



Introduction to Large Language Models (LLMs)

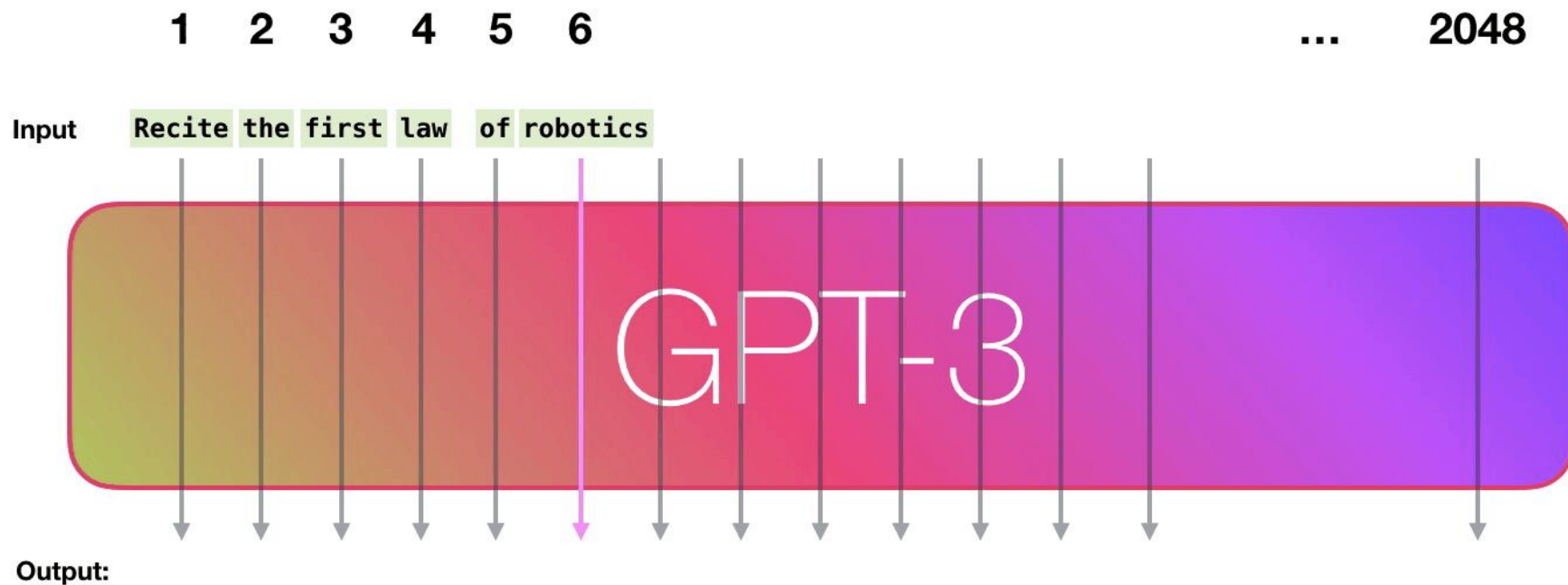
What is an LLM ?

Predict the **next** word ...



Prédiction. Touchez
une suggestion pour
l'appliquer.

... Recursively.



A prediction is a **probability**.

La planète la plus proche de la terre est Vénus

V = 57.33%

la = 20.16%

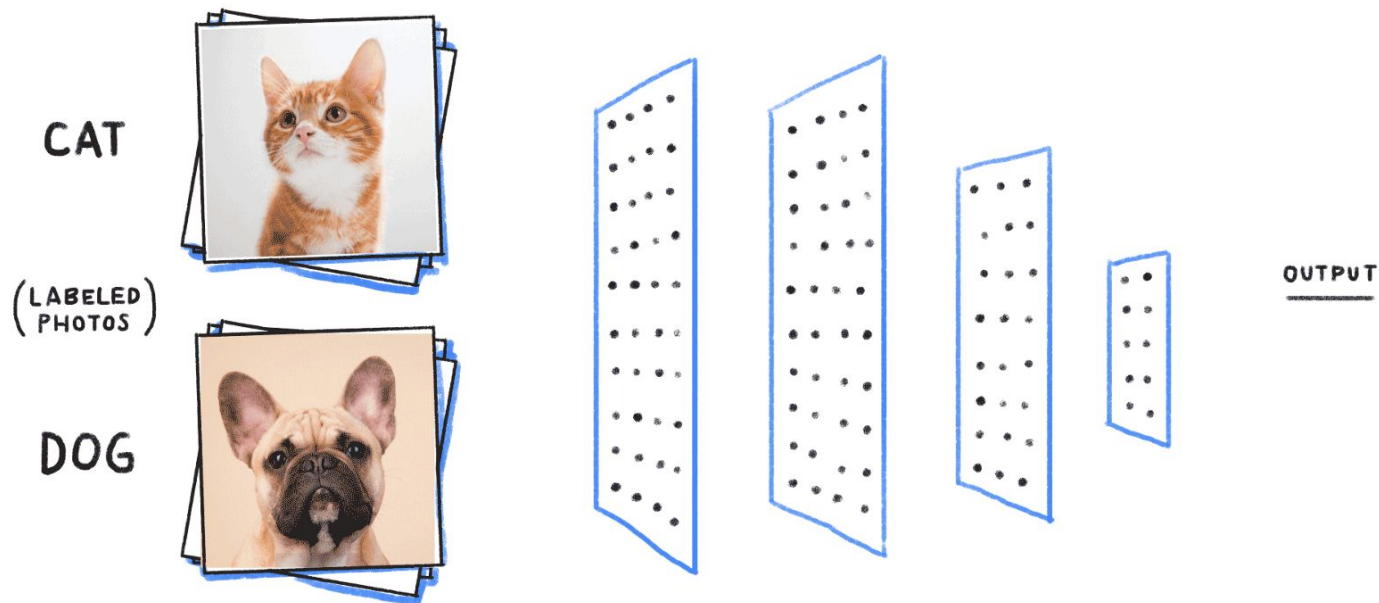
Merc = 11.21%

Venus = 6.57%

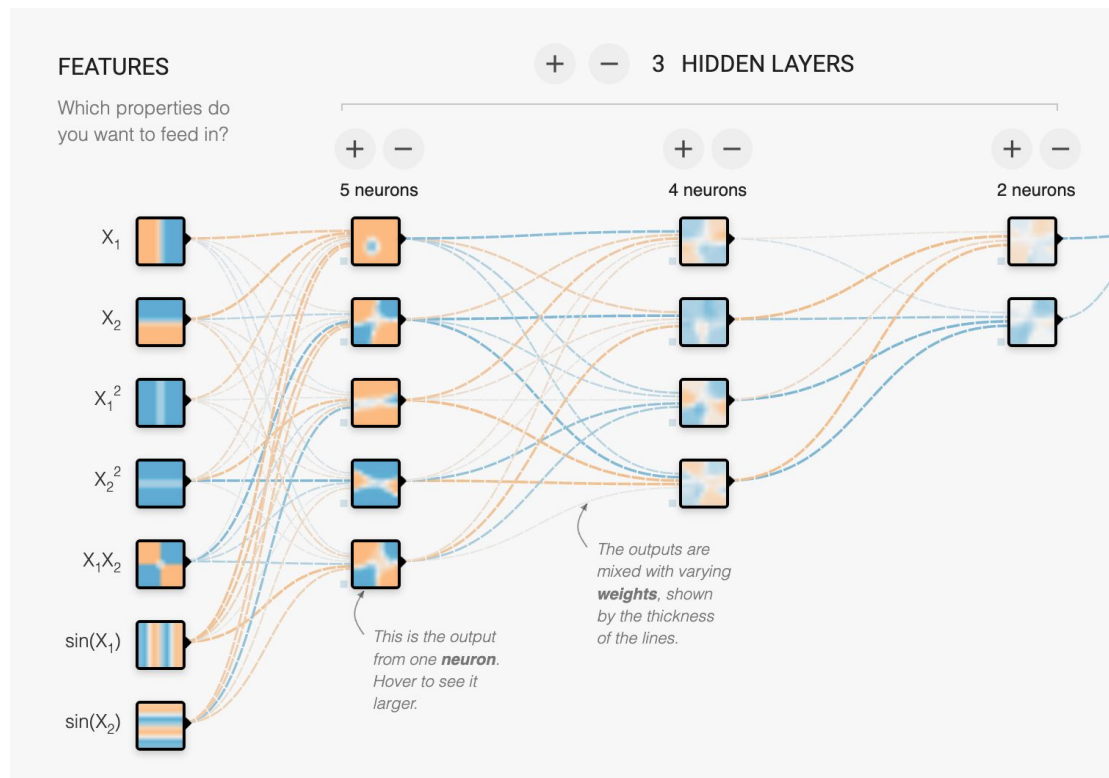
Mars = 1.51%

Total: -0.56 logprob on 1 tokens
(96.78% probability covered in top 5 logits)

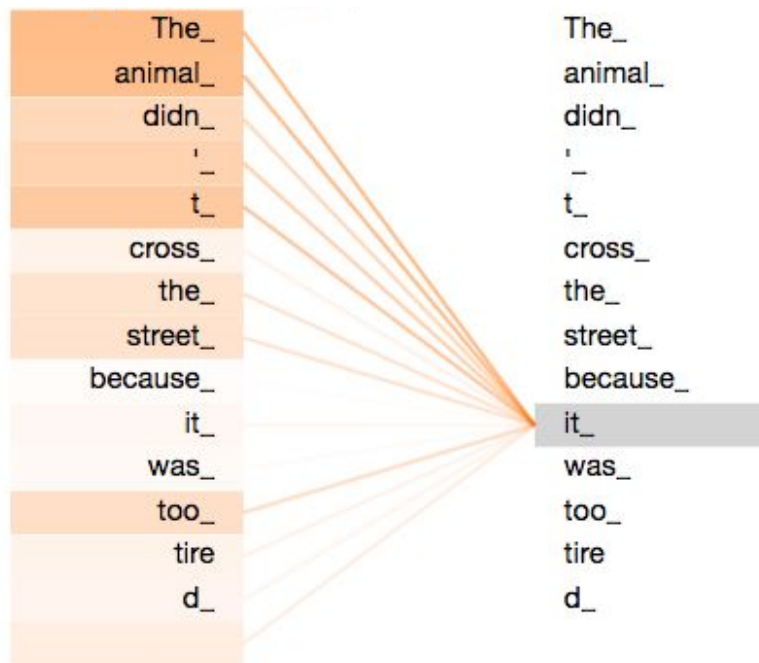
Under the hood, a **neural network**.



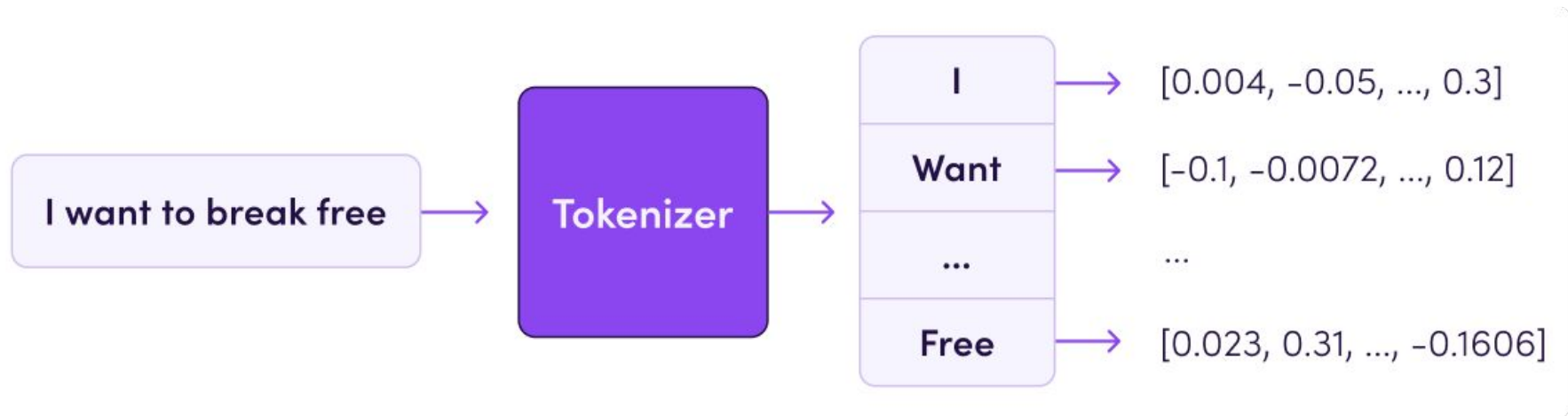
A **trained** neural network.



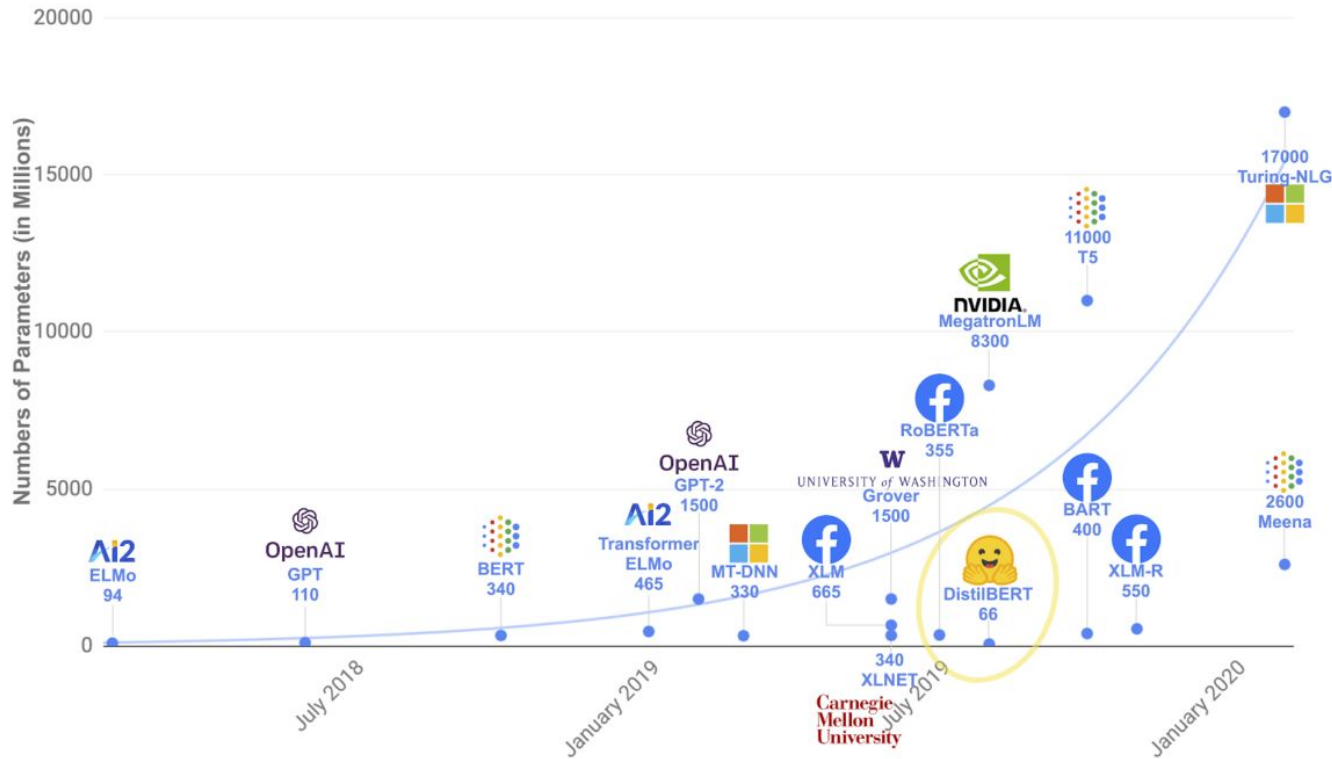
Which pays « **attention** ».



Words or **tokens** ?



Bigger and bigger models.



How many graphics cards for **GPT-3** ?

In conclusion, training OpenAI's GPT-3, a groundbreaking language model with 175 billion parameters, necessitates a significant number of GPUs. Reports suggest that the training process for GPT-3 involved anywhere from **3,000 to 5,000 GPUs**. 20 oct. 2023

And for **GPT-4** ?

25,000 Nvidia A100 GPUs

The electricity consumption of GPT-4

According to unverified information leaks, GPT-4 was trained on about 25,000 Nvidia A100 GPUs for 90–100 days [2]. Let's assume the GPUs were installed in Nvidia HGX servers which can host 8 GPUs each, meaning $25,000 / 8 = 3,125$ servers were needed.

Meta to the rescue.

Introducing Llama 2

The next generation of our
open source large language model

Llama 2 is available for free for research and commercial use.

[Download the Model](#)

GPT-4, and others

Total #models: 64. Total #votes: 299816. Last updated: Feb 15, 2024.

Contribute your vote 🗳️ at chat.lmsys.org! Find more analysis in the [notebook](#).

Rank ▲	🤖 Model ▲	★ Arena Elo ▲	📊 95% CI ▲	🗳️ Votes ▲	Organization ▲	License ▲
1	GPT-4-1106-preview	1254	+5/-5	38745	OpenAI	Proprietary
2	GPT-4-0125-preview	1253	+10/-8	6308	OpenAI	Proprietary
3	Bard (Gemini Pro)	1218	+8/-7	10313	Google	Proprietary
4	GPT-4-0314	1191	+6/-6	20430	OpenAI	Proprietary
5	GPT-4-0613	1164	+5/-6	32941	OpenAI	Proprietary

Prompt Engineering

Introduction

Prompt engineering is a relatively new discipline for developing and optimizing prompts to efficiently use language models for a wide variety of applications and research topics. Prompt engineering skills help to better understand the capabilities and limitations of large language models (LLMs).

LLM Settings

When working with prompts, you interact with the LLM via an API or directly. You can configure a few parameters to get different results for your prompts:

LLM Settings

When working with prompts, you interact with the LLM via an API or directly. You can configure a few parameters to get different results for your prompts:

- **Temperature [0 - 2]:** *In short, the lower the temperature, the more deterministic the results in the sense that the highest probable next token is always picked. Increasing temperature could lead to more randomness, which encourages more diverse or creative outputs.*
- **Top_p [0 – 1]:** *Similarly, with top_p, a sampling technique with temperature called nucleus sampling, you can control how deterministic the model is at generating a response.*
- **Max Length:** *You can manage the number of tokens the model generates by adjusting the max length.*
- **Stop Sequences:** *A stop sequence is a string that stops the model from generating tokens.*

What is a « good » prompt ?

Prompt:

The sky is

Output:

The sky is like a canvas,
ever-changing and full of wonder!

Whether it's painted in vibrant
hues during a sunset or adorned
with twinkling stars at night,
there's always something
captivating about it.

What does the sky look like right
now where you are?

What is a « good » prompt ?

This basic example **highlights the necessity to provide more context or instructions** on what specifically you want to achieve.

Prompt:

The sky is

Output:

The sky is like a canvas,
ever-changing and full of wonder!

Whether it's painted in vibrant
hues during a sunset or adorned
with twinkling stars at night,
there's always something
captivating about it.

What does the sky look like right
now where you are?

What is a « good » prompt ?

Prompt:

Complete the sentence:

The sky is

Output:

The sky is an expansive tapestry,
woven with threads of possibility
and painted with the hues of
nature's artistry.

What is a « good » prompt ?

This approach of **designing optimal prompts** to instruct the model to perform a task is what's referred to as **prompt engineering**.

Prompt:

Complete the sentence:
The sky is

Output:

The sky is an expansive tapestry,
woven with threads of possibility
and painted with the hues of
nature's artistry.

Elements of a Prompt

Elements of a Prompt

A prompt contains any of the following elements:

- **Instruction** - a specific task or instruction you want the model to perform.
- **Context** - external information or additional context that can steer the model to better responses.
- **Input Data** - the input or question that we are interested to find a response for.
- **Output Indicator** - the type or format of the output.

You do not need all the four elements for a prompt and the format depends on the task at hand.

General Tips for Designing Prompts

1) **Start Simple**

General Tips for Designing Prompts

1) **Start Simple**

As you get started with designing prompts, you should keep in mind that it **is really an iterative process** that requires a lot of experimentation to get optimal results.

When you have a **big task** that involves many different subtasks, you can try to **break down the task into simpler subtasks** and keep building up as you get better results. This avoids adding too much complexity to the prompt design process at the beginning.

General Tips for Designing Prompts

2) **The Instruction**

General Tips for Designing Prompts

2) The Instruction

You can design effective prompts for various simple tasks by using commands to instruct the model what you want to achieve, such as "Write", "Classify", "Summarize", "Translate", "Order", etc... **Try different instructions with different keywords, contexts, and data and see what works best for your particular use case and task.**

Others recommend that you **place instructions at the beginning of the prompt.** Another recommendation is to **use some clear separator like "###" to separate the instruction and context.**

General Tips for Designing Prompts

2) The Instruction

Prompt:

Instruction

Translate the text below to
Spanish:

Text: "hello!"

Output:

¡Hola!.

General Tips for Designing Prompts

2) **Specificity**

General Tips for Designing Prompts

2) **Specificity**

The more descriptive and detailed the prompt is, the better the results. In fact, providing examples in the prompt is very effective to get desired output in specific formats. Including too many unnecessary details is not necessarily a good approach. The details should be relevant and contribute to the task at hand.

General Tips for Designing Prompts

2) Specificity

Prompt:

Extract the name of places in the following text.

Desired format:

Place: <comma_separated_list_of_company_names>

Input: "Although these developments are encouraging to researchers, much is still a mystery. "We often have a black box between the brain and the effect we see in the periphery," says Henrique Veiga-Fernandes, a neuroimmunologist at the Champalimaud Centre for the Unknown in Lisbon. "If we want to use it in the therapeutic context, we actually need to understand the mechanism.""

Output:

Place: Champalimaud Centre for the Unknown, Lisbon

General Tips for Designing Prompts

3) **Avoid Impreciseness**

General Tips for Designing Prompts

3) **Avoid Impreciseness**

Given the tips above about being detailed and improving format, it's easy to fall into the trap of wanting to be too clever about prompts and potentially creating imprecise descriptions. **It's often better to be specific and direct.** The analogy here is very similar to effective communication -- the more direct, the more effective the message gets across.

General Tips for Designing Prompts

3) **Avoid Impreciseness**

For example, you might be interested in learning the concept of prompt engineering. You might try something like:

“Explain the concept of prompt engineering. Keep the explanation short, only a few sentences, and don't be too descriptive.”

It's not clear from the prompt above how many sentences to use and what style. You might still somewhat get good responses with the above prompts but the better prompt would be one that is very specific, concise, and to the point. Something like:

“Use 2-3 sentences to explain the concept of prompt engineering to a high school student.”

General Tips for Designing Prompts

4) **To do or not to do?**

General Tips for Designing Prompts

4) To do or not to do?

Another common tip **when designing prompts** is to **avoid saying what not to do but say what to do instead**. This encourages more specificity and focuses on the details that lead to good responses from the model.

General Tips for Designing Prompts

4) To do or not to do?

Prompt:

The following is an agent that recommends movies to a customer. **DO NOT ASK FOR INTERESTS. DO NOT ASK FOR PERSONAL INFORMATION.**

Customer: Please recommend a movie based on my interests.

Agent:

Output:

Sure, I can recommend a movie based on your interests. What kind of movie would you like to watch? Do you prefer action, comedy, romance, or something else?

General Tips for Designing Prompts

4) To do or not to do?

Prompt:

The following is an agent that recommends movies to a customer. The agent is responsible to recommend a movie from the top global trending movies. It should refrain from asking users for their preferences and avoid asking for personal information. **If the agent doesn't have a movie to recommend, it should respond "Sorry, couldn't find a movie to recommend today."**

Customer: Please recommend a movie based on my interests.

Agent:

Output:

Sorry, I don't have any information about your interests. However, here's a list of the top global trending movies right now: [list of movies]. I hope you find something you like!

To summarize

- Start Simple
- The Instruction
- Specificity
- Avoid Impreciseness
- To do or not to do?

Use Case - Text Summarization

One of the standard tasks in natural language generation is text summarization. Text summarization can include many different flavors and domains.

Use Case - Text Summarization

One of the standard tasks in natural language generation is text summarization. Text summarization can include many different flavors and domains.



Key-sentence:

- “*Explain the following text in X sentences:*”
- “*Summarize the above in X words:*”

...

Don't forget the `:` character at the hand the sentence for better results.

Use Case - Text Summarization

Prompt:

Antibiotics are a class of drugs that treat bacterial infections. They work by killing or slowing the growth of bacteria that cause infections. They do this by interfering with the bacterial cell wall, which causes the cell to lyse and release the bacterial contents.

Antibiotics are typically prescribed for bacterial infections, but they can also be used preventatively, or as a prophylactic measure, to reduce the risk of infection in situations where there is a higher risk of bacterial infection. For example, they are used routinely in the midstments of pregnancy and childbirth.

They are a crucial tool in the fight against bacterial infections, but it is important to use them judiciously, as some strains of bacteria are becoming resistant to antibiotics through over-use.

Explain the text above in one sentence:

Output:

Antibiotics kill or slow the growth of bacteria causing infections, interfering with their cell wall, and are used to treat or prevent bacterial infections.

Use Case - Information Extraction

While language models are trained to perform natural language generation and related tasks, it's also very capable of performing classification and a range of other natural language processing (NLP) tasks.

Use Case - Information Extraction

While language models are trained to perform natural language generation and related tasks, it's also very capable of performing classification and a range of other natural language processing (NLP) tasks.



Key-sentence:

- “**Identify** <subject> ... :”
- “**Mention** <subject> ... :”
- “**Which is** <subject> ... :”

...

You can also use When, Where, Which, ... but be careful to be specific !

*Don't forget the **:** character at the hand the sentence for better results.*

Use Case - Information Extraction

Prompt:

Author-contribution statements and acknowledgements in research papers should state clearly and specifically whether, and to what extent, the authors used AI technologies such as ChatGPT in the preparation of their manuscript and analysis. They should also indicate which LLMs were used. This will alert editors and reviewers to scrutinize manuscripts more carefully for potential biases, inaccuracies and improper source crediting. Likewise, scientific journals should be transparent about their use of LLMs, for example when selecting submitted manuscripts.

Which is the large language model based product mentioned in the paragraph above:

Output:

The answer is ChatGPT.

Use Case - Question Answering

One of the best ways to get the model to respond to specific answers is to improve the format of the prompt. As covered before, **a prompt could combine instructions, context, input, and output indicators to get improved results.** While these components are not required, it becomes a good practice as the more specific you are with instruction, the better results you will get. Below is an example of how this would look following a more structured prompt.

Use Case - Question Answering

One of the best ways to get the model to respond to specific answers is to improve the format of the prompt. As covered before, **a prompt could combine instructions, context, input, and output indicators to get improved results.** While these components are not required, it becomes a good practice as the more specific you are with instruction, the better results you will get. Below is an example of how this would look following a more structured prompt.



Key-format:

1. **Instruction:**
2. **Context:**
3. **Question:**
4. **Answer:**

Use Case - Question Answering

Prompt:

Answer the question based on the context below. Keep the answer short and concise. Respond "Unsure about answer" if not sure about the answer. // <= **Instruction** "avoid saying what not to do but say what to do instead"

Context: Teplizumab traces its roots to a New Jersey drug company called Ortho Pharmaceutical. There, scientists generated an early version of the antibody, dubbed OKT3. Originally sourced from mice, the molecule was able to bind to the surface of T cells and limit their cell-killing potential. In 1986, it was approved to help prevent organ rejection after kidney transplants, making it the first therapeutic antibody allowed for human use.

Question: What was OKT3 originally sourced from?

Answer:

Output:

Mice.

Use Case - Text Classification

Prompt:

Classify the text into **neutral**, **negative** or **positive**.

Text: I think the food was okay.

Sentiment:

Output:

Based on your text, the sentiment is neutral.

Use Case - Text Classification

Prompt:

Classify the text into **neutral**, **negative** or **positive**.

Text: I think the vacation is okay.

Sentiment: neutral

Text: I think the food was okay.

Sentiment:

Output:

neutral

Use Case - Conversation

Prompt:

The following is a conversation with an AI research assistant. The assistant tone is technical and scientific.

Human: Hello, who are you?

AI: Greeting! I am an AI research assistant. How can I help you today?

Human: Can you tell me about the creation of blackholes?

AI:

Use Case - Code Generation

Prompt:

```
""
Table departments, columns = [DepartmentId, DepartmentName]
Table students, columns = [DepartmentId, StudentId, StudentName]
Create a MySQL query for all students in the Computer Science Department
""
```

Output:

```
SELECT StudentId, StudentName
FROM students
WHERE DepartmentId IN (SELECT DepartmentId FROM departments WHERE Dep
```

Use Case - Reasoning

- Perhaps one of the most difficult tasks for an LLM today is one that requires some form of reasoning.
- Reasoning is one of most interesting areas due to the types of complex applications that can emerge from LLMs.

Use Case - Reasoning

Prompt:

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.

A:

Output:

No, the odd numbers in this group add up to an odd number: 119.

<- WRONG

Use Case - Reasoning

Prompt:

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.

Solve by breaking the problem into steps. First, identify the odd numbers, add them, and indicate whether the result is odd or even.

Output:

Odd numbers: 15, 5, 13, 7, 1
Sum: 41
41 is an odd number.

Better ! :)

Techniques - One Shot vs. Few Shot

<Question>?
or
<Instruction>

Q: <Question>?

A: <Answer>

Q: <Question>?

A: <Answer>

Q: <Question>?

A: <Answer>

Q: <Question>?

A:

Techniques - Few Shot

Prompt:

A "whatpu" is a small, furry animal native to Tanzania.

An example of a sentence that uses the word whatpu is:

We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast.

An example of a sentence that uses the word farduddle is:

Output:

Look at how cute they are! I can't stand it! I have to farduddle with excitement!

RESSOURCES

- <https://www.promptingguide.ai/>
- <https://platform.openai.com/docs/guides/prompt-engineering>

