

## 9. Conclusion

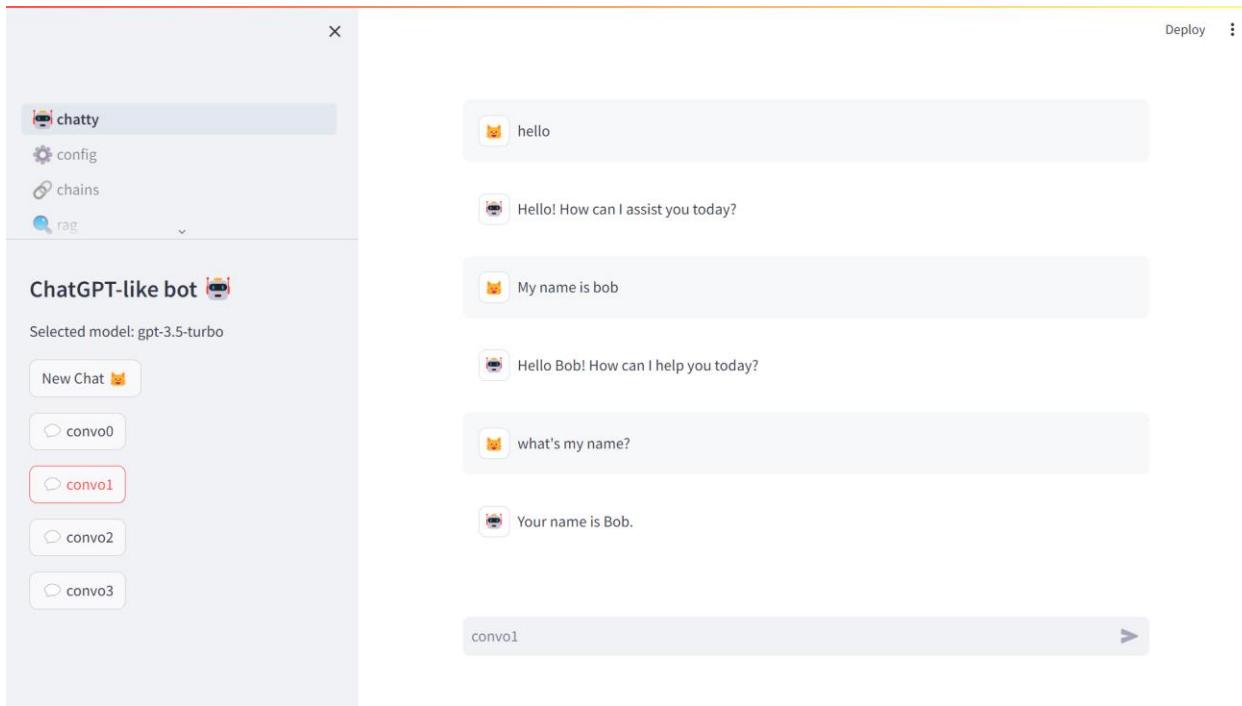
In this book we prioritized a deeper dive into how you can build your own AI-powered products via the ChatGPT API as it gives you more options than the GPT builder in the ChatGPT App (with Plus subscription).

In each chapter, we have seen one of the extended capabilities of the ChatGPT API. Let's summarize all of those into one multipage app. This should act as a companion to this book, and a template for you to start your own AI-powered product (you just have to bring your own OpenAI API key).

I'll personify each of the GPTs that we will go through in this app and give each of them a funny name.

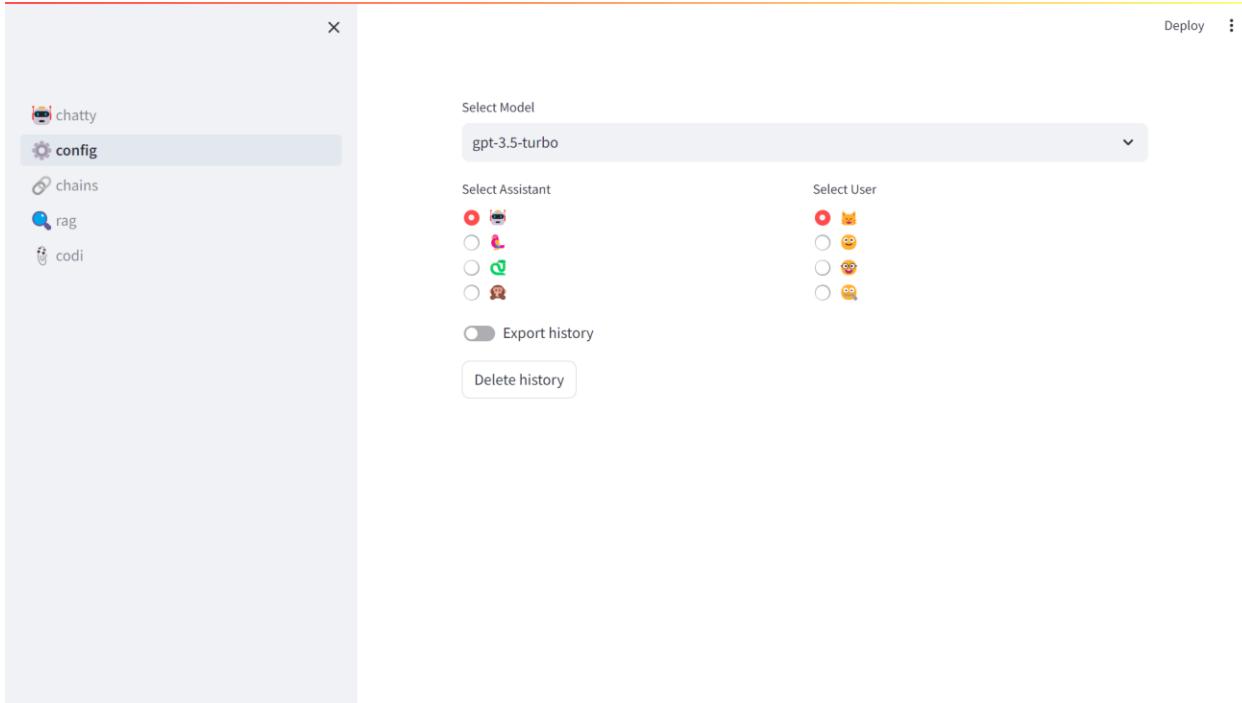
- 🤖 chatty:

Our good old chat, with the memory of the conversation saved as session state. This one is important to get the basics of how to call an LLM. In the appendix, we will see a version of chatty calling Mistral AI.



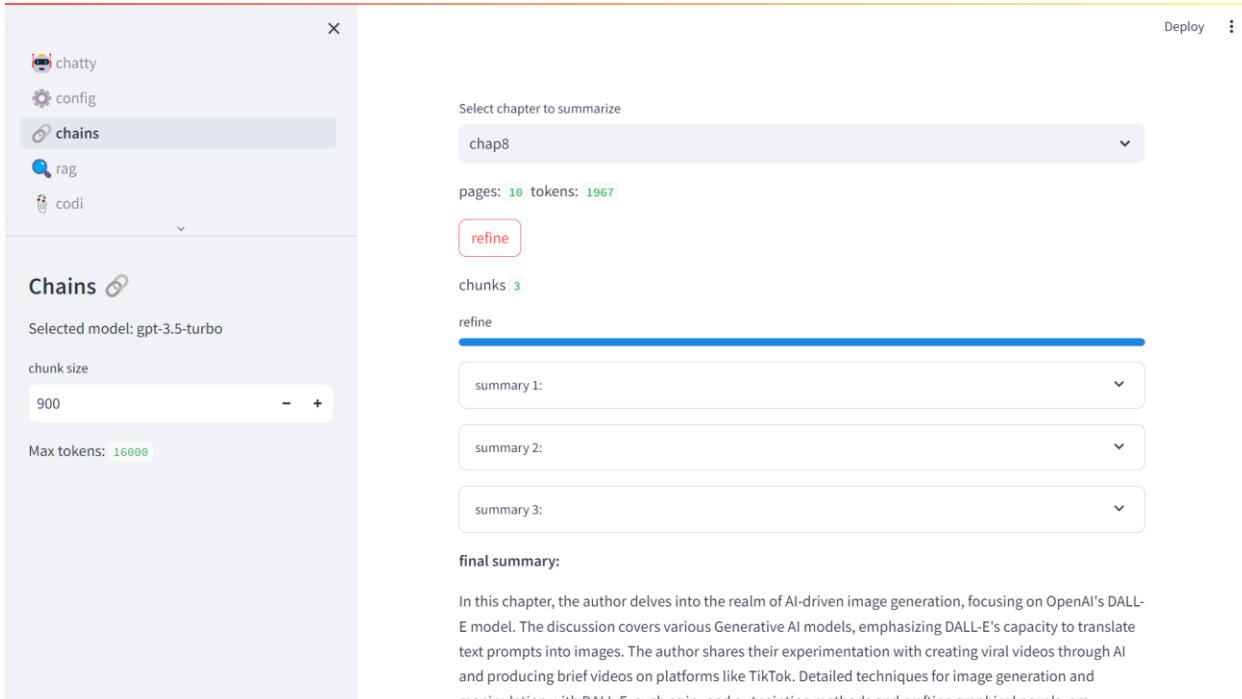
- 🛡 config:

This isn't really a GPT, but more an extension of chatty. It enables you to configure your bot, export and delete the history of messages. This way if you want to host this app online, you can give basic privacy settings to your users.



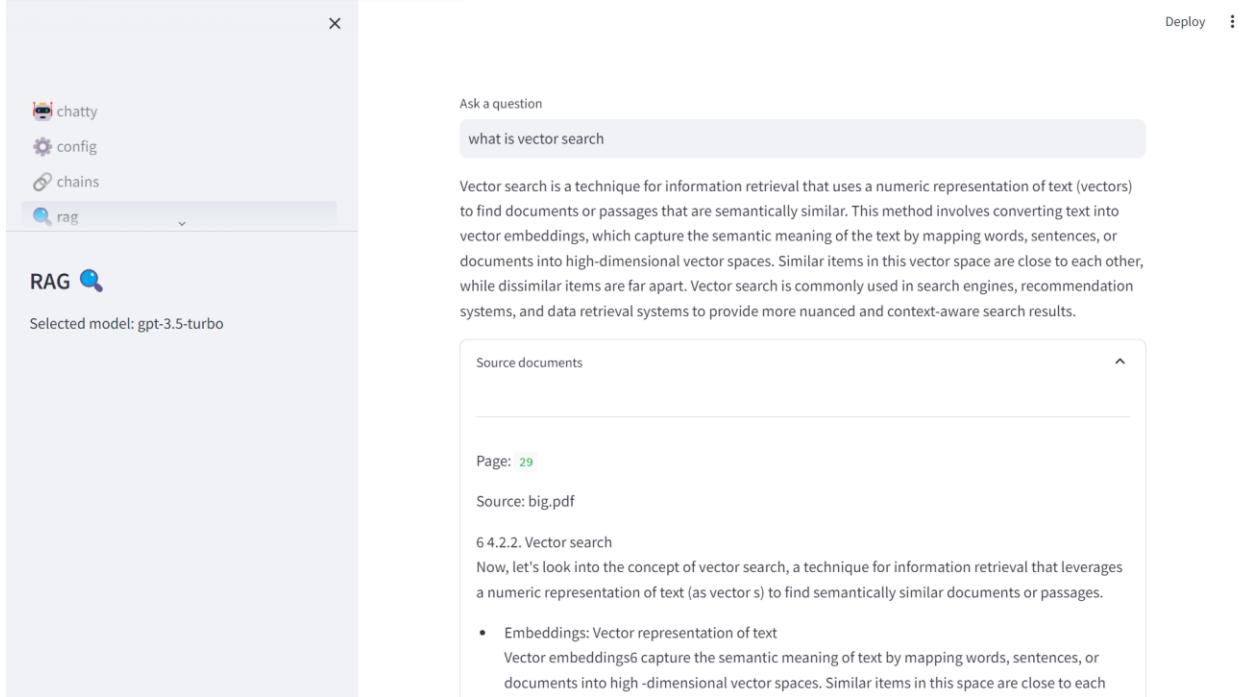
- sumi:

Summarizing is a useful use case of LLMs, and it is one of the first GPTs I created when the GPT-3.5 API came out. This version is more meant for pedagogical objective to explain how the text is broken down into chunks, and to see the impact of changing the chunk size, for instance from 1000 to 900 on chapter 8 that represents 1967 tokens in total. You can also try it out on longer chapters, to observe how the summary gets refined over the iterations.



-  **ragu:**

RAG is an important part of the excitement around the chaining of calls to LLMs. This GPT uses a vector database that contains the chapters of this book to answer questions. You can derive your own version with the document database of your choice. An alternative for this is to use OpenAI's file search service.



The screenshot shows a user interface for a RAG system. On the left, there is a sidebar with icons for 'chatty', 'config', 'chains', and 'rag'. Below this, the title 'RAG' is displayed with a magnifying glass icon. A message 'Selected model: gpt-3.5-turbo' is shown. The main area has a header 'Ask a question' with the input 'what is vector search'. Below this is a detailed explanation of vector search, mentioning its use in information retrieval, search engines, recommendation systems, and data retrieval systems. A section titled 'Source documents' lists a single source: 'big.pdf' on page 29. The text from the document is quoted, starting with '6.4.2.2. Vector search' and explaining how it uses vector representations of text to find similar documents.

Deploy :

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 vector s) 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

-  **codi:**

This GPT is one of my favorites. It is less capable than the version described in chapter 5 that enables file inputs for advanced data analysis.

**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

Plot of  $\sin(x)$

sin(x)

x

Plot of  $\sin(x)$

sin(x)

0 2 4 6 8 10

1.00  
0.75  
0.50  
0.25  
0.00  
-0.25  
-0.50  
-0.75  
-1.00

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 ➤

- 🎙 jarvis:

This is probably the prototypical chatbot that most of us think about when building our own GPT. It's a variation of chatty with added bells and whistles to take your voice as input. I didn't go the extra mile of adding a voice synthesis of the bot response, as I was worried it would bring me to the "uncanny valley". This expression represents the strange feeling that humans experience when an AI tries to mimic human functions but does not quite get it right (like when you have a weird looking human image generated).

The interface shows a sidebar with icons for chatty, config, rag, and chains. A main area titled "Voice" displays a recording interface with "Start Recording", "Stop", and "Reset" buttons, and a "Download" button below a waveform visualization. The waveform shows a red line fluctuating over time, with a progress bar at the bottom indicating 0:00 / 0:00. To the right, a conversation log is shown:

- User: Hello Jarvis, I need your help in designing a new suit.
- Jarvis: Of course! I'd be happy to help you with that. What kind of suit are you looking to design?
- User: I want one that I can take to battle.
- Jarvis: 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?
- User: I want one that has style and chrome.
- Jarvis: 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?

- **visio:**

This one sounds like a superhero name. And it does translate some of the feeling of being able to extract information from images. When this capability will scale to take video as input, it will open up a whole new dimension of copilot use cases.

The interface shows a sidebar with icons for chatty, config, chains, and rag. A main area titled "visio" displays a question "What is in the image?" above a cartoon illustration of a Corgi dog wearing a blue plaid shirt, standing in a garden with a white picket fence and flowers. Below the image, the caption "Funny corgi in a cartoon style" is shown. To the right, a larger version of the same cartoon dog image is displayed with the file name "7e3ccf47366442cb33744132859577a7212bf5cb5f1e30014c3b7132.jpg". A conversation log is shown:

- User: What is in the image?
- Jarvis: 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.

At the bottom, there is a text input field with the placeholder "Ask question about images" and a "▶" button.

- **dalle:**

This one is more artistic but can be useful for you to generate logos and images like the one on the cover of this book. If your application of AI is more for artwork or entertainment, this GPT is a good start.

