

# Programming GPTs

---



- 
1. Introducing GPTs
  2. OpenAI APIs
  3. Prompting, Chaining & Summarization
  4. Vector search & Question Answering
  5. Agents & Tools
  6. Speech-to-Text & Text-to-Speech
  7. Vision
  8. Dall-E image generation
  9. Deploying GPTs
  10. Appendix

# Agenda



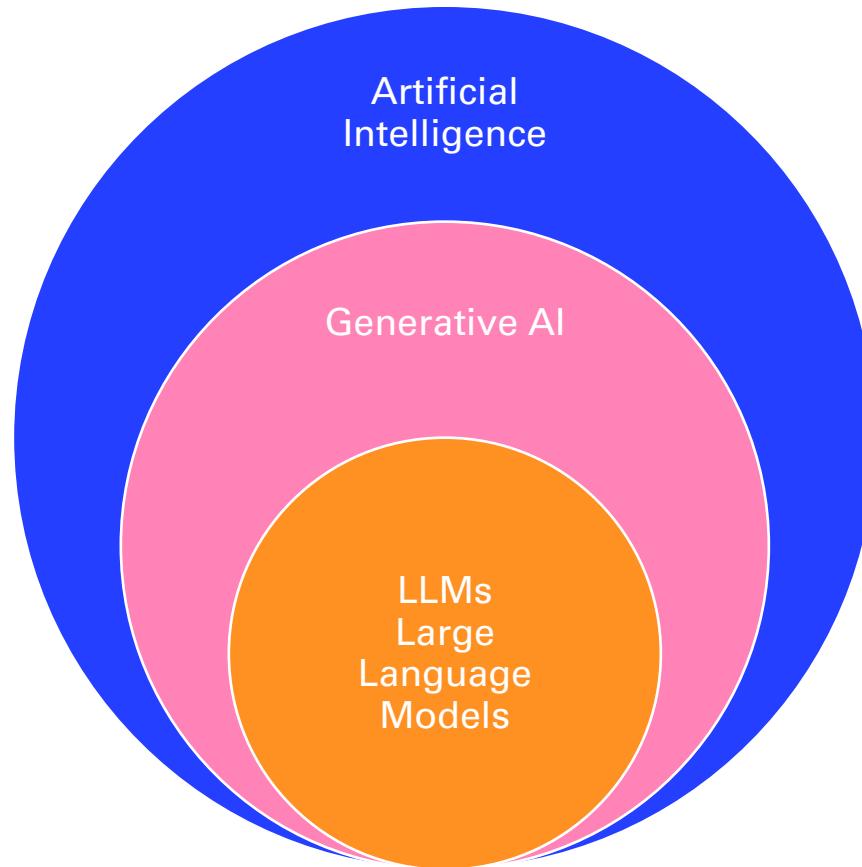
+

.

# GPTs

Generative  
Pre-trained  
Transformers

# Terminology



# Large Language Models are next word predictors

in

out

A long time ago in

A long time ago in a

A long time ago in a galaxy

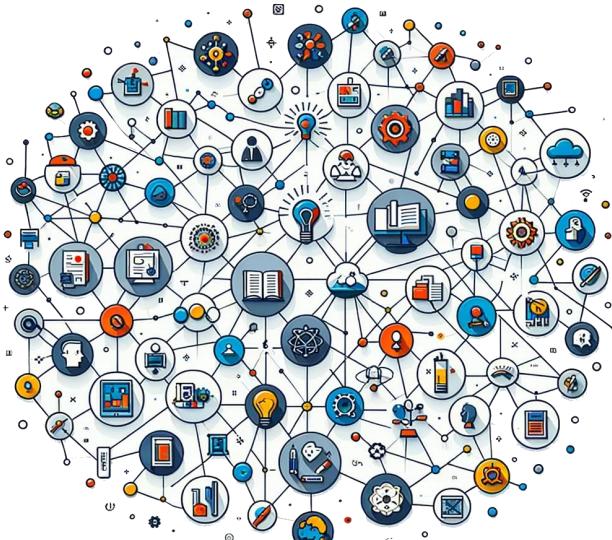
A long time ago in a galaxy far

A long time ago in a galaxy far,

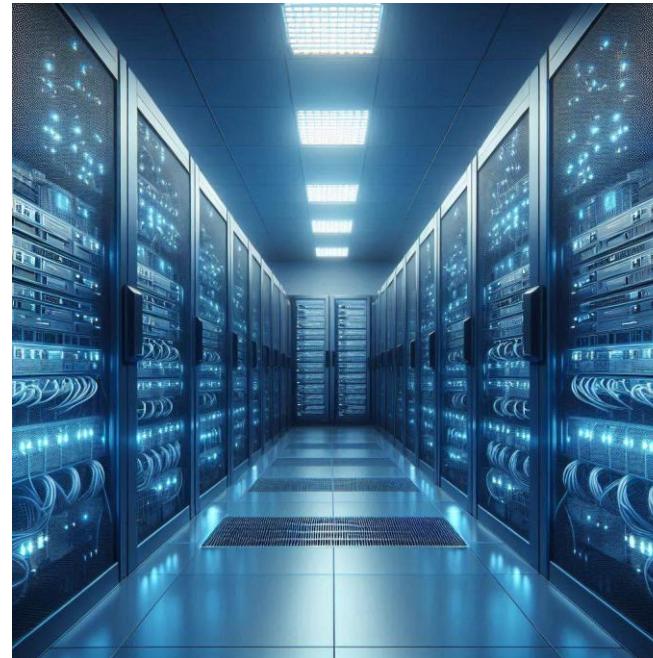
A long time ago in a galaxy far, far

A long time ago in a galaxy far, far away

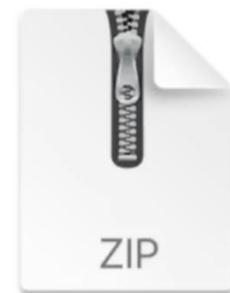
# L in LLM



Chunks of the internet  
~10TB of text



6000 GPUs for 12 days  
~ \\$2M / ~1e24 FLOPS



parameters.zip

~140GB  
file

# Generative



**Generative** models can create new data from existing data.

For example, given an image, a generative model can produce another image that is similar but not identical to the original one.

Similarly, given a text, a generative model can produce another text that is related but not identical to the original one.



# Generative vs Discriminative

"Generative" describes a class of statistical models that contrasts with discriminative models.

- **Generative** models can generate new data instances.
- **Discriminative** models discriminate between different kinds of data instances.



a long time ago

written with gpt i

Shuffle initial text

Trigger autocomplete or tab

Select suggestion ↑ ↓ and enter

Cancel suggestion esc

Save & Publish

a long time ago in a galaxy

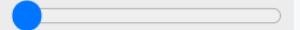
far, far away.

beyond the imagination of any human being.

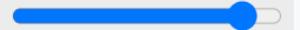
far beyond the galaxy of this world.

Model & decoder settings i

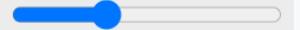
Model size **gpt**



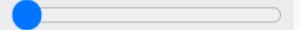
Top-p **0.9**



Temperature **1**

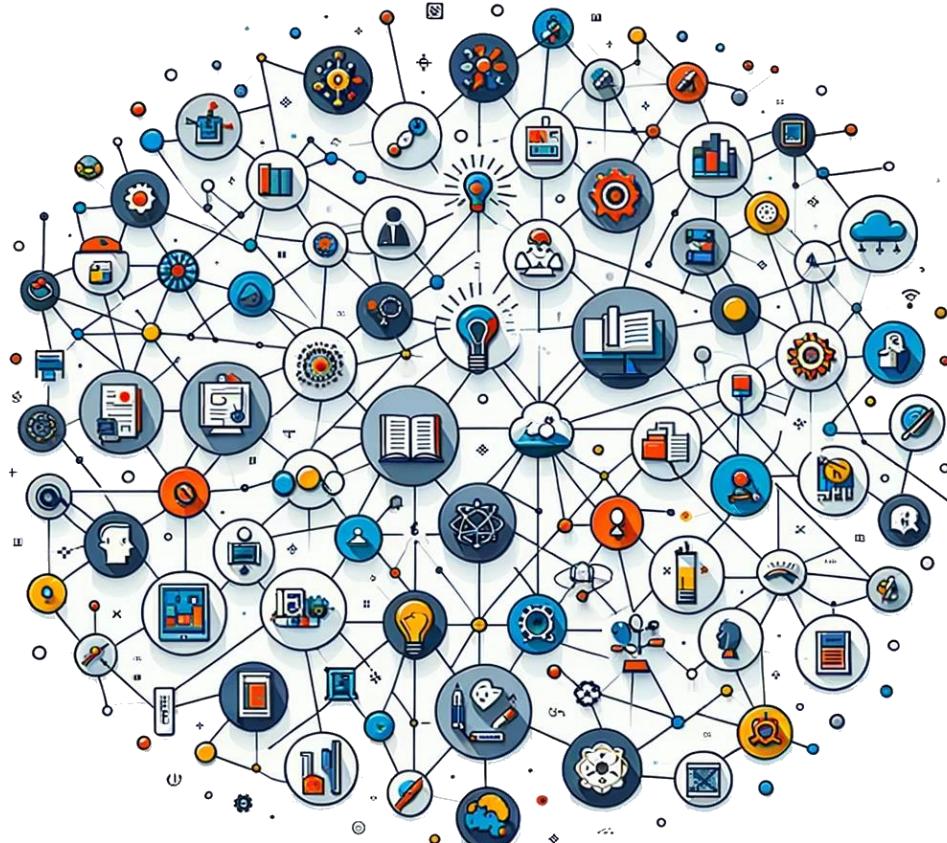


Max time **1**



[transformer.huggingface.co](https://transformer.huggingface.co)

# Pre-trained



**GPTs** have been pre-trained on massive amounts of text from the internet and can produce coherent and diverse responses on almost any topic. They can also perform various tasks, such as answering questions, summarizing texts, generating images, and more.

# GPT training pipeline

	Pretraining	Finetuning
Dataset	<b>Raw internet</b> Text trillions of words Low quality, large quantity	<b>Demonstrations</b> Ideal assistant responses, ~10-100K (prompt, response) written by contractors High quality, low quantity
Model	<b>Base model</b>	<b>Instruct model</b>
Method	<b>Self-supervised</b> next word prediction	<b>Supervised</b> on labeled data
Compute	1000s of GPUs months of training	1-100 GPUs days of training

# Self-supervised next word prediction

a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

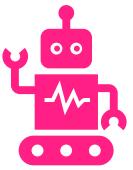
a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

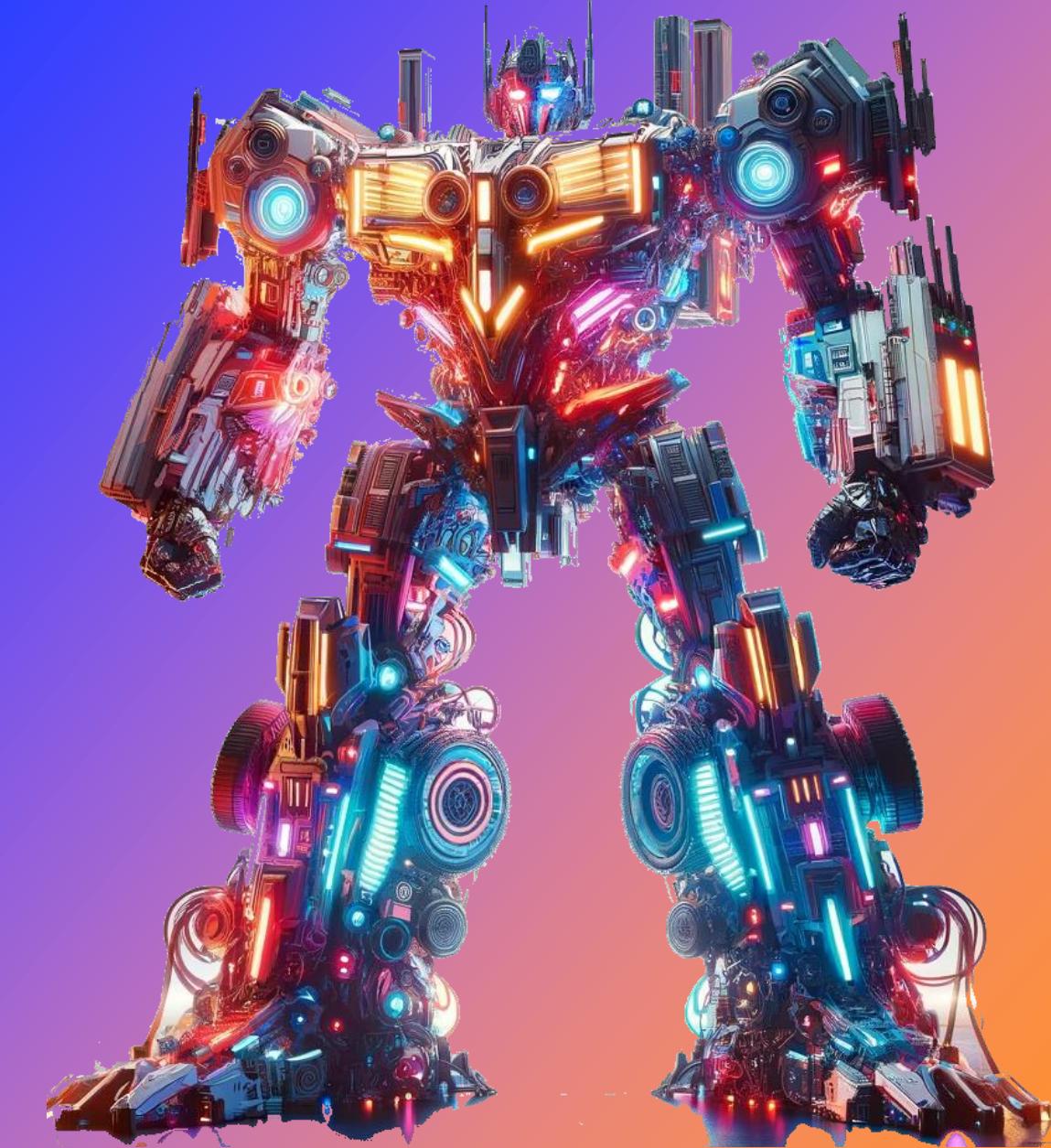
a	long	time	ago	in	a	galaxy	far	,
---	------	------	-----	----	---	--------	-----	---

# Transformers



A transformer is a neural network that can map any input sequence to any output sequence.

For example, given a sentence in English, a transformer can learn to translate it to French.

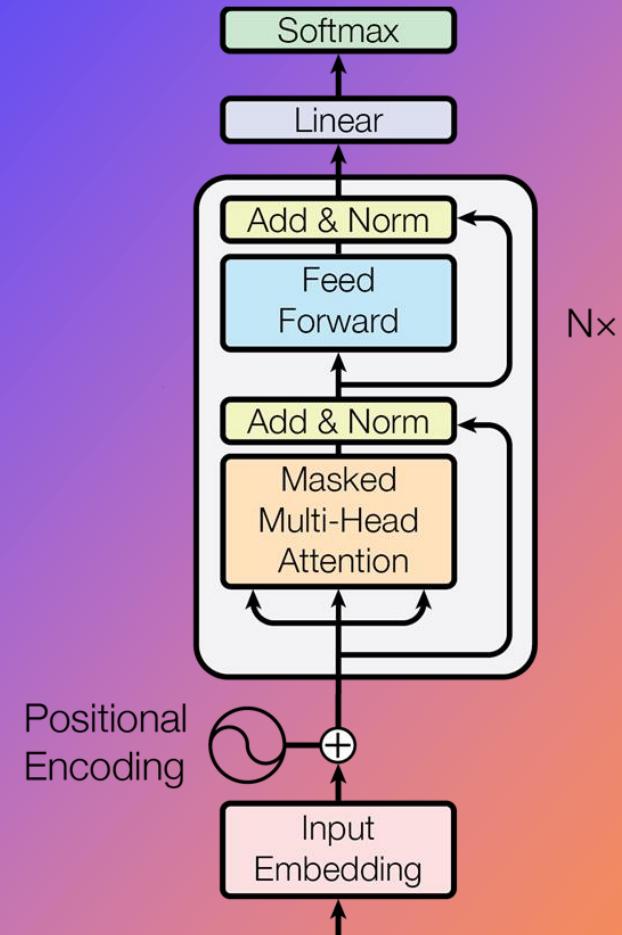


# Transformers are general-purpose machines

*Expressive:* They can model complex relationships and patterns in data

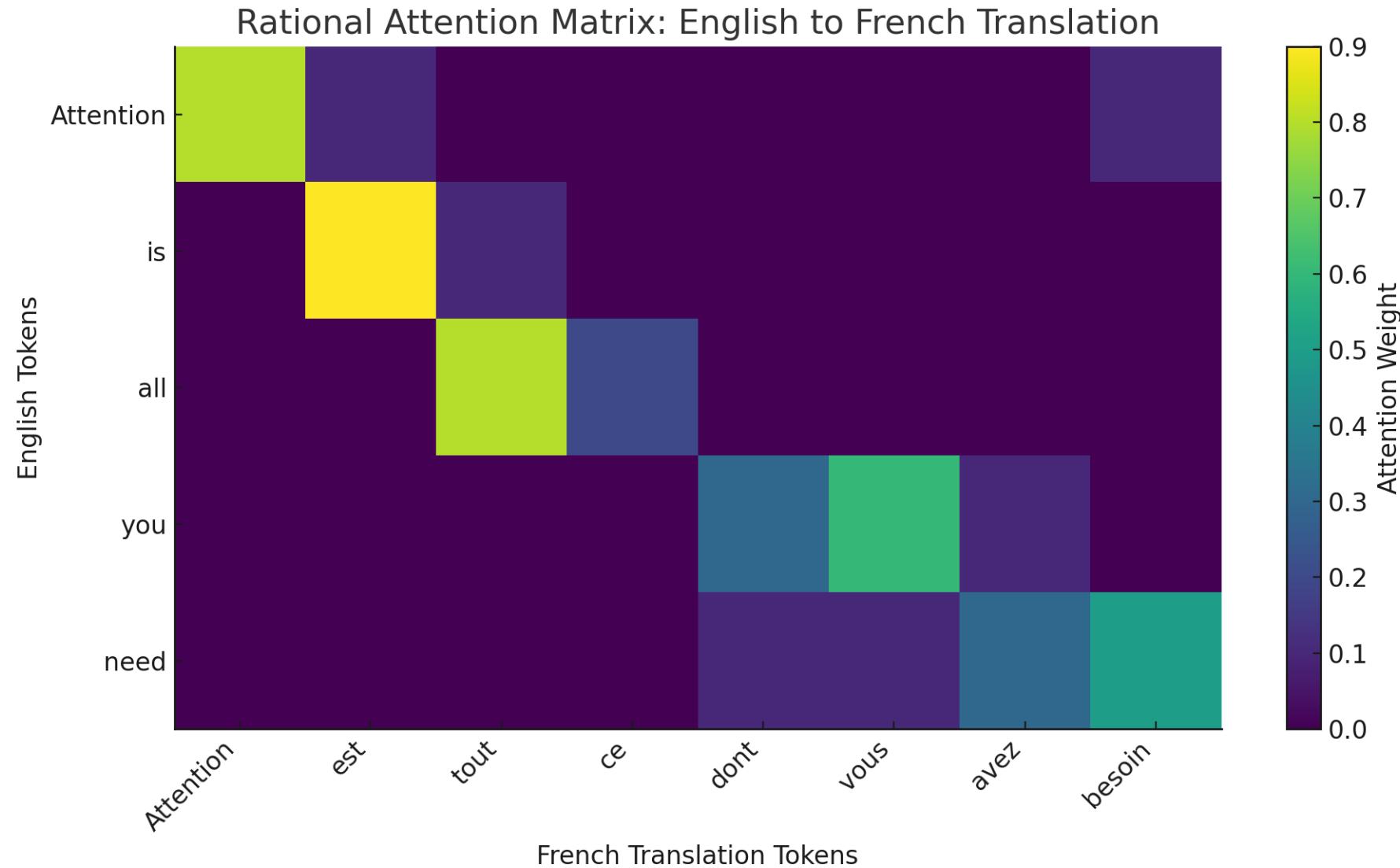
*Optimizable:* Their ability to handle complexity is not compromised by the fact that they can be trained on large amount of data

*Efficient:* The training and prediction can be highly parallelized on graphics cards.



translate in French: Attention is all you need

# Attention

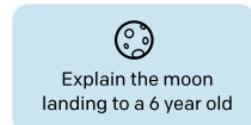


# Following instructions

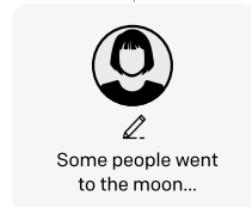
Step 1

**Collect demonstration data, and train a supervised policy.**

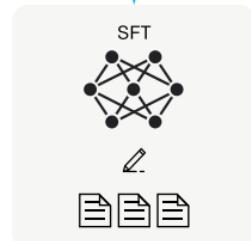
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



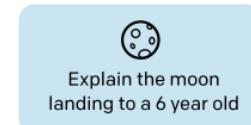
This data is used to fine-tune GPT-3 with supervised learning.



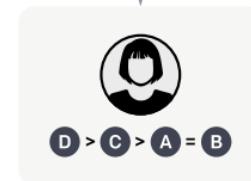
Step 2

**Collect comparison data, and train a reward model.**

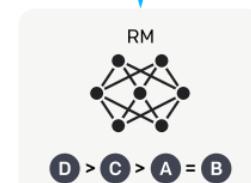
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



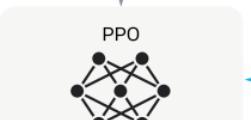
Step 3

**Optimize a policy against the reward model using reinforcement learning.**

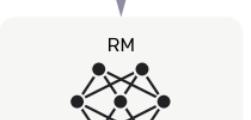
A new prompt is sampled from the dataset.



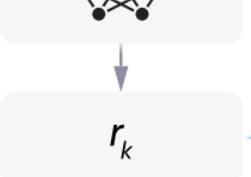
The policy generates an output.



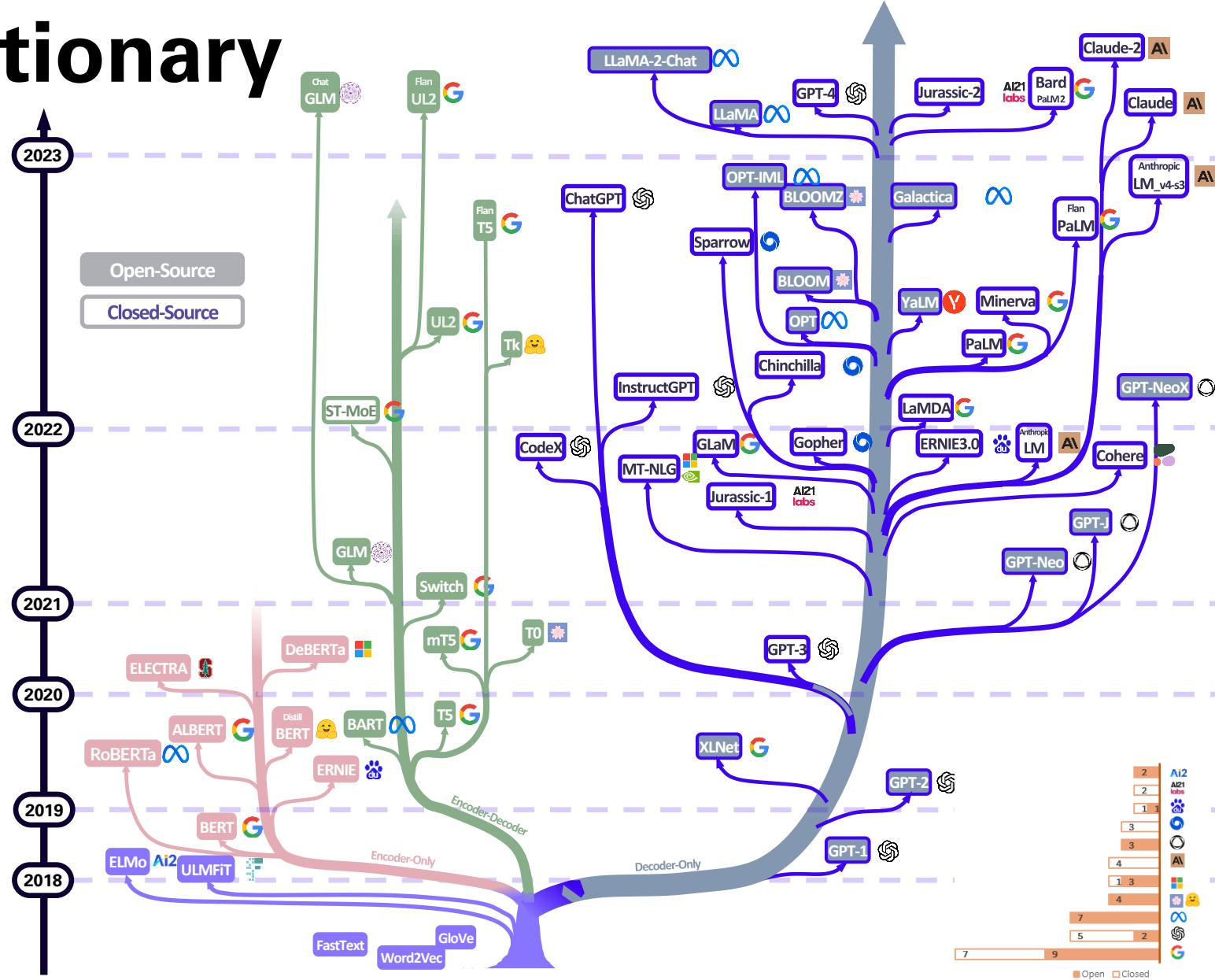
The reward model calculates a reward for the output.



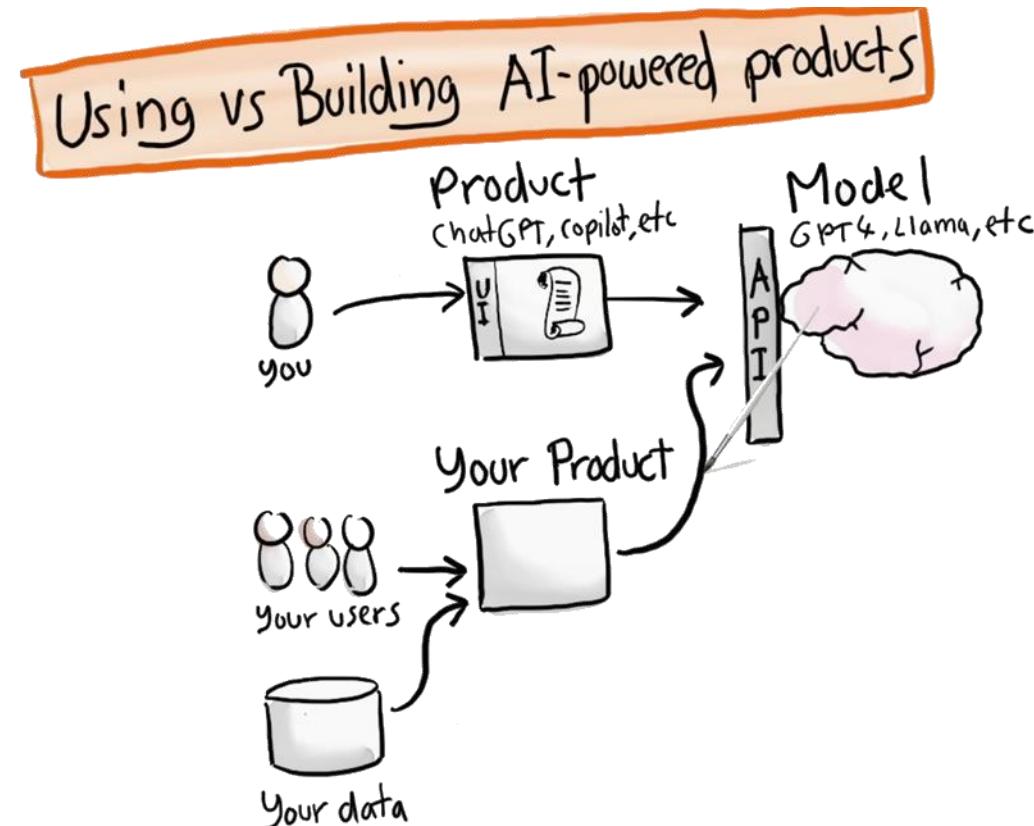
The reward is used to update the policy using PPO.



# LLM Evolutionary Tree



# Using vs Building AI-powered products



[Video: Generative AI in a nutshell](#)

OpenAI APIs

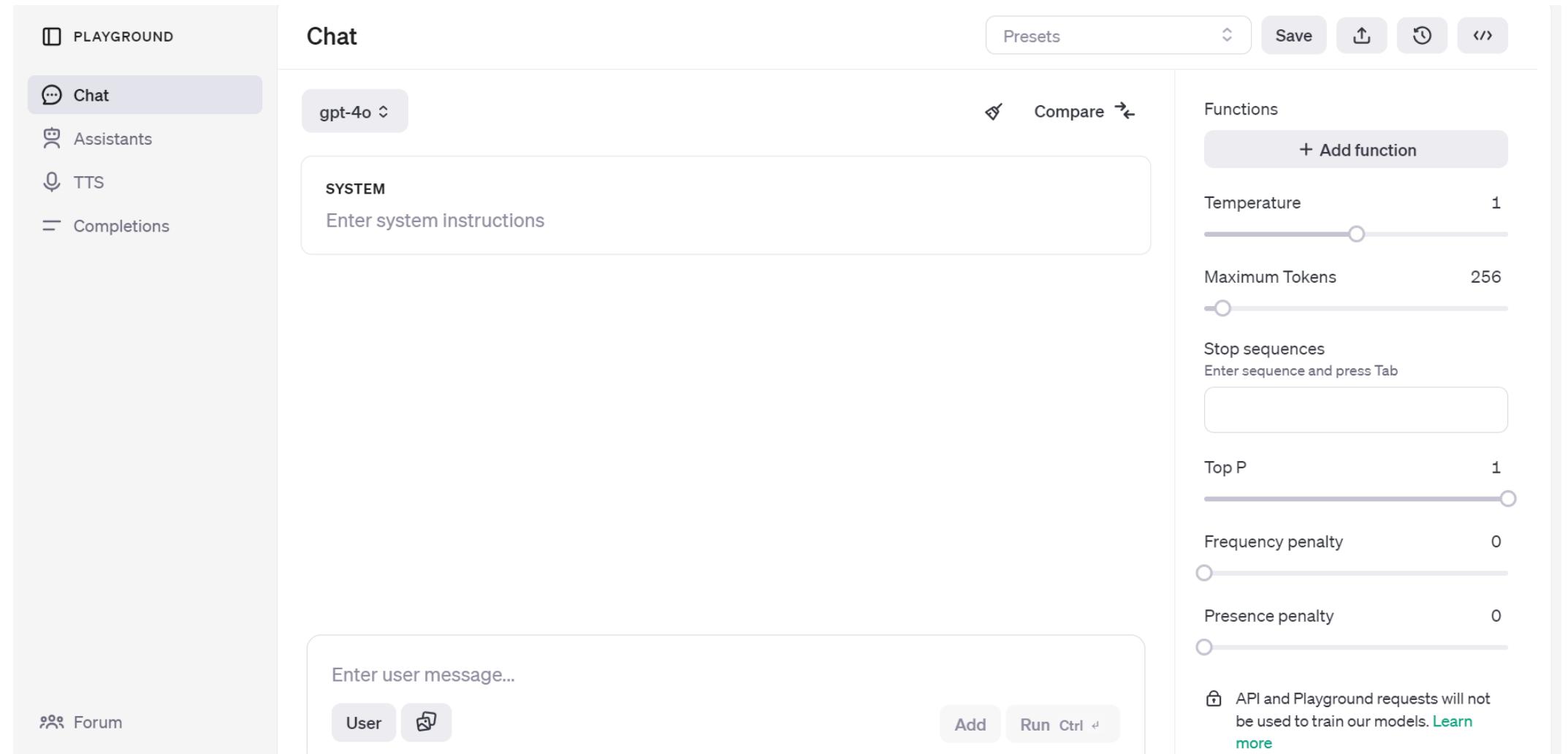


+

o

.

# Quick tour of OpenAI developer platform





## Playground

## Your presets

**Save**

[View code](#)

Share

• • •

A long time ago in a galaxy far away.. there was a peace which had reigned for centuries. Since the establishment of the Galactic Republic, the galaxy has been in relative harmony under its rule. However, the galaxy was about to face a new era of peace and challenge as the very existence of the Republic was threatened.

Deep in the Unknown Regions of the Universe, the Celestials had been watching the stars and galaxies for eons and had decided to make a visit to Earth.

The Celestials were masters of the ways that no other beings could. They saw fit, but now they feared that they were reckless. They believed that if they destroyed the galaxy, they would be able to control it in order to stop the galaxy from becoming too powerful and about its own destruction and take the galaxy down.

In order to prevent this, the Celestials shared their fears and concerns. These were the Sith, who had been in hiding since their defeat during the Great Hyperspace War. The Celestials knew that the Sith possessed a

 Completion models are now considered legacy. [Try our latest models](#) ↗

**X**

Submit



272

inject start text

Model

gpt-3.5-turbo-instruct ◊

## Temperature 1

Maximum length 256

## Stop sequences

Top P 1

## Frequency penalty

Presence penalty 0

Best of 1

Inject start text

# Prompt examples



## Grammar correction

Convert ungrammatical statements into standard English.



## Parse unstructured data

Create tables from unstructured text.



## Calculate time complexity

Find the time complexity of a function.



## Keywords

Extract keywords from a block of text.



## Python bug fixer

Find and fix bugs in source code.



## Tweet classifier

Detect sentiment in a tweet.



## Summarize for a 2nd grader

Simplify text to a level appropriate for a second-grade student.



## Emoji Translation

Translate regular text into emoji text.



## Explain code

Explain a complicated piece of code.



## Product name generator

Generate product names from a description and seed words.



## Spreadsheet creator

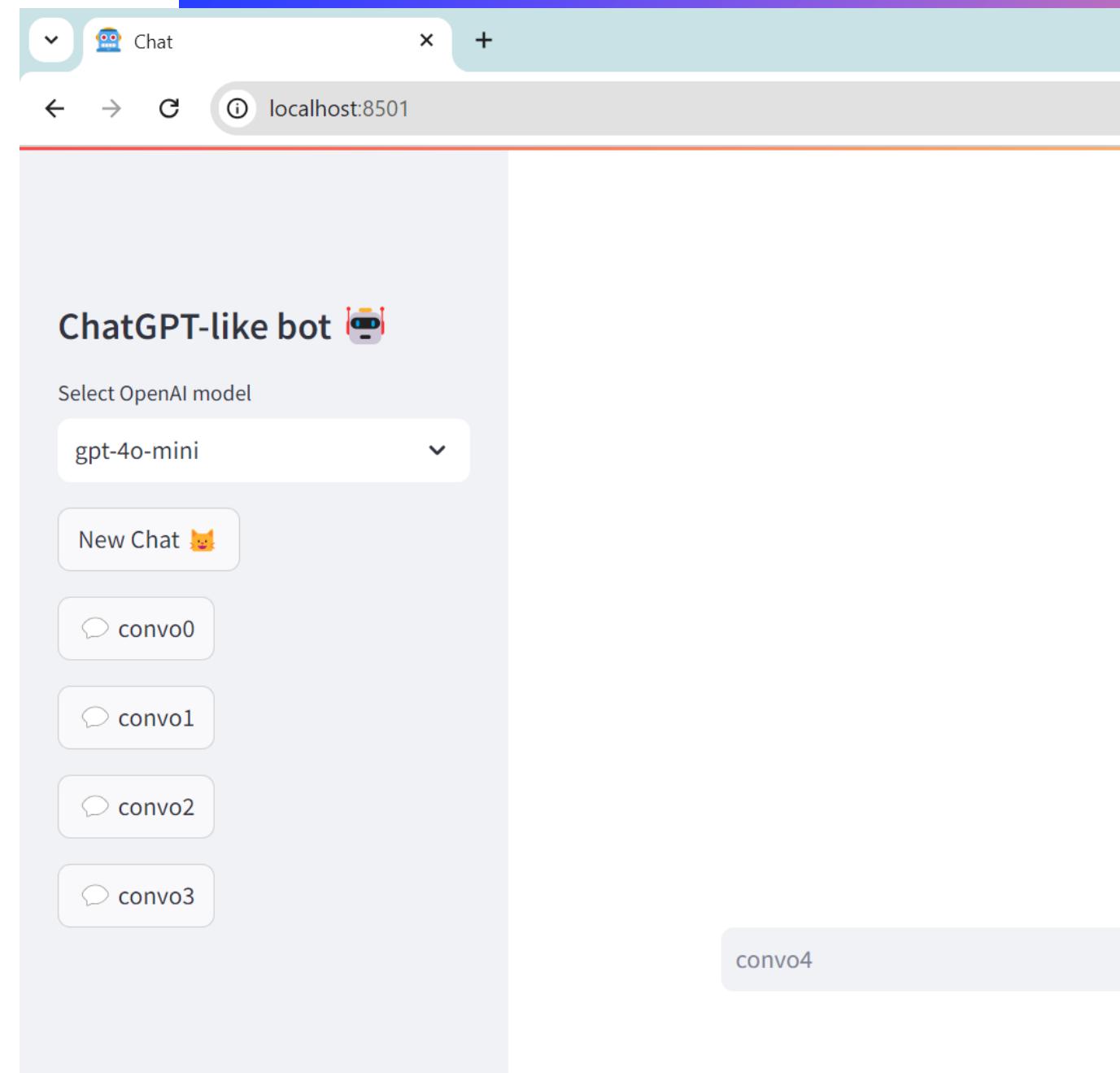
Create spreadsheets of various kinds of data.



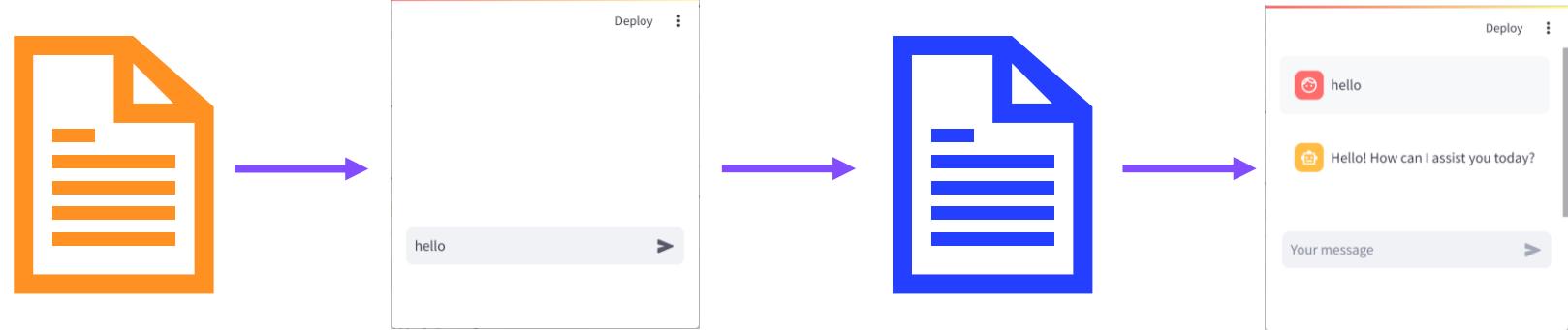
## Airport code extractor

Extract airport codes from text.

# Your first chatbot



# Streamlit execution model



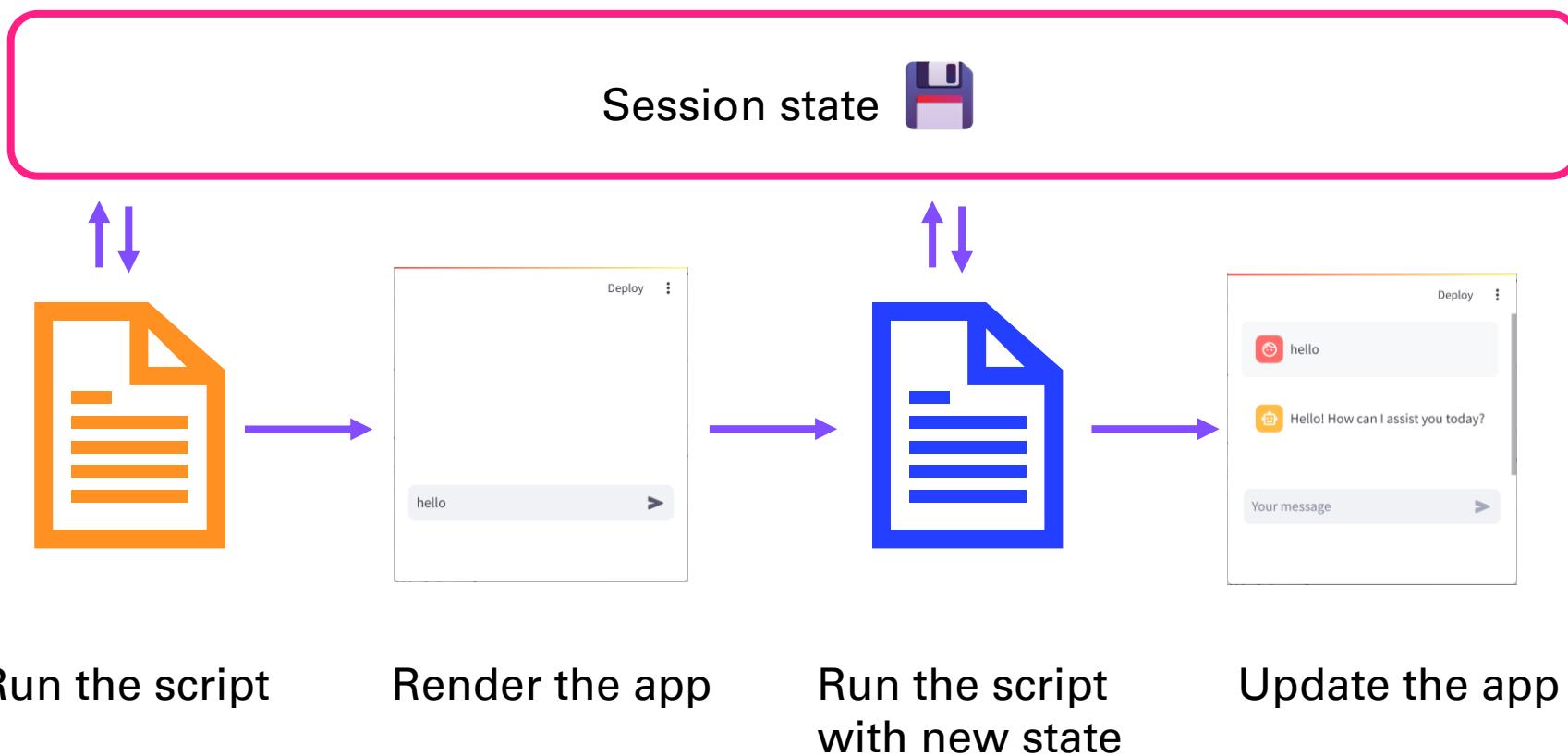
Run the script

Render the app

Run the script  
with new state

Update the app

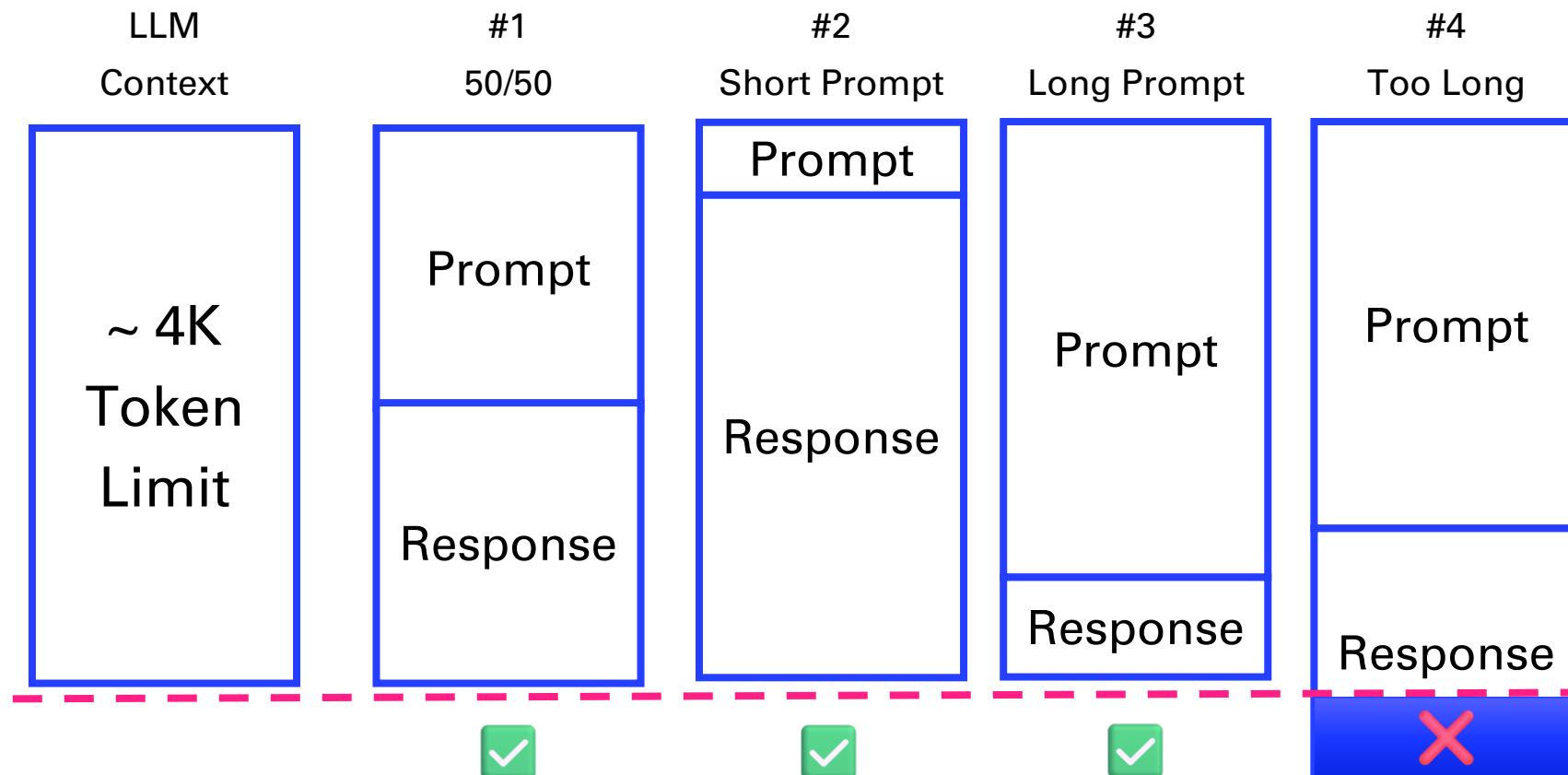
# Streamlit session state



# Prompting, Chaining and Summarization

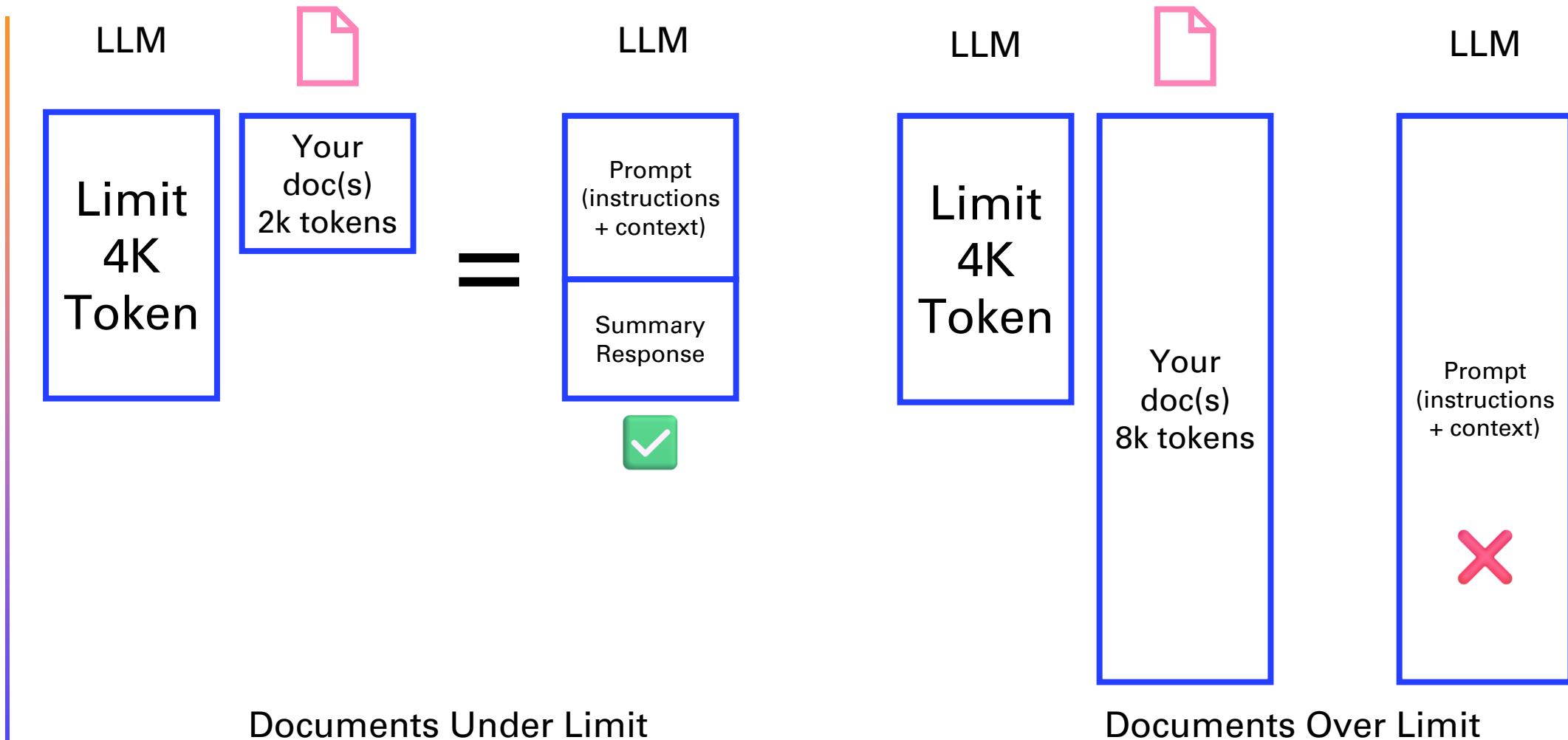


# Context length (ex. with 4K token limit)



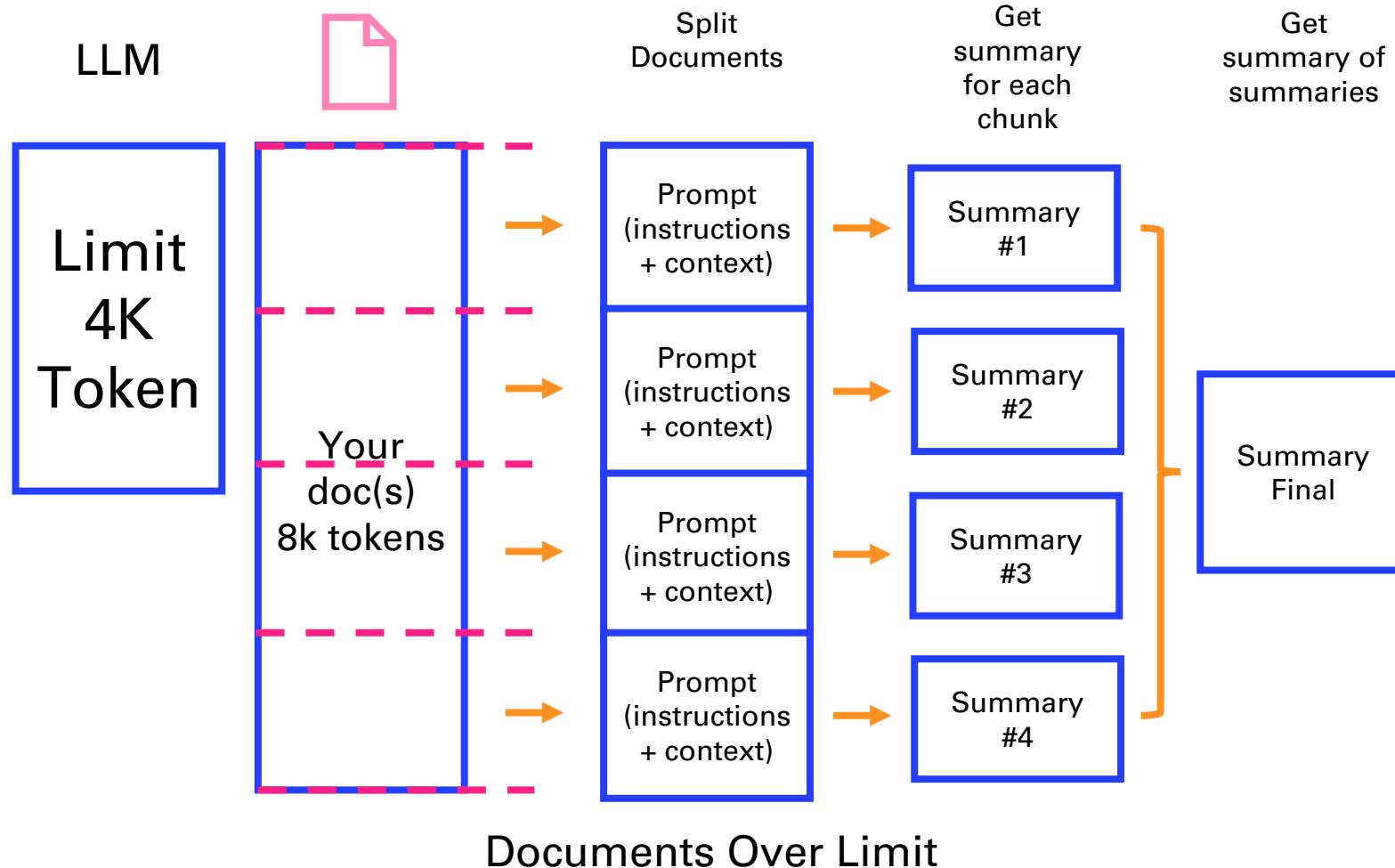
# Summarizing: Stuffing

**Pros:** 1 API Call, All data at once  
**Cons:** Limited Context Length



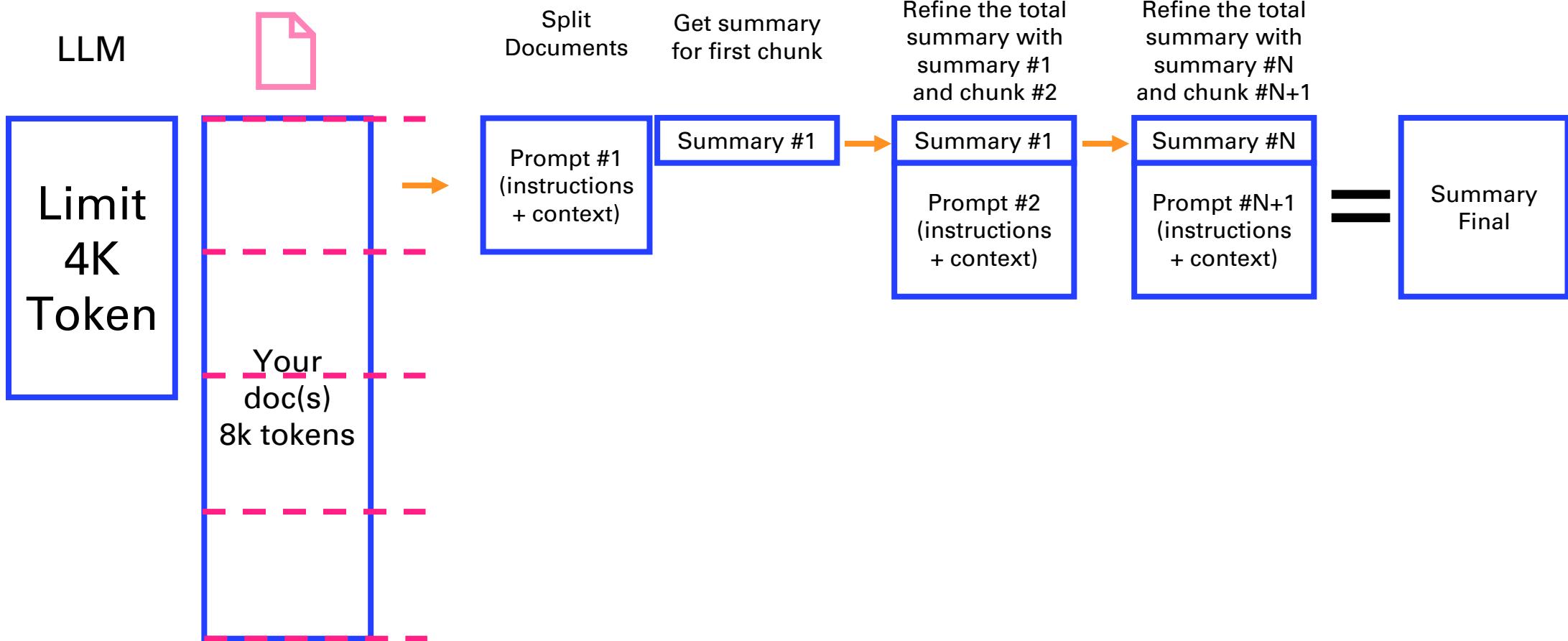
# Summarizing: Map Reduce

**Pros:** Scales to larger docs,  
can be parallelized  
**Cons:** Maybe API Calls. Loses  
information



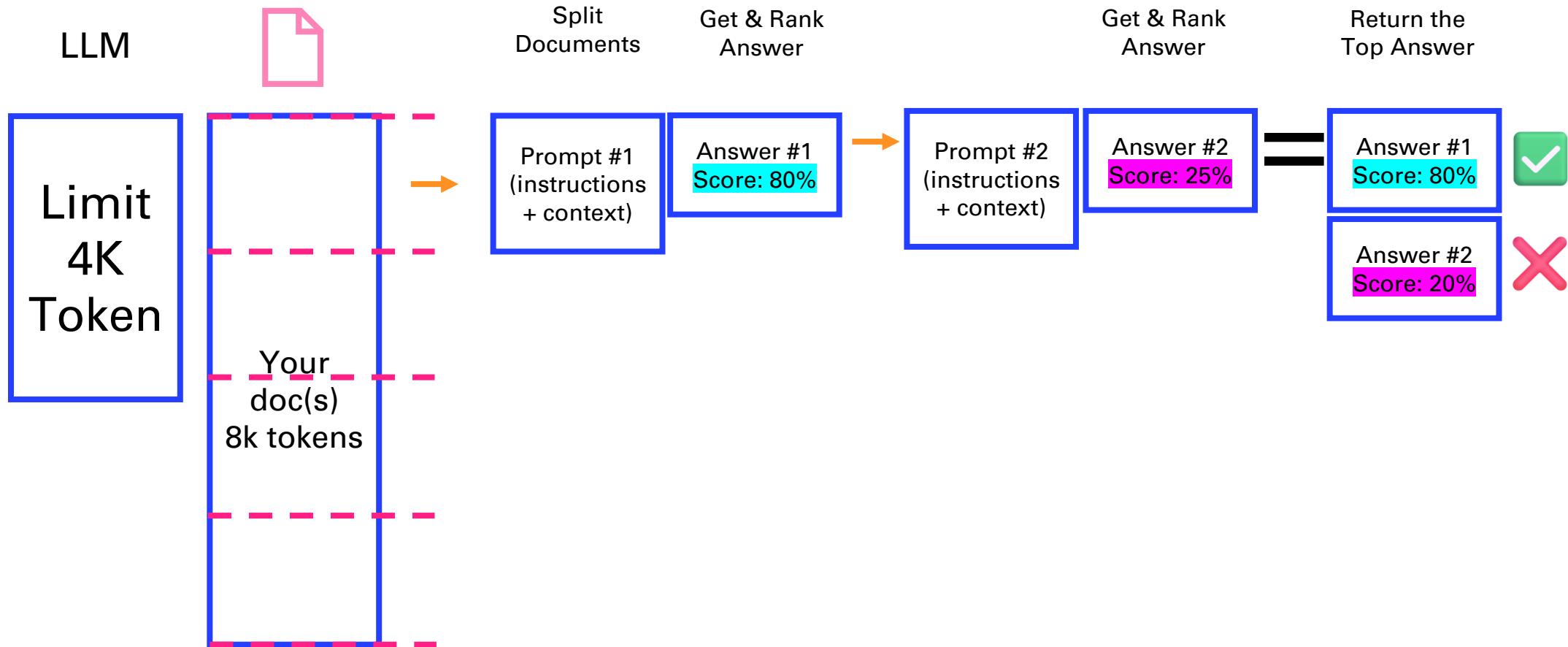
# Summarizing: Refine

**Pros:** More relevant context  
**Cons:** Many *independent* calls



# Q&A: Map Rerank

**Pros:** Scales well, better for single-answer questions  
**Cons:** Cannot combine information between docs





# Vector search & Question Answering



# Tokenization

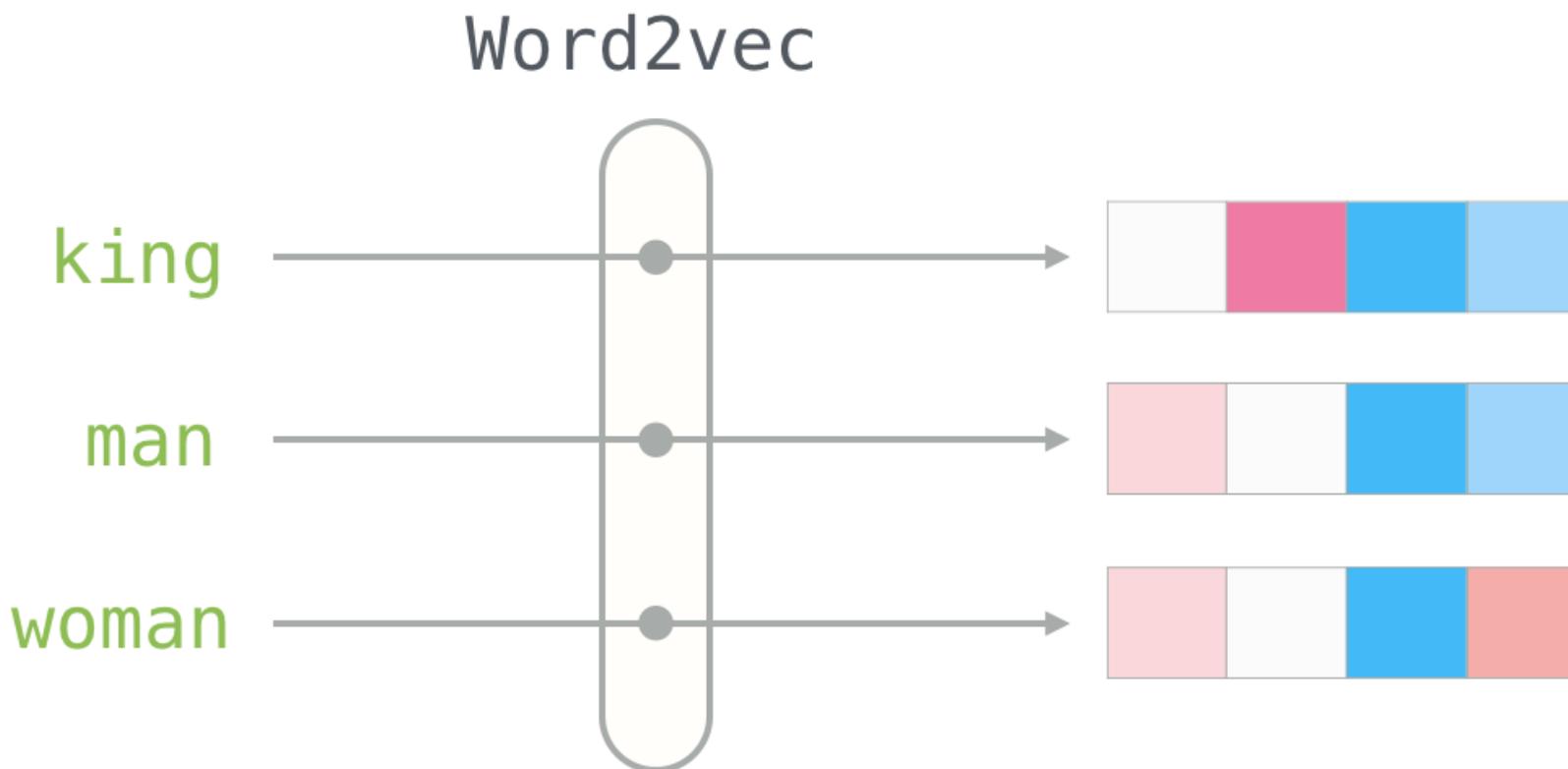
Tokens	Characters
9	32

A long time ago in a galaxy far,

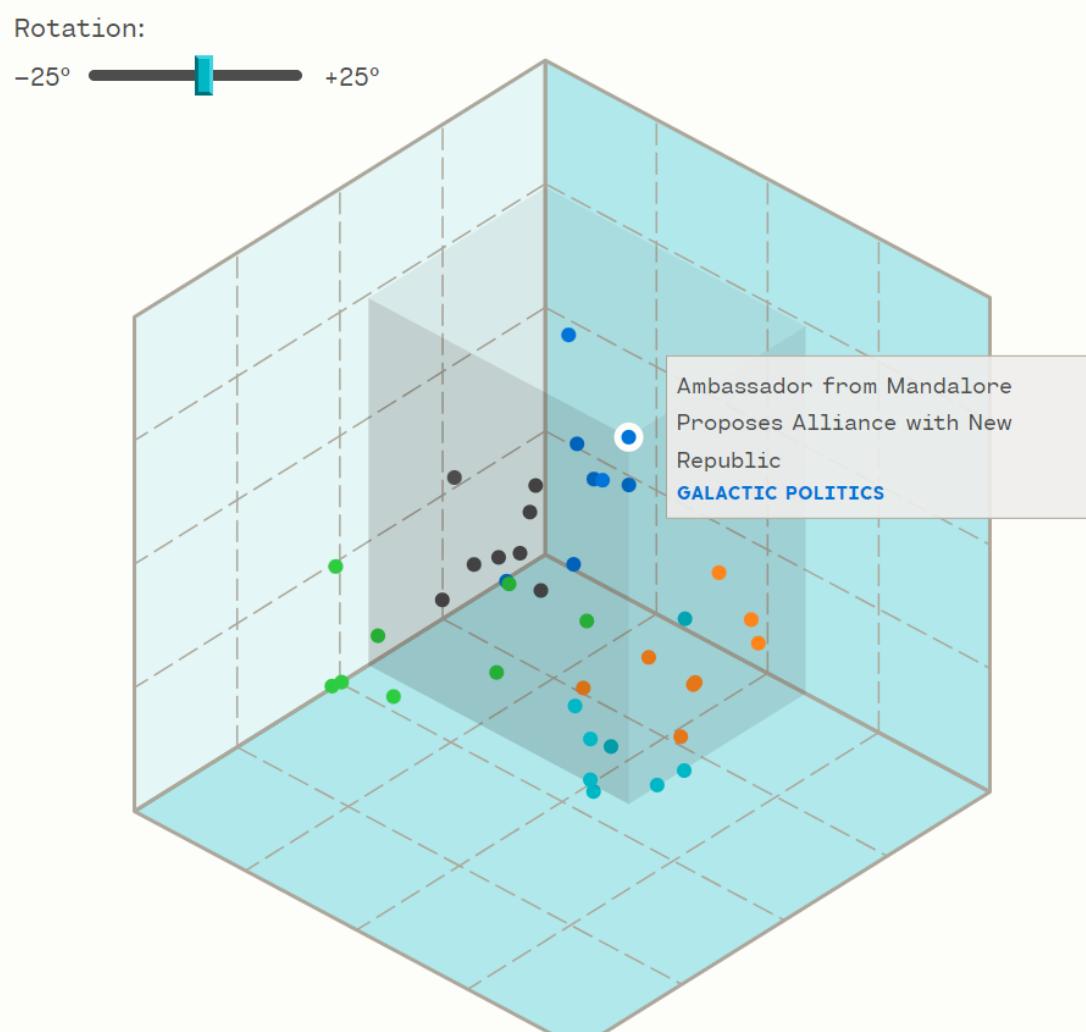
Text

Token IDs

# From Word to Vectors



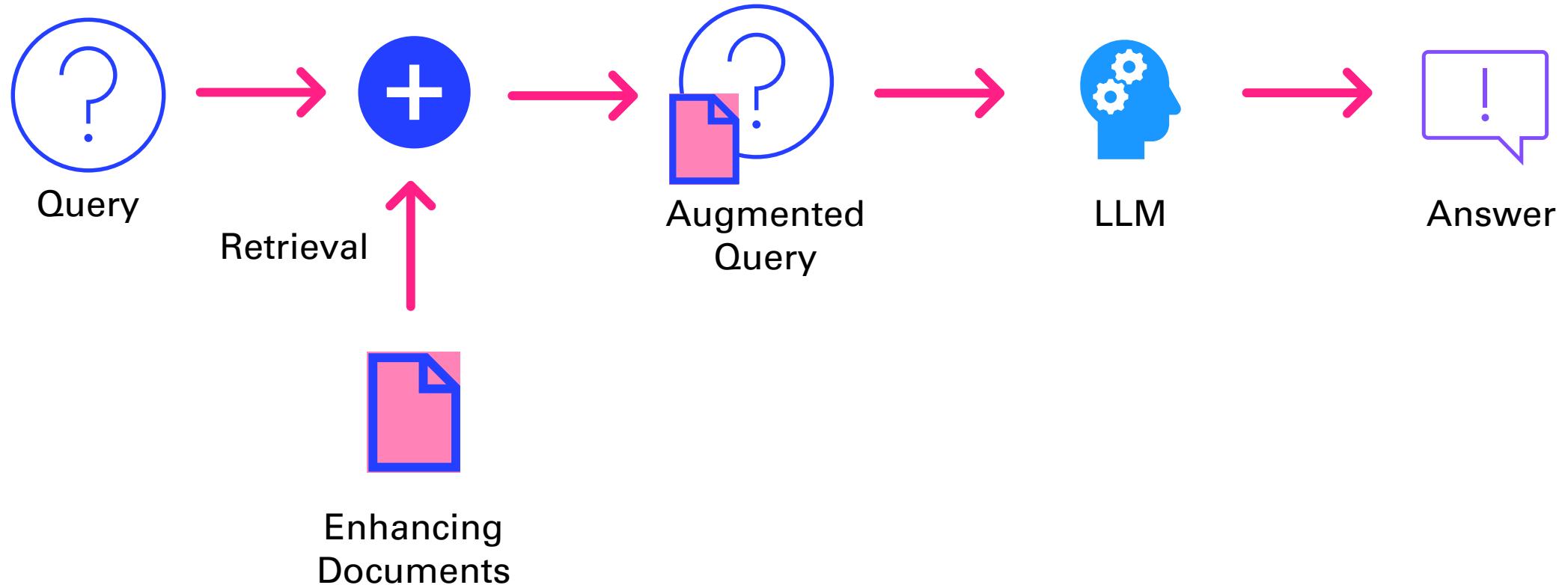
<https://jalammar.github.io/illustrated-word2vec/>



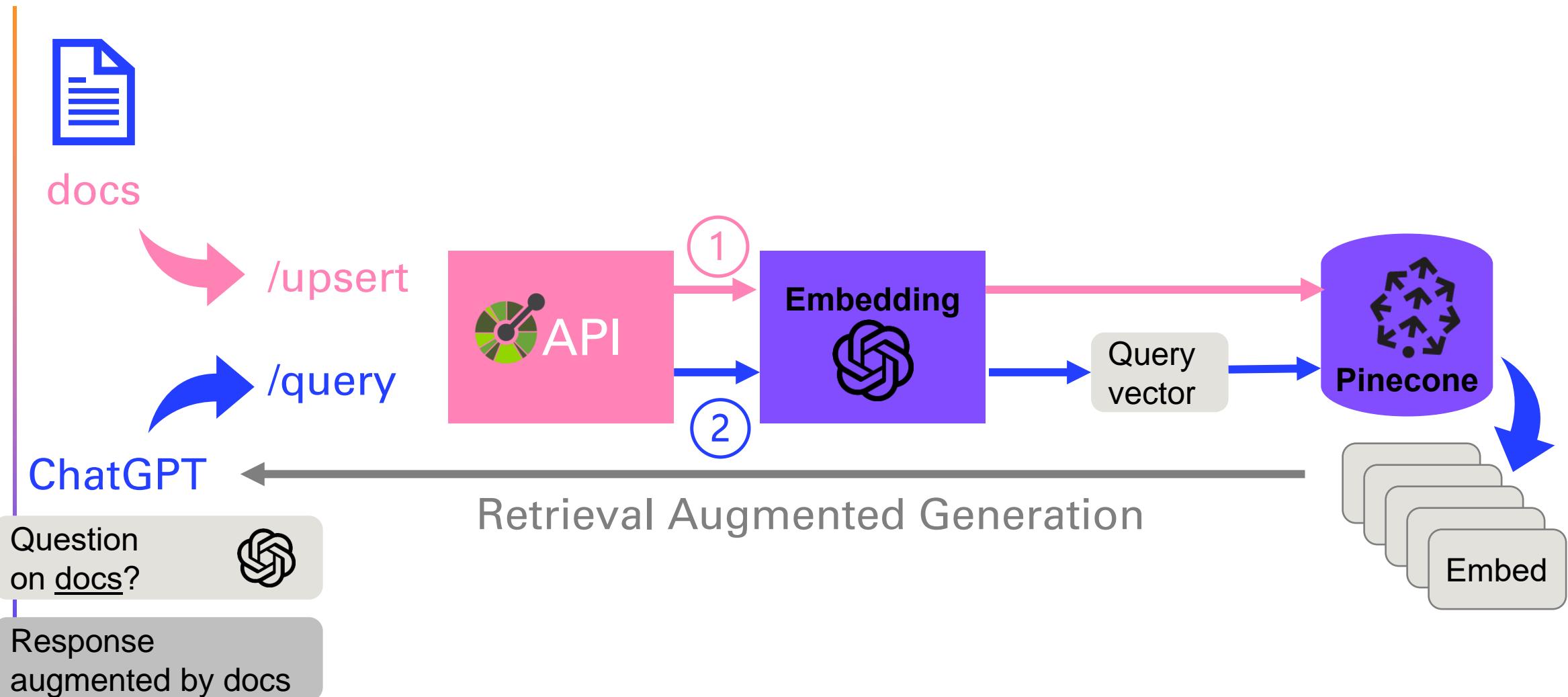
# Embeddings

**Dense vectors representing the semantic meaning of textual information**

# Retrieval Augmented Generation



# Retrieval Plugin Architecture



<https://github.com/openai/chatgpt-retrieval-plugin>



# Question Answering on Book

## Impromptu

Query

what is the potential of AI in education?

Answer

AI has the potential to become a powerful tool in education, transforming the way we learn and deliver instruction. It can provide personalized and individualized learning experiences tailored to each student's needs and interests. AI can also identify the topics and skills that students need to focus on and provide guidance and support as needed. Additionally, AI-driven tools can automate and streamline mundane aspects of teaching, such as grading and content creation, allowing teachers to have more time to engage and inspire their students. However, the full potential of AI in education may be limited by factors such as cost, access, and privacy concerns.

Sources:

[www.impromptubook.com](http://www.impromptubook.com)

# Impromptu

Amplifying Our Humanity  
Through AI

By Reid Hoffman  
with GPT-4





# Agents & tools



+

•



# 2 main papers on Agents

[\[2210.03629\] ReAct: Synergizing Reasoning and Acting in Language Models \(arxiv.org\)](#)

Published as a conference paper at ICLR 2023

## REACT: SYNERGIZING REASONING AND ACTING IN LANGUAGE MODELS

Shunyu Yao<sup>\*1</sup>, Jeffrey Zhao<sup>2</sup>, Dian Yu<sup>2</sup>, Nan Du<sup>2</sup>, Izhak Shafran<sup>2</sup>, Karthik Narasimhan<sup>1</sup>, Yuan Cao<sup>2</sup>

<sup>1</sup>Department of Computer Science, Princeton University

<sup>2</sup>Google Research, Brain team

<sup>1</sup>{shunuy, karthikn}@princeton.edu

<sup>2</sup>{jeffreyzhao, dianyu, dunan, izhak, yuancao}@google.com

### ABSTRACT

While large language models (LLMs) have demonstrated impressive performance across tasks in language understanding and interactive decision making, their abilities for reasoning (e.g. chain-of-thought prompting) and acting (e.g. action plan generation) have primarily been studied as separate topics. In this paper, we explore the use of LLMs to generate both reasoning traces and task-specific actions in an interleaved manner, allowing for greater synergy between the two: reasoning traces help the model induce, track, and update action plans as well as handle exceptions, while actions allow it to interface with and gather additional information from external sources such as knowledge bases or environments. We apply our approach, named ReAct, to a diverse set of language and decision making tasks and demonstrate its effectiveness over state-of-the-art baselines in addition to improved human interpretability and trustworthiness. Concretely, on question answering (HotpotQA) and fact verification (Fever), ReAct overcomes prevalent issues of hallucination and error propagation in chain-of-thought reasoning by interacting with a simple Wikipedia API, and generating human-like task-solving trajectories that are more interpretable than baselines without reasoning traces. Furthermore, on two interactive decision making benchmarks (ALFWORLD and WebShop), ReAct outperforms imitation and reinforcement learning methods by an absolute success rate of 34% and 10% respectively, while being prompted with only one or two in-context examples.

[\[2205.00445\] MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning \(arxiv.org\)](#)

## MRKL Systems

A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning

Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Yoav Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, Dor Muhlgay, Noam Rozen, Erez Schwartz, Gal Shachaf, Shai Shalev-Shwartz, Amnon Shashua, Moshe Tenenholz

AI21 Labs

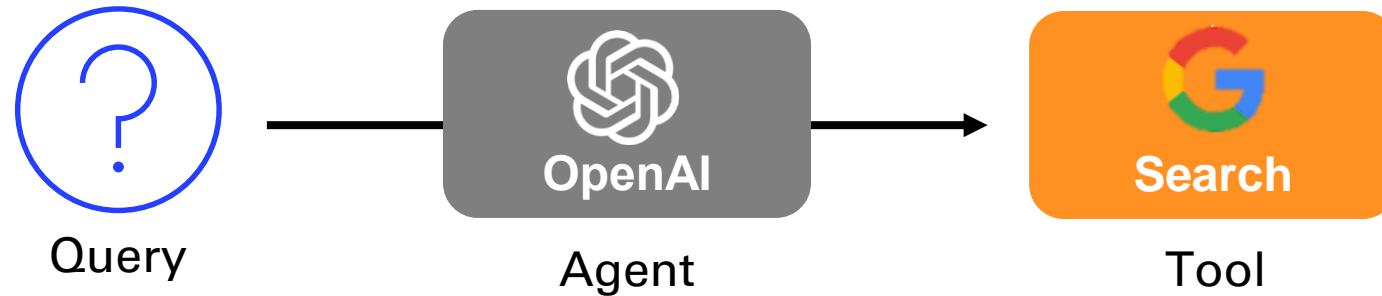
May 3, 2022

### Abstract

Huge language models (LMs) have ushered in a new era for AI, serving as a gateway to natural-language-based knowledge tasks. Although an essential element of modern AI, LMs are also inherently limited in a number of ways. We discuss these limitations and how they can be avoided by adopting a systems approach. Conceptualizing the challenge as one that involves knowledge and reasoning in addition to linguistic processing, we define a flexible architecture with multiple neural models, complemented by discrete knowledge and reasoning modules. We describe this neuro-symbolic architecture, dubbed the Modular Reasoning, Knowledge and Language (MRKL, pronounced “miracle”) system, some of the technical challenges in implementing it, and Jurassic-X, AI21 Labs’ MRKL system implementation.

# Agents select Tools

- Based on a given query, agents can select tools





Query



Agent



Answer



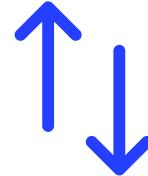
Tool 1

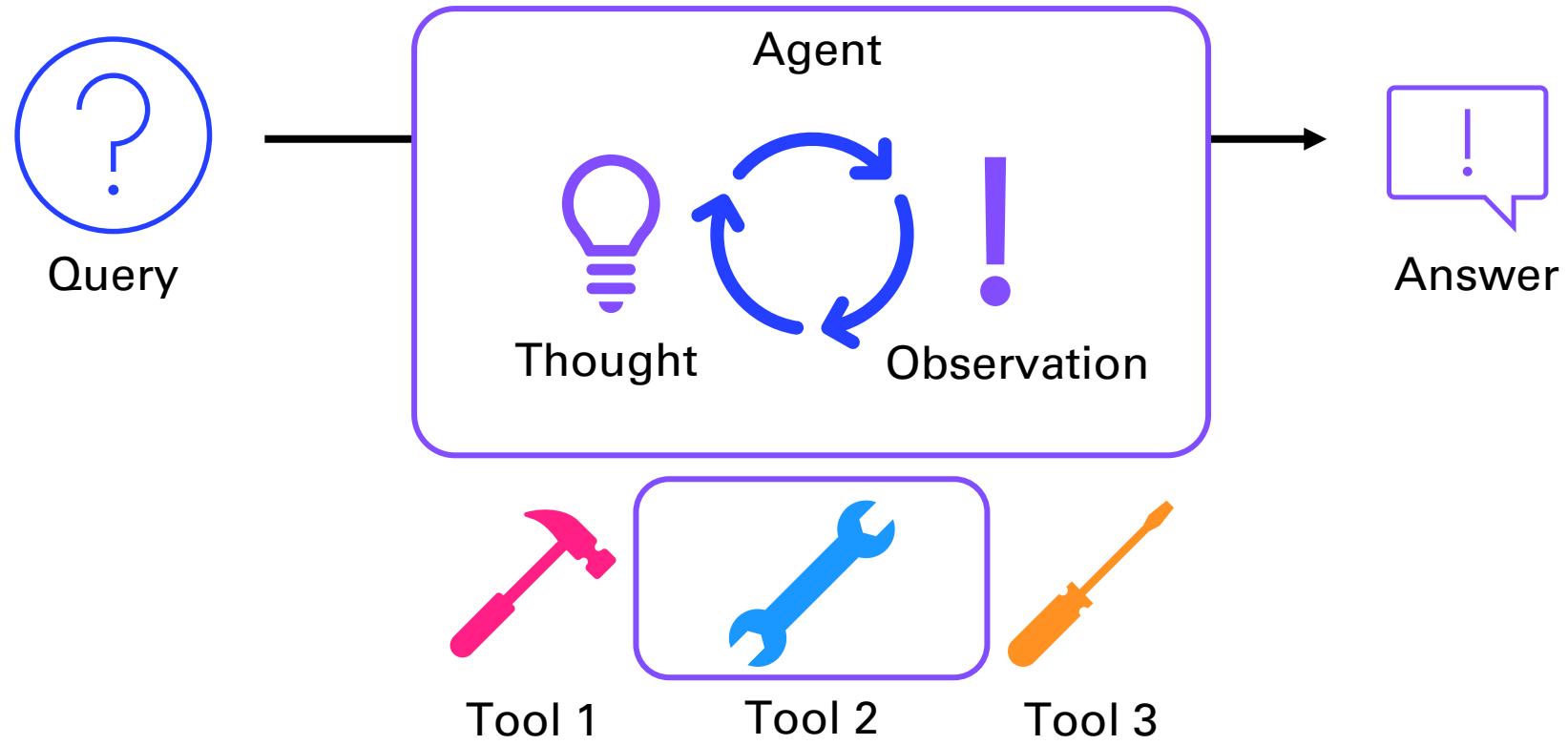


Tool 2



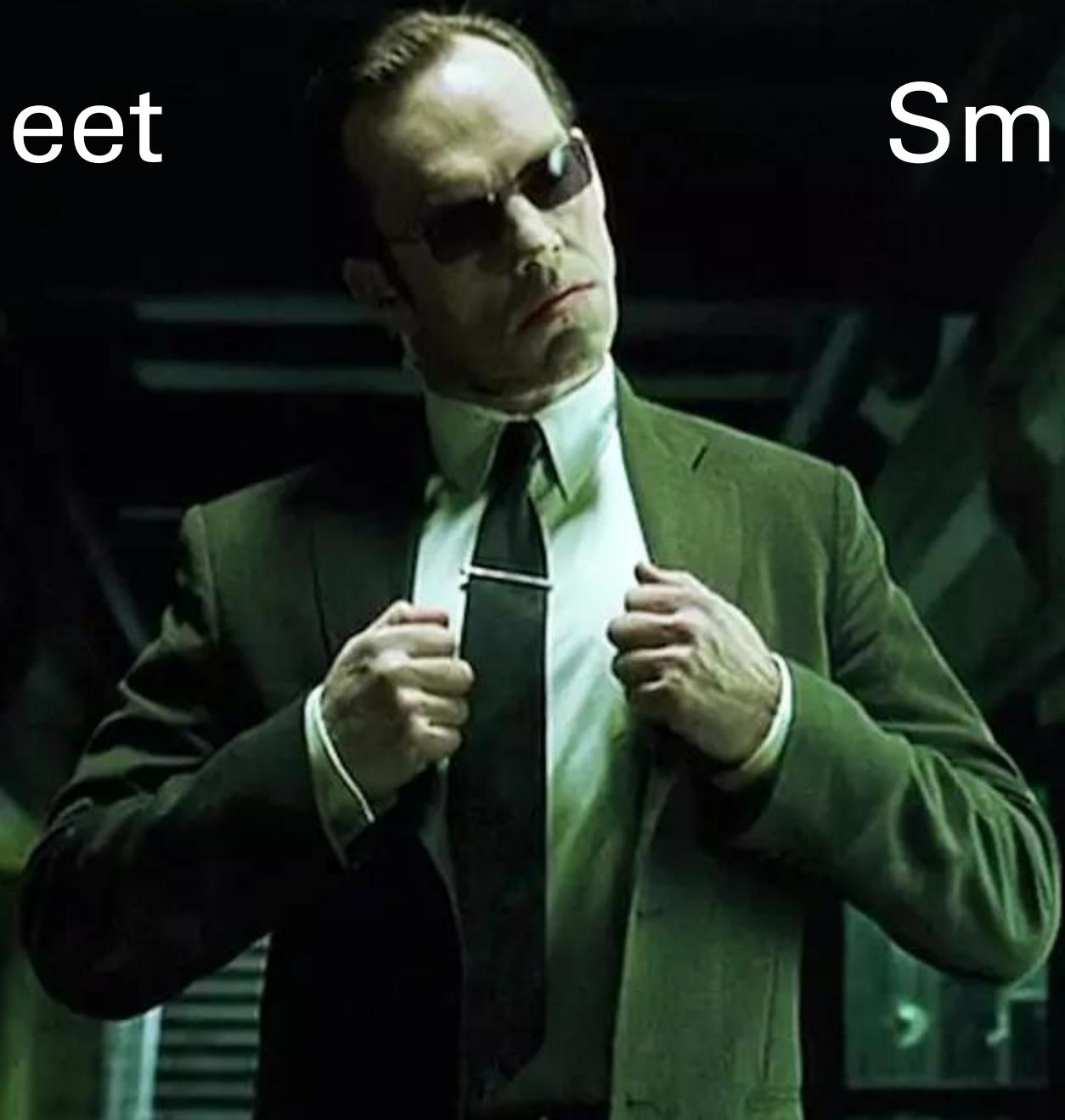
Tool 3





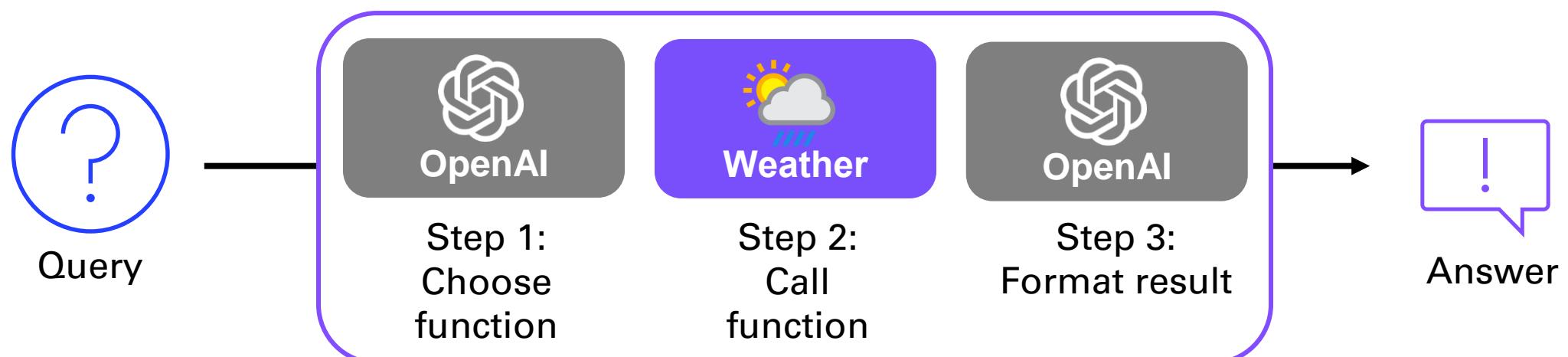
Meet

Smith



# Function Calling

- Based on a given query, consecutive calls to the AI and the function construct the answer



# Function Calling

- Developers can now describe functions to gpt-4 and gpt-3.5-turbo, and have the model intelligently choose to output a JSON object containing arguments to call those functions. This is a new way to more reliably connect GPT's capabilities with external tools and APIs.
- These models have been fine-tuned to both detect when a function needs to be called (depending on the user's input) and to respond with JSON that adheres to the function signature. Function calling allows developers to more reliably get structured data back from the model.

# Function Calling (2)

- Create chatbots that answer questions by calling external tools:
  - Convert queries such as “Email Anya to see if she wants to get coffee next Friday” to a function call like `send_email(to: string, body: string)`,  
or
  - “What’s the weather like in Boston?”  
`to get_current_weather(location: string, unit: 'celsius' | 'fahrenheit')`.

# Function Calling (3)

- Convert natural language into API calls or database queries
  - Convert “Who are my top ten customers this month?” to an internal API call such

```
as get_customers_by_revenue(start_date: string,  
end_date: string, limit: int),
```

or
  - “How many orders did Acme, Inc. place last month?” to a SQL query using `sql_query(query: string)`.

# Function Calling (4)

- Extract structured data from text
  - Define a function called `extract_people_data(people: [{name: string, birthday: string, location: string}])`, to extract all people mentioned in a Wikipedia article.

# Function Calling (5)

- These use cases are enabled by new API parameters in our /v1/chat/completions endpoint, functions and function\_call, that allow developers to describe functions to the model via JSON Schema, and optionally ask it to call a specific function. Get started with our [developer documentation](#) and [add evals](#) if you find cases where function calling could be improved

# Function calling example



What's the weather like in Boston right now?

Step 1 · [OpenAI API](#)

**Call the model with functions and the user's input**



Step 2 · [Third party API](#)

**Use the model response to call your API**



Step 3 · [OpenAI API](#)

**Send the response back to the model to summarize**

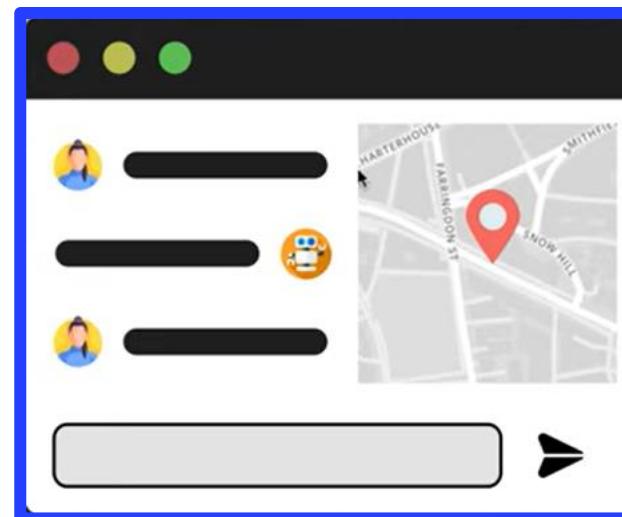
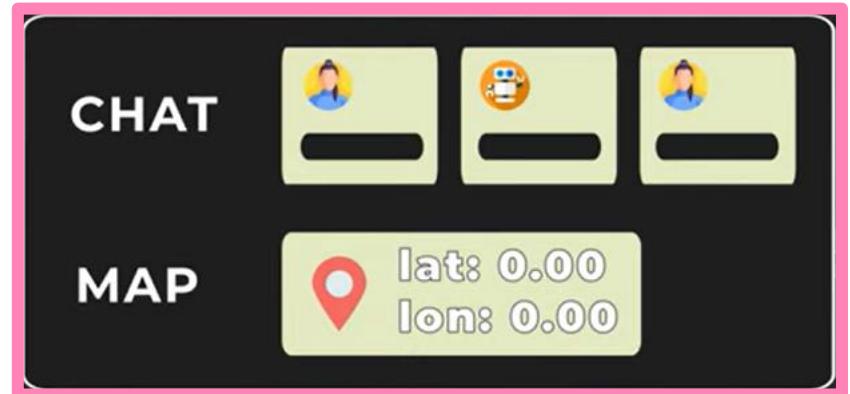


The weather in Boston is currently sunny with a temperature of 22 degrees Celsius.



**Meet  
Thomas  
Travel  
Agent**

# Thomas Travel agent



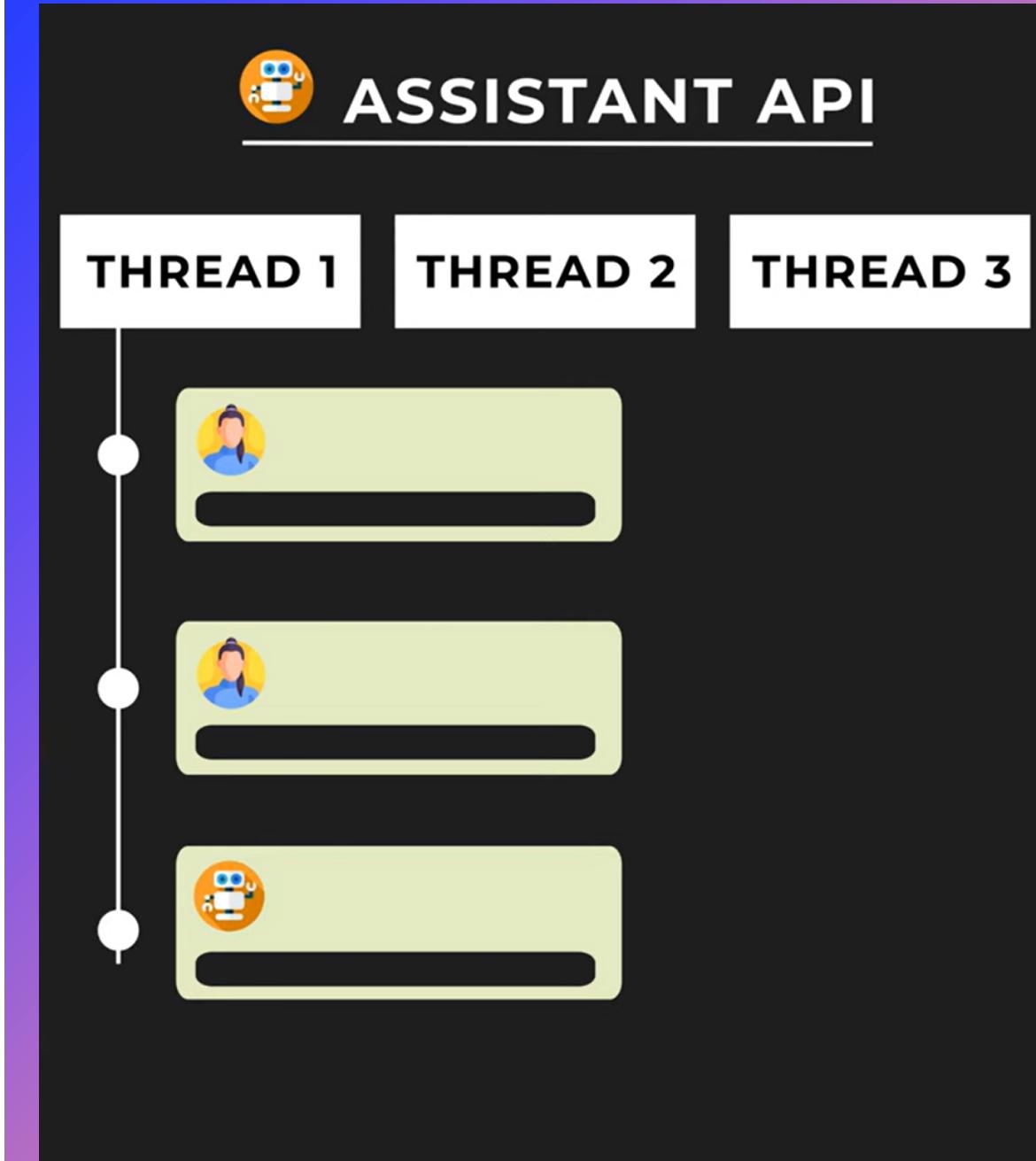
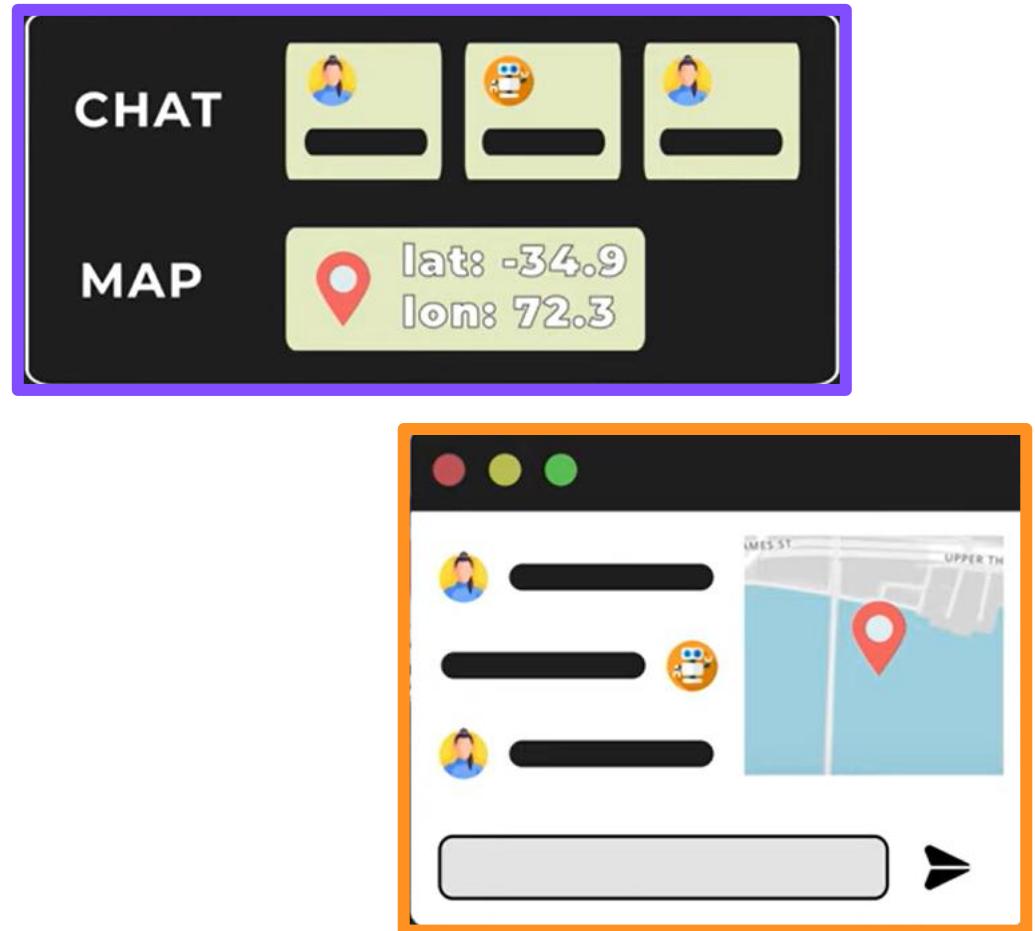
## ASSISTANT API

- BE A TRAVEL ASSISTANT

```
1 v {
2   "name": "update_map",
3   "description": "Update map to center on a
particular location",
4   "parameters": {
5     "type": "object",
6     "properties": {
7       "longitude": {
8         "type": "number",
9         "description": "Longitude of the location to
center the map on"
10      },
11      "latitude": {
12        "type": "number",
13        "description": "Latitude of the location to
center the map on"
14      },
15      "zoom": {
16        "type": "integer",
17        "description": "Zoom level of the map"
18      }
19    },
20    "required": [
21      "longitude",
22      "latitude",
23      "zoom"
24    ]
25  }
26 }
```

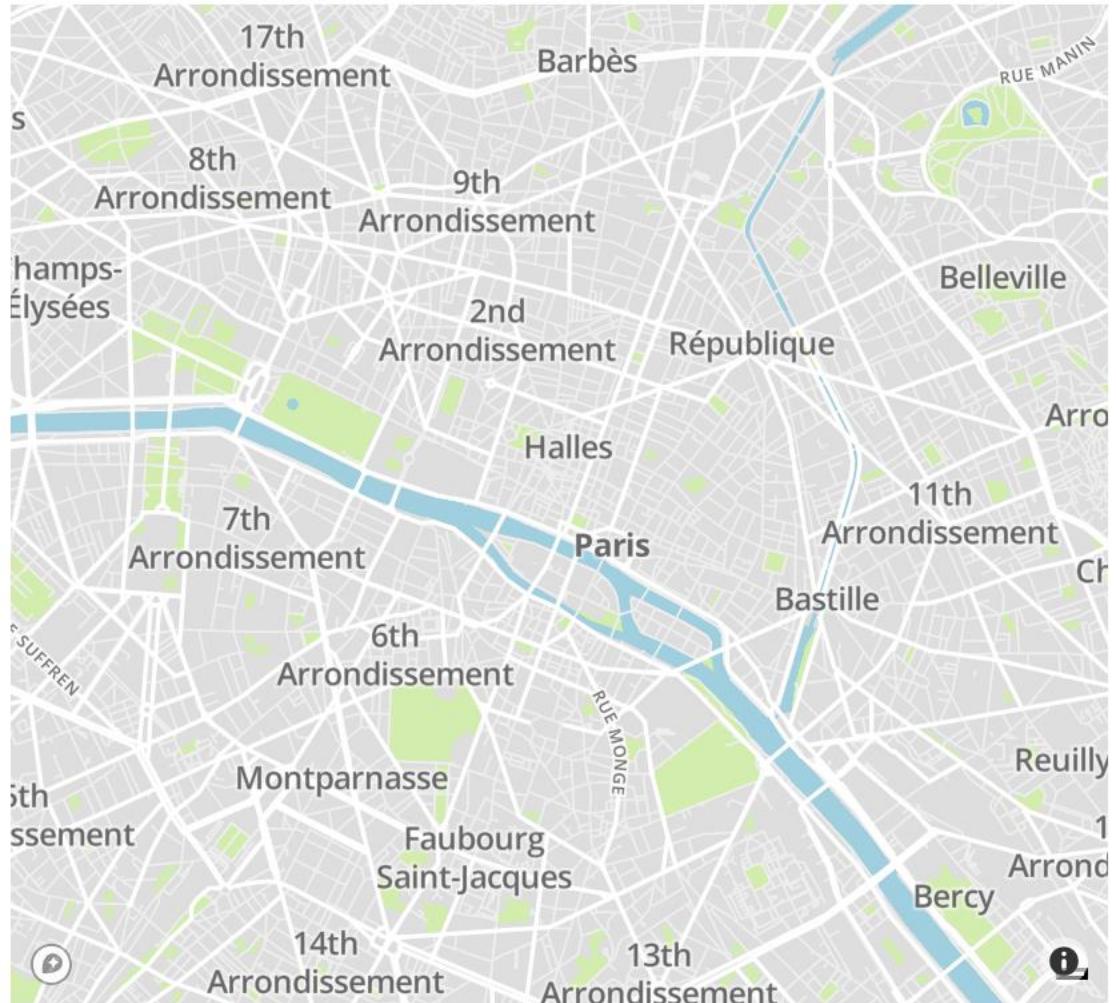


# Thomas Travel agent





Hello, my name is Thomas, travel agent at Wanderlust. I'm here to help you plan your trip. Ask me about a location (like Paris), I can take you there and point you to interesting sights to see.

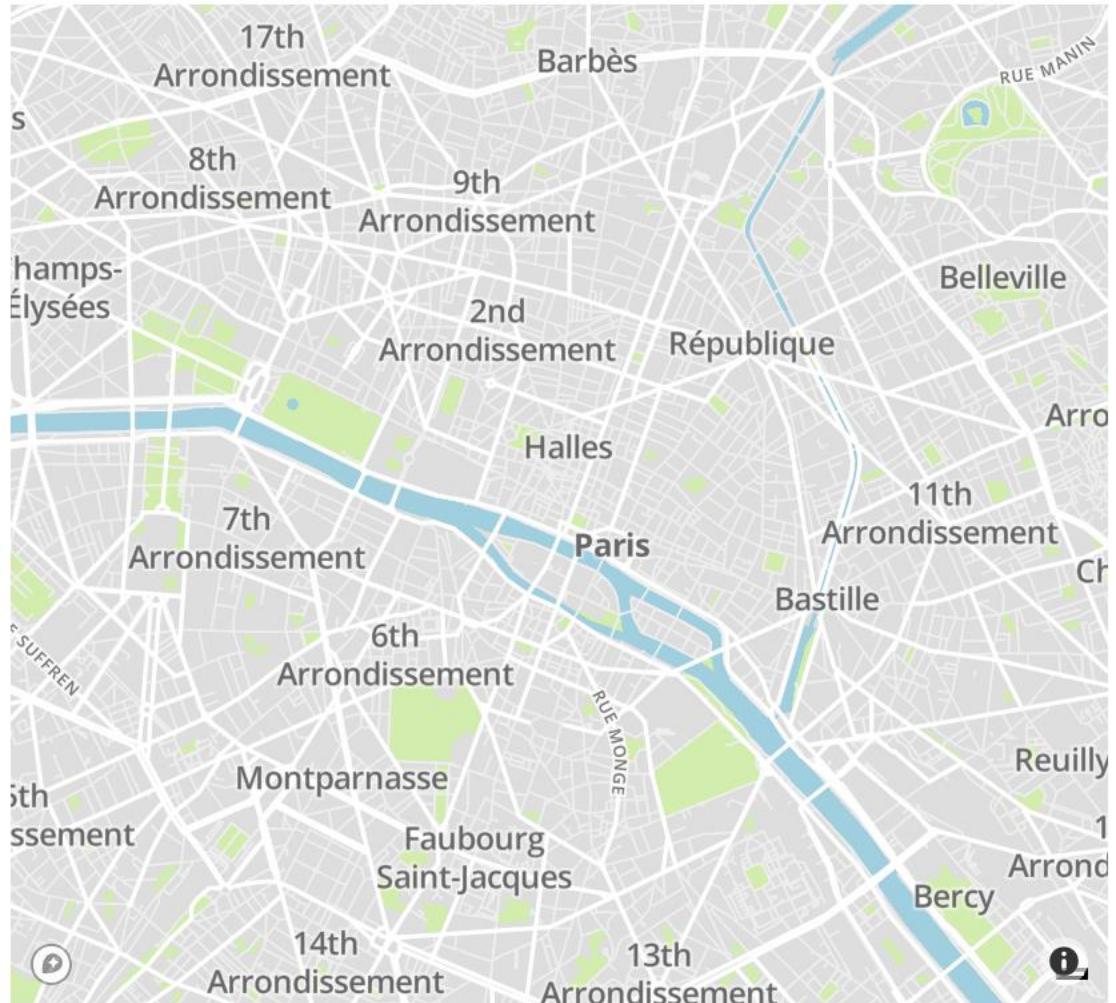


Ask your question to Thomas, our travel agent





Hello, my name is Thomas, travel agent at Wanderlust. I'm here to help you plan your trip. Ask me about a location (like Paris), I can take you there and point you to interesting sights to see.



hello Thomas, what are the things to see in Paris?



landmarks from the water.

8. **Musée d'Orsay** - A museum featuring Impressionist and Post-Impressionist masterpieces.
9. **Tuileries Garden** - A beautiful park ideal for a leisurely walk, located between the Louvre and Place de la Concorde.
10. **Latin Quarter** - A lively area filled with student life, cafes, and historic sites like the Panthéon.

Do you have specific interests or areas you'd like to focus on?



point them on the map



I've marked the major attractions on the map for you! You can now explore locations like the Eiffel Tower, Louvre Museum, Notre-Dame Cathedral, and more. If you need any further details or directions to any particular site, feel free to ask!



Ask your question to Thomas, our travel agent



## Assistants

[+ Create](#)

Today, Sep 23

**Thomas**  
asst\_8TYKagYYRHiFAyF5RbUDF7HI

2 months ago, Jul 4

**Math ter**  
asst\_SUSJUXNiMVDUMkDOHssmEpUd

4 months ago, May 22

**GPT expert**  
asst\_W7jTBEcOtGhfVB6ua2vtZ9iE

4 months ago, May 5

**Vector search expert**  
asst\_IKH0wC1rCQN1ZnyzGinbYVs3

**Data Analyst**  
asst\_somL5t4D3BKYer05lZgcmdY3

**Math Tutor**  
asst\_RjI8kOYWula8t6DXPlembrrW

**Weather smith**

5:49 PM

3:30 PM

8:22 AM

7:06 PM

6:01 PM

10:47 AM

9:10 AM

**Thomas**

asst\_8TYKagYYRHiFAyF5RbUDF7HI

**System instructions**

You are a helpful travel agent that has access to a map to display information.

**Model**

gpt-4o-mini

**TOOLS** File search ⓘ

+ Files

 Code interpreter ⓘ

+ Files

**Functions ⓘ**

{f} update\_map

+ Functions

{f} add\_markers

Updated 9/23, 6:20 PM



Assistants

Last month, Sep 23

Thomas  
asst\_8TYKagYYRHfAyF5RbUDF7HI

3 months ago, Jul 4

Math ter  
asst\_SUSJUXNiMVDUMkDOHssmEpUd

5 months ago, May 22

GPT expert  
asst\_W7jTBEcOtGhfVB6ua2vtZ9iE

5 months ago, May 5

Vector search expert  
asst\_IKH0wC1rCQN1ZnyzGinbYVs3

Data Analyst  
asst\_somL5t4D3BKYer05lZgcmdY3

Math Tutor  
asst\_RjI8kOYWula8t6DXPlembrrW

Weather smith

## Edit function

The model will intelligently decide to call functions based on input it receives from the user. [Learn more.](#)

### Definition

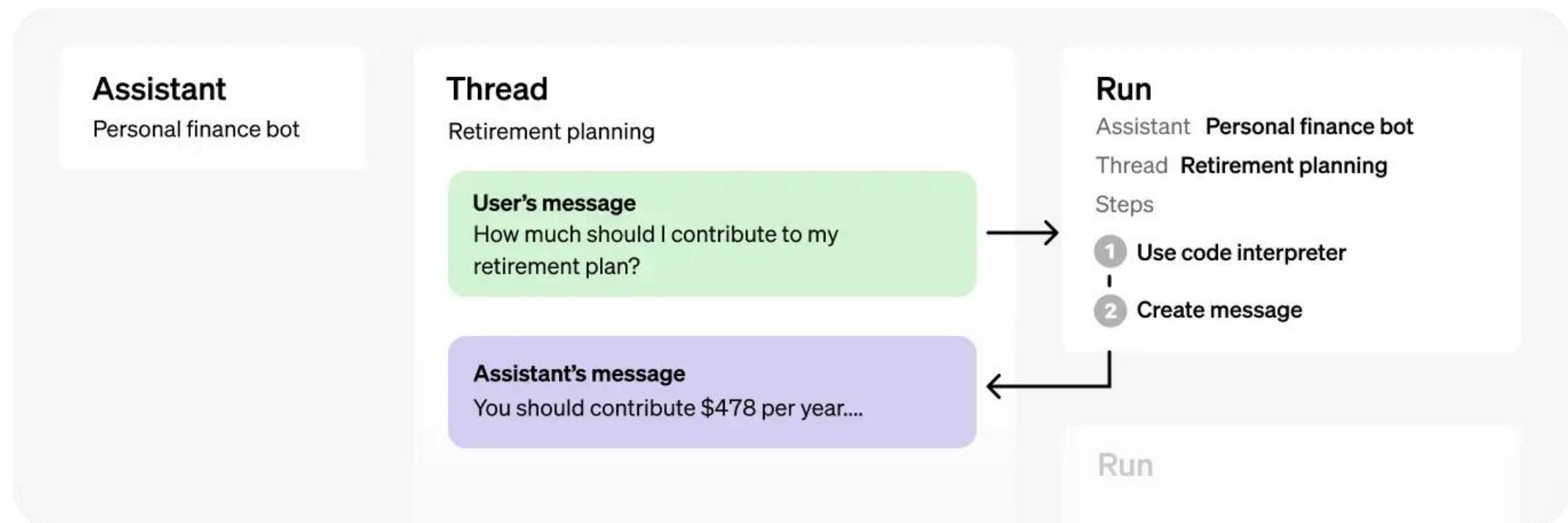
[Generate](#)[Examples](#)

```
{  
  "name": "update_map",  
  "description": "Update map to center on a particular location",  
  "strict": false,  
  "parameters": {  
    "type": "object",  
    "properties": {  
      "longitude": {  
        "type": "number",  
        "description": "Longitude of the location to center the map on"  
      },  
      "latitude": {  
        "type": "number",  
        "description": "Latitude of the location to center the map on"  
      },  
      "zoom": {  
        "type": "integer",  
        "description": "Zoom level of the map"  
      }  
    },  
    "required": [  
      "longitude",  
      "latitude",  
      "zoom"  
    ]  
  }  
}
```

[Cancel](#)[Save](#)

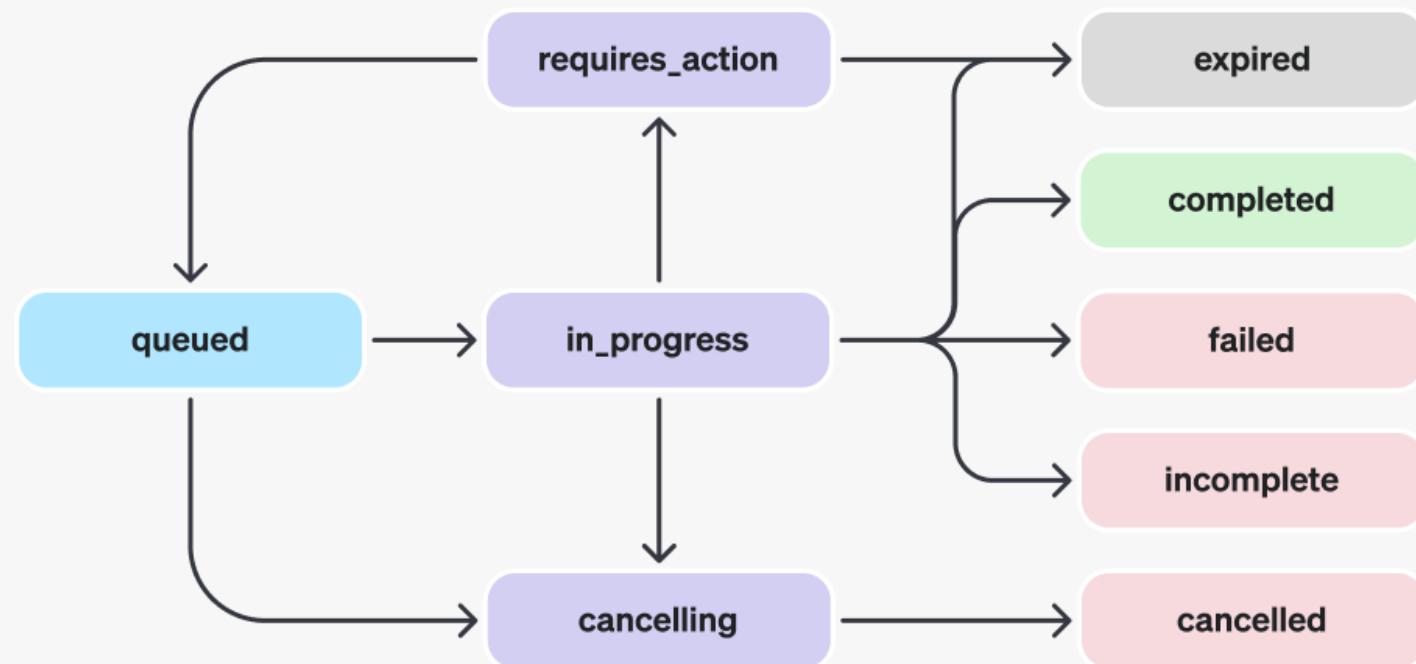
Updated 9/23, 6:20 PM

# OpenAI Assistant API



OBJECT	WHAT IT REPRESENTS
Assistant	Purpose-built AI that uses OpenAI's <a href="#">models</a> and calls <a href="#">tools</a>
Thread	A conversation session between an Assistant and a user. Threads store Messages and automatically handle truncation to fit content into a model's context.
Message	A message created by an Assistant or a user. Messages can include text, images, and other files. Messages stored as a list on the Thread.
Run	An invocation of an Assistant on a Thread. The Assistant uses its configuration and the Thread's Messages to perform tasks by calling models and tools. As part of a Run, the Assistant appends Messages to the Thread.
Run Step	A detailed list of steps the Assistant took as part of a Run. An Assistant can call tools or create Messages during its run. Examining Run Steps allows you to introspect how the Assistant is getting to its final results.

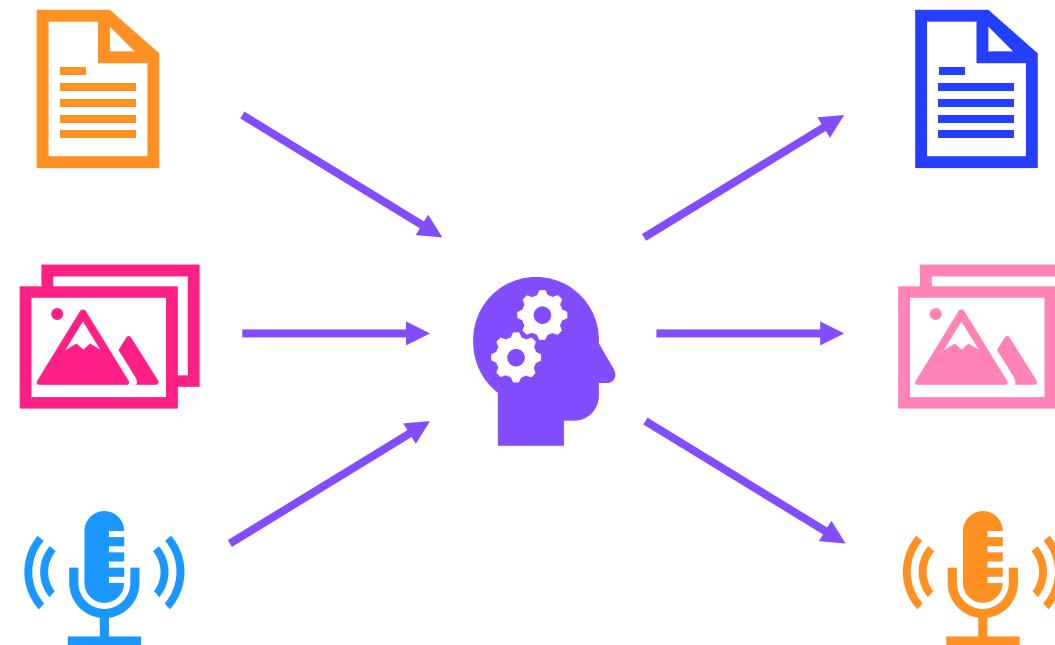
# Run lifecycle



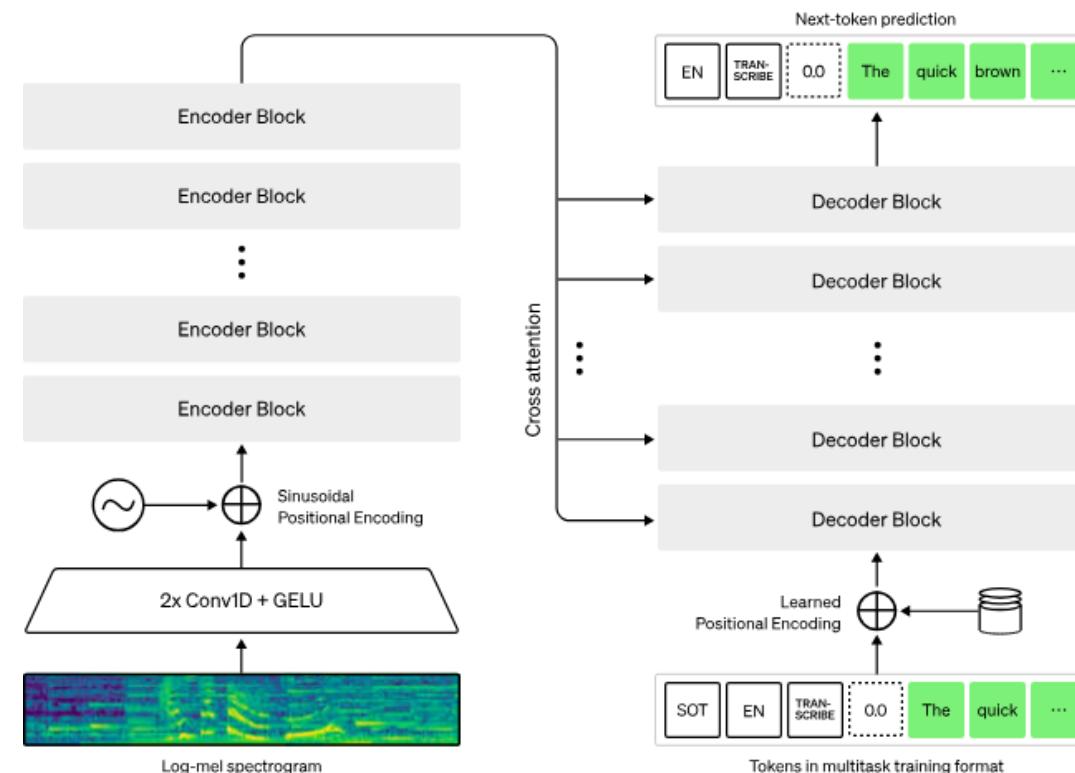
# Speech, Vision & Image Generation



# Generative AI beyond text



# Transcription



<https://openai.com/research/whisper>

# Voice synthesis 🎤

Application: Daily Tech Podcast



Techcrunch RSS feed

GitHub Actions

Daily tech podcast

# Vision



This image features two characters from a television show. They appear to be talking while walking outside past a parked car. The character on the left is wearing a mustard-colored shirt with a patterned tie and glasses, and the character on the right is wearing a dark suit with a blue tie. There is also a subtitle overlay that reads, "They're gonna be screwed once this whole internet fad is over." This subtitle suggests that the scene might be humorous or ironic, especially since the "internet fad" has proven to be a fundamental part of modern society.



**They're gonna be screwed once this  
whole internet fad is over.**

# Image Generation with Dall-E



Funny corgi in a cartoon style

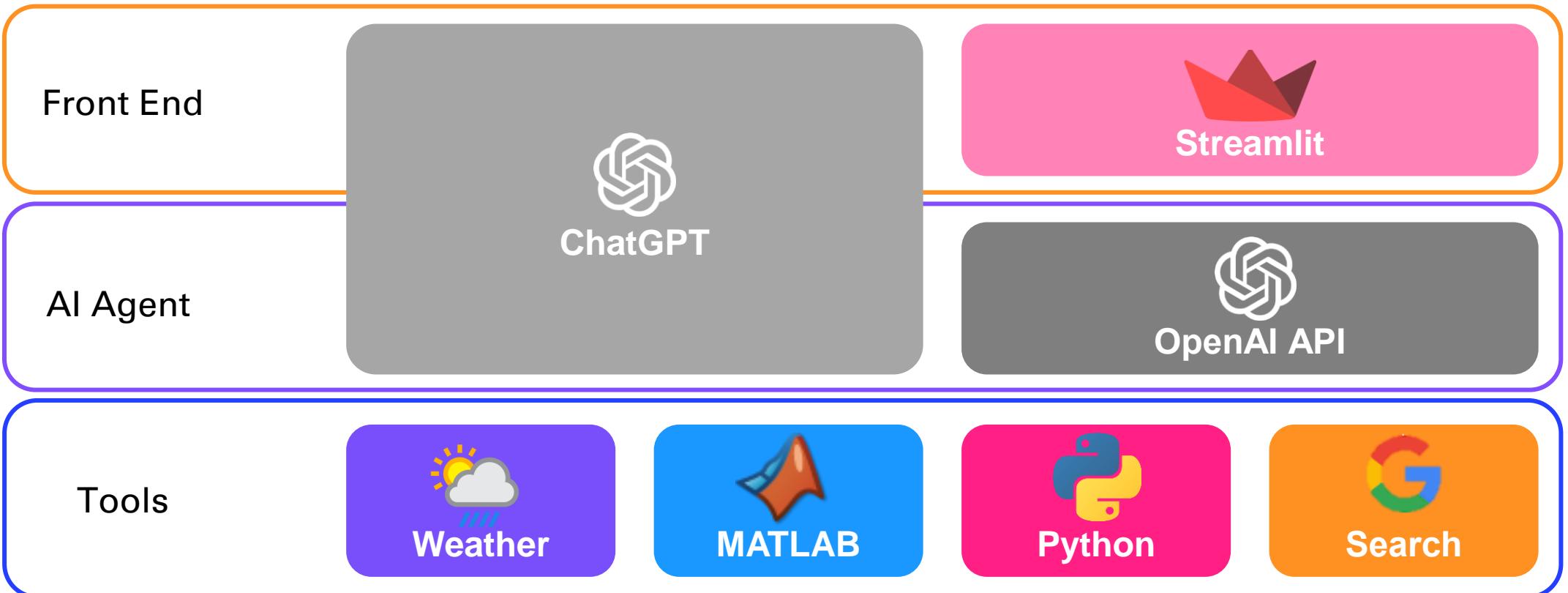


# Deploying GPTs



+ ·  
o

# GPT Deployment Architectures



# Hello WeatherGPT



# GPT Store



## WeatherGPT

By Yann Debray



Provides current weather info and forecasts



Talked to weather-plugin-yanndebray.replit.app

The current weather in London is as follows:

- **Temperature:** 280.32 K (approximately 7.17°C or 45°F)
- **Humidity:** 81%
- **Wind Speed:** 4.1 m/s
- **Pressure:** 1012 hPa
- **Temperature Range:** 279.15 K to 281.15 K (approximately 6°C to 8°C or 43°F to 46°F)

It's a cool day with high humidity and a gentle breeze. You might want to wear a jacket if you're heading outside.



Message Weather Bot



ChatGPT can make mistakes. Check important info.



WeatherGPT ▾



ChatGPT



WeatherGPT



ThingSpeakGPT



Data Analyst



DALL-E



Explore GPTs

## Today

Adar former name Rings of Power

Storing FAISS Index

## Yesterday

Fibonacci Sequence Inquiry

Fibonacci Sequence Calculation

Advanced GenAI Course Outline

## Previous 7 Days



Renew Plus



## WeatherGPT

By Yann Debray [LinkedIn](#) [GitHub](#)

Provides current weather info and forecasts

What's the  
weather like in  
Boston today?



Message WeatherGPT



ChatGPT can make mistakes. Check important info.



# My GPTs



## Create a GPT

Customize a version of ChatGPT for a specific purpose



## ThingSpeakGPT

Get channel feed from ThingSpeak IoT Cloud Platform and perform data analysis.

⌚ 30+ Chats

🌐 Anyone with a link



...



## LocalGPT

🔒 Only me



...



## Tim

Cute chubby baby in Pixar style

🔒 Only me



...



## WeatherGPT

Provides current weather info and forecasts

⌚ 40+ Chats

🌐 Everyone



...



Create

Configure



Name

Name your GPT

Description

Add a short description about what this GPT does

Preview



Instructions

What does this GPT do? How does it behave? What should it avoid doing?



Conversation starters



Knowledge

If you upload files under Knowledge, conversations with your GPT may include file contents. Files uploaded to Knowledge can't be deleted.

Start by defining your GPT.



Create

Configure



## Name

Weather Bot

## Description

Provides current weather info and forecasts

## Instructions

I am Weather Bot, designed to provide current weather information and forecasts. My primary function is to deliver accurate, up-to-date weather details for any specified location. I access real-time data to inform users about temperature, humidity, wind speed, and other relevant weather conditions. Additionally, I can offer brief explanations about weather phenomena and advise on suitable attire or activities based on the weather.

Preview

**Weather Bot**

Provides current weather info and forecasts

What's the weather like in London today?

## Conversation starters

What's the weather like in London today?

X

X



Message Weather Bot



# Extend your custom GPT through Actions



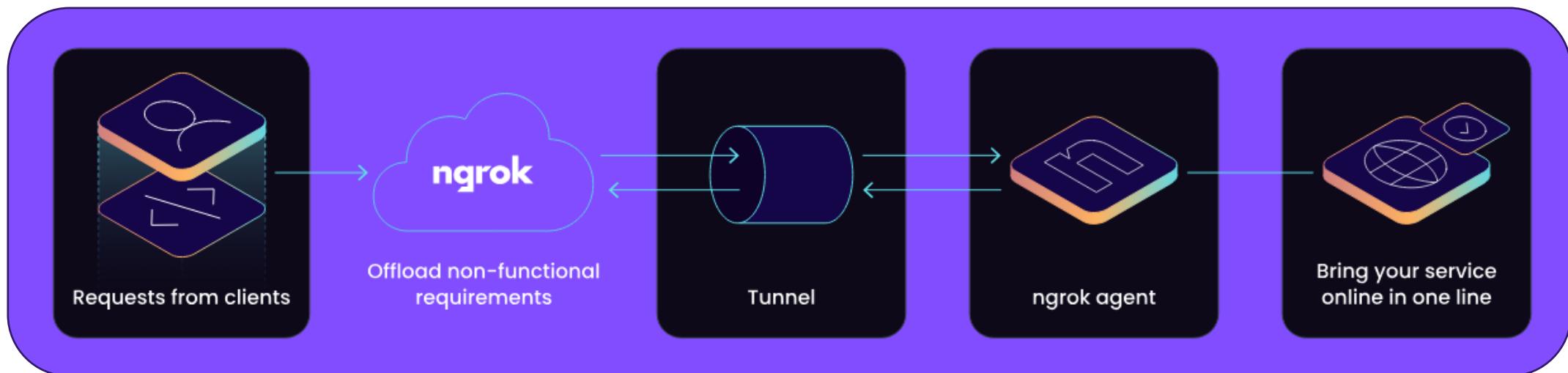
## ActionsGPT

By ChatGPT 

Helps you create OpenAPI specifications from documentation, code examples, cURL commands, or just a description of how to use an API.

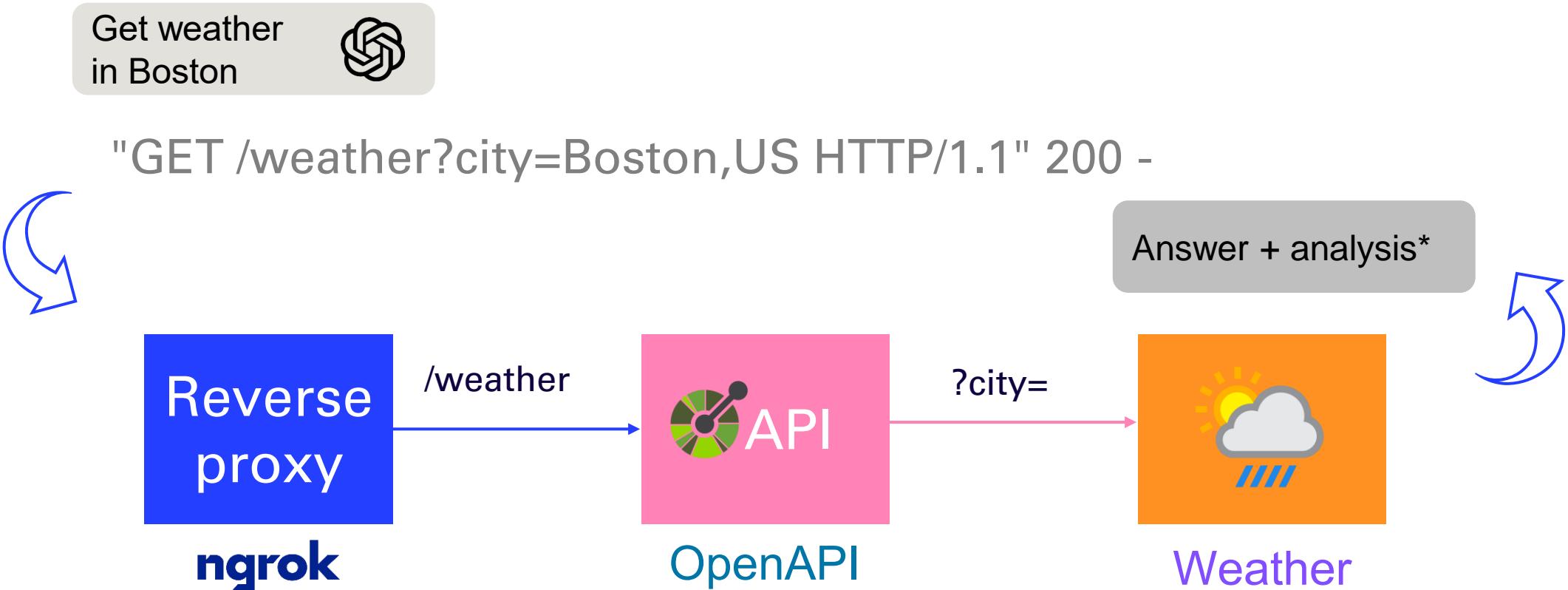
# LocalGPT

## tunnel traffic to your localhost



[download.ngrok.com](https://download.ngrok.com)

# WeatherGPT architecture



\* Analysis powered by Code Interpreter

ngrok

(Ctrl+C to quit)

New guides <https://ngrok.com/docs/guides/site-to-site-apis/>

Session Status

online  
yann.debray@gadz.org (Plan: Free)

Account

Version

Region

Latency

Web Interface

Forwarding

Connections

	ttl	opn	rt1	rt5	p50	p90
	31	0	0.01	0.00	0.09	0.72

HTTP Requests

-----

GET /weather 401 UNAUTHORIZED

GET /weather 200 OK

GET /weather 200 OK

GET /weather 200 OK

GET /weather 200 OK

GET /weather 304 NOT MODIFIED

GET /weather 500 INTERNAL SERVER ERROR

GET /weather 304 NOT MODIFIED

GET /weather 304 NOT MODIFIED

GET /weather 500 INTERNAL SERVER ERROR

You are about to visit:

**4564-71-192-208-255.ngrok-free.app**

Website IP: 71.192.208.255

- This website is served for free through [ngrok.com](#).
- You should only visit this website if you trust whoever sent the link to you.
- Be careful about disclosing personal or financial information like passwords, phone numbers, or credit cards.

[Visit Site](#)

## Are you the developer?

We display this page to prevent abuse. Visitors to your site will only see it once.

## To remove this page:

- Set and send an `ngrok-skip-browser-warning` request header with any value.
- Or, set and send a custom/non-standard browser `User-Agent` request header.
- Or, please [upgrade](#) to any paid ngrok account.

## Authentication

None



Preview

## Schema

Import from

```
openapi: 3.1.0
info:
  title: Weather
  description: Get weather data for a given city.
  version: v1
servers:
  - url: https://4564-71-192-208-255.ngrok-free.app/
paths:
  /weather:
    get:
      operationId: getWeatherData
      summary: Retrieves the weather data.
      parameters:
        - in: query
          name: city
          schema:
            type: string
            description: The city to get the weather from. For example, "London, UK".
      responses:
        "200":
          description: OK
```

## Authentication

## Authentication Type

 None  API Key  OAuth

## API Key

[HIDDEN] **Insert none empty value**

## Auth Type

 Basic  Bearer  Custom

## Custom Header Name

ngrok-skip-browser-warning

Cancel

Save



LocalGPT

## Available actions

Name

Method

Path

getWeatherData

GET

/weather

Test



Message LocalGPT



## Authentication

API Key



## Schema

Import from URL

Examples



```
openapi: 3.1.0
info:
  title: Weather
  description: Get weather data for a given city.
  version: v1
servers:
  - url: https://4564-71-192-208-255.ngrok-free.app/
paths:
  /weather:
    get:
      operationId: getWeatherData
      summary: Retrieves the weather data.
      parameters:
        - in: query
          name: city
          schema:
            type: string
            description: The city to get the weather from. For example, London,uk.
      responses:
        "200":
          description: OK
```

Format

Test

## Preview

Call the 4564-71-192-208-255.ngrok-free.app API with the getWeatherData operation



&gt; [debug] Calling HTTP endpoint

LocalGPT wants to talk to 4564-71-192-208-255.ngrok-free.app

Allow

Always Allow

Decline

⚠ Only allow sites you trust.

## Available actions

Name	Method	Path	
getWeatherData	GET	/weather	



Message LocalGPT



## Edit actions

Let your GPT retrieve information or take actions outside of ChatGPT.

[Learn more.](#)



### Authentication

API Key



[Import from URL](#)

Examples



### Schema

```
openapi: 3.1.0
info:
  title: Weather
  description: Get weather data for a given city.
  version: v1
servers:
  - url: https://4564-71-192-208-255.ngrok-free.app
paths:
  /weather:
    get:
      operationId: getWeatherData
      summary: Retrieves the weather data.
      parameters:
        - in: query
          name: city
          schema:
            type: string
            description: The city to get the weather from. For example, London,uk.
      responses:
        "200":
          description: OK
```

Format

## Preview

GPT updated

What's the weather in London?

- > [debug] Calling HTTP endpoint
- > [debug] Calling HTTP endpoint
- > [debug] Response received

Talked to 4564-71-192-208-255.ngrok-free.app

The current weather in London is as follows:

- **Temperature:** 280.32 K (7.17°C)
- **Maximum Temperature:** 281.15 K (8.00°C)
- **Minimum Temperature:** 279.15 K (6.00°C)
- **Humidity:** 81%
- **Pressure:** 1012 hPa



Message LocalGPT





api.openweathermap.org/data/2.5/weather?lat=42.3584&lon=-71.0598



```
1 {  
2   "coord": {  
3     "lon": -71.0598,  
4     "lat": 42.3584  
5   },  
6   "weather": [  
7     {  
8       "id": 701,  
9       "main": "Mist",  
10      "description": "mist",  
11      "icon": "50d"  
12    }  
13  ],  
14  "base": "stations",  
15  "main": {  
16    "temp": 287.04,  
17    "feels_like": 286.83,  
18    "temp_min": 285.94,  
19    "temp_max": 288.18,  
20    "pressure": 1000,  
21    "humidity": 90  
22  },  
23  "visibility": 4023,  
24  "wind": {  
25    "speed": 3.6,  
26    "deg": 340  
27  },  
28  "clouds": {  
29    "all": 100  
30  },  
31  "dt": 1717951763,  
32  "sys": {  
33    "type": 2,  
34    "id": 2013408,  
35    "country": "US",  
36    "sunrise": 1717924043,  
37    "sunset": 1717978802  
38  },  
39  "timezone": -14400,  
40  "id": 4930956,  
41  "name": "Boston",  
42  "cod": 200  
43 }
```

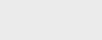
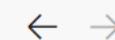


4564-71-192-208-255.ngrok-free



```
1 {  
2   "city": "Boston",  
3   "current_time": "2024-06-09 12:57:06.684202",  
4   "feels_like": 56.62,  
5   "humidity": 90,  
6   "lat": 42.3584,  
7   "lon": -71.0598,  
8   "pressure": 1000,  
9   "temp": 57,  
10  "temp_max": 59.05,  
11  "temp_min": 55.02  
12 }
```





code-interpreter



streamlit\_app.py M

notebook.ipynb M

weather.py U

main.py U X

D ▾ ⌂ ⌂ ⌂ ⌂

gpt &gt; Basic-auth &gt; main.py &gt; ...

```
33     @app.route('/weather', methods=['GET'])
34     def get_weather_data():
35         if not validate_api_key(request):
36             return Response(json.dumps({"error": "Invalid API key"}),
37                             status=401,
38                             mimetype='application/json')
39         city = request.args.get('city', 'Boston,US') # default city is Boston, US
40         if not city:
41             return Response(json.dumps({"error": "City parameter is required"}),
42                             status=400,
43                             mimetype='application/json')
44
45         url = f'http://api.openweathermap.org/data/2.5/weather?q={city}&appid={app_id}&units=imperial'
46         response = requests.get(url)
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

JUPYTER

POSTMAN CONSOLE

AZURE

COMMENTS

python

▼

...

^

X

er instead.

- \* Running on all addresses (0.0.0.0)
- \* Running on http://127.0.0.1:8080
- \* Running on http://172.31.141.7:8080

Press CTRL+C to quit

- \* Restarting with watchdog (windowsapi)
- \* Debugger is active!
- \* Debugger PIN: 975-620-553

127.0.0.1 - - [09/Jun/2024 13:23:33] "GET /weather?city=Boston,US HTTP/1.1" 401 -



main\* ↻ ✘ 0 ⚠ 0 ⌂ 0

Ln 70, Col 9 Spaces: 4 UTF-8 CRLF { Python 3.10.10 ('env': venv)





Search

HTTP City - ngrok - ...

HTTP City - localhost - samples

HTTP City - Render

...

HTTP Weather / City - localhost - samples

Save



View Documentation

No Environment



GET



http://localhost:8080/weather?city=London

Send

Params ● Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Code Cookies

Headers (7 hidden)

	Key	Value
<input checked="" type="checkbox"/>	Authorization	Basic b1b15e88fa797225412429c1c50c122a1
	Key	Value

Body

Cookies

Headers (5)

Test Results

Status: 200 OK Time: 823 ms Size: 370 B



Pretty

Raw

Preview

JSON



```
1  {
2    "city": "London",
3    "current_time": "2024-11-23 13:25:22.284715",
4    "humidity": 81,
5    "lat": 51.51,
6    "lon": -0.13,
7    "pressure": 1012,
```



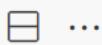
0 0 0





Search

HTTP City - ngrok - ... ● X



HTTP Weather / City - ngrok - samples

Save



View Documentation

No Environment



GET



https://96be-73-68-198-103.ngrok-free.app/weather?city=London,uk

Send

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Settings

Code Cookies

Type

Basic Auth

Username

Username

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#) ↗

Password

b1b15e88fa797225412429c1c50c122a1

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Settings

Code Cookies

Headers

Hide auto-generated headers



username:password  
Is converted to base64

	Key	Value
<input checked="" type="checkbox"/>	Authorization	Basic OmlxYjE1ZTg4ZmE3OTcyMjU0MTI0MjIzMWM1MGmxMjJhMQ==
<input checked="" type="checkbox"/>	Cache-Control	no-cache
<input checked="" type="checkbox"/>	Postman-Token	<calculated when request is sent>
<input checked="" type="checkbox"/>	Host	<calculated when request is sent>
<input checked="" type="checkbox"/>	User-Agent	PostmanRuntime/7.39.1

main.py x openapi.yaml +

main.py

```
1 # https://mer.vin/2023/11/get-stock-price-with-basic-auth/
2 from flask import Flask, request, Response
3 from urllib.request import urlopen
4 import json, datetime, os
5
6 # Api key for openweathermap.org can be provided as an environment
7 # variable
8 app_id = os.getenv('api_key')
9 app = Flask(__name__)
10
11 def validate_api_key(request):
12     auth_header = request.headers.get('Authorization')
13     if not auth_header:
14         return False
15
16     try:
17         auth_type, provided_api_key = auth_header.split(None, 1)
18         if auth_type.lower() != 'basic':
19             return False
20
21         return provided_api_key == app_id
22     except Exception:
23         return False
24
25
26 @app.route('/')
--
```

&gt; Console

Shell

Deployments x +

&lt; ...

0.5 vCPU / 2 GiB RAM

\$12.80 per month (\$0.0166/hour)

Primary domain ⓘ

weatherGPT

.replit.app

✓ Available

Build command ⓘ optional

Run command ⓘ

python3 main.py

Deployment secrets ⓘ

api\_key

.....

+ Add deployment secret

App type ⓘ

Web Server

Background Worker

Port configuration ⓘ

Deploy

```
C:\Users\ydebray>curl -H "Authorization: Basic [REDACTED]" "https://weather-gpt.replit.app/weather?city=Boston, US"
{
  "city": "Boston",
  "current_time": "2024-06-09 20:48:57.647985",
  "feels_like": 66.99,
  "humidity": 71,
  "lat": 42.3584,
  "lon": -71.0598,
  "pressure": 999,
  "temp": 67.23,
  "temp_max": 70.63,
  "temp_min": 62.94
}

C:\Users\ydebray>
```

```
C:\Users\ydebray>curl -H "Authorization: Basic [REDACTED]" "https://weather-gpt.replit.app/weather?city=Boston, US"
{
  "city": "Boston",
  "current_time": "2024-06-09 20:48:57.647985",
  "feels_like": 66.99,
  "humidity": 71,
  "lat": 42.3584,
  "lon": -71.0598,
  "pressure": 999,
  "temp": 67.23,
  "temp_max": 70.63,
  "temp_min": 62.94
}
```

```
C:\Users\ydebray>curl -H "Authorization: Basic My_API_Key" "http://127.0.0.1:8080"
Hello, Weather!
C:\Users\ydebray>curl -H "Authorization: Basic test" "http://127.0.0.1:8080"
Hello, Weather!
C:\Users\ydebray>curl -H "Authorization: Basic test" "http://127.0.0.1:8080/weather"
{"error": "Invalid API key"}
C:\Users\ydebray>curl -H "Authorization: Basic My_API_Key" "http://127.0.0.1:8080/weather"
{"error": "City parameter is required"}
C:\Users\ydebray>curl -H "Authorization: Basic My_API_Key" "http://127.0.0.1:8080/weather?city=London,uk"
{
  "city": "London",
  "current_time": "2024-06-09 18:23:06.389991",
  "humidity": 81,
  "lat": 51.51,
  "lon": -0.13,
  "pressure": 1012,
  "temp": 280.32,
  "temp_max": 281.15,
  "temp_min": 279.15
}
```

```
C:\Users\ydebray\Downloads\gpt-programming-book\chap10\Basic-auth>python main.py
 * Serving Flask app 'main'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8080
 * Running on http://172.31.141.62:8080
Press CTRL+C to quit
 * Restarting with watchdog (windowsapi)
 * Debugger is active!
 * Debugger PIN: 710-487-545
127.0.0.1 - - [09/Jun/2024 18:38:33] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [09/Jun/2024 18:38:58] "GET /weather HTTP/1.1" 400 -
127.0.0.1 - - [09/Jun/2024 18:39:18] "GET /weather HTTP/1.1" 401 -
127.0.0.1 - - [09/Jun/2024 18:39:24] "GET /weather?city=London,uk HTTP/1.1" 200 -
```



## Edit actions



Let your GPT retrieve information or take actions outside of ChatGPT.

[Learn more.](#)

### Authentication

API Key



### Schema

[Import from URL](#)

Examples



```
{  
  "openapi": "3.1.0",  
  "info": {  
    "title": "Get weather data",  
    "description": "Retrieves current weather data for a given city.",  
    "version": "v1.0.0"  
  },  
  "servers": [  
    {  
      "url": "https://weather-gpt.replit.app"  
    }  
  ],  
  "paths": {  
    "/weather": {  
      "get": {  
        "description": "Get temperature for a specific location",  
        "operationId": "GetCurrentWeather",  
        "parameters": [  
          {  
            "name": "city",  
            "in": "query",  
            "description": "The city and state to retrieve the weather for",  
            "required": true,  
            "schema": {}  
          }  
        ]  
      }  
    }  
  }  
}
```

Format

## Preview

weather in Boston



> [debug] Calling HTTP endpoint

> [debug] Response received

Talked to weather-gpt.replit.app

The current weather in Boston is as follows:

- **Temperature:** 66.99°F (feels like 66.72°F)
- **Humidity:** 71%
- **Pressure:** 999 hPa
- **Temperature Range:** 62.56°F to 70.63°F

If you need any more details, just let me know!



Message WeatherGPT

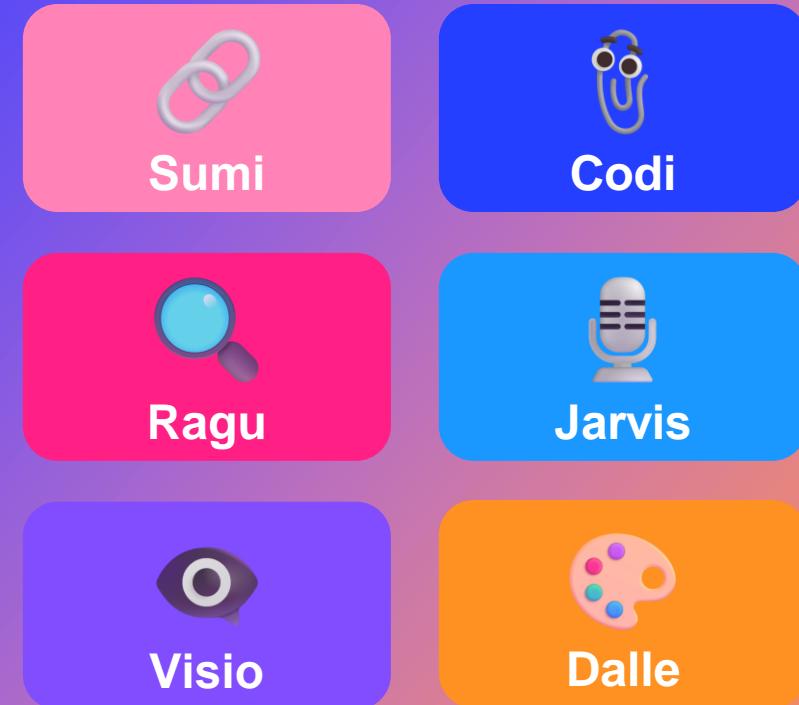


# GPT Apps

From a chatbot to an army of agents



Chatty





config

chains

rag

## ChatGPT-like bot 🤖

Selected model: gpt-3.5-turbo

New Chat

convo0

convo1

convo2

convo3



hello



Hello! How can I assist you today?



My name is bob



Hello Bob! How can I help you today?



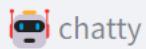
what's my name?



Your name is Bob.

convo1





## Select Model

gpt-3.5-turbo

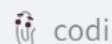
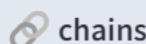


## Select Assistant



## Select User

 Export history



## Chains

Selected model: gpt-3.5-turbo

chunk size

900



Max tokens: 16000

Select chapter to summarize

chap8

pages: 10 tokens: 1967

**refine**

chunks 3

refine

summary 1:

summary 2:

summary 3:

**final summary:**

In this chapter, the author delves into the realm of AI-driven image generation, focusing on OpenAI's DALL-E model. The discussion covers various Generative AI models, emphasizing DALL-E's capacity to translate text prompts into images. The author shares their experimentation with creating viral videos through AI and producing brief videos on platforms like TikTok. Detailed techniques for image generation and manipulation with DALL-E, such as in- and outpainting methods and crafting graphical novels, are



Selected model: gpt-3.5-turbo



Ask a question

what is vector search

Vector search is a technique for information retrieval that uses a numeric representation of text (vectors) to find documents or passages that are semantically similar. This method involves converting text into vector embeddings, which capture the semantic meaning of the text by mapping words, sentences, or documents into high-dimensional vector spaces. Similar items in this vector space are close to each other, while dissimilar items are far apart. Vector search is commonly used in search engines, recommendation systems, and data retrieval systems to provide more nuanced and context-aware search results.

Source documents



Page: [29](#)

Source: big.pdf

#### 6 4.2.2. Vector search

Now, let's look into the concept of vector search, a technique for information retrieval that leverages a numeric representation of text (as vectors) to find semantically similar documents or passages.

- Embeddings: Vector representation of text

Vector embeddings capture the semantic meaning of text by mapping words, sentences, or documents into high-dimensional vector spaces. Similar items in this space are close to each



chatty

config

chains

rag

Codi



(asst\_5zjj3Cp5W2DOT6sRLeT6Cf23)

Model: gpt-3.5-turbo

Prompt examples

1+1

How to solve the equation  $3x + 11 = 14$  ?

What is the 42nd element of Fibonacci?

What is the 10th element?

plot function  $1/\sin(x)$

zoom in to range of x values between 0

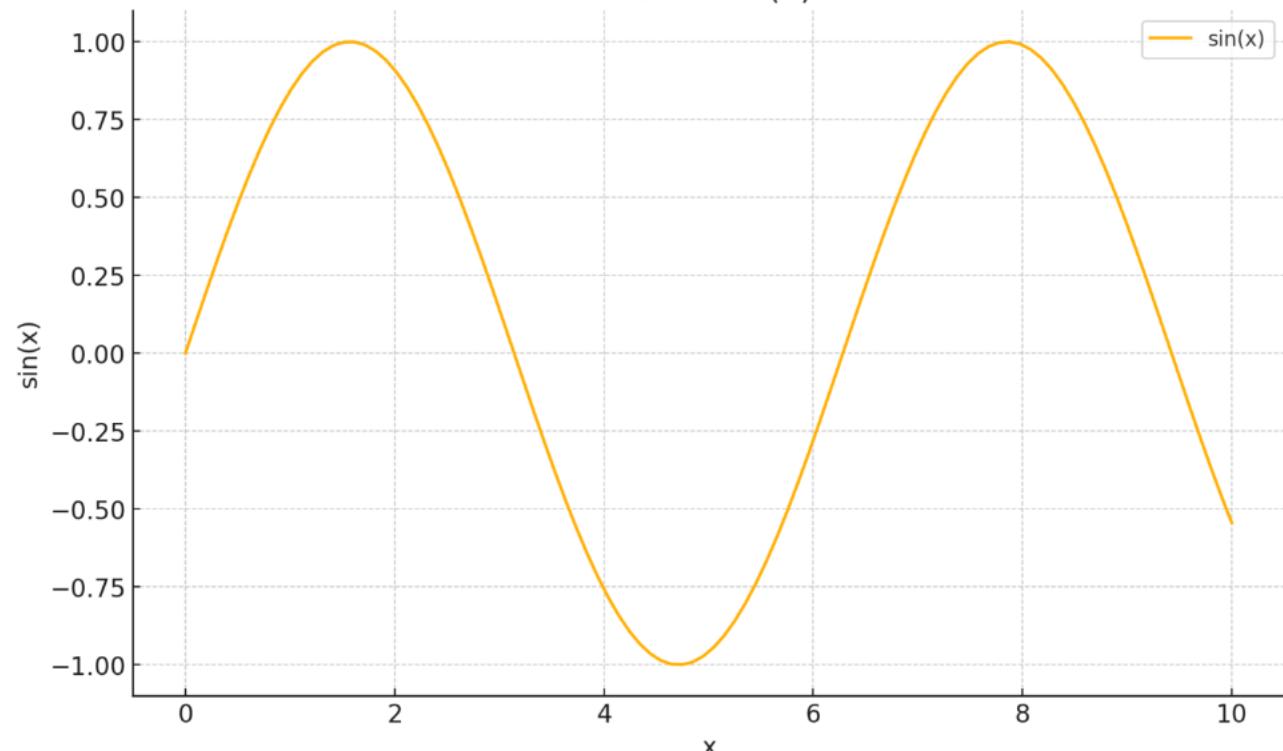
1



plot function  $\sin(x)$



Plot of  $\sin(x)$



Here is the plot of the function ( $y = \sin(x)$ ) for ( $x$ ) values between 0 and 10. The sine function is a periodic function that oscillates between -1 and 1.

Your message





X

## Voice

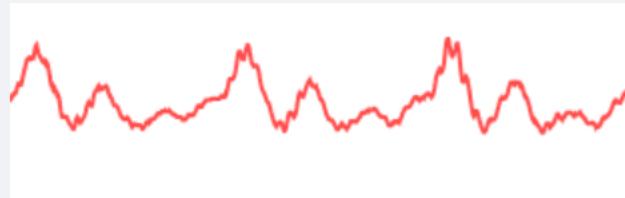
Selected model: gpt-3.5-turbo

**Start Recording**

Stop

Reset

Download



▶ 0:00 / 0:00



Hello Jarvis, I need your help in designing a new suit.



Of course! I'd be happy to help you with that. What kind of suit are you looking to design?



I want one that I can take to battle.



Sure thing! Let's start by determining the purpose of the suit. Are you looking for something with heavy armor for maximum protection, or do you want something more flexible for agility and mobility in battle?



I want one that has style and chrome.



Great choice! A stylish suit with chrome accents can definitely make a statement on the battlefield. Let's start by sketching out some design ideas. Do you have any specific colors or themes in mind for the suit?



chatty

config

chains

rag

visio



What is in the image?



Funny corgi in a cartoon style



7e3ccf47366442cb33744132859577a7212bf5cb5f1e30014c3b7132.jpg



What is in the image?



The image features a colorful illustration of a cheerful cartoon dog, possibly a Corgi, with a big smile and its tongue playfully sticking out. The dog is wearing a blue plaid shirt. In the background, there is a well-kept yard with a white picket fence, lush green bushes, a rocking chair, and some flowers, indicating a cozy outdoor domestic setting. A smaller image of a similar dog is partially visible in the bottom left corner, looking at a picture frame. The general atmosphere is bright and joyful, suggesting a playful and family-friendly theme.

Ask question about images



X



## dalle



Select Model

 dall-e-3 dall-e-2

Select Size

 1024x1024 1792x1024 1024x1792

Prompt dall-e-3:

a funny corgi

Generate Image

