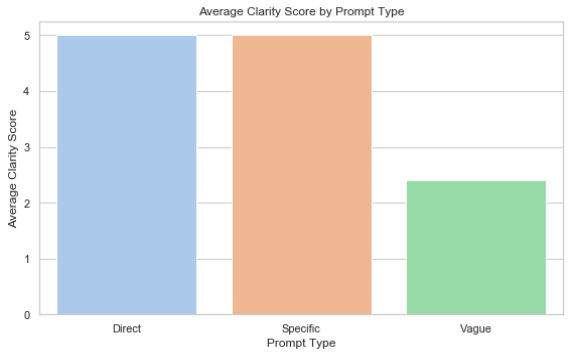

Figure 5.1: *Average Clarity Score by Prompt Type*

This confirms that prompts which provide clear, concise, or well-structured instructions help the model produce more readable and relevant responses. Vague prompts often led to generic or off-topic outputs, reducing their clarity significantly.

5.2 Prompt Structure and Accuracy

The chart below shows the percentage of correct responses by prompt type across all tasks. Direct and Specific prompts each contributed to 41.7% of all correct answers, while Vague prompts accounted for only 16.7%.

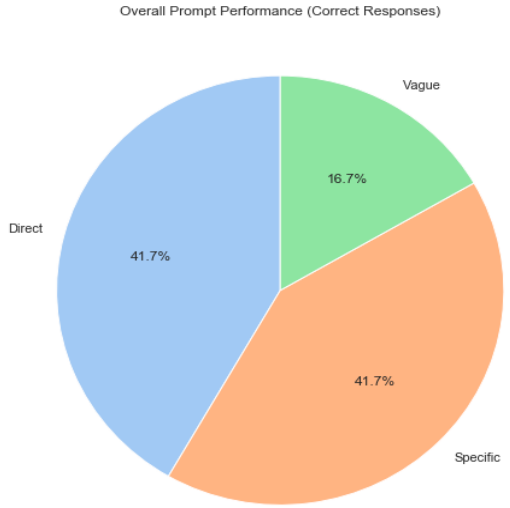

Figure 5.2: *Overall Prompt Performance (Correct Responses)*

This suggests that when instructions are clearly articulated—either through directness or added specificity—the model is more likely to interpret the task accurately and produce a correct response.

5.3 Heatmap of Task Vs Prompt Clarity

The following heatmap visualizes average clarity scores across each task and prompt type. It clearly shows that Direct and Specific prompts maintain perfect clarity (score of 5.0) across all tasks. Vague prompts, however, performed inconsistently:

- Summarization: 2.0
- Sentiment: 2.5
- QA: 2.7

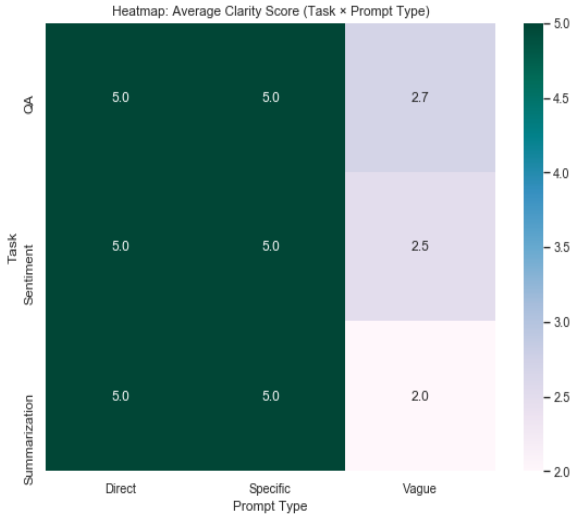

Figure 5.3: *Heatmap: Average Clarity Score (Task × Prompt Type)*

This again highlights that vague phrasing confuses the model, especially for summarization tasks, where the lack of structure leads to incomplete or generic outputs.

5.4 Interpretation of Results

The combined analysis clearly indicates that prompt structure has a substantial influence on both accuracy and clarity of LLM outputs. Direct and specific instructions consistently led to better results across all tasks, with negligible differences between the two. Vague prompts, while sometimes sounding more conversational, resulted in lower performance and should be avoided in high-stakes or precision-driven applications.

V. CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

This study explored the impact of prompt structure—specifically Direct, Specific, and Vague styles—on the output quality of a Large Language Model (GPT-3.5) across three widely used NLP tasks: summarization, sentiment analysis, and question answering.

Through a structured evaluation using 90 generated outputs, it was observed that:

- Direct and Specific prompts consistently yielded higher clarity and accuracy scores across all tasks.
- Vague prompts, while more conversational in tone, often resulted in incomplete, off-topic, or less useful responses.
- Task performance varied slightly, with question answering achieving the highest overall clarity.

These results confirm that prompt phrasing is a critical factor in guiding LLM behaviour, and that well-structured prompts significantly improve model performance, even without additional training or tuning.

6.2 Future Work

While the current study was limited to GPT-3.5 and three prompt types, future research can build on this foundation by:

- Expanding to more complex NLP tasks such as dialogue generation, code generation, or reasoning-based problem solving.
- Testing across multiple LLMs (e.g., Claude, Gemini, LLaMA) to observe model-specific prompt sensitivities.
- Exploring multilingual prompt structures to evaluate consistency across languages.
- Investigating automatic prompt optimization techniques or user-adaptive prompting systems.

As Large Language Models continue to evolve, understanding how to effectively communicate with them through prompt engineering will become increasingly important in both academic and practical applications.

REFERENCES

[1] T. Brown et al., “Language Models are Few-Shot Learners,” arXiv preprint arXiv:2005.14165, 2020. Available: https://arxiv.org/abs/2005.14165

[2] L. Reynolds and K. McDonell, “Prompt Programming for Large Language Models,” GitHub, 2021. Available: https://github.com/norcross/awesome-prompt-engineering

[3] P. Liu et al., “Pre-train Prompt Tuning: A Unified Paradigm for Prompt Engineering,” arXiv preprint arXiv:2107.13586, 2023. Available: https://arxiv.org/abs/2107.13586

[4] J. Wei et al., “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models,” arXiv preprint arXiv:2201.11903, 2022. Available: https://arxiv.org/abs/2201.11903

[5] OpenAI, “OpenAI API Documentation,” 2024. Available: https://platform.openai.com/docs

[6] Z. Jiang et al., “How Can We Know What Language Models Know?” arXiv preprint arXiv:2012.03803, 2020. Available: https://arxiv.org/abs/2012.03803

[7] W. Zhao et al., “Calibrate Before Use: Improving Few-Shot Performance of Language Models,” arXiv preprint arXiv:2102.09690, 2021. Available: https://arxiv.org/abs/2102.09690

[8] T. Schick and H. Schütze, “Exploiting Cloze Questions for Few-Shot Text Classification,” arXiv preprint arXiv:2001.07676, 2021. Available: https://arxiv.org/abs/2001.07676

[9] OpenAI, “Prompt Engineering Guide,” OpenAI Docs, 2024. Available: https://platform.openai.com/docs/guides/prompt-engineering

[10] S. Mishra et al., “Reframing Instructional Prompts for Multi-Task NLP,” arXiv preprint arXiv:2104.08773, 2022. Available: https://arxiv.org/abs/2104.08773