



AWS ML Heroes in 15:

Unlock the Power of Generative AI

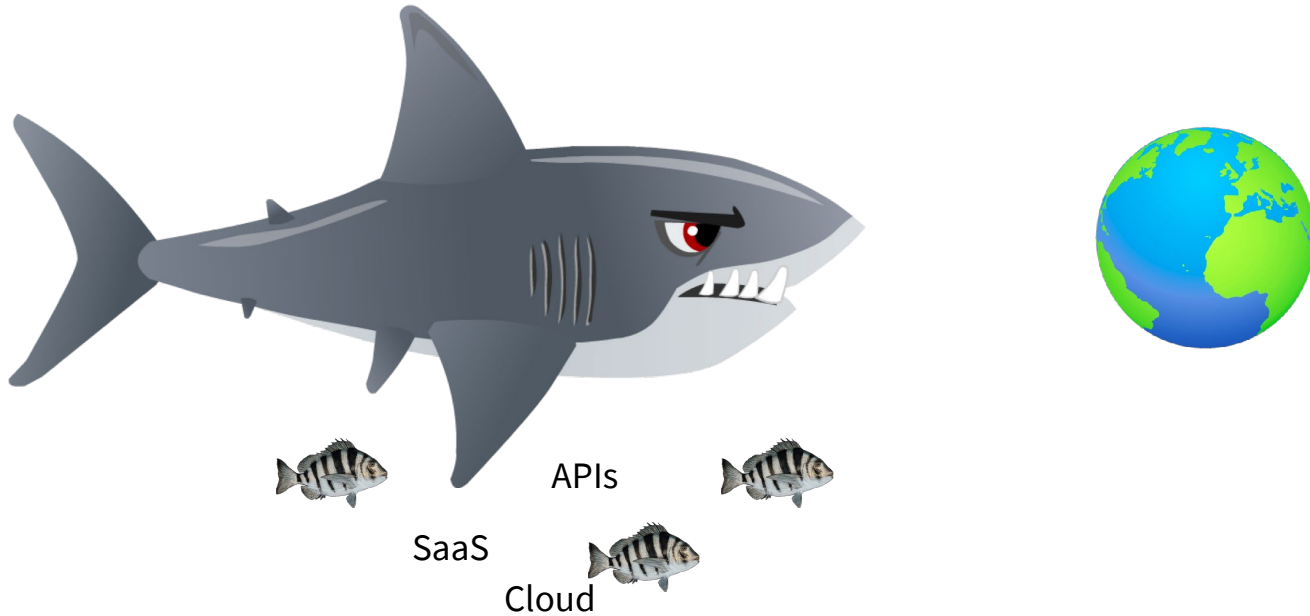
Philipp Schmid

Technical Lead
Hugging Face

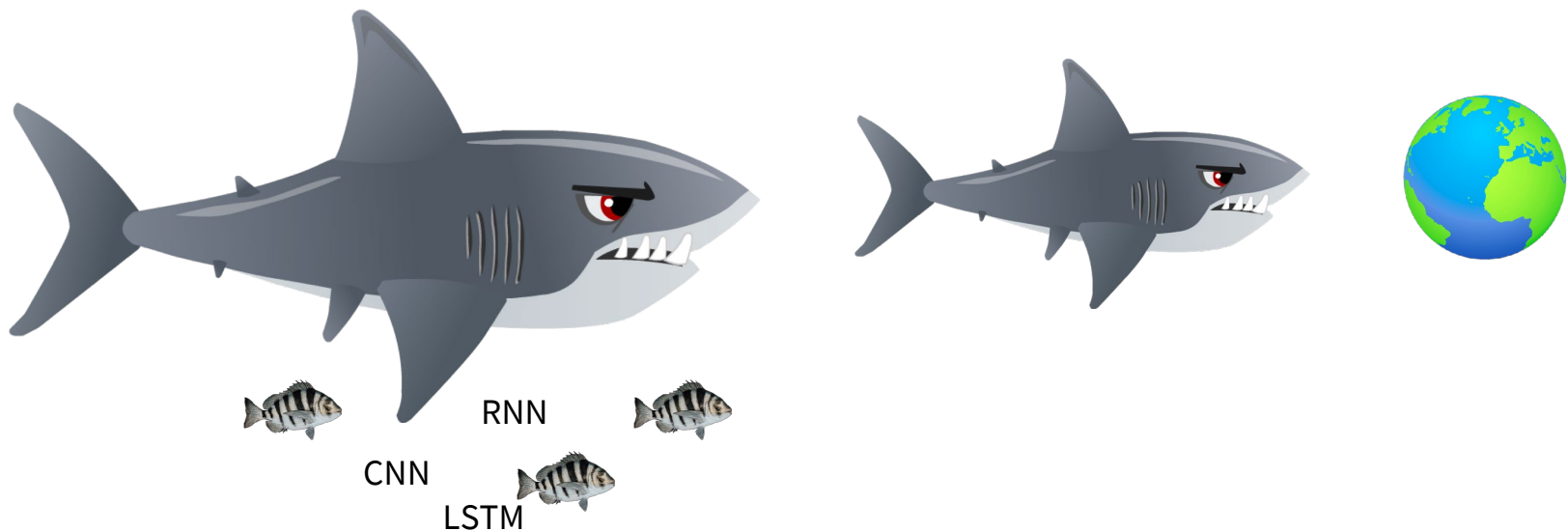
Outline

- A brief history
- What is Generative AI?
- Hugging Face on AWS
- Applied Generative AI Example

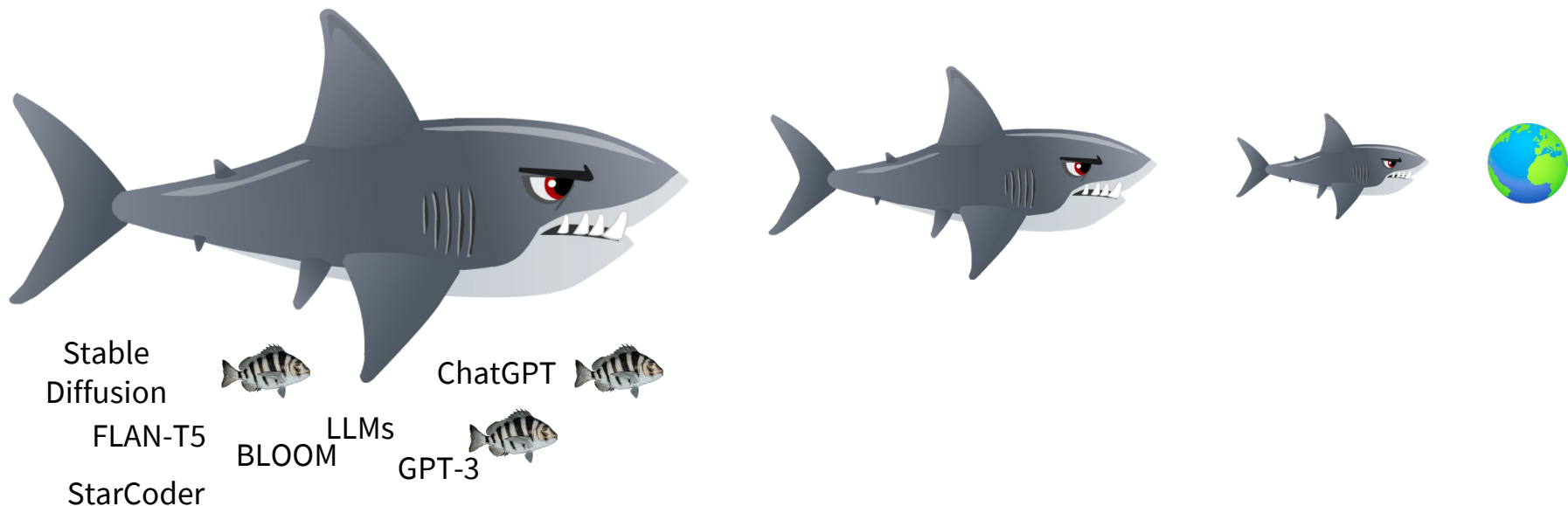
2011: Software is eating the world



2017: Deep Learning is eating Software



2023: Generative AI is eating Deep Learning



But what is Generative AI?

“Generative models are a key enabler of machine creativity, allowing machines to go beyond what they’ve seen before and create something new.”

~Ian Goodfellow

“Generative models represent the next phase of artificial intelligence, where machines move beyond simple pattern recognition to create new and unique content.”

~Alex Krizhevsky

“Generative models allow machines to learn from data and then create new, original content, and they have the potential to revolutionize industries from music to fashion to gaming.”

~Yann LeCun

But what is Generative AI?

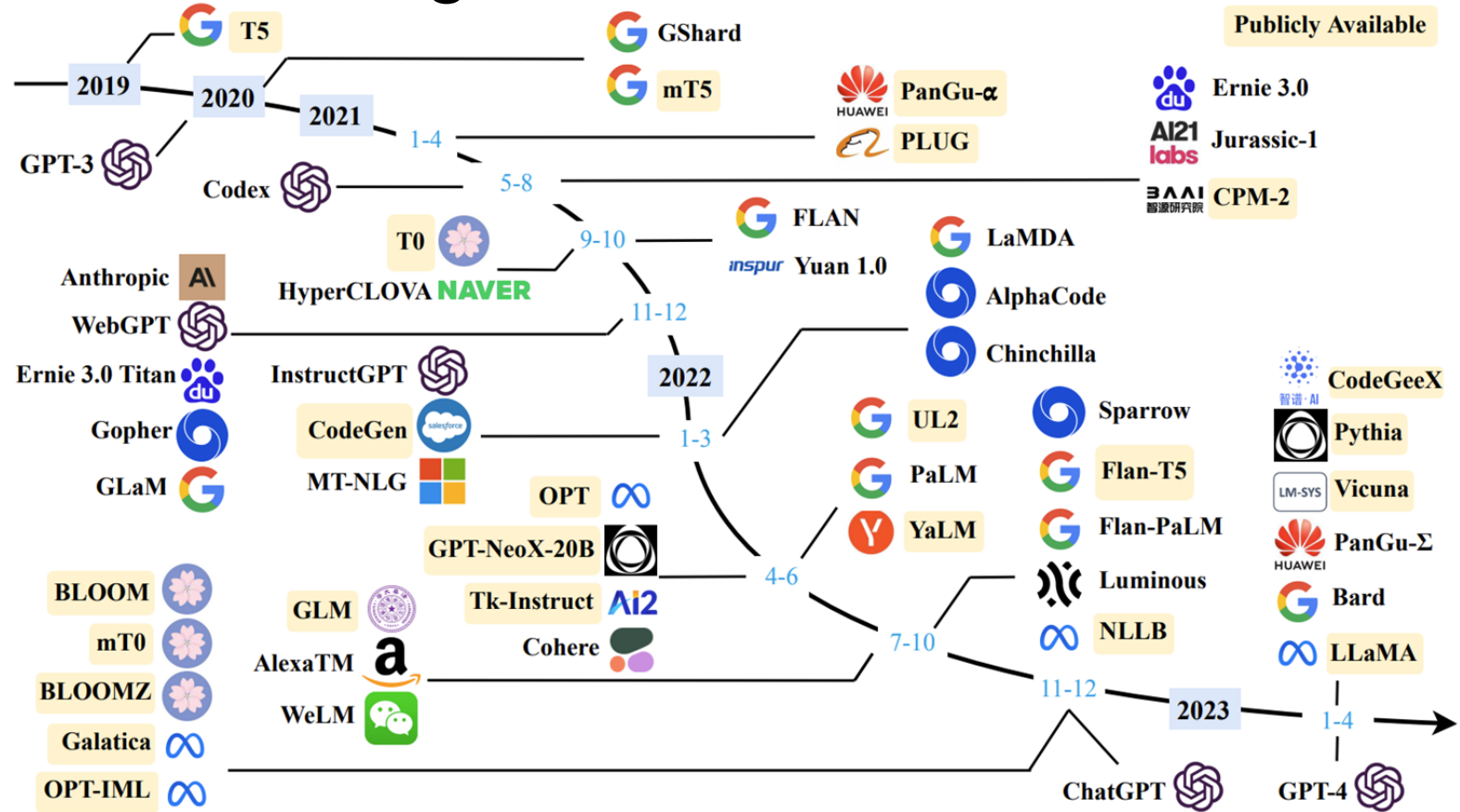
“Generative AI is an artificial intelligence that uses machine learning to generate new content such as text, images, video, and audio.” ~Claude

“Generative AI is a type of machine learning that uses large language models to create entirely new and original outputs based on the information it has learned from the examples it has been shown.” ~ChatGPT

Basic Principles of Generative AI

- Neural networks trained to generate new content like text, images, music, etc.
- These networks are trained on a dataset and learn to create new data by generating similar patterns and structures.
- The output generated by the model can be used for various applications, such as generating images, text, or even music.
- Large language models (LLMs) are subset of Generative AI for Text
- Responsibly is crucial we need to consider bias, toxicity, privacy, job disruption, and control.

LLMs since GPT-3



Risk of closed-source LLMs

- Lack of transparency (Data, Model, Security)
- Model Versioning/ Backwards Compatibility
- Risk of “Improvements”
- Reduced Innovation
- Lock-in effects



Generative AI in Action

<https://huggingface.co/spaces/runwayml/stable-diffusion-v1-5>



```
from diffusers import StableDiffusionPipeline

pipe = StableDiffusionPipeline.from_pretrained(
    "CompVis/stable-diffusion-v1-4")

prompt = "in the desert, a corvette parked \
in front of an old-school diner at sundown"

image = pipe(prompt).images[0]
image.save("picture.png")
```



Generative AI in Action

<https://huggingface.co/spaces/DeepFloyd/IF>

DeepFloyd IF

The DeepFloyd IF model has been initially released as a non-commercial research-only model. Please make sure you read and abide to the [LICENSE](#) before using it.

[Duplicate Space to skip the queue](#)

a llama standing in front on the CN tower in Toronto holding a shield with "TMLS"

Generate

Enter a negative prompt



Back to selection

Generative AI in Action

<https://huggingface.co/spaces/sanchit-gandhi/whisper-jax>

Microphone Audio File YouTube

Whisper JAX: The Fastest Whisper API ⚡

Whisper JAX is an optimised implementation of the [Whisper model](#) by OpenAI. It runs on JAX with a TPU v4-8 in the backend. Compared to PyTorch on an A100 GPU, it is over **70x faster**, making it the fastest Whisper API available.

Note that at peak times, you may find yourself in the queue for this demo. When you submit a request, your queue position will be shown in the top right-hand side of the demo pane. Once you reach the front of the queue, your audio file will be transcribed, with the progress displayed through a progress bar.

To skip the queue, you may wish to create your own inference endpoint, details for which can be found in the [Whisper JAX repository](#).

inputs

0:05 / 0:05

Task

transcribe translate

Return timestamps

Löschen Absenden

Transcription

Hello, my name is Philip. I'm super excited to be here in Toronto today.

Transcription Time (s)

1.5278284549713135

Whisper large-v2 model by OpenAI. Backend running JAX on a TPU v4-8 through the generous support of the [TRC](#) programme. Whisper JAX [code](#) and Gradio demo by 🤗 Hugging Face.

Generative AI in Action

<https://huggingface.co/spaces/bigcode/bigcode-playground>



StarCoder

Can you help me filter a list of dicts in python based on a key

- You can use the function `filter()` to filter a list of dictionaries based on a specific key. Here's an example:

```
my_list = [{'name': 'Alice', 'age': 28 },{'name': 'Bob', 'age': 32 }]  
filtered_list = filter(lambda x: x['age'] == 28, my_list)  
  
print(list(filtered_list))
```

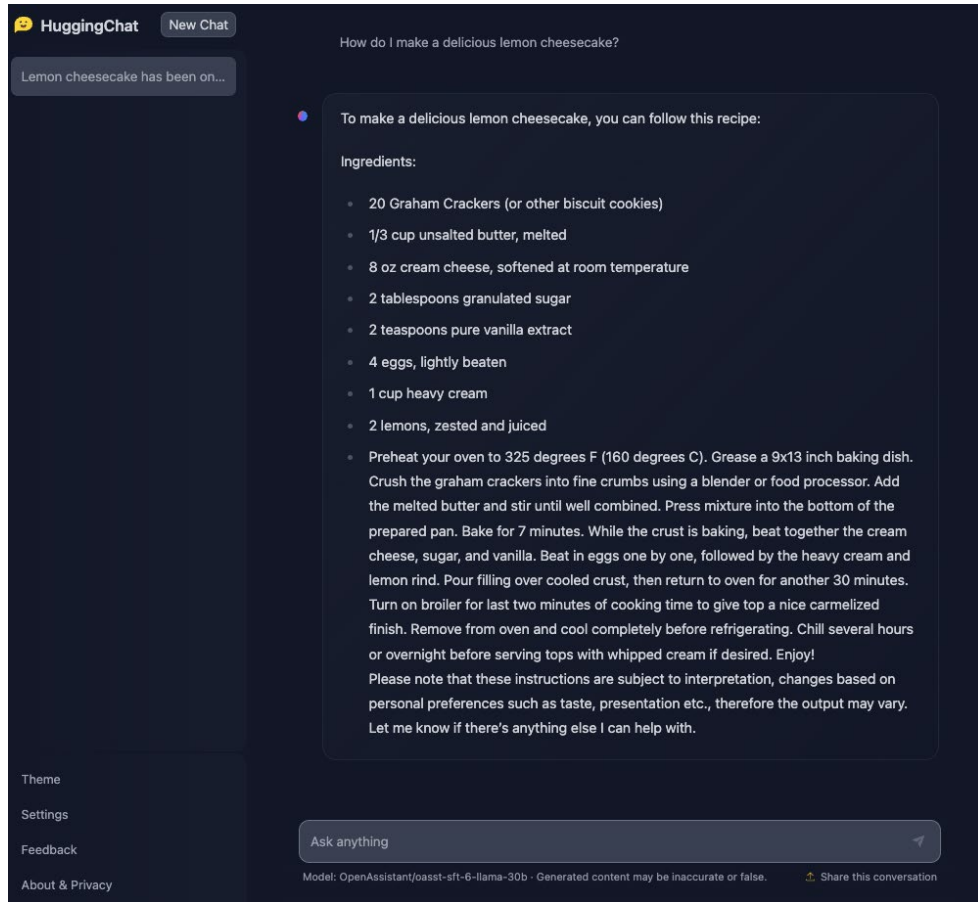
This code creates a list of dictionaries called `my_list`, and then uses the `filter()` function to filter the list on the `'age'` key. The lambda function in the `filter()` function filters the list to only include dictionaries where the `'age'` key has a value of 28.

parse_expenses.py write_sql.go sentiment.ts addresses.rb

```
1 cll  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17
```

Generative AI in Action

[HuggingChat](#)



The screenshot shows the HuggingChat web interface. At the top, there's a header with the HuggingChat logo and a 'New Chat' button. Below the header, a chat history entry shows a user asking 'How do I make a delicious lemon cheesecake?'. The chat area displays a response from the AI assistant, which provides a detailed recipe for lemon cheesecake. The response includes a list of ingredients and step-by-step instructions. At the bottom of the chat area, there's a text input field with the placeholder 'Ask anything' and a send button. The footer of the interface contains links for 'Theme', 'Settings', 'Feedback', and 'About & Privacy', along with a model identifier 'Model: OpenAssistant/basst-sft-6-llama-30b' and a disclaimer 'Generated content may be inaccurate or false.' and a 'Share this conversation' link.

HuggingChat New Chat

How do I make a delicious lemon cheesecake?

Lemon cheesecake has been on...

To make a delicious lemon cheesecake, you can follow this recipe:

Ingredients:

- 20 Graham Crackers (or other biscuit cookies)
- 1/3 cup unsalted butter, melted
- 8 oz cream cheese, softened at room temperature
- 2 tablespoons granulated sugar
- 2 teaspoons pure vanilla extract
- 4 eggs, lightly beaten
- 1 cup heavy cream
- 2 lemons, zested and juiced

Preheat your oven to 325 degrees F (160 degrees C). Grease a 9x13 inch baking dish. Crush the graham crackers into fine crumbs using a blender or food processor. Add the melted butter and stir until well combined. Press mixture into the bottom of the prepared pan. Bake for 7 minutes. While the crust is baking, beat together the cream cheese, sugar, and vanilla. Beat in eggs one by one, followed by the heavy cream and lemon rind. Pour filling over cooled crust, then return to oven for another 30 minutes. Turn on broiler for last two minutes of cooking time to give top a nice caramelized finish. Remove from oven and cool completely before refrigerating. Chill several hours or overnight before serving tops with whipped cream if desired. Enjoy!

Please note that these instructions are subject to interpretation, changes based on personal preferences such as taste, presentation etc., therefore the output may vary. Let me know if there's anything else I can help with.

Theme

Settings

Feedback

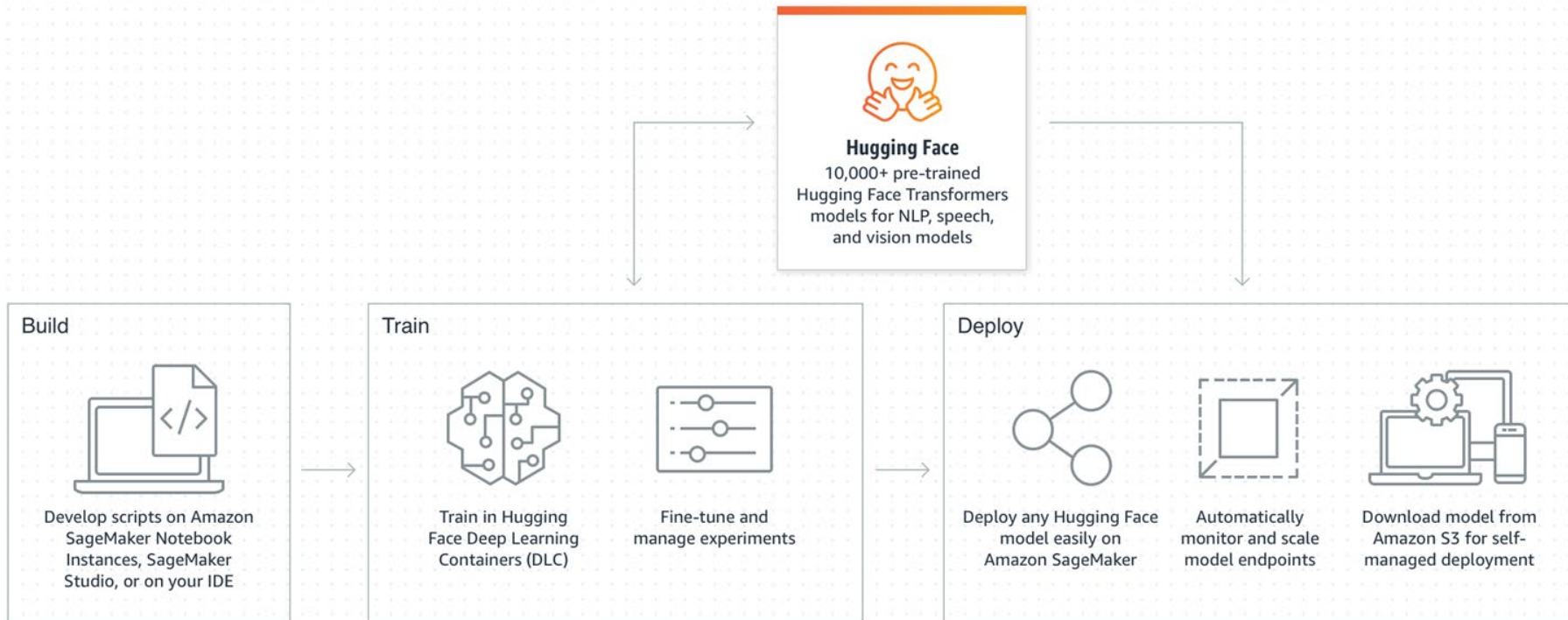
About & Privacy

Ask anything

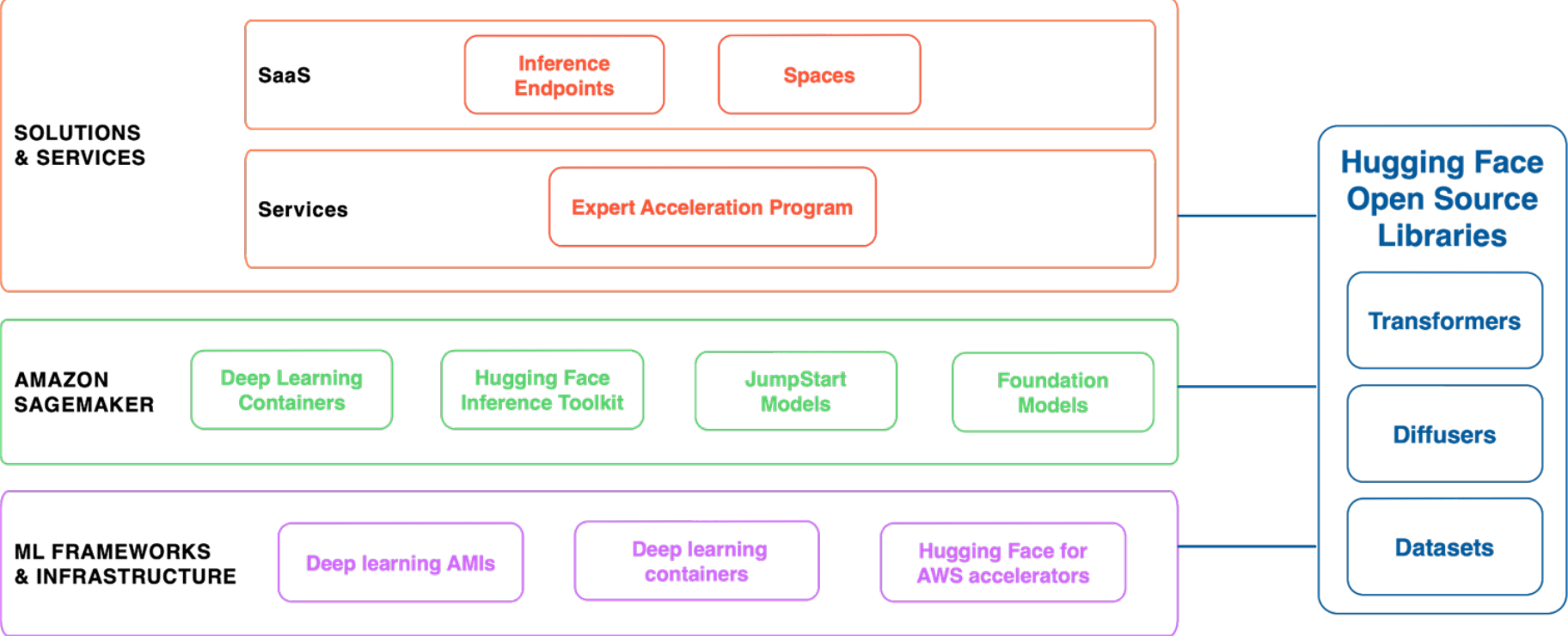
Model: OpenAssistant/basst-sft-6-llama-30b · Generated content may be inaccurate or false. [Share this conversation](#)

Hugging Face on AWS

Hugging Face on AWS

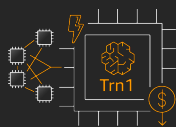


Hugging Face on AWS

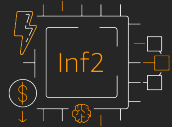


Hugging Face on AWS

AWS ML Infrastructure



AWS
Trainium



AWS
Inferentia2



Neuron
SDK



Neuron
AMI



Hugging Face

Model Hub



Models



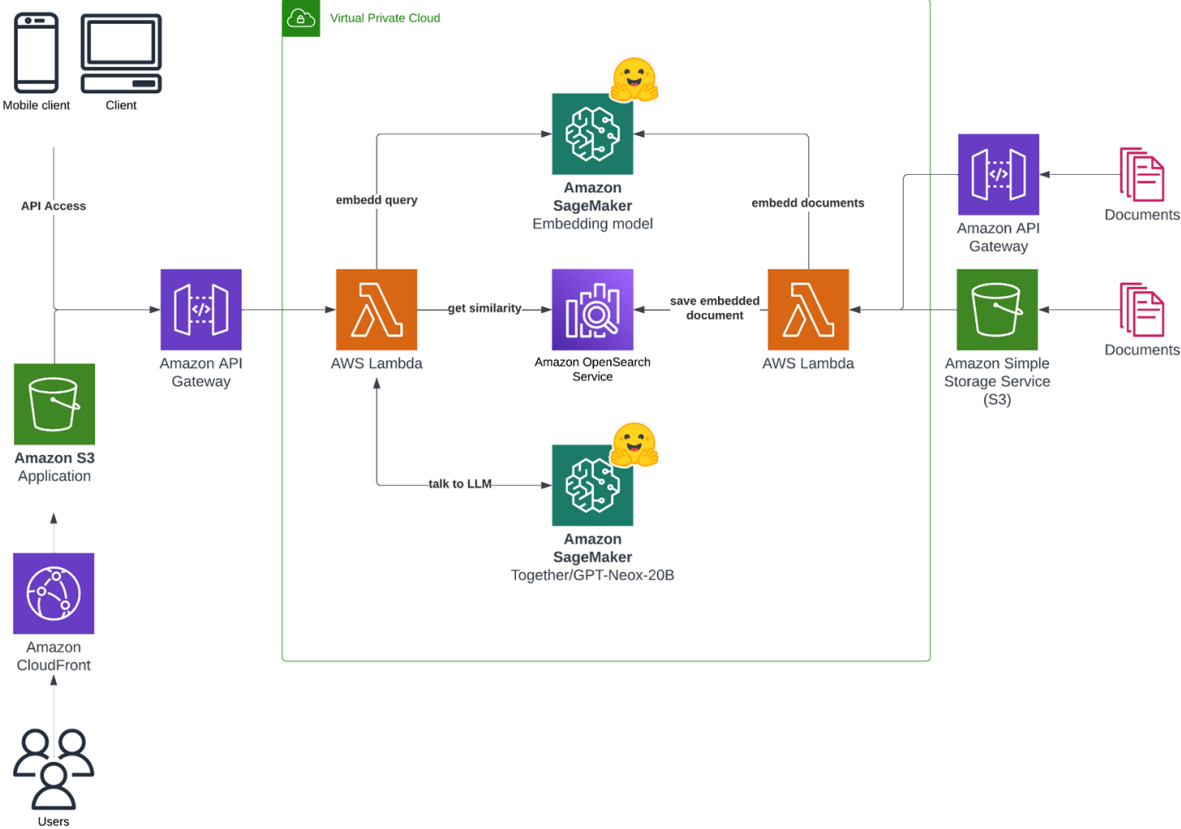
Datasets



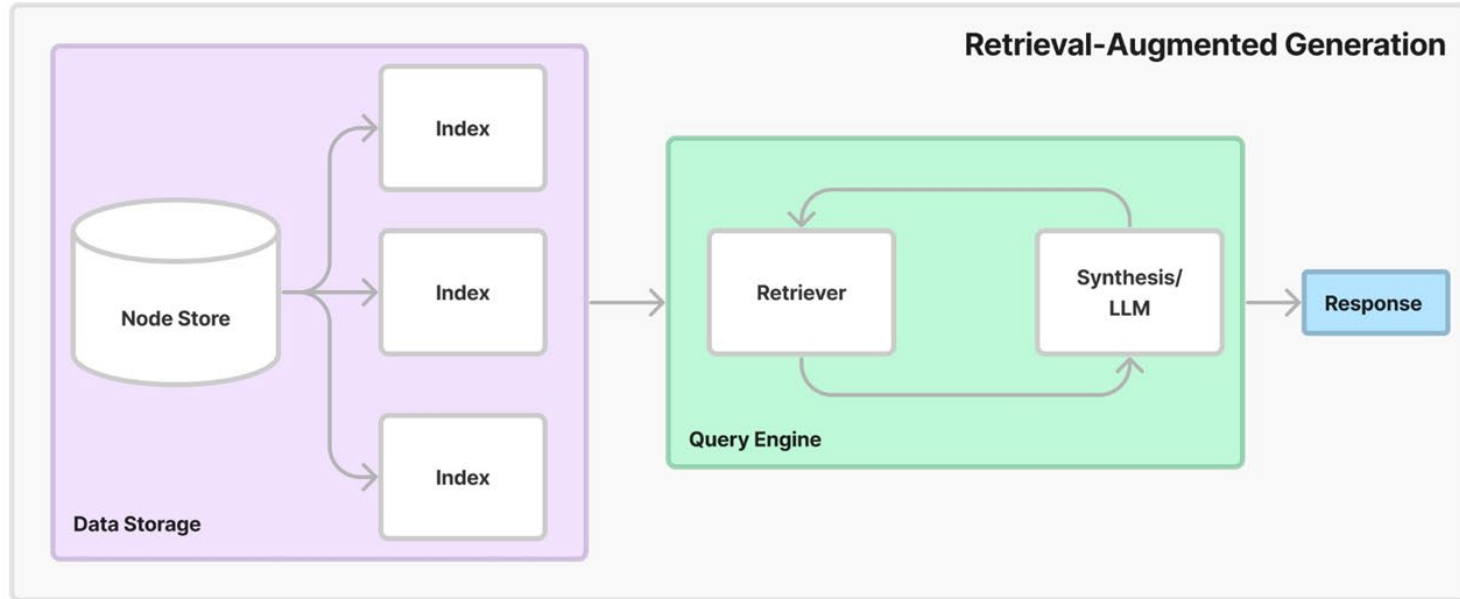
Libraries

Applied Generative AI Example

Enterprise search powered by Generative AI



What is Retrieval-Augmented Generation?



What are the benefits?

Improved search accuracy:

The combination of Generative AI and knowledge retrieval to provide more accurate and relevant search results. By understanding the context of the search query, RAG can generate more targeted responses that are tailored to the user's search query.

Enhanced productivity and decision-making:

Enterprises save time and improve productivity by providing faster and more accurate search results. This can help employees find the information they need more quickly and efficiently, allowing them to focus on other critical tasks.

Security and compliance:

Deployed on AWS with strong data encryption, access controls, and monitoring. This ensures that sensitive data is protected and handled securely at all times and being compliant with industry standards such as GDPR, HIPAA, and PCI DSS.

Technical details



Amazon Cloudfront



Amazon OpenSearch



Amazon SageMaker



AWS CDK



Amazon S3



AWS Lambda

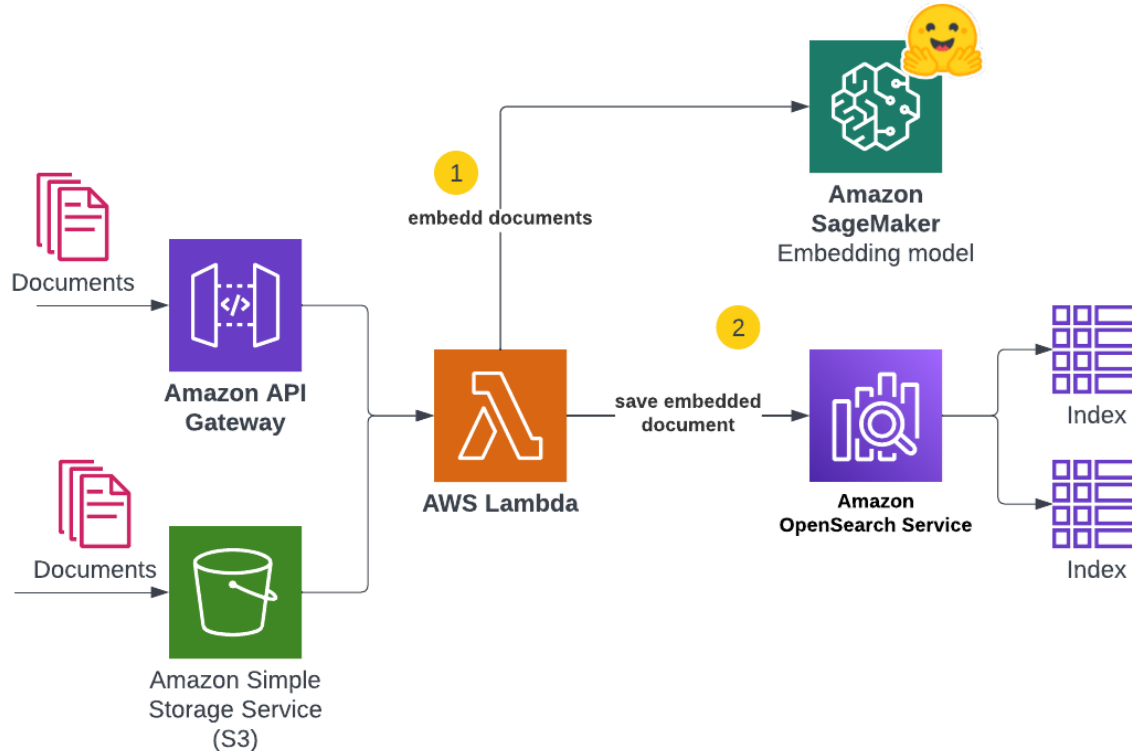


Amazon API Gateway

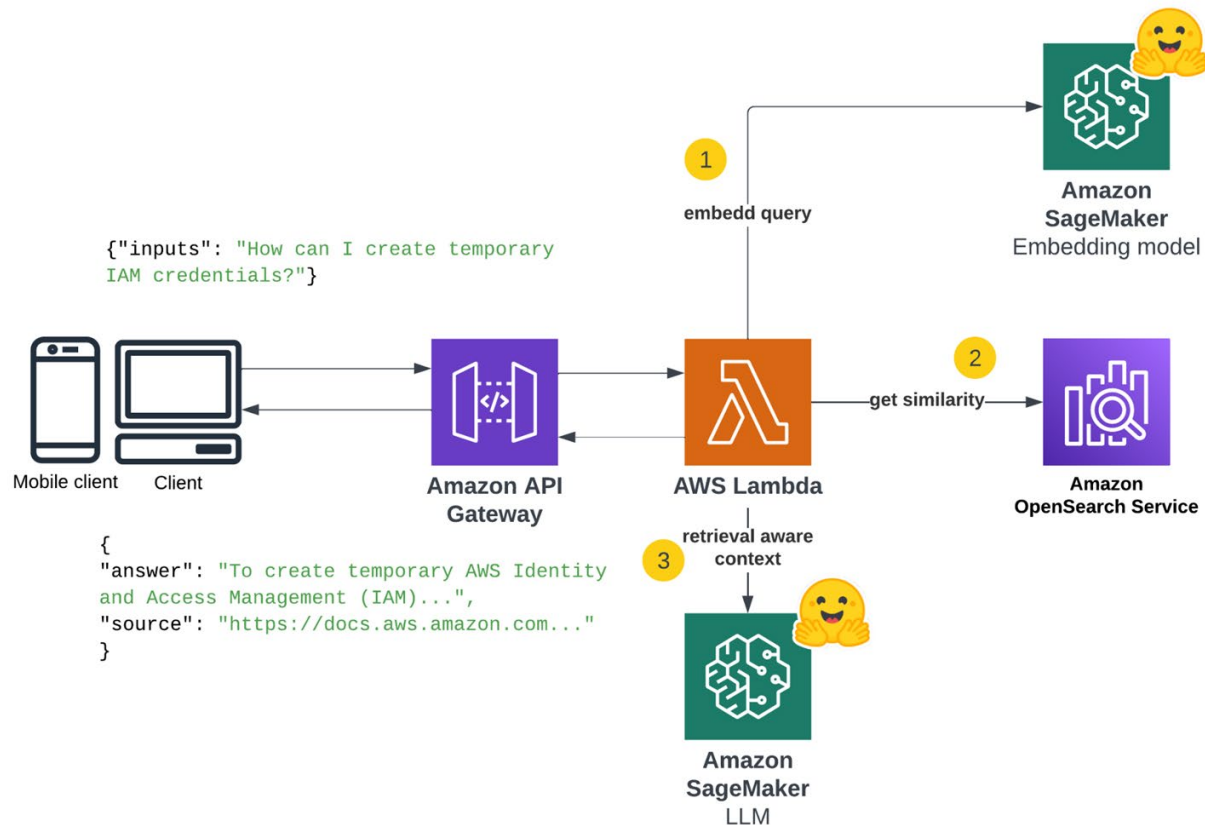


Amazon VPC

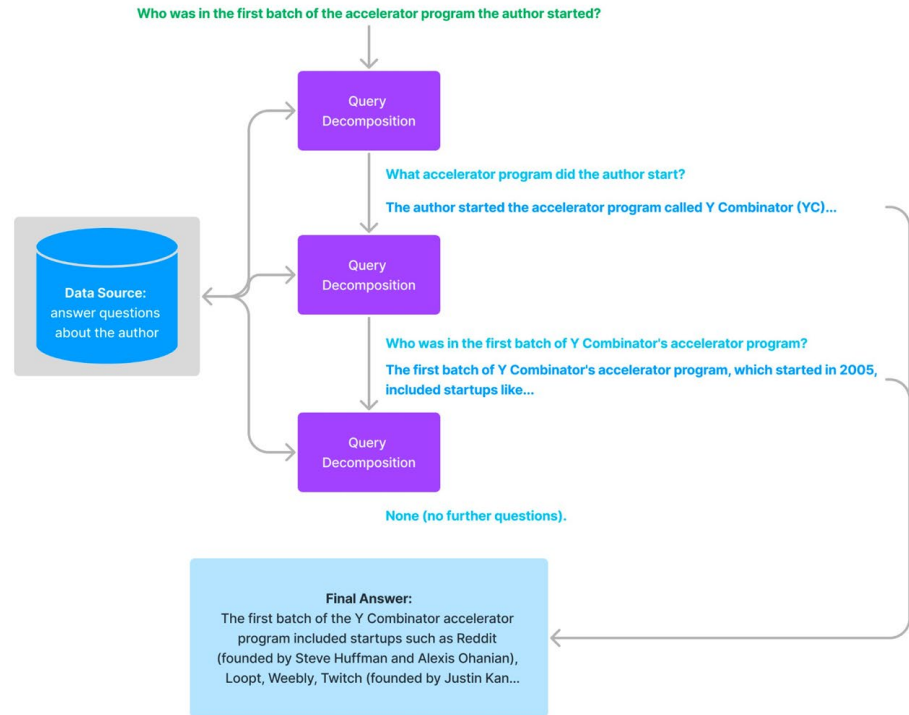
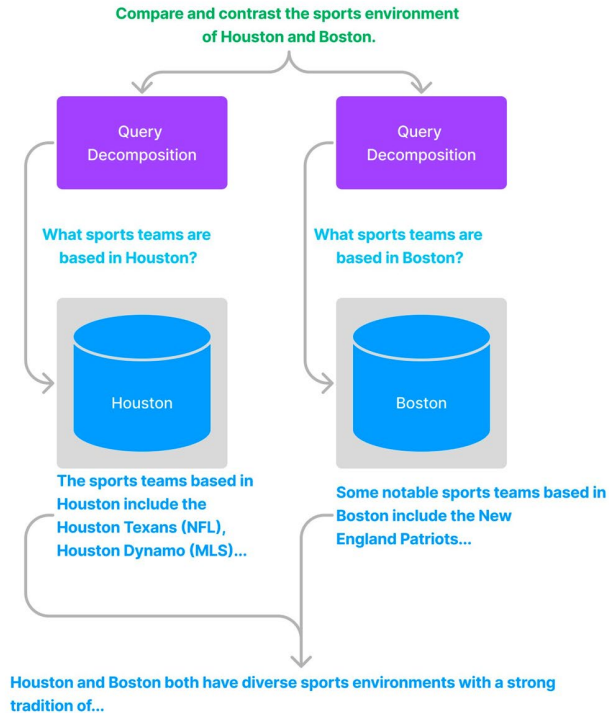
Technical details: Knowledge Ingestion



Technical details: Retrieval-Augmented Generation



Further improvements





Thank you!