Prompt Engineering

Notes-<https://docs.google.com/document/d/1e5UdugYiEHcK7uKK8oKgZzb_y4TsY9pI52AaQDD0Z0A/edit?usp=sharing>

<https://docs.google.com/document/d/1e5UdugYiEHcK7uKK8oKgZzb_y4TsY9pI52AaQDD0Z0A/edit?usp=sharing>

[Prompt Engineering for AI Guide](https://cloud.google.com/discover/what-is-prompt-engineering)

[Prompt engineering - OpenAI API](https://platform.openai.com/docs/guides/prompt-engineering)

[What is Prompt Engineering? A Detailed Guide For 2024](https://www.datacamp.com/blog/what-is-prompt-engineering-the-future-of-ai-communication)

[Awesome-Prompt-Engineering](https://github.com/promptslab/Awesome-Prompt-Engineering)

<https://github.com/langflow-ai/langflow>

[Prompt Engineering Guide](https://www.promptingguide.ai/)

[Prompting Techniques](https://www.promptingguide.ai/techniques)

[How to Become a Prompt Engineer: A Comprehensive Guide](https://www.datacamp.com/blog/how-to-become-a-prompt-engineer)

[Understanding oLLaMa](https://www.hostinger.in/tutorials/what-is-ollama#:~:text=and%20enhance%20performance.-,Local%20AI%20model%20management,who%20prioritize%20strict%20data%20security.)

[LoRA](https://huggingface.co/docs/diffusers/en/training/lora)

[rag-fusion](https://python.langchain.com/v0.1/docs/templates/rag-fusion/)

**[LM StudioDocs](https://lmstudio.ai/)**

[Prompt Engineering with LangChain](https://www.datacamp.com/tutorial/prompt-engineering-with-langchain)

[ConversationBufferMemory](https://python.langchain.com/api_reference/langchain/memory/langchain.memory.buffer.ConversationBufferMemory.html)

[LangChain Python API Reference](https://python.langchain.com/api_reference/index.html)

[Introduction to Using Transformers and Hugging Face](https://www.datacamp.com/tutorial/an-introduction-to-using-transformers-and-hugging-face)

A black rectangle with white text

Description automatically generated

**Tools for Prompt Design and Optimization**

1. **OpenAI API**: [OpenAI API](https://platform.openai.com/)
2. **Hugging Face Transformers**: Hugging Face Transformers
3. **LangChain**: [LangChain](https://www.langchain.com/)
4. **PromptPerfect**: PromptPerfect

**Tools for Prompt Testing and Debugging**

1. **WeightWatcher**: [WeightWatcher GitHub Repository](https://github.com/CalculatedContent/WeightWatcher)
2. **EleutherAI's GPT-Neo**: [EleutherAI](https://www.eleuther.ai/)

**Tools for Visualization and Workflow Integration**

1. **Streamlit**: [Streamlit](https://streamlit.io/)
2. **Gradio**: [Gradio](https://gradio.app/)

**Technologies for Advanced Prompt Engineering**

1. **Google Gemini**: [Google Gemini](https://inthecloud.withgoogle.com/)​

[KDnuggets](https://www.kdnuggets.com/3-new-prompt-engineering-resources)

.

1. **LLMLingua**: [LLMLingua GitHub](https://github.com/)​

[KDnuggets](https://www.kdnuggets.com/3-new-prompt-engineering-resources)

.

1. **ChatGPT Plugins**: [ChatGPT Plugins](https://openai.com/chatgpt/plugins/)

**Tools for Fine-Tuning and Customization**

1. **PyTorch**: [PyTorch](https://pytorch.org/)
2. **TensorFlow**: [TensorFlow](https://www.tensorflow.org/)
3. **AI21 Studio**: AI21 Studio

**Specialized Prompt Libraries and Repositories**

1. **PromptBase**: [PromptBase](https://promptbase.com/)
2. **OpenPrompt**: [OpenPrompt GitHub](https://github.com/thunlp/OpenPrompt)

**Types and Techniques of Prompt Engineering**

Prompt Engineering involves designing tailored inputs for AI models to get accurate and effective outputs. Understanding its **types** and **techniques** will help you optimize the performance of AI systems.

**Types of Prompts in Prompt Engineering**

**1. Instruction-Based Prompts**

These prompts give direct commands or instructions to the AI.

* **Examples:**
  + "Summarize this article in 3 sentences."
  + "Explain Newton's laws in simple terms."

**2. Role-Based Prompts**

Assign a role to the AI to narrow its behavior and style.

* **Examples:**
  + "You are a math tutor. Explain the Pythagorean theorem."
  + "Act as a project manager and create a risk management plan."

**3. Contextual Prompts**

Provide additional context or background information to the model to generate better responses.

* **Examples:**
  + "Given the rise in electric vehicles, explain how lithium demand will impact the economy."
  + "Using this data [insert table], write an analysis of sales trends."

**4. Example-Driven Prompts**

Also called **few-shot prompts**, these include examples to set a pattern for the AI to follow.

* **Examples:**
  + "Translate the following into French:
    - Hello -> Bonjour
    - Good morning -> Bon matin
    - Goodbye -> [Your response]"

**5. Chain-of-Thought Prompts**

Encourage the AI to think step-by-step for solving complex problems.

* **Examples:**
  + "Solve this problem step-by-step: If a train travels 50 miles in 1 hour, how long will it take to travel 200 miles?"

**6. Open-Ended Prompts**

Used for generating creative or exploratory outputs without strict constraints.

* **Examples:**
  + "Write a short story about a dragon who learns to fly."
  + "Generate ideas for a sustainable startup."

**7. Specific Prompts**

Narrowly focused prompts to control the output format, style, or content.

* **Examples:**
  + "List the advantages and disadvantages of cloud computing in a table format."
  + "Write a professional email apologizing for a delay in delivery."

**8. Iterative Prompts**

Involves using follow-up questions or refining previous outputs to reach the desired result.

* **Example:**
  + First Prompt: "Summarize this article."
  + Second Prompt: "Rewrite the summary to make it more engaging."

**9. Multimodal Prompts**

Used for AI systems that can handle both text and other input types (e.g., images, audio).

* **Example:**
  + "Describe the key features of this image [upload image]."

**Techniques in Prompt Engineering**

**1. Few-Shot Learning**

Provide a few examples in the prompt to train the AI on the desired output style and structure.

* **Example:**
  + "Classify the following as 'Positive' or 'Negative':
    - The weather is amazing. -> Positive
    - I feel terrible today. -> Negative
    - The movie was fantastic. -> [Your response]"

**2. Zero-Shot Learning**

Ask the AI to perform a task without providing any examples, relying entirely on the prompt's clarity.

* **Example:**
  + "Write a haiku about the moon."

**3. Chain-of-Thought (CoT) Prompting**

Encourage the model to reason step-by-step for complex problems.

* **Example:**
  + "If a worker builds 5 chairs in a day, how many chairs will 3 workers build in 4 days? Break your answer into steps."

**4. Iterative Prompt Refinement**

Improve results by analyzing and tweaking the prompt multiple times.

* **Steps:**
  1. Test the prompt.
  2. Identify shortcomings in the output.
  3. Add context, constraints, or examples.

**5. Role Specification**

Assign a persona to the AI for targeted outputs.

* **Example:**
  + "You are a historian. Write about the significance of the French Revolution."

**6. Multimodal Prompting**

Incorporate non-text data (images, graphs) for AI systems like GPT-4V or DALL·E.

* **Example:**
  + "Analyze this chart and summarize its key trends: [chart link]."

**7. Contextualization**

Provide a detailed background to improve relevance.

* **Example:**
  + "Given this scenario: A company faces declining sales due to competition. Suggest 3 strategies to increase market share."

**8. Task Decomposition**

Break complex problems into smaller tasks and tackle them sequentially.

* **Example:**
  1. "List the main points in the article."
  2. "Expand each point into a paragraph."
  3. "Write a conclusion based on the paragraphs."

**9. Instruction Tuning**

Use explicit instructions for fine-grained control over output.

* **Example:**
  + "Write a Python function to reverse a string. Include comments explaining each step."

**10. Formatting and Constraints**

Guide the AI by specifying structure or length.

* **Examples:**
  + "Generate a 100-word summary of this text."
  + "List 5 bullet points about the advantages of solar energy."

**11. Ethical and Bias-Aware Prompting**

Design prompts that reduce bias or lead to neutral, inclusive outputs.

* **Example:**
  + Avoid: "Why are women better at multitasking?"
  + Better: "Explain the science behind multitasking and how it varies among individuals."

**12. Temperature Control in API Prompting**

Adjust the **temperature parameter** in API-based models to control randomness:

* Lower values (e.g., 0.2): More deterministic and focused responses.
* Higher values (e.g., 0.8): Creative but less predictable outputs.