AI Builder | Prompt Builder

A Prompt engineering guide

Prompts are how you build custom generative AI capabilities in Power Platform — like summarizing a body of text, drafting a response, or categorizing an incoming email. Think of prompts as a way of building custom GPT functions using only natural language.

1. What can you build with prompts?

Prompts are the key to many AI solutions, depending on what you need to solve. Here are six common uses:

- ✓ **Classification** of text:
  "Which category does this [blog] belong to: technology, lifestyle, or education?"
- ✓ **Sentiment** analysis:
  "Is this customer [review] positive, negative, or neutral?"
- ✓ **Rewriting** content:
  "Rewrite this technical [guide] in simpler terms for beginners."
- ✓ **Summarize** information:
  "Summarize this [article] in a few paragraphs."
- ✓ **Extract** information :
  "Find the main event date and location in this news [report]."
- ✓ Drafting a **Response**:
  "Draft a reply to this customer's [complaint] about delivery delay."

2. What is prompt engineering?

Prompt engineering is the art of crafting effective prompts to guide AI behavior. It involves designing inputs that clearly convey the task to the AI, ensuring accurate and relevant responses. This process is crucial for maximizing the potential of AI tools, as the quality of the prompt directly influences the output.

Create a product feedback summary based on recent customer [reviews]. The summary should identify key themes, overall sentiment, and any recurring issues or praises. Format the summary to be concise, and present the information in a clear, easy-to-understand manner suitable for a marketing team meeting.

Try Prompt Builder at aka.ms/tryprompts

December 2023

AI Builder | Prompt Builder

A Prompt engineering guide

3. What can I do with prompts?

Prompts are fundamental artifacts in Power Platform. Think of them as custom AI functions and can be invoked via Power Fx across the platform.

Power Apps

In Power Apps, you can easily add prompts as AI models, similar to other AI models. By using **Power Fx**, you can invoke these custom prompts, bringing smart features into your business applications.

Copilot Studio

Prompts can be saved as **Plugins** to enhance your Copilot, much like adding new skills to a toolbox. These plugins, build with prompts, help extend your Copilot with additional functions and expertise.

Power Automate

Prompts in Power Automate are available as AI Builder **actions**, enabling you to automate complex workflows with ease.

Things to keep in mind

Consume AI Credits.

Prompts are AI models as a core concept and hence consume credits at runtime.

Shareable.

Prompts are shareable within the environment.

Solution-aware.

Prompts support ALM and hence are solution-aware.

Azure OpenAI models.

Prompts use Azure OpenAI models backed by Azure enterprise promises.

Your data is yours.

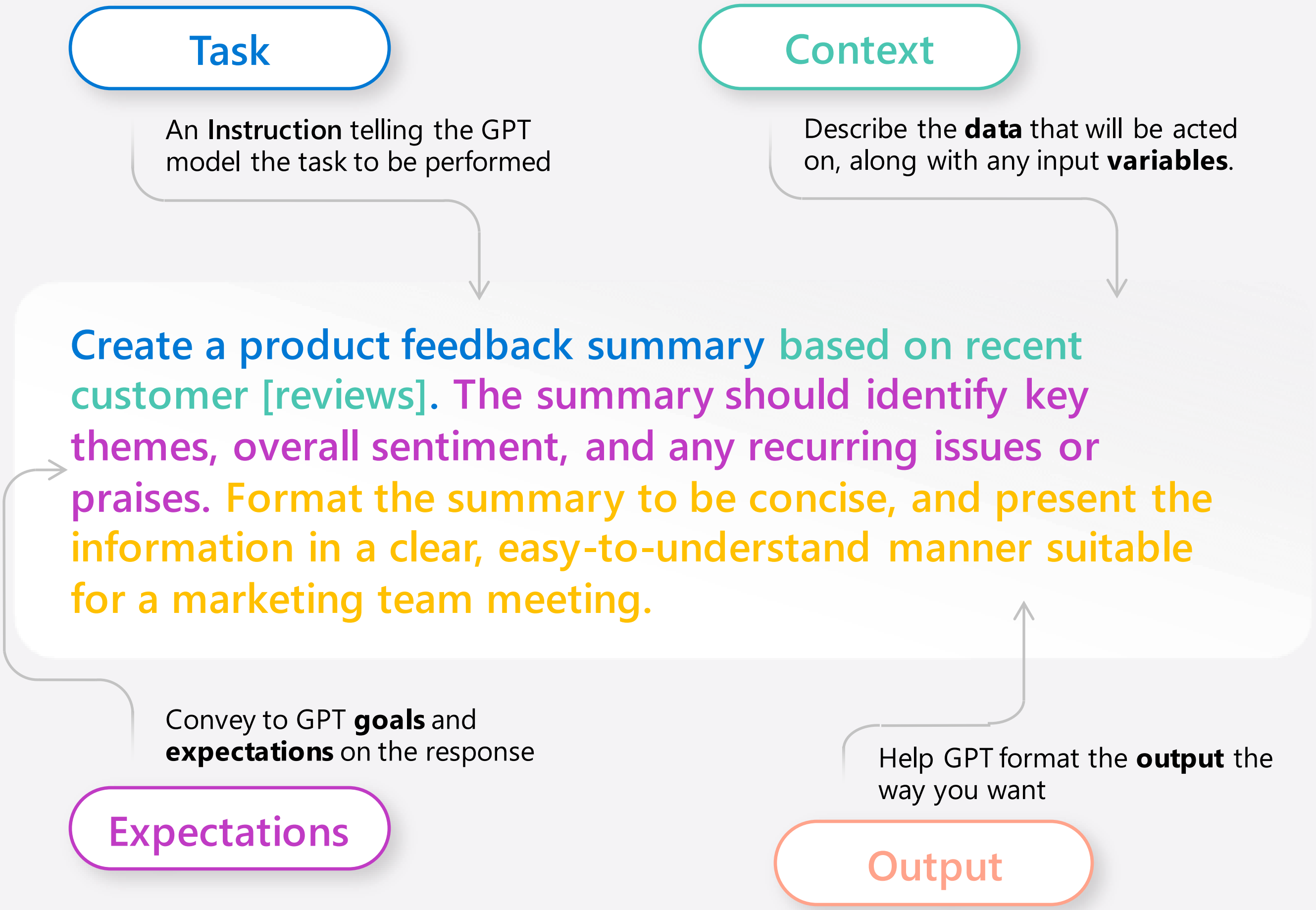
Azure OpenAI models don’t train on your data. Models are only used for inferencing.

Learn more about Prompts on our documentation page.

AI Builder | Prompt Builder
A Prompt engineering guide

Prompt Ingredients

Prompts are how you ask AI model (GPT) to do something for you — it largely has two parts, instruction and data context. To get the best response, it's important to **focus on some of the key elements below** when building you custom AI capabilities.



Task

An **Instruction** telling the GPT model the task to be performed

Context

Describe the **data** that will be acted on, along with any input **variables**.

Create a product feedback summary based on recent customer [reviews]. The summary should identify key themes, overall sentiment, and any recurring issues or praises. Format the summary to be concise, and present the information in a clear, easy-to-understand manner suitable for a marketing team meeting.

Convey to GPT **goals** and **expectations** on the response

Expectations

Help GPT format the **output** the way you want

Output

Tips

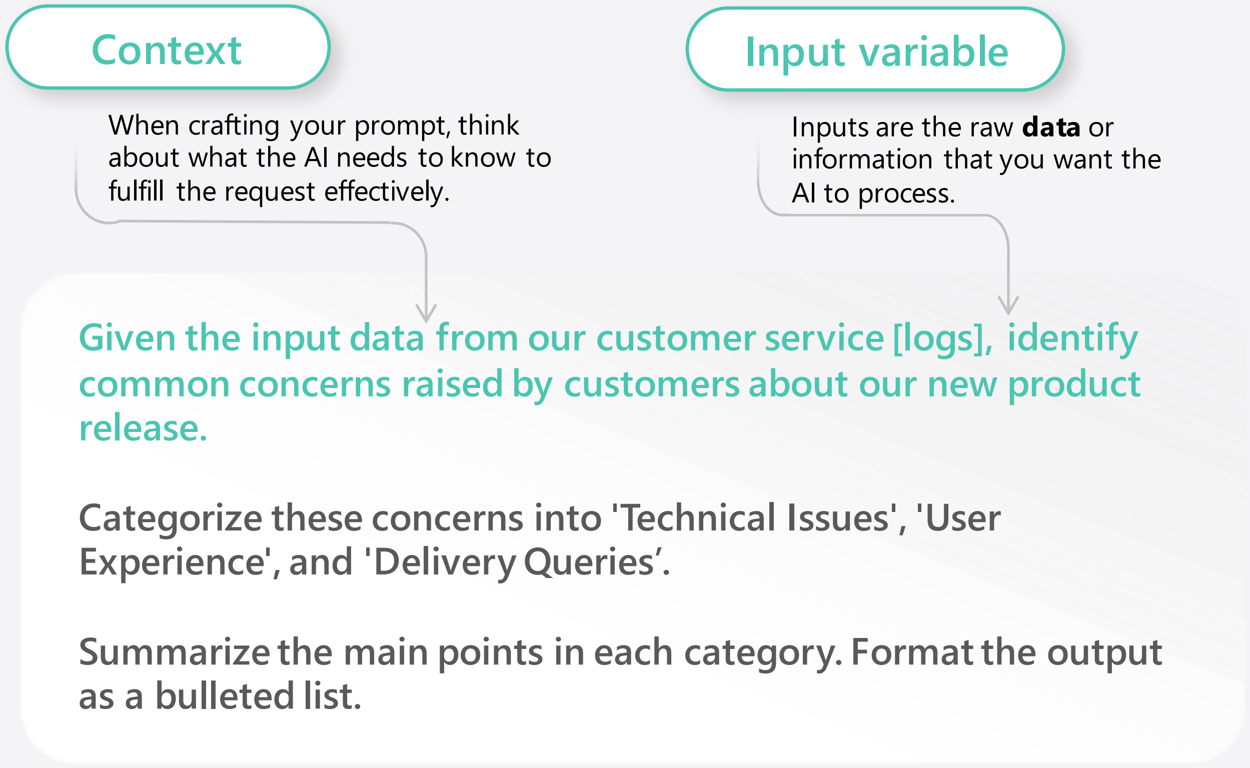
- When creating a prompt, think of it as if you were instructing to a helpful assistant to do something on your behalf – the more **prescriptive** you are the more consistent would be the output.
- Remember prompt engineering is an **iterative process**, so don't shy away from experimentation and trying different ways of asking the question and providing the context.

December 2023

Prompt Strategies

Context and Input

Every prompt you create should be contextualized as a way of setting the stage for the task at hand. The context helps the AI understand not just the 'what' but the **'why' behind your prompt**, leading to more accurate and relevant responses. Custom prompts now allow multiple dynamic values which helps in bringing input data at runtime. Think of these as parameters of a function.



Context

When crafting your prompt, think about what the AI needs to know to fulfill the request effectively.

Input variable

Inputs are the raw **data** or information that you want the AI to process.

Given the input data from our customer service [logs], identify common concerns raised by customers about our new product release.

Categorize these concerns into 'Technical Issues', 'User Experience', and 'Delivery Queries’.

Summarize the main points in each category. Format the output as a bulleted list.

Tips

- Custom prompt can have **multiple input variables**. Keyboard shortcut to add variables in the prompt builder experience is “/”.
- **Be explicit** about what is input data, as the variable name will be replaced at runtime with the actual data, and the context will be lost if not explicitly stated in the prompt.
- **Domain ambiguity** should be clearly clarified in the prompt. Don’t assume AI knows all the domains nuances.

Output formatting

Output formatting in prompt engineering refers to specifying how you want the AI's response to be structured, such as bullet points, paragraphs, or lists, and the style or tone, like formal, casual, or persuasive. GPT can also generate structured outputs like JSON and XML. This ensures the AI's output meets your **downstream processing goals**.

> Analyze the customer [feedback] data and summarize key insights.
>
> Format the response as a JSON object, with fields for 'TotalFeedbackCount', 'PositiveCount', 'NegativeCount', and 'CommonThemes', each containing appropriate values and summaries.

This guides the AI to respond with the exact JSON format, ensuring the output is ready for use in a subsequent output processing step.

Output

Tips

- **Be Specific:** Clearly define the format you need, such as JSON, XML, bullet points, or paragraphs.
- **Match the Use Case:** Align the format with the intended use of the output, like data integration or readability.
- **Include Essential Details:** For structured formats, specify necessary fields or elements.
- **Test and Refine:** Experiment with prompt and generated output to find the most optimal response.

Prompt Strategies

In-Context Learning

In-context learning (ICL), also known as **few-shot learning**, refers to an AI's ability to learn from a few examples or scenarios provided within the prompt. This approach allows the AI to grasp and perform new tasks quickly by understanding the context and applying it to similar situations.

Help me classify incoming customer [feedback] as Positive, Negative, or Neutral. Here are three examples of customer feedback and how I would classify them.

1. 'I love this product; it has changed my life for the better!'
Sentiment: Positive
2. 'I am not happy with my purchase. It stopped working after one day.'
Sentiment: Negative
3. 'The product is okay, not what I expected but it does the job.'
Sentiment: Neutral

The examples help AI analyze and identify patterns to make informed decisions about new, similar tasks.

Examples

Tips

- Provide clear, **high-quality and diverse examples**, similar examples don't help the model generalize a concept.
- Ensure the examples are **closely related** to the task at hand, unrelated examples don't guide the model.
- Use **concise** and unambiguous language to help the AI understand and generalize from those examples to new data.

Section delimiting

Section boundaries or delimiters can provide task segmentation, reduce ambiguity, efficient data handling, execution sequencing and response structuring. There are several delimiting options to choose from e.g., bullet points for highlighting key points to the AI, numbered list to give step by step instructions, line breaker with distinct marking to separate out important sections.

```
Task: Help me write a Market Analysis Report
Sections Required:
1. Executive Summary - Summarize overall sales trends and performance.
2. Product Performance - List the top 5 performing products and their sales figures.
3. Customer Feedback - Highlight common feedback themes from customer surveys.
4. Recommendations - Provide three actionable steps based on the analysis.
Data Source: [Last Quarter Sales Data.]
Output Format: Structured Report with Bullet Points
Tone: **Professional** and Concise
```

This section uses multi-delimiting strategy. Note that each important section is identifies itself and hence convey clearly the purpose. The number list give specific directions on expectations and in what sequence they should be generated

Section delimiting

Tips

- **Explicit separators:** There are many possible explicit section delimiter options like '----' or '####'
- **XML like markers:** Marks clearly beginning and ending of the section. Start-and-end marking will never be ambiguous with this approach. E.g.
  <sectionname>
  Instruction related to the section goes in here.
  </sectionname>
- **Spotlighting:** if you want the AI to really take note of an instruction when enclose it in double asterisk '**'.

Chain of Thought

Chain-of-thought (COT) prompting is about leading the AI through a step-by-step reasoning process to solve a problem or answer a question. This approach mirrors human problem-solving, where each step builds on the previous one. It helps the AI break down the task into logical sequences that leads to improved response.

Given a customer [complaint] about a late delivery.
- First, identify possible reasons for the delay based on our [shipping policy].
- Next, outline appropriate responses we can offer the customer, including compensation options.
- Lastly, draft a final response incorporating these elements.

The prompt provides a step-by-step guidance on how to break down the problem and produce a comprehensive answer just like a human would do.

Chain-of-Thought

Tips

- **Outline steps:** Pose each step as a question or statement that leads to the next part of the process.
- **Leading approach:** Ensure the prompts flow in a logical order, guiding the AI towards the final answer.
- **Encourage Transparency:** Ask the AI to show its work, providing a traceable and auditable path to the final response or decision.
- COT phrases: Some of the other self reflection COT phrases have been seen to product superior results e.g., “Let’s think step by step” or "Consider each aspect one by one.“ or "Firstly, secondly, finally..."

AI Builder | Prompt Builder

A Prompt engineering guide

Get the most out of your prompts and avoid common pitfalls by learning what to do and what not to do when writing prompts.

Do’s

✓ Be clear and specific.

Provide specific instructions, about the task to be performed, explain the data context, and output requirements.

✓ Human oversight.

Human agency should be preserved. Ensure that humans have opportunity to review the AI response, in line with goals of the scenario.

✓ Give examples.

Use high-quality and diverse examples to guide the AI to generate more relevant and accurate responses.

✓ Define output format.

Help guide the AI to generate the output that is aligned with your goals and your postprocessing steps of the response.

✓ Test and iterate.

Prompt engineering is an iterative process. First draft of the prompt might not result in the desired outcome. Hence Prompt Builder UX support fast experimentation with prompts.

✓ Check for accuracy.

Occasionally, AI may make mistakes. Always check AI responses for accuracy, grammar, and style, and watch out for irrelevant or inappropriate content.

✓ Transparency.

Create transparency in you end user experience, to inform that results might be AI generate and hence can have inaccuracies or bias.

✓ Adhere to terms of use.

Use of Microsoft generative AI technology should in compliance with Microsoft’s of code of conduct and service terms of usage.

✓ Incorporate feedback.

If an output isn't quite right, tweak your prompt and try again, rather than starting from scratch.

✓ LLM Drift.

Be aware of LLM Drift. Over a period, your AI output quality can degrade, because your incoming data has drifted. Update your prompt or add new examples to align the AI response.

Don’ts

✗ Be vague.

When prompting, avoid using vague language, and be as clear as possible to receive better-quality responses.

✗ Request inappropriate or unethical content.

AI Builder is not responsible for the content or the consequences of your writing. You should respect local laws, rules, the rights of others and term of service.

✗ Use slang, jargon, or informal language.

This may cause AI to give low-quality, inappropriate or unprofessional responses.

✗ Give conflicting instructions.

Prompts with conflicting pieces of information in the same request can confuse the AI and result in lower quality responses.

✗ Overcomplication.

Simplify your prompts to ensure clarity for the AI, focusing on single-task capabilities. For complex, multi-step tasks, consider creating a Power Automate Flow instead for better results.

✗ Assume Prior Knowledge.

Avoid assuming the AI has knowledge outside the given prompt, this gives room for hallucination. Always give context and set expectations.

December 2023

AI Builder | Prompt Builder

A Prompt engineering guide

Creating safe and ethical AI solutions is a collective responsibility. We play our part, and we count on you to do yours.

Responsible AI

✓ Fairness.

AI systems should treat all people fairly. This involves ensuring that your prompts do not discriminate against or express bias toward any group or individual based on gender, race, sexual orientation, or religion.

"Analyze these job applications and rank the candidates based on their qualifications, ensuring the analysis is free from gender, race, or age biases."

✓ Privacy and Security.

Protecting the data used in AI systems is paramount. When designing prompts, it's important to ensure that they do not compromise individual privacy or security.

"Summarize the patient health trends from this anonymized dataset without revealing any personal or sensitive information."

✓ Transparency.

AI systems must be clear and their decision-making process transparent. As a developer, it's crucial to ensure that users understand how and why AI is used in generating responses, decisions, or content.

✓ Reliable and Safety.

AI systems need to be reliable and safe. This means that prompts should be designed to produce consistent and safe outcomes, and AI systems should be rigorously tested to handle edge cases and new situations safely.

"Generate an article on the features of our new product, ensuring that all claims are based on verified data and tested results."

✓ Inclusiveness.

AI should be inclusive of all human races and experiences. This principle encourages the use of inclusive design practices in prompt engineering, ensuring that prompts do not unintentionally exclude people.

"Create an overview of our website's content, suggesting changes to make it more accessible to users with visual and hearing impairments."

✓ Accountability.

People should be accountable for AI systems. This principle emphasizes the need for oversight and governance in the development and ensuring that there is human accountability for the outputs generated by AI.

Microsoft Responsible AI Principles [Documentation]

Azure OpenAI Service [Code of Conduct]

Read Service [Terms of Use]

December 2023