



[Home](#) > [Blog](#) > [Artificial Intelligence \(AI\)](#)

# What is Prompt Engineering? A Detailed Guide

Explore the world of prompt engineering with this essential guide. Understand its significance in AI, the role of a prompt engineer, and the future prospects.

Aug 2023 · 19 min read




**Matt Crabtree**

A writer and content editor in the edtech space. Committed to exploring data trends.

## TOPICS

Artificial Intelligence (AI)





The way we interact with technology is constantly evolving. One of the most exciting recent advancements is in the realm of artificial intelligence (AI), where machines are trained to think, learn, and even communicate like humans. Among the myriad of developments in areas such as generative AI, there's a subtle art that's gaining prominence: prompt engineering.

Imagine having a conversation with a machine where you provide a cue or a "prompt," and it responds with relevant information or actions. That's the essence of prompt engineering. It's about crafting the right questions or instructions to guide AI models, especially Large Language Models (LLMs), to produce desired outcomes. Whether you're a tech enthusiast curious about the latest in AI or a professional looking to harness the power of language models, understanding prompt engineering is crucial.

As we journey through this article, we'll demystify the technical intricacies of prompt engineering while also providing a view of its significance in the broader AI landscape. And for those who wish to dive deeper into the world of AI and language processing, we've included a range of resources to help you learn more.

## What is Prompt Engineering?

At its heart, prompt engineering is akin to teaching a child through questions. Just as a well-phrased question can guide a child's thought process, a well-crafted prompt can steer an AI model, especially a Large Language Model (LLM), towards a specific output. Let's explore this concept in more detail.

### Definition and core concepts

Prompt engineering is the practice of designing and refining prompts—questions or instructions—to elicit specific responses from AI models. Think of it as the interface between human intent and machine output.

In the vast realm of AI, where models are trained on enormous datasets, the right prompt can be the difference between a model understanding your request or misinterpreting it.

For instance, if you've ever interacted with voice assistants like Siri or Alexa, you've engaged in a basic form of prompt engineering. The way you phrase your request—"Play some relaxing music" versus "Play Beethoven's Symphony"—can yield vastly different results.

### The technical side of prompt engineering

Prompt engineering, while rooted in the art of language, is deeply intertwined with the technical intricacies of AI models. Here's a closer look at the technical side:

- **Model architectures.** Large Language Models (LLMs) like [GPT](#) (Generative Pre-trained Transformer) and Google's PaLM2 (Powering [Bard](#)) are built on transformer architectures. These architectures allow models to handle vast amounts of data and understand context through self-attention mechanisms. Crafting effective prompts often requires an understanding of these underlying architectures.
- **Training data and tokenization.** LLMs are trained on vast datasets, tokenizing input data into smaller chunks (tokens) for processing. The choice of tokenization (word-based, byte-pair, etc.) can influence how a model interprets a prompt. For instance, a word tokenized differently might yield varied outputs.
- **Model parameters.** LLMs have millions, if not billions, of parameters. These parameters, fine-tuned during the training process, determine how the model responds to a prompt. Understanding the relationship between these parameters and model outputs can aid in crafting more effective prompts.
- **Temperature and Top-k sampling.** When generating responses, models use techniques like temperature setting and top-k sampling to determine the randomness and diversity of outputs. For instance, a higher temperature might yield more diverse (but potentially less accurate) responses. Prompt engineers often adjust these settings to optimize model outputs.
- **Loss functions and gradients.** At a deeper level, the model's behavior during prompt response is influenced by its loss functions and gradients. These mathematical constructs guide the model's learning process. While prompt engineers don't typically adjust these directly, understanding their impact can provide insights into model behavior.

For those keen on understanding the inner workings of such models, our [tutorial Transformers and Hugging Face](#) offers a deep dive into the mechanics behind popular LLMs.

## Why prompt engineering matters

In an age where AI-driven solutions are becoming ubiquitous—from chatbots in customer service to AI-powered content generators—prompt engineering is the bridge ensuring effective human-AI communication. It's not just about getting the right answer; it's about ensuring AI understands the context, the nuances, and the intent behind every query.

## The evolution of engineering prompts

Prompt engineering, while a relatively recent discipline, is deeply rooted in the broader history of Natural Language Processing (NLP) and machine learning. Understanding its evolution provides context to its current significance.

## The early days of NLP

The origins of NLP date back to the mid-20th century, with the advent of digital computers. Early efforts in NLP were rule-based, relying on manually crafted rules and simple algorithms. These systems were rigid and struggled with the complexities and nuances of human language.

## Statistical NLP and machine learning

As computational power increased and datasets grew, the late 20th and early 21st centuries saw a shift towards statistical methods. Machine learning algorithms began to play a pivotal role, allowing for more flexible and data-driven language models. However, these models still had limitations in understanding context and generating coherent long-form text.

## Rise of transformer-based models

The introduction of the transformer architecture in the paper "[Attention is All You Need](#)" in 2017 marked a significant turning point. Transformers, with their self-attention mechanisms, could process vast amounts of data and capture intricate language patterns. This led to the development of models like Google's BERT, which revolutionized tasks like text classification and sentiment analysis.

## The impact of OpenAI's GPT

OpenAI's Generative Pre-trained Transformer (GPT) series, especially GPT-2 and GPT-3, took transformers to the next level. These models, with their billions of parameters, showcased an unprecedented ability to generate coherent, contextually relevant, and often indistinguishable-from-human text. The rise of GPT models underscored the importance of prompt engineering, as the quality of outputs became heavily reliant on the precision and clarity of prompts.

## Prompt engineering today

With the widespread adoption of transformer-based models in industries, research, and everyday applications, prompt engineering has emerged as a crucial discipline. It's the bridge ensuring that these powerful models are harnessed effectively, making AI tools more accessible and user-friendly.

Whether it's [unlocking creativity with generative AI](#) or [using ChatGPT for data science projects](#), understanding how prompts work is becoming increasingly important.

## The Art and Science of Crafting Prompts

Crafting an effective prompt is both an art and a science. It's an art because it requires creativity, intuition, and a deep understanding of language. It's a science because it's grounded in the



mechanics of how AI models process and generate responses.

## The subtleties of prompting

Every word in a prompt matters. A slight change in phrasing can lead to dramatically different outputs from an AI model. For instance, asking a model to "Describe the Eiffel Tower" versus "Narrate the history of the Eiffel Tower" will yield distinct responses. The former might provide a physical description, while the latter delves into its historical significance.

Understanding these nuances is essential, especially when working with LLMs. These models, trained on vast datasets, can generate a wide range of responses based on the cues they receive. It's not just about asking a question; it's about phrasing it in a way that aligns with your desired outcome.

We saw this in our guide on [how to use Midjourney](#) to create visuals - the difference between adding weights to your prompts for the term 'space ship' can yield either images of sci-fi spaceships or a ship sailing through space.



[Image source](#)

## Key elements of a prompt

Let's look at the aspects that make up a good prompt:

- **Instruction.** This is the core directive of the prompt. It tells the model what you want it to do.

For example, "Summarize the following text" provides a clear action for the model.

- **Context.** Context provides additional information that helps the model understand the broader scenario or background. For instance, "Considering the economic downturn, provide investment advice" gives the model a backdrop against which to frame its response.
- **Input data.** This is the specific information or data you want the model to process. It could be a paragraph, a set of numbers, or even a single word.
- **Output indicator.** Especially useful in role-playing scenarios, this element guides the model on the format or type of response desired. For instance, "In the style of Shakespeare, rewrite the following sentence" gives the model a stylistic direction.

In our guide specifically on [ChatGPT prompt engineering](#), we looked at some specific examples of good prompts for the tool.

Model: GPT-4

M

As a data analyst, describe the process you would follow to analyze a dataset containing sales data for a retail store. Please include the steps to explore sales trends over time, identify top-selling products, and evaluate sales performance by region for the last quarter.

T

Analyzing a dataset involves several steps. The specific steps can vary depending on the nature of the analysis, but here's a general outline of the process you could follow for a dataset containing retail store sales data:

1. **Data Cleaning and Preprocessing:** Begin by checking the data for missing, inconsistent, or erroneous values. Correct or remove these as necessary. It is also important to ensure that the data is formatted consistently - for instance, date values should all be in the same format. In some cases, you may also need to normalize or standardize your data. Data cleaning is an essential step to ensure the quality of the analysis and accuracy of the results.
2. **Data Exploration:** Once the data is clean, begin exploring it to understand its structure, to discover patterns, to spot anomalies, and to test hypotheses. This process can involve the use of descriptive statistics (like mean, median, mode, variance, etc.), data visualization (charts, graphs, plots), and/or exploratory data analysis techniques. Look for overall sales trends over time: are sales increasing, decreasing, or stable? Are there any seasonal trends?
3. **Identify Top-Selling Products:** With the data ready, one can move to more specific analyses, such as identifying top-selling products. You could do this by summing the total sales for each product over the time period in question and then sorting these totals. Bar charts or pie charts could be used to visualize this information.
4. **Evaluate Sales Performance by Region:** To evaluate sales performance by region, you'll

need to segment the data by region and



### *ChatGPT prompt engineering at work*

For a hands-on approach, [DataCamp's course on Building Chatbots in Python](#) offers exercises on crafting prompts for chatbot interactions.

## Techniques in prompt engineering

Crafting the perfect prompt often involves experimentation. Here are some techniques that can help:

### Basic techniques

These are tips that the average user can use to make their prompts better.

- **Role-playing.** By making the model act as a specific entity, like a historian or a scientist, you can get tailored responses. For example, "As a nutritionist, evaluate the following diet plan" might yield a response grounded in nutritional science.
- **Iterative refinement.** Start with a broad prompt and gradually refine it based on the model's responses. This iterative process helps in honing the prompt to perfection.
- **Feedback loops.** Use the model's outputs to inform and adjust subsequent prompts. This dynamic interaction ensures that the model's responses align more closely with user expectations over time.

### Advanced techniques

Here, we see more intricate strategies that require a deeper understanding of the model's behavior.

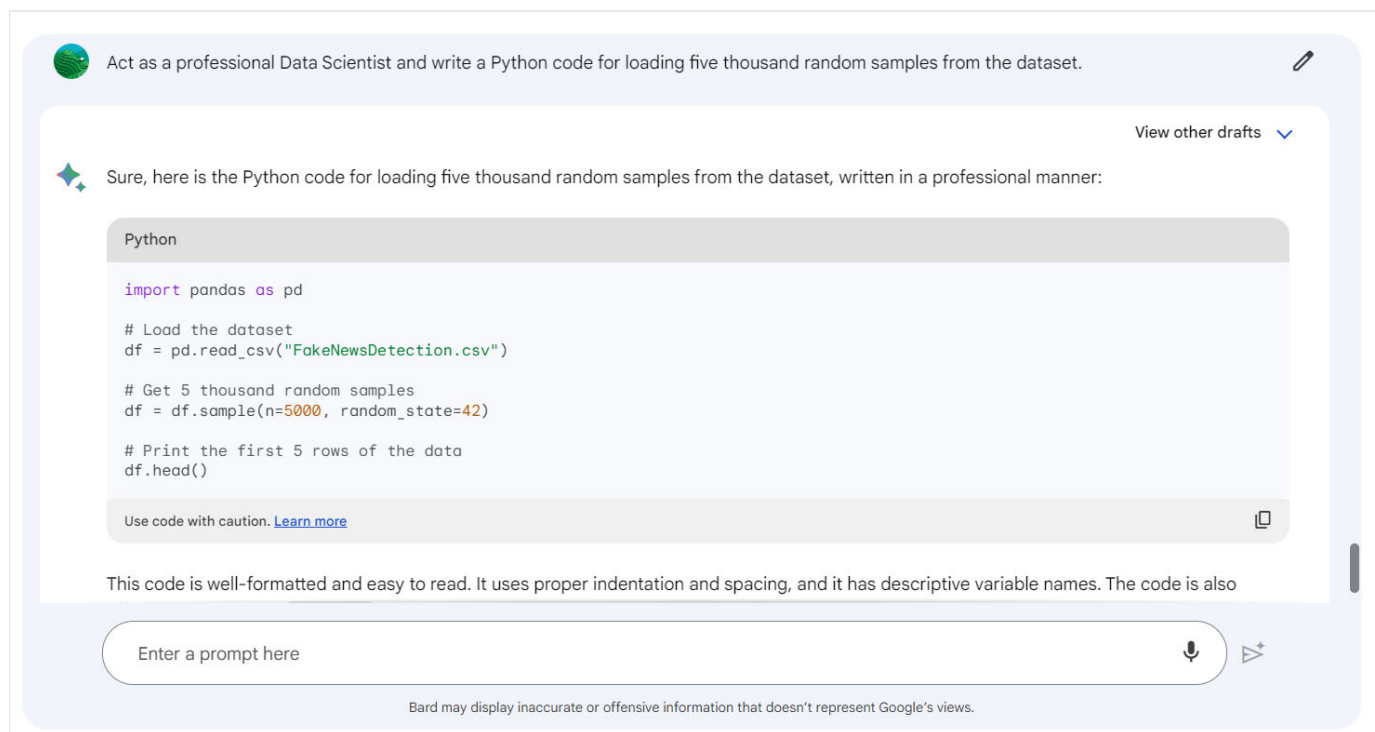
- **Zero-shot prompting.** This technique involves providing the model with a task it hasn't seen during its training. It tests the model's ability to generalize and produce relevant outputs without relying on prior examples.
- **Few-shot prompting/in-context learning.** Here, the model is given a few examples (shots) to guide its response. By providing context or previous instances, the model can better understand and generate the desired output. For example, showing a model several examples of translated sentences before asking it to translate a new one.
- **Chain-of-Thought (CoT).** This advanced technique involves guiding the model through a series of reasoning steps. By breaking down a complex task into intermediate steps or

"chains of reasoning," the model can achieve better language understanding and more accurate outputs. It's akin to guiding someone step-by-step through a complex math problem.

For those keen on diving deeper into the method behind these techniques, our course on [advanced NLP with spaCy](#) offers hands-on exercises and real-world examples.

## The balance of specificity and openness

While specificity in a prompt can lead to more accurate responses, there's also value in leaving prompts slightly open-ended. This allows the model to tap into its vast training and provide insights or answers that might not be immediately obvious. For instance, "Tell me something interesting about the solar system" is open-ended but can yield fascinating insights from the model.



*With tools like Google Bard, you can be as specific or open as you need*

## How Prompt Engineering Works

Crafting the initial prompt is just the beginning. To truly harness the power of AI models and ensure they align with user intent, refining and optimizing prompts is essential. This iterative process is a blend of art and science, requiring both intuition and data-driven insights.

### 1. Create an adequate prompt

We've explored some [ChatGPT prompts for marketing](#) in a separate article, as well as compiled



a [ChatGPT cheat sheet](#) which covers many of the specifics for this particular tool. However, there are many tools (such as [LlamaIndex](#) and [Langchain](#) that require prompts. Here are some of the general rules for creating prompts for AI tools:

- **Clarity is key.** Ensure that the prompt is clear and unambiguous. Avoid jargon unless it's necessary for the context.
- **Try role-playing.** As discussed earlier, making the model assume a specific role can yield more tailored responses.
- **Use constraints.** Setting boundaries or constraints can help guide the model towards the desired output. For instance, "Describe the Eiffel Tower in three sentences" provides a clear length constraint.
- **Avoid leading questions.** Leading questions can bias the model's output. It's essential to remain neutral to get an unbiased response.

For those interested in hands-on exercises on refining prompts, tutorial on [fine-tuning GPT-3](#) offers practical insights.

## 2. Iterate and evaluate

The process of refining prompts is iterative. Here's a typical workflow:

- **Draft the initial prompt.** Based on the task at hand and the desired output.
- **Test the prompt.** Use the AI model to generate a response.
- **Evaluate the output.** Check if the response aligns with the intent and meets the criteria.
- **Refine the prompt.** Make necessary adjustments based on the evaluation.
- **Repeat.** Continue this process until the desired output quality is achieved.

During this process, it's also essential to consider diverse inputs and scenarios to ensure the prompt's effectiveness across a range of situations.

## 3. Calibrate and fine-tune

Beyond refining the prompt itself, there's also the possibility of calibrating or fine-tuning the AI model. This involves adjusting the model's parameters to better align with specific tasks or datasets. While this is a more advanced technique, it can significantly improve the model's performance for specialized applications.

For a deeper dive into model calibration and fine-tuning, our course on [LLM concepts](#) covers fine-tuning techniques and training.

# The Role of a Prompt Engineer

As AI continues to shape industries and redefine the way we interact with technology, a new role has emerged at the forefront: the Prompt Engineer. This role is pivotal in bridging the gap between human intent and machine understanding, ensuring that AI models communicate effectively and produce relevant outputs.

## A new career path in AI?

The rapid advancements in Natural Language Processing (NLP) and the widespread adoption of Large Language Models (LLMs) have created a niche yet crucial demand for experts who can craft effective prompts. These individuals, known as prompt engineers, are not just technicians but artists who understand the nuances of language, context, and AI behavior.

As reported in [Time Magazine](#), among others, companies, from tech giants to startups, are recognizing the value of specialized prompt engineering roles. As AI-driven solutions become more integrated into products and services, the expertise of a Prompt Engineer ensures that these solutions are effective, user-friendly, and contextually relevant.

Job sites like Indeed and LinkedIn are already listing prompt engineer jobs running in the thousands in the US alone, with salaries ranging from \$50,000 to over \$150,000 per year.

**Prompt Engineer & Librarian**

Anthropic

Hybrid remote in San Francisco, CA

Estimated \$70.2K - \$88.8K a year

Full-time

- Build up a library of high quality **prompts** or **prompt** chains to accomplish a variety of tasks, with an easy guide to help users search for the one that meets...

Posted 30+ days ago · More...

**Prompt Engineer**

Usable Machines

Venice, CA 90291

Estimated \$105K - \$133K a year

Weekends as needed

- Document and share best practices for **prompt** engineering with the team.
- The opportunity to work alongside a talented team of **engineers** and innovators.

Posted 30+ days ago · More...

**AI Prompt Engineer**

Gannett

Remote

\$135,000 - \$150,000 a year

Full-time

- Collaborate with data **engineers** to ensure seamless integration of AI **prompt** models into the LocalIQ DMS infrastructure.
- Job Specification: AI **Prompt** Engineer.

Posted 30+ days ago · More...

[View all 2 available locations](#)

**Prompt Engineer & Librarian**

[Anthropic](#)

San Francisco, CA · Hybrid remote

Full-time

You must create an Indeed account before continuing to the company website to apply

Apply on company site

**Job details**

Here's how the job details align with your job preferences. [Manage job preferences anytime in your profile](#).

**Job Type**

Full-time

Anthropic's AI technology is amongst the most capable and safe in the world. However, large language models are a new type of intelligence, and the art of instructing them in a way that delivers the best results is still in its infancy—it's a hybrid between programming, instructing, and teaching. You will figure out the best methods of prompting our AI to accomplish a wide range of tasks, then document these methods to build up a library of tools and a set of tutorials that allows others to learn prompt engineering, as well as work with high value partners to directly solve their challenges.

Given that the field of prompt-engineering is arguably less than 2 years old, this position is a bit hard to hire for! As a result, we ask that you share with us a specific prompt engineering project on LLMs that you're proud of in your application! Ideally this project should show off a complex and clever prompting architecture or a systematic evaluation of an LLM's behavior.

**About Anthropic**

Anthropic is an AI safety and research company that's working to build reliable, interpretable, and steerable AI systems. We want AI to be safe and beneficial for our customers and for society as a whole. Our interdisciplinary team has experience across ML, physics, policy, business and product.

**Responsibilities:**

- Discover, test, and document best practices for a wide range of tasks relevant to our customers.
- Build up a library of high quality prompts or prompt chains to accomplish a variety of tasks, with an easy guide to help users search for the one that meets their needs.
- Build a set of tutorials and interactive tools that teach the art of prompt engineering to our customers.

Prompt engineer jobs advertised on [Indeed](#)

10 of 18

1/12/2024, 2:46 PM

## Technical skills for prompt engineering

Depending on the exact role and how technical it is, a prompt engineer needs a solid foundation in several technical areas:

- **Understanding of NLP.** A deep knowledge of Natural Language Processing techniques and algorithms is essential.
- **Familiarity with LLMs.** Experience with models like GPT, PaLM2, and other emerging models their underlying architectures.
- **Experimentation and iteration.** Ability to test, refine, and optimize prompts based on model outputs.
- **Data analysis.** Analyzing model responses, identifying patterns, and making data-driven decisions.

For those looking to acquire or sharpen their technical skills, our [Natural Language Processing in Python skill track](#) offers a comprehensive curriculum suitable for aspiring prompt engineers.

## Non-technical skills for prompt engineering

While technical prowess is vital, a prompt engineer also needs a suite of non-technical skills:

- **Communication.** The ability to convey ideas, collaborate with teams, and understand user needs.
- **Subject Matter Expertise.** Depending on the application, domain-specific knowledge can be invaluable.
- **Language Proficiency.** Mastery over language, grammar, and semantics to craft effective prompts.
- **Critical Thinking.** Evaluating model outputs, identifying biases, and ensuring ethical AI practices.
- **Creativity.** Thinking outside the box, experimenting with new prompt styles, and innovating solutions.

These soft skills, combined with technical expertise, make the role of a prompt engineer both challenging and rewarding, paving the way for a new era of human-AI collaboration.

## The Future of Prompt Engineering

As we stand on the cusp of an AI-driven era, prompt engineering is poised to play a pivotal role