



# PROMPT ENGINEERING

**Whitepaper by**

**Sathish KR**, AI-ML CoE Lead, Mphasis.AI Lead, TTL Cluster  
**Gokulrajan Nagarajan**, Mphasis.AI Teammate, TTL Cluster  
**Kamali Ramraj**, Mphasis.AI Teammate, TTL Cluster  
**Dronadula Priya**, Mphasis.AI Teammate, TTL Cluster



# Contents

1. Abstract	1
2. Background	1
3. Objectives	1
4. Problem Statement	1
5. Structure of the Whitepaper	1
6. What is a Large Language Model?	1
7. How do LLMs Work?	1
8. Large Language Model Settings	2
9. What is Prompt Engineering?	2
10. What is Linguistic Analysis and its Role in Prompt Engineering?	2
11. Importance of Prompt Engineering	2
12. Techniques of Prompt Engineering	3
13. Best Practices	4
14. Basic Level of Prompts	5
15. Intermediate Level of Prompts	6
16. Image Prompting Techniques	7
17. Potential Risks of Prompt Engineering	8
18. Myths and Fact About Prompt Engineering	8
19. Conclusion	8

# 1. Abstract

This whitepaper explores the fundamentals of prompt engineering in the field of Artificial Intelligence (AI). Prompt engineering involves providing explicit instructions to AI systems to achieve desired outputs. The paper emphasizes the significance of prompt engineering in AI and highlights the role of linguistic analysis in creating effective prompts. Additionally, the paper discusses diverse applications of prompt engineering in various AI domains.

## 2. Background

Artificial Intelligence (AI) has transformed the way we interact with technology, becoming an indispensable part of our daily lives. From speech recognition to image and voice recognition, AI systems have revolutionized information access and task completion.

## 3. Objectives

The purpose of this whitepaper is to explore the fundamentals of prompt engineering in the context of AI. It aims to provide insights into the importance of prompt engineering and its impact on the performance of AI systems. Additionally, the whitepaper seeks to examine the role of linguistic analysis in creating effective prompts.

## 4. Problem Statement

Despite the advancements in AI, effectively instructing AI systems to produce desired outputs remains a challenge. Explicit instructions, known as prompts, play a crucial role in guiding AI systems towards the desired outcomes. However, prompt engineering techniques have not been extensively explored and understood.

## 5. Structure of the Whitepaper

The whitepaper is organized with the following sections for any novice users to understand prompt engineering.

- What is a Language Model? - Fundamentals of Large Language Model
- What is Prompt Engineering?
- Techniques, Best Practices and Various Levels of Prompt Engineering in AI
- Potential Risks of Prompt Engineering and Myths
- Conclusion

## 6. What is a Large Language Model?

A large language model is a Machine Learning model that is trained to predict the next word in a sequence of words. These models are trained on large amounts of text data, allowing them to learn the patterns and relationships between words in the language.

## 7. How do LLMs Work?

LLMs work by using a process called unsupervised learning. The model is fed large amounts of text data and uses statistical analysis to identify patterns and relationships between words. After the model has undergone training, it becomes capable of generating text that bears similarity to the text it was trained on.

# 8. Large Language Model Settings

Large language models have various settings that can be configured to customize their behavior and performance. While the specific settings may depend on the implementation and platform, here are some common options:

## Context Length

Language models process text in chunks or “context windows.” The context length determines how much text the model considers at a time. Longer context lengths allow for more context-aware responses, but may increase response time. Users can adjust the context length based on the desired trade-off between response quality and speed.

## Temperature

The temperature setting controls the randomness or creativity of the model’s responses. Higher temperature values (e.g., 0.8) make the responses more diverse and unpredictable, while lower values (e.g., 0.2) make the responses more focused and deterministic. Users can adjust the temperature setting to achieve the desired level of creativity in the model’s output.

## Top-K Sampling

These settings determine how the model selects the next word in its response. Top-K sampling limits the selection to the Top-K most likely words sampling limits the selection to the smallest set of words whose cumulative probability exceeds a certain threshold. These settings can be used to control the diversity and length of the model’s responses.

# 9. What is Prompt Engineering?

Prompt engineering is an AI technique that serves several purposes. It encompasses the process of refining large language models or LLMs with specific prompts and refining input to various generative AI services to generate text or images. The process involves creating prompts that accurately capture the intent of the user and produce the desired outputs.

# 10. What is Linguistic Analysis and its Role in Prompt Engineering?

Linguistic analysis is a crucial component in the field of artificial intelligence. It involves studying language and how it is used in various contexts to identify patterns and relationships between words and phrases. Linguistic analysis is essential in creating effective prompts that accurately capture the user’s intent in prompt engineering. It is also important in creating natural language processing systems that can accurately interpret and respond to user input in AI applications, such as speech recognition and text-to-speech systems. By understanding the nuances of language and using linguistic analysis to create effective prompts, we can improve the accuracy and relevance of AI-generated outputs.

# 11. Importance of Prompt Engineering

Prompt engineering is crucial for ensuring that AI and Machine Learning models produce accurate and reliable outputs. The quality of a model’s output depends on the quality of the prompts or instructions that are provided to it. By providing clear and concise instructions, prompt engineering can help optimize the performance of AI and Machine Learning models, enabling them to produce the desired outputs. Prompt engineering is particularly important in Natural Language Processing (NLP), where the nuances of language and context can significantly impact the accuracy and reliability of the model’s outputs. For example, a model that is trained to generate responses to customer service inquiries must be able to understand the intent behind the inquiry and provide a relevant and accurate response. By crafting effective prompts that accurately capture the intent of the user, prompt engineering can help to ensure that the model produces accurate and reliable outputs.

# 12. Techniques of Prompt Engineering

Different types of prompting can be used in prompt engineering, depending on the specific use case and requirements of the model.

## Zero-shot prompting

Zero-shot prompting involves training a model to generate responses to tasks that it has not been trained on. This can be useful in situations where the model needs to generalize to new tasks or domains.

**For example:**

**Prompt:** "Translate 'Hello' to Spanish."

**Response:** The translation of 'Hello' to Spanish is 'Hola'.

## One-shot prompting

One-shot prompting involves providing a model with a single example of a task and asking it to generate a response. This can be useful in situations where there is limited data available for training the model.

**For example:**

**Prompt:** Paris is the capital of France.

**Prompt:** What is the capital of Australia?"

**Response:** "Canberra."

## Few-shot prompting

Few-shot prompting involves providing a model with a few examples of a task and asking it to generate a response. This can be useful in situations where there is some data available for training the model, but not enough to fully train it.

**For example:**

**Prompt:**

Example 1: An apple is a fruit.

Example 2: A carrot is a vegetable.

Example 3: A cow is an animal.

A rose is a..."

**Response:** "Flower."

- ✓ A prompt created by a prompt engineer is intentionally designed to guide the model's output.  
(Persona – Prompt engineer)
- ✓ User input is the natural input provided by individuals interacting with the model in real-world scenarios.  
(Persona – End customer)
- ✓ Both prompt engineering and user input play crucial roles in shaping the behavior and responses of the language model.

# 13. Best Practices

- **Clear instructions:** When providing information, make sure to use clear and concise language so the user can easily understand. Try to use simple language and avoid technical terms unless they are necessary.  
**Example:** User wants to find a hotel in London with a swimming pool instruction: “I am looking for a hotel in London with a swimming pool. Can you recommend some options, please?”
- **Adopt a person:** It can be helpful to approach information-seeking as if you are adopting the user and taking them under your wing. Consider their needs, preferences and abilities to tailor your assistance to them.  
**Example:**
  - **Prompt:** “I want to learn a programming language, but I have no prior experience.”
  - **Adoptive Prompt Response:** “That’s great! I’m here to guide you through the process. Let’s start with a beginner-friendly programming language like Python. Would you like me to recommend learning resources or provide step-by-step tutorials to get you started?”
- **Specify the format:** When providing information, specify the format in which it is presented.  
**Example:** If providing a website link, indicate that it is a URL. If providing a document, specify the file type.
- **Avoid leading the answer:** Avoid leading the user to a specific answer or conclusion. Instead, present all relevant information and let them draw their own conclusions.  
**Example:**  
**Initial Prompt:** “What are the adverse impacts of spending too much time on screens?”  
**Prompt Response:** “Spending too much time on screens can result in higher levels of sedentary behavior, eye strain and disturbances in sleep patterns.”  
**Adjusted Prompt:** Examine the possible adverse consequences of spending excessive time on screens.  
**Adjusted Prompt Response:**  
Excessive screen time can have several potential negative effects, including:
  - **Physical Health Issues:** Long hours of screen time can lead to obesity due to a lack of physical activity. It can also lead to other health issues like cardiovascular disease. Constant exposure to screens can strain the eyes causing headaches, blurred vision and long-term vision issues.
  - **Sleep Disorders:** The blue light emitted by screens can interfere with the production of melatonin, a hormone that regulates sleep. This can lead to difficulties falling asleep, insomnia and overall poor sleep quality.
  - **Mental Health Problems:** Excessive screen time, especially on social media, can contribute to feelings of anxiety and depression. It can also lead to low self-esteem and loneliness, especially among young people.
  - **Behavioral Problems:** In children, excessive screen time can lead to attention problems, school difficulties and learning disorders. It can also contribute to aggressive behavior and other social issues.
  - **Poor Posture:** Staring at a screen for a long time often leads to slouching, which can cause back, neck and shoulder pain.
  - **Reduced Productivity:** Spending too much time on screens can reduce productivity as it can be a major distraction and time-consuming.
  - **Social Isolation:** Excessive screen time can lead to reduced interpersonal and communication skills, leading to social isolation.
  - **Digital Eye Strain:** This can cause symptoms like dry and irritated eyes, fatigue, and headaches.These are just some of the potential negative effects of excessive screen time. It’s important for individuals to manage their screen time effectively and take regular breaks to mitigate these risks.
- **Limit the scope:** When assisting with information-seeking, it is important to limit the scope of the search to avoid overwhelming the user with too much information. Ask clarifying questions to understand exactly what they are looking for and focus on providing relevant information.  
**Example:** Limited Scope: “Could you recommend a budget-friendly laptop with at least 8 GB RAM and a minimum of 256 GB storage?”
- **Positive instructions:** Use positive language and instructions to guide the user.  
**Example:** Instead of saying “Don’t click on that link,” say “Please avoid clicking on that link.”
- **Iterative process:** Information-seeking can be an iterative process, so be prepared to adjust your approach as needed based on the user’s feedback and additional questions.



**Example:**

**Prompt 1:** “I would like to gather some information about the solar system.”

**Response 1:** “Sure, the solar system is a fascinating subject. It includes the Sun and everything that orbits around it, such as planets, moons, asteroids and comets. What specific information are you looking for?”

**Prompt 2:** “I’m interested in the planets. Could you tell me more about them?”

**Response 2:** “Certainly, our solar system consists of a total of eight planets, which are arranged in order of their proximity to the sun: Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus and Neptune. Each planet possesses distinct characteristics. Is there a particular planet you would like to explore further?”

**Prompt 3:** “I’d like to know more about Mars.”

**Response 3:** “Mars, commonly referred to as the Red Planet, is positioned as the fourth planet from the Sun. It’s known for its red appearance, which is due to iron oxide (or rust) on its surface. Mars has the tallest volcano and deepest, longest canyon in the solar system. Scientists are also exploring the possibility of life on Mars. Would you like information on its atmosphere, geography, exploration missions or something else?”

- **Provide examples:** Providing examples can be helpful in clarifying information or demonstrating how to complete a task. Use relevant and clear examples to help the user understand better.

*Example: Initial Prompt:* “What are the symptoms of a common cold?”

*Example-providing Prompt:* “For instance, what are some common symptoms of a cold such as cough, congestion or runny nose?”

**Addressing bias and inclusivity:** When providing information, it is important to address bias and promote inclusivity. Ensure that the information you provide is diverse, inclusive and representative of different perspectives and experiences. Be aware of your own biases and strive to present information in a fair and balanced manner.

**Example:**

**Persona** – Prompt Engineer.

**Biased Prompt:** “Why do men excel in science?”

- **Inclusive Prompt:** “What factors contribute to success in science across diverse individuals?” Promoting inclusiveness and reducing bias in generated responses can be achieved by formulating prompts in a manner that doesn’t make assumptions about an individual’s characteristics or abilities based on factors such as gender or ethnicity. This approach helps to mitigate bias and fosters a more inclusive environment in the generated responses.
- **Usage of pre-trained models:** Prompt engineering is that it can be time-consuming and resource-intensive, particularly when working with complex models. Using a pre-trained model, we can save time and reduce the need for extensive prompt engineering.

**Example: User:** “Translate ‘Hello’ to French.”

**Efficient Prompt Engineering:** “Can you assist me in translating the word ‘Hello’ into French?”

## 14. Basic Level of Prompts

### How to summarize the text using a prompt?

**Prompt:** “Summarize the following content in two lines: {user input}”

**User Input:** Atoms of radioactive elements can undergo a process known as nuclear fission, where they split and release energy. This phenomenon, as described by Albert Einstein, involves the conversion of a small amount of mass into energy. While the energy released may seem insignificant at first, the emission of neutrons during the process triggers a chain reaction. These neutrons collide with other radioactive elements, releasing more neutrons and generating a significant amount of energy.

This energy is predominantly converted into heat, leading to an uncontrollable expansion and ultimately resulting in an explosion. Therefore, while atoms themselves don’t explode, they produce energy that can cause explosions.

**Output:** Atoms of radioactive elements can undergo nuclear fission, converting a small amount of mass into energy. This energy, released through a chain reaction, can cause explosions due to the uncontrollable expansion of heat.

## How to create an email using a prompt?

**Prompt:** My Name: X Manager name: Y Write a **serious**, professional email to my manager saying that I will be out of the office today since I am sick". Be concise.

**Output: Subject:** Sick Leave - Out of Office Today

Dear [Manager's Name],

I wanted to inform you that I will be unable to come to the office today due to illness. I am not feeling well and need to take a sick day to rest and recover. I will ensure that any urgent tasks or pending work are handled promptly upon my return. Please feel free to reach out to me via email if there are any critical matters that require my attention.

Thank you for your understanding.

Best regards,

[Your Name]

## How to generate code using prompt?

**Prompt:** Generate a Python code snippet to calculate the Arithmetic series of a given number.

**Output:**

Certainly! Here's a Python code snippet to calculate the arithmetic series of a given number:

```
def arithmetic_series(n):
    return (n * (n + 1)) // 2

number = int(input("Enter a number: "))
result = arithmetic_series(number)
print("The arithmetic series of", number, "is", result)
```

In this code snippet, we define a function `arithmetic_series()` that takes an integer `n` as input. The formula to calculate the arithmetic series is  $(n * (n + 1)) // 2$ . We then prompt the user to enter a number, calculate its arithmetic series using the `arithmetic_series()` function, and print the result. The `//` operator is used for integer division, which ensures that the result is an integer.

# 15. Intermediate Level of Prompts

## Chain of Thought (CoT)

Chain of Thought (CoT) prompting is a recently developed prompting method, which encourages the LLM to explain its reasoning. CoT is achieved by a few shot prompting.

**Example:**

Which is a faster way to get home?

Option 1: Take a 20-minute bus, then a 50-minute bus and finally a 20-minute train.

Option 2: Take a 100-minute train, then 35-minute bike ride and finally a 20-minute bus.

Option 1 will take  $20+50+20 = 90$  minutes.

Option 2 will take  $100+35+20 = 155$  minutes.

Since Option 1 takes 90 minutes and Option 2 takes 155 minutes, Option 1 is faster.

Which is a faster way to get to work?

Option 1: Take a 700-minute bus, then an hour train and finally a 30-minute bike ride.

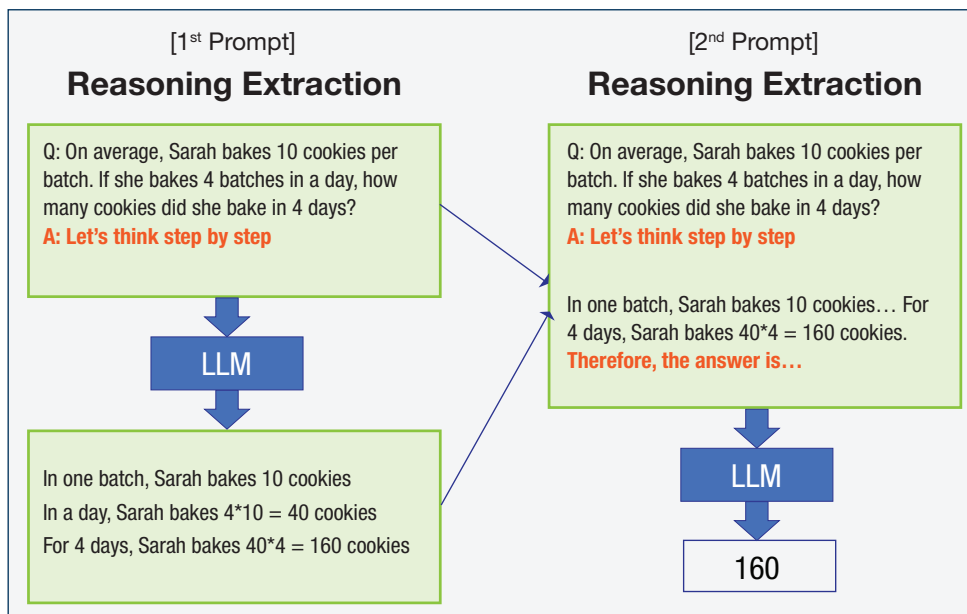
Option 2: Take a 900-minute bus, then a half hour train and finally a 20-minute bike ride.



## Zero-shot Chain of Thought

Zero-shot Chain of Thought (Zero-shot-CoT) prompting is an extension of CoT prompting that incorporates a remarkably simple zero-shot prompt. They find that by appending the words *Let's Think Step by Step* to the end of a question, LLMs can generate a chain of thought that answers the question. From this chain of thought, they can extract more accurate answers.

Technically, the full Zero-shot-CoT process involves two distinct prompts/completions. In the below image, the leftmost top bubble generates a chain of thought, while the rightmost top bubble utilizes the output from the first prompt (including the prompt itself) to extract the answer from the chain of thought. This second prompt is a self-augmented prompt. Through this chain of thought, they can extract more accurate answers.



## Self-consistency

Self-consistency, as a technique, indeed offers a promising approach to enhance the performance of chain-of-thought (CoT) prompting, particularly in tasks requiring arithmetic and commonsense reasoning. By sampling multiple diverse reasoning paths through few-shot CoT and selecting the most consistent answer, self-consistency aims to overcome the limitations of naive greedy decoding, thereby improving the overall robustness and accuracy of the model's responses.

# 16. Image Prompting Techniques

Image prompting is a technique used in AI and Machine Learning to generate new images based on existing images or prompts.

Several techniques can be used to improve the quality and accuracy of the generated images:

- **Style Modifier:** This technique involves adjusting the style or aesthetic of the generated images to achieve a desired look. It can include modifications such as changing the color palette, adding filters or applying artistic effects.
- **Quality Booster:** This technique aims to enhance the overall quality of the generated images. It may involve using algorithms or neural networks to refine details, reduce noise, improve sharpness or enhance resolution.
- **Repetition:** Repetition refers to generating multiple variations of an image and selecting the best one based on pre-defined criteria. By generating multiple versions and comparing them, the system can choose the image with the highest quality or accuracy.
- **Weighted Terms:** This technique involves assigning different weights or importance to certain terms or features during the image generation process. By giving higher weights to specific attributes or characteristics, the system can prioritize their inclusion or emphasize their presence in the generated images.
- **Fix Deformed Generations:** Sometimes, the generated images may contain deformities or errors. This technique involves identifying and correcting these issues to improve the accuracy and quality of the generated images.

It may involve using post-processing techniques or additional algorithms to rectify any distortions or inconsistencies.

- **Shot Type:** Shot type refers to the perspective or framing of the image, such as close-up, wide-angle or overhead view. By specifying the desired shot type during the image generation process, the system can generate images that align with the intended composition or viewpoint, improving their quality and visual appeal.

## 17. Potential Risks of Prompt Engineering

While prompt engineering can help to optimize the performance of AI and Machine Learning models, there are also potential risks associated with its use. One of the most significant risks is AI hallucination, which occurs when a language model generates false information. This can have serious consequences in certain applications, such as in the medical or legal fields. It is important to approach prompt engineering with caution and to develop robust protocols for ensuring the accuracy and reliability of AI and Machine Learning models.

## 18. Myths and Facts About Prompt Engineering

### **Myth: Prompt engineering is a one-time task - Wrong**

**Fact - Continuous optimization is key:** Prompt engineering is not a one-time task. It requires ongoing optimization to achieve success and maintain competitiveness. Continuous evaluation, analysis and refinement are necessary to improve performance over time.

### **Myth: It is just about keywords - Wrong**

**Fact - Prompt engineering is a symphony of elements:** Prompt engineering goes beyond keywords. It involves various elements such as understanding user intent, crafting effective prompts, considering context, utilizing relevant data and incorporating natural language processing techniques. It is a holistic approach that considers multiple factors to generate meaningful and relevant responses.

### **Myth: More prompts more results - Wrong**

**Fact - Quality over quantity:** It is not necessarily true that more prompts lead to more results. The focus should be on quality rather than quantity. Well-crafted and targeted prompts are more likely to generate desired outcomes and provide a better user experience.

### **Myth: Prompt engineering is set and forget - Wrong**

**Fact - Continuous monitoring is important:** Prompt engineering is not a set-and-forget task. Continuous monitoring is crucial to assess the performance of prompts, identify any issues or gaps and make necessary adjustments. Regular evaluation and optimization ensure that the prompts remain effective and aligned with the desired goals.

### **Myth: Experts only do prompt engineering - Wrong**

**Fact - Learning and iteration lead to mastery:** Prompt engineering is not limited to experts only. While expertise can enhance the effectiveness of prompt engineering, continuous learning and iteration are key to mastering this skill. It requires staying updated with industry trends, experimenting with different approaches, analyzing feedback and making improvements based on insights gained.

## 19. Conclusion

Prompt engineering is a crucial technique for optimizing the performance of AI and Machine Learning models. By providing clear and concise instructions, prompt engineering can help to ensure that AI and Machine Learning models produce accurate and reliable outputs. However, there are potential risks associated with its use, such as AI hallucination. It is important to approach prompt engineering with caution and to develop robust protocols for ensuring the accuracy and reliability of AI and Machine Learning models. As AI and Machine Learning continue to advance, prompt engineering will become increasingly important for optimizing the performance of these technologies and ensuring their safe and effective use.

## Acknowledgments

### Special thanks to –

- Sarangan V (Chief Architect, Mphasis.ai Lead, TTL Cluster)
- Mohammed Rupawalla (CTO, Portfolio Account)
- Ramanathan Srikumar (Chief Solutions Officer, Mphasis)
- Robert Down (Digital DTM)

**AI will not replace your job as an engineer. An engineer who is using AI will.**

## About Mphasis

Mphasis' purpose is to be the “*Driver in the Driverless Car*” for Global Enterprises by applying next-generation design, architecture and engineering services, to deliver scalable and sustainable software and technology solutions. Customer centricity is foundational to Mphasis, and is reflected in the Mphasis' Front2Back™ Transformation approach. Front2Back™ uses the exponential power of cloud and cognitive to provide hyper-personalized ( $C = X2C_{tm}^2 = 1$ ) digital experience to clients and their end customers. Mphasis' Service Transformation approach helps 'shrink the core' through the application of digital technologies across legacy environments within an enterprise, enabling businesses to stay ahead in a changing world. Mphasis' core reference architectures and tools, speed and innovation with domain expertise and specialization, combined with an integrated sustainability and purpose-led approach across its operations and solutions are key to building strong relationships with marquee clients. [Click here](#) to know more. (BSE: 526299; NSE: MPHASIS)

For more information, contact: [marketinginfo.m@mphasis.com](mailto:marketinginfo.m@mphasis.com)

### USA

Mphasis Corporation  
41 Madison Avenue  
35<sup>th</sup> Floor, New York  
New York 10010, USA  
Tel: +1 (212) 686 6655

### UK

Mphasis UK Limited  
1 Ropemaker Street, London  
EC2Y 9HT, United Kingdom  
T : +44 020 7153 1327

### INDIA

Mphasis Limited  
Bagmane World Technology Center  
Marathahalli Ring Road  
Doddanakundhi Village, Mahadevapura  
Bangalore 560 048, India  
Tel.: +91 80 3352 5000



WAS 27/05/24 US LETTER B&S L 8890