

Prompt engineering techniques

These techniques aren't recommended for reasoning models like gpt-5 and o-series models.

Prompt construction can be difficult. In practice, the prompt acts assist the model complete the desired task, but it's more of an art than a science, often requiring experience and intuition to craft a successful prompt. The goal of this article is to help get you started with this learning process. It attempts to capture general concepts and patterns that apply to all GPT models. However it's important to understand that each model behaves differently, so the learnings might not apply equally to all models.

Basics

This section covers the basic concepts and elements of GPT prompts.

Text prompts are how users interact with GPT models. As with all generative language models, GPT models attempt to produce the next series of words that are most likely to follow from the previous text. It's as if we're saying *What is the first thing that comes to your mind when I say <prompt>*? The examples below demonstrate this behavior. Given the first words of famous content, the model is able to accurately continue the text.

[+] Expand table

Prompt	Completion
Four score and seven years ago our	fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal. [...]
"Call me Ishmael."	"Some years ago—never mind how long precisely—having little or no money in my purse, and nothing particular to interest me on shore, I thought I would sail about a little and see the watery part of the world." [...]

As you develop more complex prompts, it's helpful to keep this fundamental behavior in mind. Regardless of the prompt that is provided, the model is simply responding with what it determines is most likely (given its training data and training targets). If you ask a question in your prompt, the model isn't following a separate "Q&A" code path, but rather it appears to answer the question because an answer is the most likely sort of response for the given question as input.

Prompt components

When using the Completion API while there's no differentiation between different parts of the prompt, it can still be useful for learning and discussion to identify underlying prompt components. With the Chat Completion API there are distinct sections of the prompt that are sent to the API in the form of an array of dictionaries with associated roles: system, user, and assistant. This guidance focuses more generally on how to think about prompt construction rather than providing prescriptive guidance that is specific to one API over another.

It's also important to understand that while there could be other valid ways to dissect prompts, the goal of this breakdown is to provide a relatively simple way to think about prompt construction. With the Completion API, all the components are optional, but at least one must be present and most prompts include more than one component. There can be some grey area between components as well. The order presented below roughly corresponds to how commonly each component is used, from most to least.

Instructions

Instructions are likely the most commonly used prompt component. Instructions are simply that: instructions to the model on what to do. While simple in concept, they can be complex in practice. The following table illustrates this by showing both simple and more complex instructions, in two different forms.

Expand table

Instruction Type	Prompt	Completion
Simple instructions	Write an introduction for a weekly newsletter.	Welcome to our weekly newsletter. This week we have a special focus on productivity. We'll be sharing tips and tricks on how to be productive in all areas of your life. We hope you enjoy and please share with your friends!
Complex instructions	Write an introduction for a weekly newsletter to my company named Contoso. It should mention the great all hands meeting we had last week and thank the team for all their hard work over the tough past few months.	Welcome to Contoso's weekly newsletter! We had a great all hands meeting last week and I want to thank the team for all their hard work over the tough past few months. We're looking forward to a bright future and I'm confident that with our team's dedication and commitment, we'll achieve great things. Thank you for being a part of our team and I look forward to working with you all in the coming weeks and months. Sincerely, [Your name].
Complex Instructions	Write an introduction for a weekly newsletter, including the following: - My company is named	Hello Contoso team! I wanted to take a moment to thank you for all your hard work over the past few months. It's been a tough couple of months, but you've all pulled together and done an amazing job. I'm really

Instruction	Prompt	Completion
Type		
contoso	<ul style="list-style-type: none"> - An exuberant greeting - Thanking the team for their hard work over a couple tough months - Positive outlook for the coming quarter - Signed by the SLT 	proud of you all. Looking ahead to the next quarter, I'm feeling really positive. I think we're going to achieve some great things. Thanks again, The Senior Leadership Team.

Primary Content

Primary content refers to some sort of text that is being processed or transformed by the model. Primary content is typically used in conjunction with instructions. A simple example would be language translation. In the example below, the English text would be considered the primary content while "Translate to French:" is an instruction.

[] Expand table

Prompt	Completion
Can you please tell me how to get to the museum?	Pouvez-vous s'il vous plaît me dire comment aller au musée?
Translate to French:	

Primary content can also be much longer. In the following example, the primary content is the introduction section from the Wikipedia entry on time complexity , which is almost 300 words long. This content is abbreviated in our table for ease of display.

[] Expand table

Prompt	Completion
In computer science, the time complexity is the computational complexity that describes the amount of computer time it takes to run an algorithm. Time complexity is commonly estimated by [...] Summarize the previous text in a few sentences:	Time complexity is a measure of how long an algorithm takes to run. It's typically expressed as a function of the size of the input. The most common measure is worst-case time complexity, which is the maximum amount of time required for inputs of a given size.
In computer science, the time complexity is the computational	The time complexity of an algorithm is the amount of time it takes to run an algorithm. Time complexity is commonly

Prompt	Completion
complexity that describes the amount of computer time it takes to run an algorithm. Time complexity is commonly estimated by [...] TL;DR (Too long didn't read)	estimated by counting the number of elementary operations performed by the algorithm, supposing that each elementary operation takes a fixed amount of time to perform. The time complexity is generally expressed as a function of the size of the input.

GPT models can also handle primary content that is structured. In the example below, a TSV about beers in Belgium (abbreviated for display in this article) is passed to the model as part of the prompt. It's capable of properly interpreting the content and answering questions about the data.

[+] [Expand table](#)

Prompt	Completion
Belgian Beer Brands Beer name Beer style color ABV Brewery "Chimay Gold: ""Doree"" trappist pale ale gold 4.80% Chimay Brewery (official trappist) "Chimay Blue: ""Grande Reserve"" trappist dark ale dark 9.00% [...] Referencing the table above, how many beers are less than 6% ABV?	There are 10 beers in the table that are less than 6% ABV.

Examples

Successful prompts often rely on the practice of “one-shot” or “few-shot” learning. This refers to the inclusion of one or more examples of the desired behavior of the model, typically by including input and output pairs. This isn't learning in the sense that the model is permanently changed, but rather that the examples better condition the model to respond as desired for only the current inference. The use of prompts with no examples is sometimes referred to as “zero-shot” learning. Note that with the Chat Completion API few-shot learning examples are typically added to the messages array in the form of example user/assistant interactions after the initial system message.

[+] [Expand table](#)

Learning Type	Prompt	Completion
Zero-shot	Headline: Coach confident injury won't	The coach is confident that the injury won't derail the Warriors' season. The team is still focused on their goals and that they'll

Learning Type	Prompt	Completion
	derail Warriors Topic:	continue to work hard to achieve them.
Few-shot	Headline: Twins' Correa to use opt-out, test free agency Topic: Baseball Headline: Qatar World Cup to have zones for sobering up Topic: Soccer Headline: Yates: Fantasy football intel for Week 6 Topic: Football Headline: Coach confident injury won't derail Warriors Topic:	Basketball

The example above illustrates the utility of few-shot learning. Without the examples, the model seems to be guessing at the desired behavior, while the examples cleanly show the model how to operate. This also demonstrates the power of the model, it can infer the category of label that is wanted, even without a "basketball" label in the examples.

Cue

Cues act as the "jumpstart" for the output of the model, helping to direct the model to the desired output. It's often a prefix that the model can build onto. In the example below, we show the cue directing the story to include key elements. Cues are often used along with instructions, but not always. In the examples below, the model is being used to summarize a long announcement email. A cue is used in one case to help focus the output of the model and to suggest a specific output format (bullet points) in the other.

In the context of the Chat Completion API instructions take the form of the system message, and few-shot examples are indicated by providing a series of example user/assistant interactions to help prime the model.

[] Expand table

Number of Cues	Prompt	Completion
0	<p>Performance reviews are a core component of our Performance & Development approach at Contoso. [...]</p> <p>Summarize the above email message:</p>	<p>The purpose of this email is to inform employees of recent changes to the performance review experience, which is part of Contoso's Performance & Development approach. The changes include an improved structure, upgraded in-tool user experience, and focused prompts. Additionally, there's now an optional Conversation Starters section for employees to flag topics they'd like to discuss with their managers.</p>
1	<p>Performance reviews are a core component of our Performance & Development approach at Contoso. [...]</p> <p>Summarize the above email message:</p> <p>The key takeaway of the message is</p>	<p>that Contoso is improving the performance review experience by making it more focused on core priorities, expectations, and impact. Additionally, they're upgrading the in-tool user experience with tips, resources, and guidance. Finally, they're adding guidance on how to activate company-wide core priorities.</p>
2	<p>Performance reviews are a core component of our Performance & Development approach at Contoso. [...]</p> <p>Summarize the above email message:</p> <p>Key Points:</p> <ul style="list-style-type: none"> • 	<p>Contoso is introducing changes to the performance review experience, based on feedback from employees.</p> <ul style="list-style-type: none"> • The changes include a new focus on core priorities, expectations, and impact; an upgraded user experience; and new guidance on company-wide core priorities. • Performance reviews are a time to reflect on past performance and learning as well as to plan for future priorities and growth opportunities. • Employees and managers should write comments in advance of the performance review meeting. • There should be at least two performance reviews per year. • The time investment for each review should remain about the same.

Supporting content

Supporting content is information that the model can utilize to influence the output in some way. It differs from primary content in that it isn't the main target of task, but it's typically used along with primary content. Common examples include contextual information such as the current date, the name of the user, user preferences, and so on. The example below uses supporting content to help organize a set of planned workshops for the user. Without the supporting (important topics) the model simply lists off the workshops (truncated for display), when informed of my important topics, the model is able to accurately group the meetings.

[] Expand table

Supporting Content	Prompt	Completion
Without	<p>Workshop list:</p> <p>Title Description</p> <p>Planning Workshop Kickoff Please join us to kick off our workshop! Hear from our leaders about cross group priorities and learnings for the year ahead. The invitation to this event has already been sent to everyone in [...]</p> <p>Summarize the workshops above, grouped by topic:</p>	<p>Workshop list:</p> <p>Planning</p> <p>Workshop Kickoff</p> <p>-Planning</p> <p>Workshop Wrap-up</p> <p>- Core MeetUp</p> <p>- Customer Driven Perspectives</p> <p>-Impact and Key Focus Areas</p> <p>-Best practices</p> <p>-Accelerating app adoption [...]</p>
With	<p>Workshop list:</p> <p>Title Description</p> <p>Planning Workshop Kickoff Please join us to kick off our workshop! Hear from our leaders about cross group priorities and learnings for the year ahead. The invitation to this event has already been sent to everyone in [...]</p> <p>My Important Topics: prompt engineering, search, GPT models</p> <p>Summarize the workshops above, grouped by my important topics:</p>	<p>1. Prompt Engineering:</p> <ul style="list-style-type: none"> - New prompt engineering tips - The art of prompt engineering <p>2. Search:</p> <ul style="list-style-type: none"> -Intro to vector search with embedding <p>3. GPT Models:</p> <ul style="list-style-type: none"> - Intro to GPT-4 - GPT-35-Turbo in-depth.

Scenario-specific guidance

The techniques in this section will teach you strategies for increasing the accuracy and grounding of responses you generate with a Large Language Model (LLM). It is, however, important to remember that even when using prompt engineering effectively you still need to validate the responses the models generate. Just because a carefully crafted prompt worked well for a particular scenario doesn't necessarily mean it will generalize more broadly to certain use cases. Understanding the limitations of LLMs, is just as important as understanding how to leverage their strengths.

This guide doesn't go in-depth into the mechanics behind the message structure for Chat Completions. If you aren't familiar with interacting with Chat Completions models programmatically, we recommend reading our how-to guide on the Chat Completion API first.

(!) Note

All of the examples in this section of the guide were tested against a base GPT-4 model in English. Some techniques may produce different results with newer models like gpt-4o, and gpt 4.5. If you are reading a localized version of this article in another language, these responses represent a localized translation of the English results. To learn more about potential limitations depending on what language you are using to prompt a model, please consult our [Responsible AI transparency note](#).

Few-shot learning

A common way to adapt language models to new tasks is to use few-shot learning. In few-shot learning, a set of training examples is provided as part of the prompt to give additional context to the model.

When using the Chat Completions API, a series of messages between the User and Assistant (written in the new prompt format), can serve as examples for few-shot learning. These examples can be used to prime the model to respond in a certain way, emulate particular behaviors, and seed answers to common questions.

 Expand table

System message	Few-shot examples
Assistant is an intelligent chatbot designed to help users answer their tax related questions.	User - "When do I need to file my taxes by?" Assistant - "In 2023, you'll need to file your taxes by April 18th. The date falls after the usual April 15th deadline because April 15th falls on a Saturday in 2023. For more details, see https://www.irs.gov/filing/individuals/when-to-file "
Instructions - Only answer questions related to taxes. - If you're unsure of an answer, you can say "I don't know" or "I'm not sure" and recommend users go to the IRS website for more information.	User - "How can I check the status of my tax refund?" Assistant - "You can check the status of your tax refund by visiting https://www.irs.gov/refunds "

The table above describes the basic prompt structure, but for a detailed explanation of the exact prompt format you'll need to refer to the Chat Completion guide.

Non chat scenarios

While the Chat Completion API is optimized to work with multi-turn conversations, it also can be used for non chat scenarios. For example, for a sentiment analysis scenario, you might use the following prompt:

[+] Expand table

System message	User	Assistant
You're an assistant designed to analyze sentiment from speech data. Users will paste in a string of text and you'll respond with an assessment of the response of the speaker. Rate on a scale of 1-10 (10 being highest). Explain why this rating was given.	hi thank you for calling contoso pharmacy who am i speaking with today i understand your frustration mary um can you tell me what exactly you're trying to accomplish OK uh certainly happy to check that for you one moment please i see here that you were on a generic form of lipitor is that right OK uh so i see that your doctor stopped prescribing these drugs in twenty twenty one actually OK uh yeah i'd be happy to check that for you uh because sometimes there's a gap in the system and it just doesn't get reported so let me take a look here just a moment so i'm seeing here that your doctor had these drugs prescribed to you from twenty twelve through twenty twenty one OK um yeah according to the latest records uh provided by doctor's office you're now on a different drug metformin would you like us to go ahead and fill that prescription for you for pick up a bit of form in five hundred milligrams uh you're very welcome mary please let us know if there's anything else we can do for you today.	Sentiment rating: 8 The speaker in this text seems to be polite, helpful, and patient while addressing the concerns of the customer. They take the time to check the information, provide clear explanations, and offer assistance. However, the rating isn't a 10 because there's still a hint of uncertainty in some parts of the conversation, which might not convey complete confidence to the customer.

Start with clear instructions

The sequence information appears in the prompt matters. This is because GPT style models are built in a certain way that defines how they process the input. Our research suggests that telling the model the task you want it to do at the beginning of the prompt, before sharing additional contextual information or examples, can help produce higher-quality outputs.

(!) Note

Although following this technique is still generally recommended, in contrast to previous model versions (GPT-3 and prior), our testing showed that the model response with ChatGPT and GPT-4 models was the same regardless of whether the technique is utilized. In the case of the below example, we see that adding the statement "Several sources ...

"eruption" at the beginning or the end of the prompt did not cause any change in the final model response.

[+] Expand table

System message	User	Assistant
You're an AI assistant that helps people find information.	Your task is to verify if the statement "Several sources mention a chance of another large eruption" is supported by a specific quote from the following set of snippets. --- SNIPPETS [1] 14 percent chance of megaquake hitting Seattle, experts say SEATTLE - There's a 14 percent chance of a magnitude 9 Cascadia earthquake hitting Seattle in the next 50 years, the U.S. Geological Survey estimates. "Unfortunately, we're unable to... [2] Earthquake experts lay out latest outlook for Seattle's 'Really Big One' "We say that there's approximately a 14% chance of another approximately magnitude-9 earthquake occurring in the next 50 years," said a geophysicist at the University of Washington... --- Is the statement "Several sources mention a chance of another large eruption" directly implied or stated by the snippets?	No, the statement isn't directly implied or stated by the snippets. The snippets mention a chance of a "megaquake" and a "magnitude 9 Cascadia earthquake" hitting Seattle in the next 50 years, but don't mention a chance of another large eruption.

Repeat instructions at the end

Models can be susceptible to **recency bias**, which in this context means that information at the end of the prompt might have more significant influence over the output than information at the beginning of the prompt. Therefore, it's worth experimenting with repeating the instructions at the end of the prompt and evaluating the impact on the generated response.

Prime the output

This refers to including a few words or phrases at the end of the prompt to obtain a model response that follows the desired form. For example, using a cue such as “Here’s a bulleted list of key points:\n- ” can help make sure the output is formatted as a list of bullet points.

 Expand table

System message	User	Assistant
You're an AI assistant that helps people find information. Answer in as few words as possible.	John Smith is married to Lucy Smith. They have five kids, and he works as a software engineer at Microsoft. What search queries should I do to fact-check this? ## One possible search query is:	"John Smith married Lucy Smith five kids software engineer Microsoft"

In the above prompt, the text *One possible search query is:* primes the model to produce a single output. Without this cue the model produces several search queries as output.

Add clear syntax

Using clear syntax for your prompt—including punctuation, headings, and section markers—helps communicate intent and often makes outputs easier to parse.

In the example below, separators (--- in this case) have been added between different sources of information or steps. This allows the use of --- as a stopping condition for generation. In addition, section headings or special variables are presented in uppercase to differentiate them.

 Expand table

System message	User	Assistant
You're an AI assistant that helps people find information.	You'll read a paragraph, and then issue queries to a search engine in order to fact-check it. Also explain the queries. --- PARAGRAPH John Smith is married to Lucy Smith. They have five kids, and he works as a software engineer at Microsoft. What search queries should I do to fact-check this? --- QUERIES	1. "John Smith Microsoft" - To check if John Smith is indeed employed at Microsoft. 2. "John Smith Lucy Smith" - To check if John Smith is married to Lucy Smith. 3. "John Smith children" - To check if John Smith has five children.

If you're not sure what syntax to use, consider using Markdown or XML. The models have been trained on a large quantity web content in XML and Markdown, which might provide better results.

Break the task down

Large language models (LLMs) often perform better if the task is broken down into smaller steps. For example, in the search query prompt referenced earlier, the prompt can be restructured so that the model is first instructed to extract relevant facts, and then instructed to generate search queries that can be used to verify those facts.

 Expand table

System message	User	Assistant
You're an AI assistant that helps people find information.	You'll read a paragraph, and then issue queries to a search engine in order to fact-check it. --- PARAGRAPH John Smith is married to Lucy Smith. They have five kids, and he works as a software engineer at Microsoft. What search queries should I do to fact-check this? --- Now you'll extract factual claims, and issue queries to fact-check them. When issuing a query, use the function SEARCH("query") FACTUAL CLAIMS	1. John Smith is married to Lucy Smith 2. They have five kids 3. He works as a software engineer at Microsoft QUERIES SEARCH("John Smith married Lucy Smith") SEARCH("John Smith family size") SEARCH("John Smith Microsoft software engineer")

Notice the use of clear syntax to differentiate the sections and prime the output. In this simple example, breaking the task down from one to two steps isn't very dramatic, but when trying to do this for a larger piece of text with many factual claims, breaking the task down can make a significant difference.

Use of affordances

Sometimes we can get the model to use an affordance instead of relying on its own parameters for information and answers. Search, for example, can be an affordance to help mitigate against fabricated answers, and to get up-to-date information.

One simple way to use an affordance is to stop generation once the affordance calls are generated by the model, then paste the outcomes back into the prompt. Below is an example

of what the follow-up call could be after doing the **SEARCH** calls above. Notice how we pasted the search results into the prompt and replaced the previous **SEARCH** calls.

 [Expand table](#)

System message	User	Assistant
You're an AI assistant that helps people find information.	<p>You'll read a paragraph, extract factual claims, and then use search engine results to fact-check them</p> <p>---</p> <p>PARAGRAPH</p> <p>John Smith is married to Lucy Smith. They have five kids, and he works as a software engineer at Microsoft. What search queries should I do to fact-check this?</p> <p>---</p> <p>FACTUAL CLAIMS</p> <ul style="list-style-type: none">- John Smith is married to Lucy Smith- John and Lucy have five kids- John works as a software engineer at Microsoft <p>---</p> <p>Here are various search queries issued to research the claims above:</p> <p>SEARCH QUERIES</p> <ul style="list-style-type: none">- John Smith married to Lucy Smith- John Smith number of children- John Smith software engineer Microsoft <p>---</p> <p>Here are snippets of the search results:</p> <p>SNIPPETS:</p> <ul style="list-style-type: none">[1] ... John Smith's wedding was on September 25, 2012 ...[2] ... John Smith was accompanied by his wife Lucy to a party[3] John was accompanied to the soccer game by his two daughters and three sons[4] ... After spending 10 years at Microsoft, Smith founded his	<p>- John Smith is married to Lucy Smith: Confirmed. Snippet [1] and [2] both mention John Smith's wife as Lucy.</p> <p>- John and Lucy have five kids: Confirmed. Snippet [3] mentions that John was accompanied by his two daughters and three sons.</p> <p>- John works as a software engineer at Microsoft: Partially confirmed. Snippet [4] mentions that John spent 10 years at Microsoft, but it doesn't explicitly state that he currently works there. Further research might be needed to confirm his current employment status.</p>

System message	User	Assistant
	<p>own startup, Tailspin Toys</p> <p>[5] John M is the town smith, and he married Fiona. They have a daughter named Lucy</p> <p>---</p> <p>Given the snippets, fact check each of the factual claims above:</p>	

Chain of thought prompting

ⓘ Important

This technique is only applicable to non-reasoning models. Attempting to extract model reasoning through methods other than the reasoning summary parameter are not supported, may violate the Acceptable Use Policy, and may result in throttling or suspension when detected.

This is a variation on the **break the task down** technique. Instead of splitting a task into smaller steps, in this approach, the model response is instructed to proceed step-by-step and present all the steps involved. Doing so reduces the possibility of inaccuracy of outcomes and makes assessing the model response easier.

 Expand table

System message	User	Assistant
You're an AI assistant that helps people find information.	<p>Who was the most decorated (maximum medals) individual athlete in the Olympic games that were held at Sydney? Take a step-by-step approach in your response, cite sources and give reasoning before sharing final answer in the below format: ANSWER is: <name></p>	<p>Step 1: Determine the year of the Sydney Olympic games. The Sydney Olympic games were held in the year 2000.</p> <p>Step 2: Find a list of individual athletes who won multiple medals in the 2000 Sydney Olympics.</p> <p>According to the official website of the Olympic Games, there were several individual athletes who won multiple medals in the 2000 Sydney Olympics. Some of the notable names include:</p> <ul style="list-style-type: none"> • Ian Thorpe (Australia) - 3 gold, 2 silver, 0 bronze • Pieter van den Hoogenband (Netherlands)

System message	User	Assistant
		<ul style="list-style-type: none"> - 2 gold, 0 silver, 2 bronze • Lenny Krayzelburg (USA) - 3 gold, 0 silver, 0 bronze • Inge de Bruijn (Netherlands) - 3 gold, 1 silver, 0 bronze <p>Step 3: Determine the athlete with the most medals. After reviewing the list of individual athletes who won multiple medals in the 2000 Sydney Olympics, we can see that the athlete with the most medals is Ian Thorpe from Australia, who won a total of 5 medals (3 gold and 2 silver).</p> <p>Therefore, the ANSWER is: Ian Thorpe.</p>

Specifying the output structure

Using your prompt to specify the structure of the output can have a significant effect on the nature and quality of the results. Sometimes system message inputs as “only write true facts” or “don’t fabricate information” might not be adequate mitigations. Instead, asking for the model response to also include citations can help result in a reduction of the prevalence of incorrect responses.

If you instruct the model to cite the source material when it makes statements, those statements are much more likely to be grounded. Asking for citations makes it so that the model must make two errors every time it generates a response: the first error is the fabricated response, and the second is the bad citation. Note that the closer the citation is to the text it supports, the shorter the distance ahead the model needs to anticipate the citation, which suggests that inline citations are better mitigations for false content generation than citations at the end of the content.

Similarly, if you ask the model to extract factual statements from a paragraph, it might extract compound statements such as ‘X is doing Y AND Z’ (which can be harder to verify). This can be avoided by specifying an output structure such as (entity1, relationship, entity2).

The examples below show the use of citations and directing the model response to fit a defined structure.

System message	User	Assistant
<p>You're an AI assistant that helps people find information</p>	<p>List down the different behaviors exhibited by the Customer Support agent in the following conversation.</p> <p>---</p> <p>hi thank you for calling contoso pharmacy who am i speaking with today i understand your frustration mary um can you tell me what exactly you're trying to accomplish OK uh certainly happy to check that for you one moment please i see here that you were on a generic form of lipitor is that right OK uh so i see that your doctor stopped prescribing these drugs in twenty twenty one actually OK uh yeah i'd be happy to check that for you uh because sometimes there's a gap in the system and it just doesn't get reported so let me take a look here just a moment so i'm seeing here that your doctor had these drugs prescribed to you from twenty twelve through twenty twenty one OK um yeah according to the latest records uh provided by doctor's office you're now on a different drug metformin would you like us to go ahead and fill that prescription for you for pick up a bit of form in five hundred milligrams uh you're very welcome mary please let us know if there's anything else we can do for you today.</p> <p>---</p> <p>How was the response of the Customer Support agent? Classify and list down their various behaviors and cite the reason behind the classification, using the format BEHAVIOR("reason")</p>	<p>1. Polite Greeting BEHAVIOR("Agent starts the conversation with a polite greeting and asks for the caller's name")</p> <p>2. Empathetic BEHAVIOR("Agent acknowledges Mary's frustration and tries to understand her concern")</p> <p>3. Attentive BEHAVIOR("Agent listens to the customer's problem and asks for more details")</p> <p>4. Proactive BEHAVIOR("Agent offers to check the issue for the customer")</p> <p>5. Informative BEHAVIOR("Agent provides the relevant information about the customer's prescription history")</p> <p>6. Thorough BEHAVIOR("Agent checks for any possible gaps in the system and provides a complete response")</p> <p>7. Confirmatory BEHAVIOR("Agent confirms the current medication with the customer")</p> <p>8. Solution-oriented BEHAVIOR("Agent offers to fill the new prescription for the customer")</p>

System message	User	Assistant
		9. Courteous BEHAVIOR("Agent thanks the customer and offers further assistance if needed")

[] Expand table

System message	User	Assistant
You're an AI assistant that helps people find information.	<p>Cluster the following news headlines into topic categories based on patterns seen within the text. Also mention reasoning behind how these categories were defined.</p> <p>Output format:</p> <pre>{ "TOPIC_NAME": "", "HEADLINES": [], "REASONING": "" }</pre> <p>Input news headlines:</p> <ol style="list-style-type: none"> 1. "From books to presentations in 10s with AR + ML" 2. "Demo from 1993 of 32-year-old Yann LeCun showing off the World's first Convolutional Network for Text Recognition" 3. "First Order Motion Model applied to animate paintings" 4. "Robinhood and other brokers literally blocking purchase of \$GME, \$NOK, \$BB, \$AMC; allow sells" 5. "United Airlines stock down over 5% premarket trading" <p>Output:</p>	<pre>{ "ARTIFICIAL_INTELLIGENCE": { "HEADLINES": ["From books to presentations in 10s with AR + ML", "Demo from 1993 of 32-year-old Yann LeCun showing off the World's first Convolutional Network for Text Recognition", "First Order Motion Model applied to animate paintings"], "REASONING": "These headlines are related to artificial intelligence, machine learning, and their applications in various fields." }, "FINANCIAL_MARKETS": { "HEADLINES": ["Robinhood and other brokers literally blocking purchase of \$GME, \$NOK, \$BB, \$AMC; allow sells", "United Airlines stock down over 5% premarket trading", "Bitcoin was nearly \$20,000 a year ago today"], "REASONING": "These headlines are related to financial markets, stocks, cryptocurrencies, and trading platforms." } }</pre>

Temperature and Top_p parameters

Changing the temperature parameter changes the output of the model. The temperature parameter can be set between 0 and 2. A higher value, for example 0.7 makes the output more random and produce more divergent responses, while a lower value, like 0.2, makes the output more focused and concrete. A fictional story could be generated using a higher temperature. Whereas to generate a legal document it's recommended to use a much lower temperature. Top_probability is another parameter and is similar to Temperature in that it also controls the randomness of the model response, but it does so in a different way. The general recommendation is to alter one of these two parameters at a time, not both.

Provide grounding context

One of the most effective ways to provide reliable answers is to give the model data to draw its responses from (grounding data). If your use case relies on up-to-date, reliable information and isn't purely a creative scenario, we strongly recommend providing grounding data. In general, the closer you can get your source material to the final form of the answer you want, the less work the model needs to do, which means there's less opportunity for error. Here's an example where the system is provided a recent blog describing the launch of GPT-4 in Azure OpenAI in Azure AI Foundry Models [,](#) and asked to name some of its early customers.

[] Expand table

System message	User	Assistant
<p>You're an AI assistant that helps people find information. You'll provide answers exclusively from below text and respond in 100 words or less:</p> <p>"At Microsoft, we're constantly discovering new ways to unleash creativity, unlock productivity, and up level skills so that more people can benefit from using AI. This is allowing our customers to ...</p> <p>... thereby reducing the likelihood that the model generates a harmful response."¹</p> <p>Answer in 100 words or less. Use bullet lists wherever possible.</p>	<p>Who are some of the early customers of the Azure OpenAI GPT-4 models?</p>	<p>Here are some of the early customers of the Azure OpenAI GPT-4 models:</p> <ul style="list-style-type: none">- Epic Healthcare: They plan to use GPT-4 to help physicians and nurses spend less time on data entry and investigation.- Coursera: They're using Azure OpenAI with GPT-4 to create a new AI-powered learning experience on their platform.- Coca-Cola: They're using Azure OpenAI to build a knowledge hub and plan to leverage GPT-4's multimodal capabilities for marketing, advertising, public relations, and customer relations.- Other companies of all sizes are using Azure AI to improve customer experiences, summarize long-form content, help write software, and

System message	User	Assistant
		reduce risk by predicting the right tax data.

¹ Text from Azure OpenAI GPT-4 launch blog [.](#)

Best practices

- **Be Specific.** Leave as little to interpretation as possible. Restrict the operational space.
- **Be Descriptive.** Use analogies.
- **Double Down.** Sometimes you might need to repeat yourself to the model. Give instructions before and after your primary content, use an instruction and a cue, etc.
- **Order Matters.** The order in which you present information to the model might impact the output. Whether you put instructions before your content ("summarize the following...") or after ("summarize the above...") can make a difference in output. Even the order of few-shot examples can matter. This is referred to as recency bias.
- **Give the model an "out".** It can sometimes be helpful to give the model an alternative path if it's unable to complete the assigned task. For example, when asking a question over a piece of text you might include something like "respond with "not found" if the answer isn't present." This can help the model avoid generating false responses.

Space efficiency

While the input size increases with each new generation of GPT models, there will continue to be scenarios that provide more data than the model can handle. GPT models break words into "tokens." While common multi-syllable words are often a single token, less common words are broken in syllables. Tokens can sometimes be counter-intuitive, as shown by the example below which demonstrates token boundaries for different date formats. In this case, spelling out the entire month is more space efficient than a fully numeric date.

Recent work has demonstrated substantial gains on many NLP tasks and benchmarks by pre-training on a large corpus of text followed by fine-tuning on a specific task. While typically task-agnostic in architecture, this method still requires task-specific fine-tuning datasets of thousands or tens of thousands of examples.	October, 18th 2022 October 18 2022 2022/10/18 10-18-2022 10-18-22	
---	---	--

Given this limited space, it's important to use it as efficiently as possible.

- **Tables** – As shown in the examples in the previous section, GPT models can understand tabular formatted data quite easily. This can be a space efficient way to include data, rather than preceding every field with name (such as with JSON).

- White Space – Consecutive whitespaces are treated as separate tokens, which can be an easy way to waste space. Spaces preceding a word, on the other hand, are typically treated as part of the same token as the word. Carefully watch your usage of whitespace and don't use punctuation when a space alone will do.

Related content

- Learn more about Azure OpenAI.
 - Get started with the ChatGPT model with the ChatGPT quickstart.
 - For more examples, check out the Azure OpenAI Samples GitHub repository
-

Last updated on 09/30/2025