

Prompt engineering

Prompt engineering is the practice of crafting clear, precise, and effective prompts to guide generative AI models in producing accurate, relevant, and high-quality outputs. It bridges the gap between human intent and AI understanding by structuring input commands or questions in a way that the AI can best interpret and respond to.

What is a prompt?

A prompt is the input text or instruction given to an AI model to elicit a response. Prompts can range from simple questions to complex instructions or multi-step tasks.

Importance of Prompt Engineering

- Improves the specificity and relevance of AI outputs.
- Reduces ambiguity and unintended outputs.
- Saves time by minimizing post-processing or re-queries.
- Enables models to perform specific tasks like summarization, translation, code generation, or reasoning effectively.

Common Techniques with Examples

- Basic Prompting: Simple, direct questions.
 - Prompt: *What is the world's largest forest?*
 - Output: *The world's largest forest is the Amazon Rainforest.*
- Chain-of-Thought Prompting: Guides the AI to break down complex problems into logical steps.
 - Prompt: *Explain step-by-step how photosynthesis works.*
 - Output: An explanation that sequentially details the process.
- Tree-of-Thought Prompting: The AI explores multiple pathways or solutions before selecting the best.
 - Prompt: *List environmental and social impacts of climate change separately.*
 - Output: Two detailed lists elaborating effects in each category.
- Self-Refine Prompting: The model critiques its previous answers and improves them iteratively.
 - Prompt: *Write a short essay on climate change and then improve it by adding statistics.*
 - Output: Initial essay followed by a refined version with data.

- Directional-Stimulus Prompting: Embeds specific keywords or styles to guide the generation.
 - Prompt: *Write a poem about love including the words "heart," "passion," and "eternal".*
 - Output: A poem incorporating these words elegantly.

Summary

Prompt engineering is both art and science—requiring creativity and linguistic skill to design prompts that elicit desired AI behaviors. It plays a critical role in maximizing the utility of LLMs and generative AI systems, making outputs more meaningful and actionable with less trial-and-error.

This practice continues to evolve with emerging techniques to harness advanced AI capabilities more effectively.

Show concise prompt templates for common tasks

Here are concise prompt templates for common tasks that help guide AI models effectively:

Task	Prompt Template
Summarization	"Summarize the following text in 3 bullet points: [Insert Text]"
Email Writing	"Write a polite and professional email to [recipient] about [topic], including next steps."
Code Generation	"Generate Python code to [describe functionality, e.g., parse JSON and output keys]"
Text Translation	"Translate the following English text to French: [Insert Text]"
Question Answering	"Answer the following question based on the context: [Insert context]. Question: [Insert question]"
Project Planning	"Create a detailed project timeline for [project type] with key milestones and responsible teams."
Data Analysis	"Analyze this dataset and summarize the top three trends with business impact."
Creative Writing	"Write a short story about [topic], using a friendly and imaginative style."
Step-by-Step Reasoning	"Explain step-by-step how to solve [problem description]."
Social Media Post	"Create a catchy social media post for [product/event], targeting [audience]."

Each template can be adapted with placeholders like [Insert Text] or [topic] for flexibility. Using clear roles, task descriptions, and formats improves AI output quality by providing explicit instructions and context.

Convert these templates into reusable parameterized prompts

Here are the concise prompt templates converted into reusable parameterized prompt formats with placeholders:

1. Summarization

"Summarize the following text in {num_points} bullet points: {text}"

2. Email Writing

"Write a polite and professional email to {recipient} about {topic}, including next steps."

3. Code Generation

"Generate {language} code to {functionality_description}."

4. Text Translation

"Translate the following {source_language} text to {target_language}: {text}"

5. Question Answering

"Answer the following question based on the context: {context}. Question: {question}"

6. Project Planning

"Create a detailed project timeline for {project_type} with key milestones and responsible teams."

7. Data Analysis

"Analyze this dataset and summarize the top {num_trends} trends with business impact."

8. Creative Writing

"Write a short story about {topic}, using a {style} style."

9. Step-by-Step Reasoning

"Explain step-by-step how to solve {problem_description}."

10. Social Media Post

"Create a catchy social media post for {product_or_event}, targeting {audience}."

Example usage in Python format style:

```
prompt = "Summarize the following text in {num_points} bullet points: {text}".format(  
    num_points=3,  
    text="Artificial Intelligence is transforming industries..."  
)
```

This approach separates static instructions from dynamic content, enabling scalable and maintainable prompt engineering.

Provide parameter schema and examples for each template

Here is a parameter schema and example usage for each reusable prompt template, expressed as JSON-style schemas for clarity and programmatic use:

Template	Parameter Schema	Example Parameters
Summarization	{ "num_points": "int", "text": "string" }	{ "num_points": 3, "text": "AI is transforming industries..." }
Email Writing	{ "recipient": "string", "topic": "string" }	{ "recipient": "John Doe", "topic": "Quarterly report deadlines" }
Code Generation	{ "language": "string", "functionality_description": "string" }	{ "language": "Python", "functionality_description": "parse JSON and output keys" }
Text Translation	{ "source_language": "string", "target_language": "string", "text": "string" }	{ "source_language": "English", "target_language": "French", "text": "Hello, world!" }
Question Answering	{ "context": "string", "question": "string" }	{ "context": "Photosynthesis is process X...", "question": "What is photosynthesis?" }
Project Planning	{ "project_type": "string" }	{ "project_type": "Mobile app development" }
Data Analysis	{ "num_trends": "int" }	{ "num_trends": 3 }
Creative Writing	{ "topic": "string", "style": "string" }	{ "topic": "space exploration", "style": "imaginative" }
Step-by-Step Reasoning	{ "problem_description": "string" }	{ "problem_description": "Solving quadratic equations" }
Social Media Post	{ "product_or_event": "string", "audience": "string" }	{ "product_or_event": "New smartphone launch", "audience": "tech enthusiasts" }