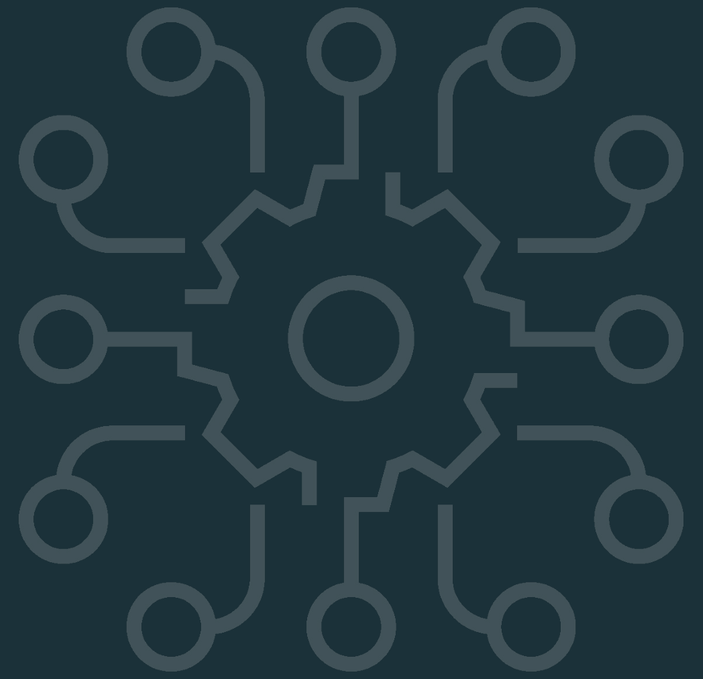
databricks

Accelerating LLM Apps to Production



Brian Law - Sr Specialist Solutions Architect

Quick Recap

Recap

From words to math

English Language	Tokens	Vectors
A language structured with: - an alphabet - Words - sentences - paragraphs	Mathematical encoding of parts of words:	Mathematical encoding of entire sentences and paragraphs:
Ex - Cat - Running	Ex - [40] - [10 12]	Ex: - [0 12 32 127]

LLM Flavors

Thinking of building your own modern LLM application?

Open-Source Models

- Use as **off-the-shelf** or **fine-tune**
- Provides flexibility for customizations
- Can be smaller in size to save cost
- **Commercial / Non-commercial use**

Open-source LLMs:

Non-commercial Use

Meta AI
LLaMA

Commercial Use

databricks
Dolly

mosaicML
MPT

Proprietary Models

- Usually offered as **LLMs-as-a-service**
- Some can be **fine-tuned**
- Restrictive licenses for usage and modification

Proprietary LLMs:

ANTHROP\C

OpenAI

ChatGPT

PaLM 2

Three Phases to Learning

How LLMs learn

Pretraining

Rote Learn text

Finetuning

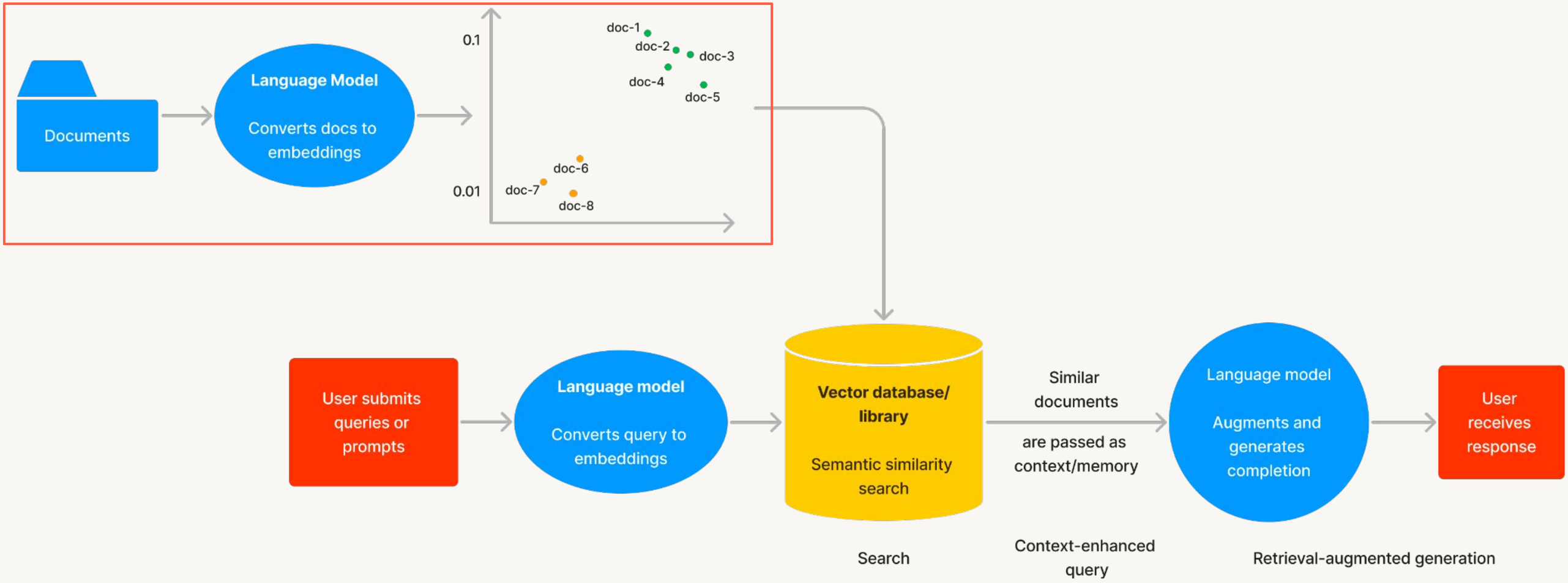
Learning General Instructions

Prompting

How to perform a task

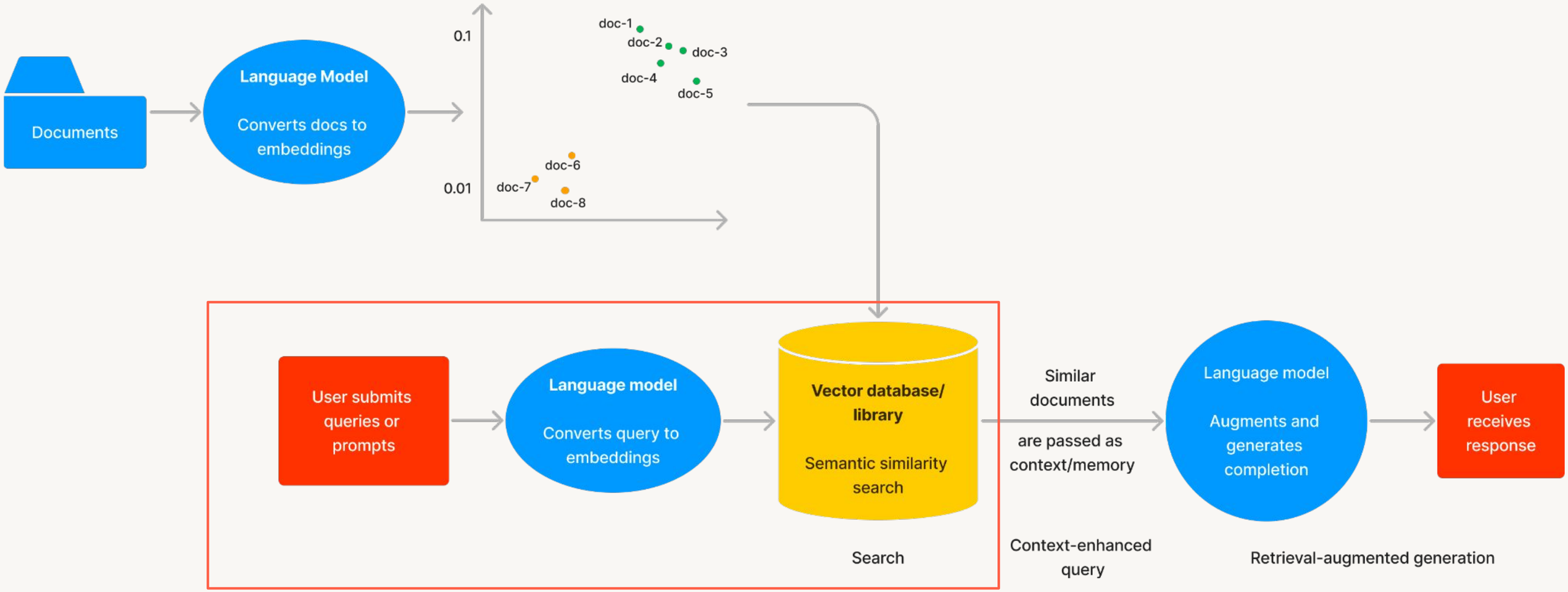
Search and Retrieval-Augmented Generation

The RAG workflow



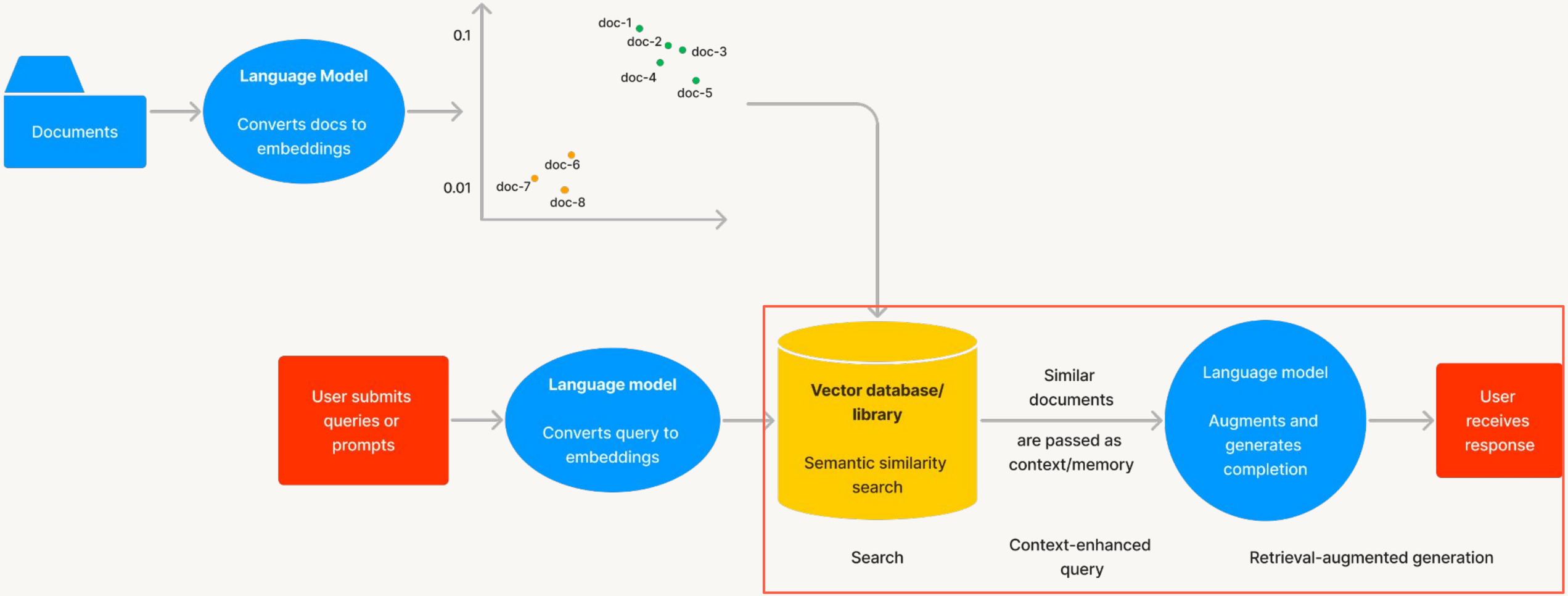
Search and Retrieval-Augmented Generation

The RAG workflow



Search and Retrieval-Augmented Generation

The RAG workflow



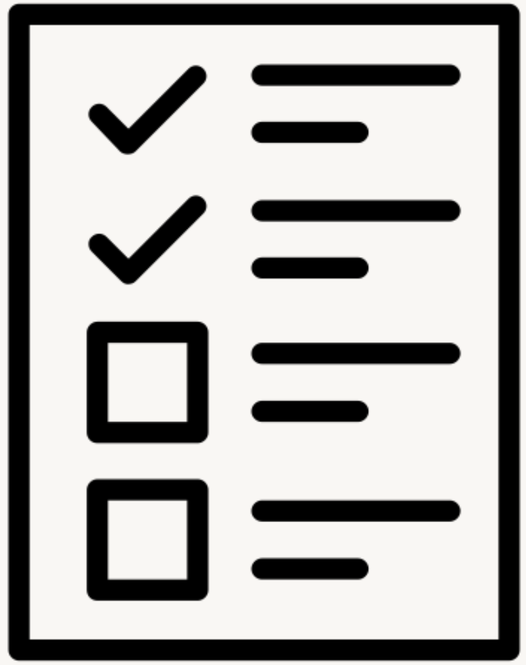
Evaluating LLM Apps

Analysing and comparing outputs

How can we evaluate LLMs?...

What does LLM performance mean?

EVALUATION TIME!



Common LLM metric tables

Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions	17.8	18.1	22.7	28.0	23.0	29.5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Common LLM metric tables

Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA						78.6	84.5	85.0
Natural Questions	17.8	18.1	22.7	28.0	23.0	29.5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Massive Multitask Language Understanding:
https://paperswithcode.com/dataset/mmlu

University general knowledge type questions

Common LLM metric tables

Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions	17					5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Trivia Question and Answers:
https://nlp.cs.washington.edu/triviaqa/

Pub Trivia

Common LLM metric tables

Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions							31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Natural Questions
https://paperswithcode.com/dataset/natural-questions

Wikipedia questions and answers

Common LLM metric tables

Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions	17.					5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Grade School Math Problems
https://paperswithcode.com/dataset/gsm8k

High school math questions

Common LLM metric tables

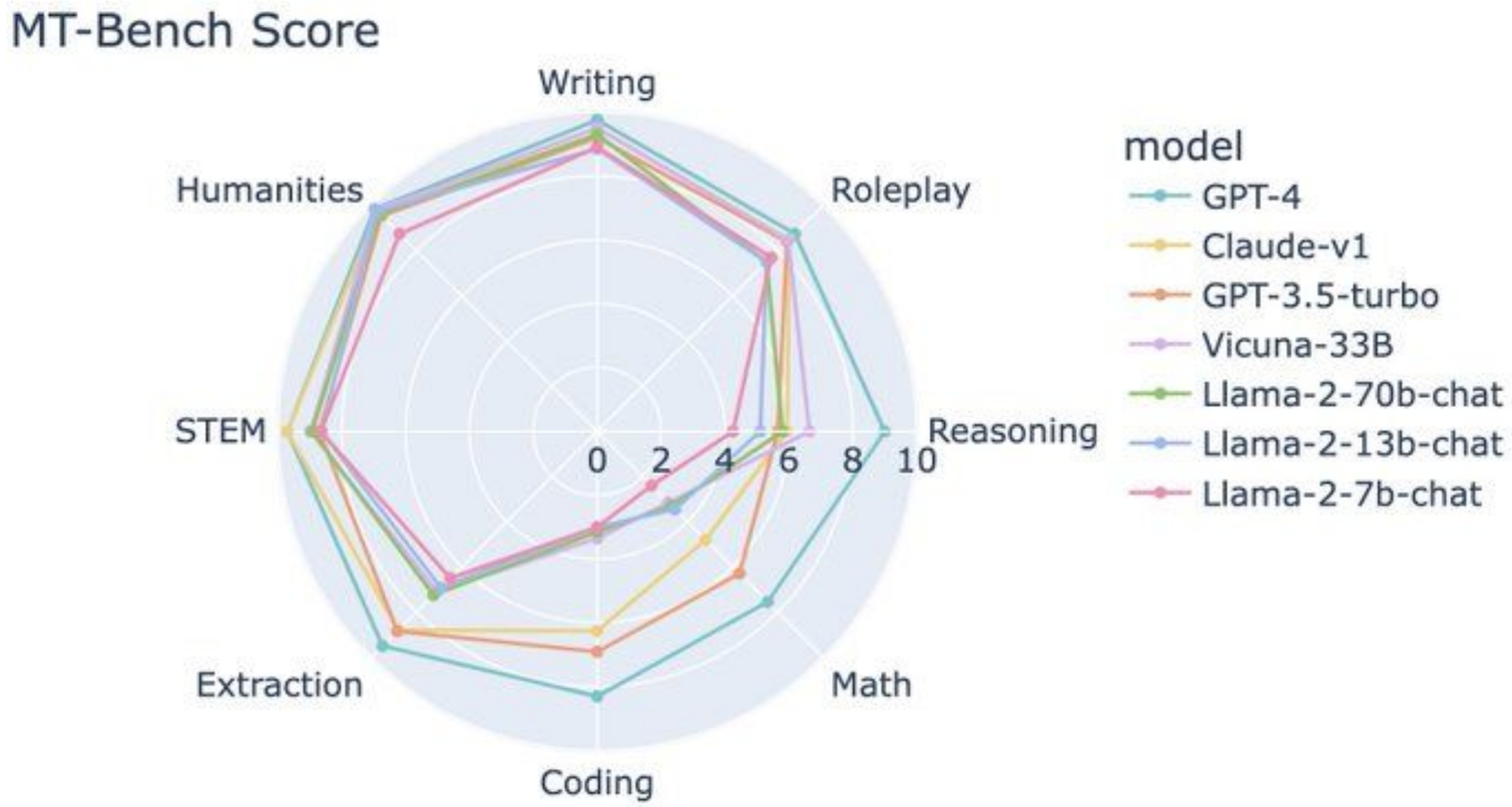
Source: https://ai.meta.com/llama/

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions	17.8	18.1	22.7	28.0	23.0	29.5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8

Does your business problem resemble one of these?

Comparing OSS to Paid providers

Source: https://huggingface.co/spaces/lmsys/mt-bench



Evaluating a RAG

How can we assess a RAG?

First let us look at the process

User Asks a Question

- Could be Specific
- Could be Random
- Could be Broad

Generation

- Given context we need to generate a fluid reply

Retrieval

- Given a question we need to find the right context

How can we assess a RAG?

First let us look at the process

User Asks a Question

- Could be Specific
- Could be Random
- Could be Broad

Questions will vary on use case!

Generation

- Given context we need to generate a fluid reply

Retrieval

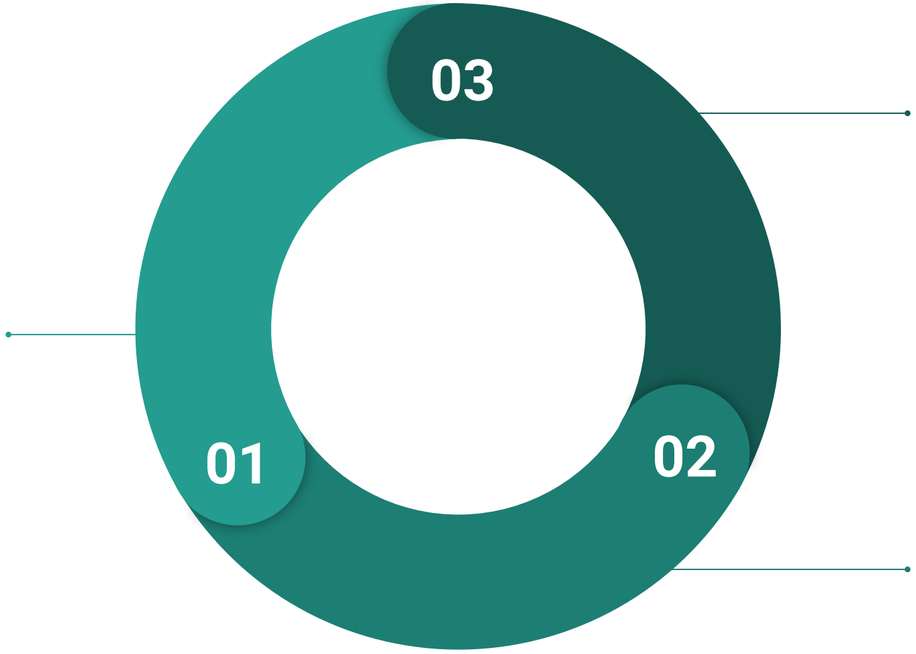
- Given a question we need to find the right context

How can we assess a RAG?

First let us look at the process

User Asks a Question

- Could be Specific
- Could be Random
- Could be Broad



Generation

- Given context we need to generate a fluid reply

Retrieval

- Given a question we r
to find the right conte

Retrieval depends on ingest

How can we assess a RAG?

First let us look at the process

User Asks a Question

- Could be Specific
- Could be Random
- Could be Broad

Generation

- Given context w
  generate a

Generation depends on context

Retrieval

- Given a question we need to find the right context

How to evaluate a RAG architecture?

Ragas approach: https://github.com/explodinggradients/ragas

Retrieval

Did I find what I wanted?

- Relevance of context

- Ability to find the right context

Generation

Did answer make sense?

- Given the context, was the answer correct?

- Did the LLM just answer from the context?

Requirements to apply metrics

- Representative Questions!
  - With associated answers
- Answers with relevant context mapping
- Manual is hard to scale!

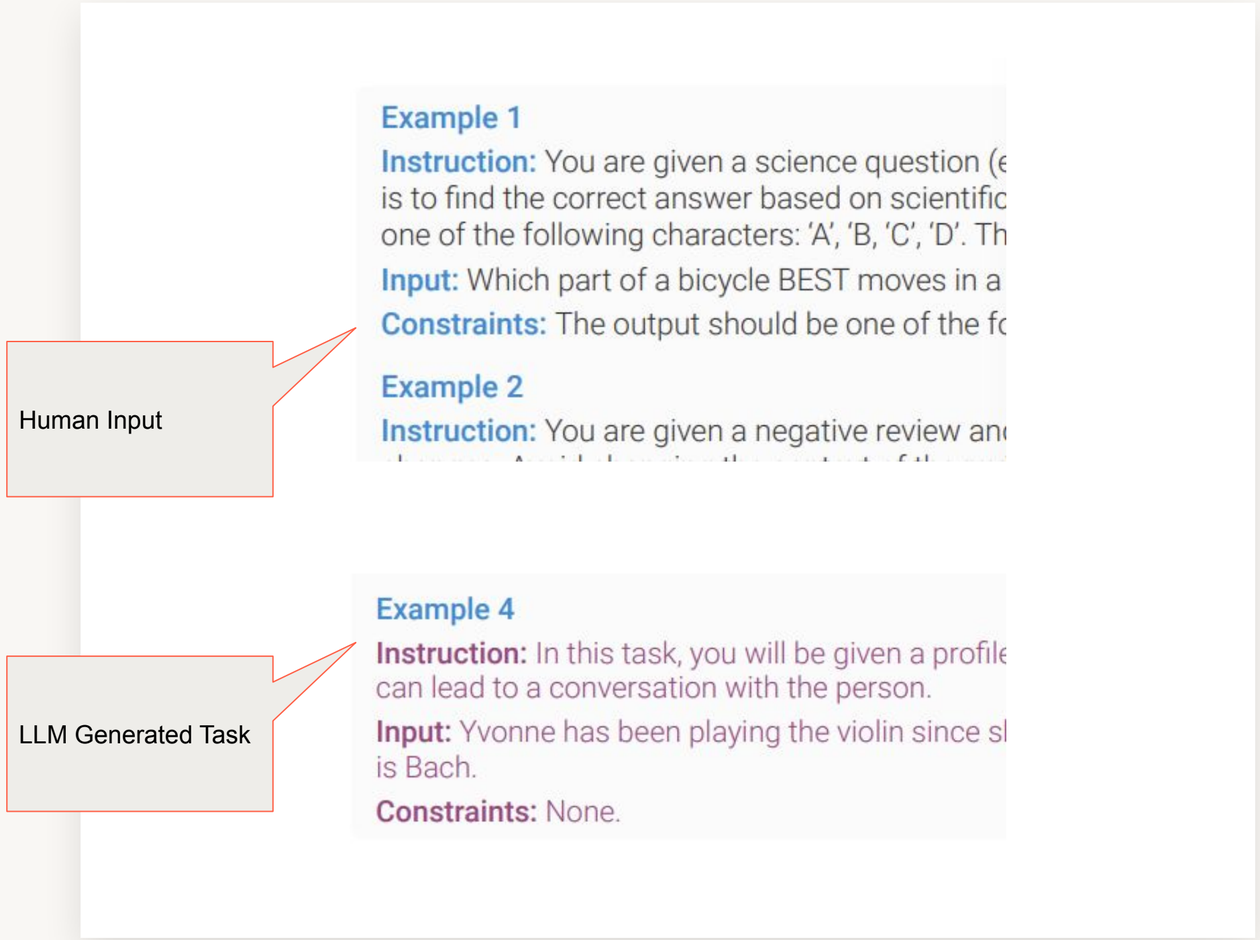
Scaling up example generation

Unnatural Instructions: https://arxiv.org/pdf/2212.09689.pdf

Key Ingredients

- Examples
  - Examples of the full types of outputs that are required including structure

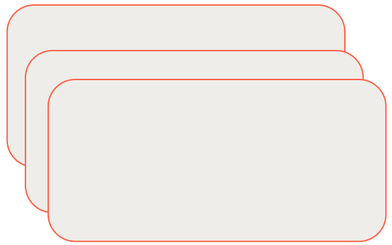
- Existing decent LLM:



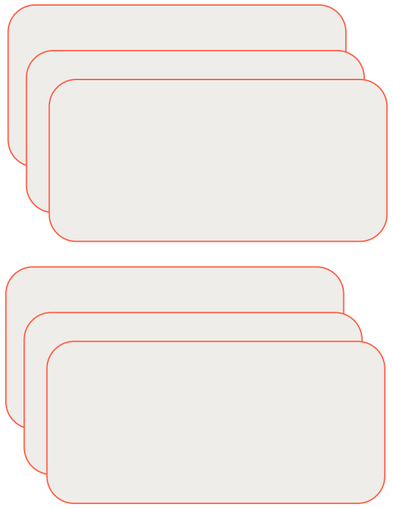
Scaling up example generation

Unnatural Instructions: https://arxiv.org/pdf/2212.09689.pdf

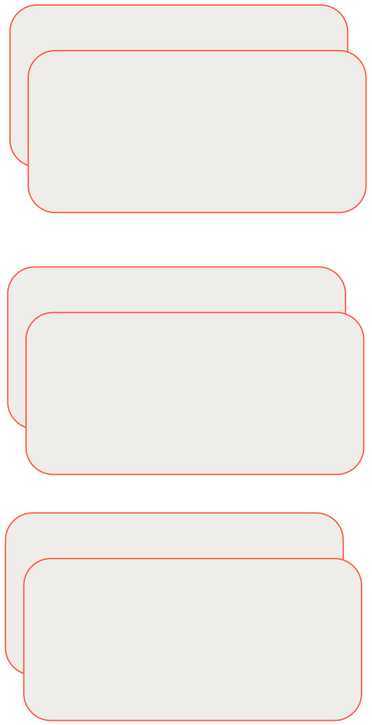
Human Generated Examples - 15



LLM Generated Tasks - 64000

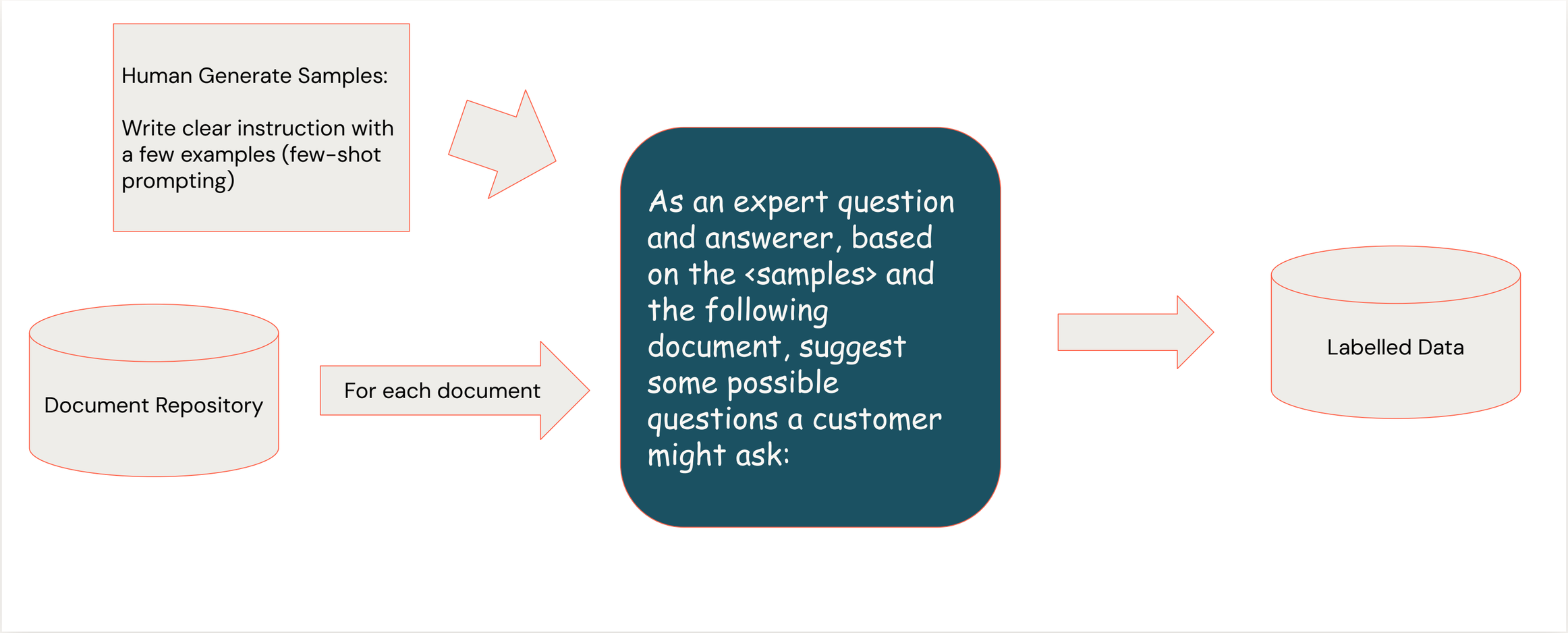


LLM Generated Task Examples - 240000



Use LLMs to generate evaluation labels

How to quickly scale:

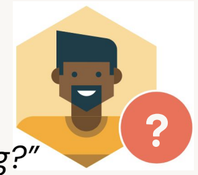


Enhancing your RAG!

Possible approaches to improvement

1 Data preparation

2 Inferences



Embedding serverless endpoint
Transform text to vector

"How can I track Databricks billing?"

doc

raw doc

Clean dataset
Doc chunks

0.1 0.2 1.1 0.8 0.3 1.20 1 ...
0.1 0.2 1.1 0.8 0.3 1.20 1 ...
0.1 0.2 1.1 0.8 0.3 1.20 1 ...

Databricks Vector Search index
automatically sycn your delta table

Databricks chatbot model endpoint

Engineer custom prompt with extended context

Knowing this context :
- <*Similar doc 1 from chroma*>
- <*Similar doc 2 from chroma*>..

Answer the question:
How to track Databricks billing?

Download docs
doc.databricks.com

Parse and split doc
Split html documents and extract text

Real time Similarity search
Find similar docs as context to the prompt

Databricks AI Gateway
Proxy to models
route/credential/throughput/logging

MosaicML Endpoint (llama2)

"Here is how you can easily track your billing: ..."

1 Data preparation

Embedding serverless endpoint
Transform text to vector

doc

raw doc

Clean dataset
Doc chunks

0.1 0.2 1.1 0.8 0.3 1.20 1 ...
0.1 0.2 1.1 0.8 0.3 1.20 1 ...
0.1 0.2 1.1 0.8 0.3 1.20 1 ...

Databricks Vector Search index
automatically sycn your delta table

Download docs
doc.databricks.com

Parse and split doc
Split html documents and extract text

Real time Similarity search
Find similar docs as context to the prompt

2 Inferences

"How can I track Databricks billing?"

Databricks chatbot model endpoint

Engineer custom prompt with extended context

Knowing this context :
- *<Similar doc 1 from chroma>*
- *<Similar doc 2 from chroma>..*

Answer the question:
How to track Databricks billing?

Databricks AI Gateway
Proxy to models
route/credential/throughput/logging

MosaicML Endpoint (llama2)

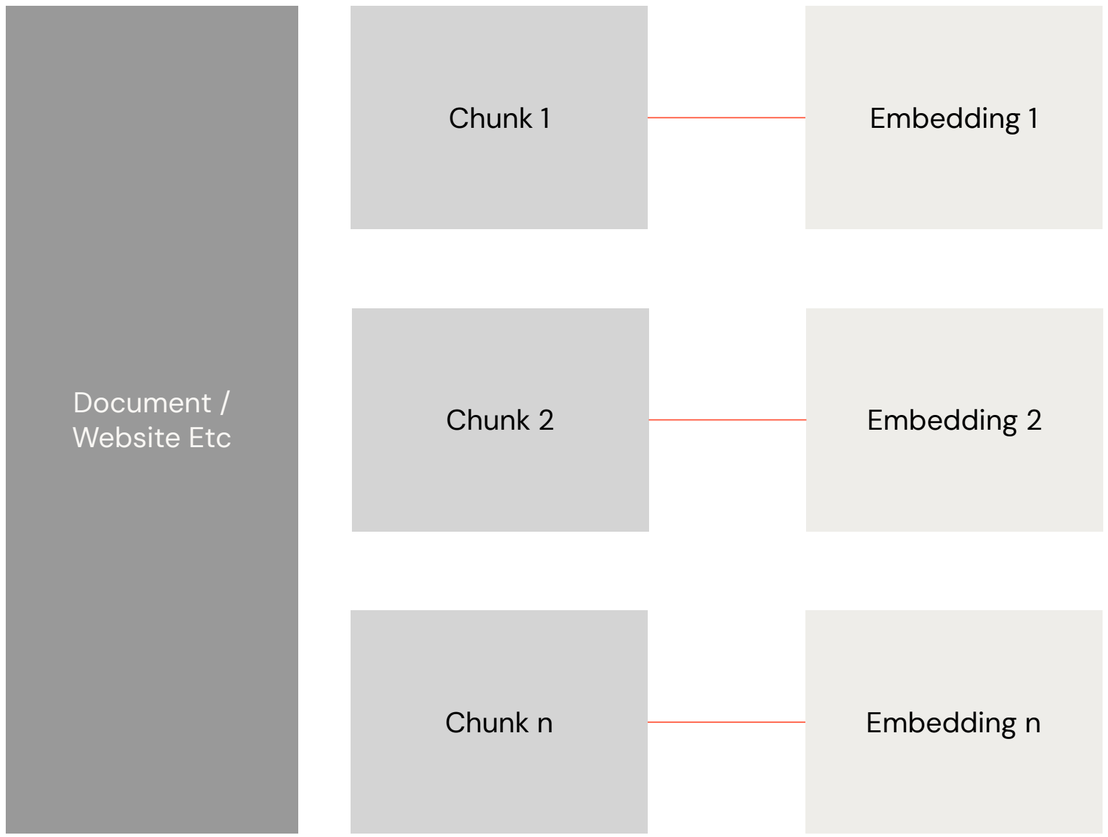
"Here is how you can easily track your billing: ..."

Advanced Chunking

Key Techniques to boost retrieval

Basic Pattern

A first pass at chunking



Features:

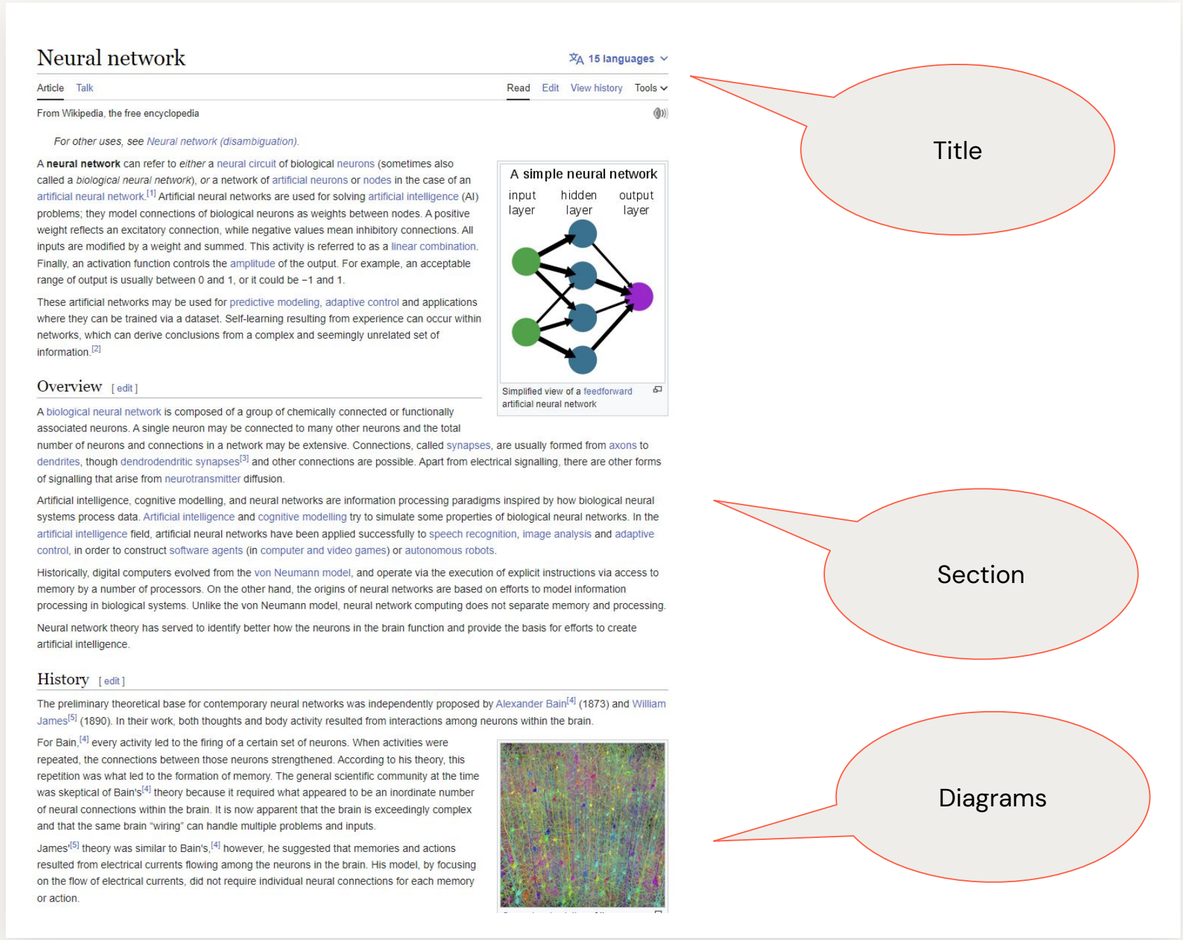
- Even chunks
- 1:1 mapping of chunk to embedding

Constraints:

- Chunks could be out of Context
- No sense of organisation in chunk

Lets Look Deeper

How should we organise it?



Neural network

15 languages

Article Talk | Read Edit View history Tools

From Wikipedia, the free encyclopedia

For other uses, see Neural network (disambiguation).

A **neural network** can refer to *either* a neural circuit of biological neurons (sometimes also called a *biological neural network*), *or* a network of artificial neurons or nodes in the case of an artificial neural network.[1] Artificial neural networks are used for solving artificial intelligence (AI) problems; they model connections of biological neurons as weights between nodes. A positive weight reflects an excitatory connection, while negative values mean inhibitory connections. All inputs are modified by a weight and summed. This activity is referred to as a linear combination. Finally, an activation function controls the amplitude of the output. For example, an acceptable range of output is usually between 0 and 1, or it could be −1 and 1.

These artificial networks may be used for predictive modeling, adaptive control and applications where they can be trained via a dataset. Self-learning resulting from experience can occur within networks, which can derive conclusions from a complex and seemingly unrelated set of information.[2]

A simple neural network
input layer
hidden layer
output layer

Simplified view of a feedforward artificial neural network

Overview [edit]

A biological neural network is composed of a group of chemically connected or functionally associated neurons. A single neuron may be connected to many other neurons and the total number of neurons and connections in a network may be extensive. Connections, called synapses, are usually formed from axons to dendrites, though dendrodendritic synapses[3] and other connections are possible. Apart from electrical signalling, there are other forms of signalling that arise from neurotransmitter diffusion.

Artificial intelligence, cognitive modelling, and neural networks are information processing paradigms inspired by how biological neural systems process data. Artificial intelligence and cognitive modelling try to simulate some properties of biological neural networks. In the artificial intelligence field, artificial neural networks have been applied successfully to speech recognition, image analysis and adaptive control, in order to construct software agents (in computer and video games) or autonomous robots.

Historically, digital computers evolved from the von Neumann model, and operate via the execution of explicit instructions via access to memory by a number of processors. On the other hand, the origins of neural networks are based on efforts to model information processing in biological systems. Unlike the von Neumann model, neural network computing does not separate memory and processing.

Neural network theory has served to identify better how the neurons in the brain function and provide the basis for efforts to create artificial intelligence.

History [edit]

The preliminary theoretical base for contemporary neural networks was independently proposed by Alexander Bain[4] (1873) and William James[5] (1890). In their work, both thoughts and body activity resulted from interactions among neurons within the brain.

For Bain,[4] every activity led to the firing of a certain set of neurons. When activities were repeated, the connections between those neurons strengthened. According to his theory, this repetition was what led to the formation of memory. The general scientific community at the time was skeptical of Bain's[4] theory because it required what appeared to be an inordinate number of neural connections within the brain. It is now apparent that the brain is exceedingly complex and that the same brain "wiring" can handle multiple problems and inputs.

James'[5] theory was similar to Bain's,[4] however, he suggested that memories and actions resulted from electrical currents flowing among the neurons in the brain. His model, by focusing on the flow of electrical currents, did not require individual neural connections for each memory or action.

Title

Section

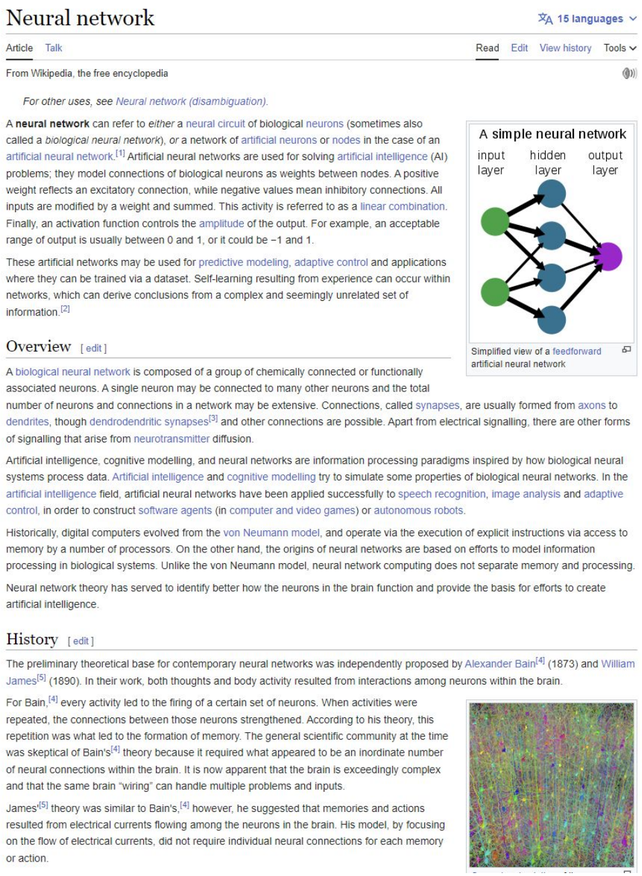
Diagrams

Some Ideas:

- Break up into sections
- Include metadata tags
- Chunk by paragraph

More Advanced Methods

We can use LLMs to help



Neural network

15 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

For other uses, see Neural network (disambiguation).

A **neural network** can refer to *either* a neural circuit of biological neurons (sometimes also called a *biological neural network*), *or* a network of artificial neurons or nodes in the case of an artificial neural network.[1] Artificial neural networks are used for solving artificial intelligence (AI) problems; they model connections of biological neurons as weights between nodes. A positive weight reflects an excitatory connection, while negative values mean inhibitory connections. All inputs are modified by a weight and summed. This activity is referred to as a linear combination. Finally, an activation function controls the amplitude of the output. For example, an acceptable range of output is usually between 0 and 1, or it could be −1 and 1.

These artificial networks may be used for predictive modeling, adaptive control and applications where they can be trained via a dataset. Self-learning resulting from experience can occur within networks, which can derive conclusions from a complex and seemingly unrelated set of information.[2]

A simple neural network

input layer hidden layer output layer

Simplified view of a feedforward artificial neural network

Overview [edit]

A biological neural network is composed of a group of chemically connected or functionally associated neurons. A single neuron may be connected to many other neurons and the total number of neurons and connections in a network may be extensive. Connections, called synapses, are usually formed from axons to dendrites, though dendrodendritic synapses[3] and other connections are possible. Apart from electrical signalling, there are other forms of signalling that arise from neurotransmitter diffusion.

Artificial intelligence, cognitive modelling, and neural networks are information processing paradigms inspired by how biological neural systems process data. Artificial intelligence and cognitive modelling try to simulate some properties of biological neural networks. In the artificial intelligence field, artificial neural networks have been applied successfully to speech recognition, image analysis and adaptive control, in order to construct software agents (in computer and video games) or autonomous robots.

Historically, digital computers evolved from the von Neumann model, and operate via the execution of explicit instructions via access to memory by a number of processors. On the other hand, the origins of neural networks are based on efforts to model information processing in biological systems. Unlike the von Neumann model, neural network computing does not separate memory and processing.

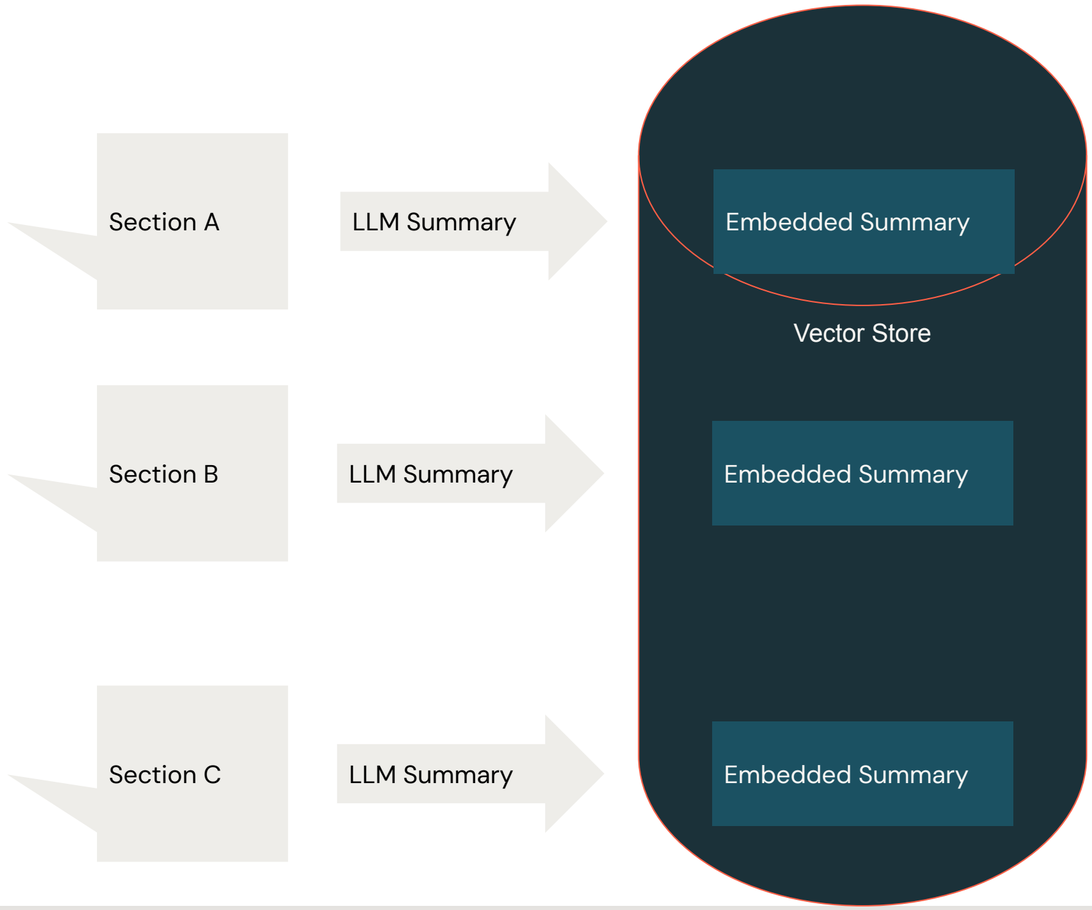
Neural network theory has served to identify better how the neurons in the brain function and provide the basis for efforts to create artificial intelligence.

History [edit]

The preliminary theoretical base for contemporary neural networks was independently proposed by Alexander Bain[4] (1873) and William James[5] (1890). In their work, both thoughts and body activity resulted from interactions among neurons within the brain.

For Bain,[4] every activity led to the firing of a certain set of neurons. When activities were repeated, the connections between those neurons strengthened. According to his theory, this repetition was what led to the formation of memory. The general scientific community at the time was skeptical of Bain's[4] theory because it required what appeared to be an inordinate number of neural connections within the brain. It is now apparent that the brain is exceedingly complex and that the same brain "wiring" can handle multiple problems and inputs.

James[5] theory was similar to Bain's,[4] however, he suggested that memories and actions resulted from electrical currents flowing among the neurons in the brain. His model, by focusing on the flow of electrical currents, did not require individual neural connections for each memory or action.

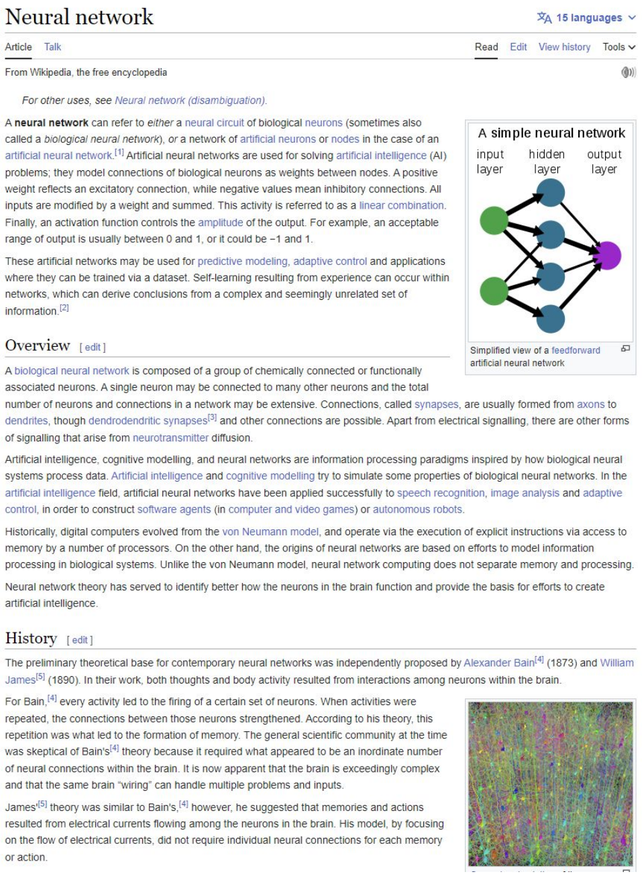


Key steps:

- Summarise Sections
- Embed Summaries
- Retrieve summaries with vector search but insert full section into prompt

Another example

https://gpt-index.readthedocs.io/en/latest/examples/metadata_extraction/MetadataExtraction_LLMSurvey.html



Neural network

15 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

For other uses, see Neural network (disambiguation).

A **neural network** can refer to *either* a neural circuit of biological neurons (sometimes also called a *biological neural network*), *or* a network of artificial neurons or nodes in the case of an artificial neural network.[1] Artificial neural networks are used for solving artificial intelligence (AI) problems; they model connections of biological neurons as weights between nodes. A positive weight reflects an excitatory connection, while negative values mean inhibitory connections. All inputs are modified by a weight and summed. This activity is referred to as a linear combination. Finally, an activation function controls the amplitude of the output. For example, an acceptable range of output is usually between 0 and 1, or it could be −1 and 1.

These artificial networks may be used for predictive modeling, adaptive control and applications where they can be trained via a dataset. Self-learning resulting from experience can occur within networks, which can derive conclusions from a complex and seemingly unrelated set of information.[2]

A simple neural network

input layer hidden layer output layer

Simplified view of a feedforward artificial neural network

Overview [edit]

A biological neural network is composed of a group of chemically connected or functionally associated neurons. A single neuron may be connected to many other neurons and the total number of neurons and connections in a network may be extensive. Connections, called synapses, are usually formed from axons to dendrites, though dendrodendritic synapses[3] and other connections are possible. Apart from electrical signalling, there are other forms of signalling that arise from neurotransmitter diffusion.

Artificial intelligence, cognitive modelling, and neural networks are information processing paradigms inspired by how biological neural systems process data. Artificial intelligence and cognitive modelling try to simulate some properties of biological neural networks. In the artificial intelligence field, artificial neural networks have been applied successfully to speech recognition, image analysis and adaptive control, in order to construct software agents (in computer and video games) or autonomous robots.

Historically, digital computers evolved from the von Neumann model, and operate via the execution of explicit instructions via access to memory by a number of processors. On the other hand, the origins of neural networks are based on efforts to model information processing in biological systems. Unlike the von Neumann model, neural network computing does not separate memory and processing.

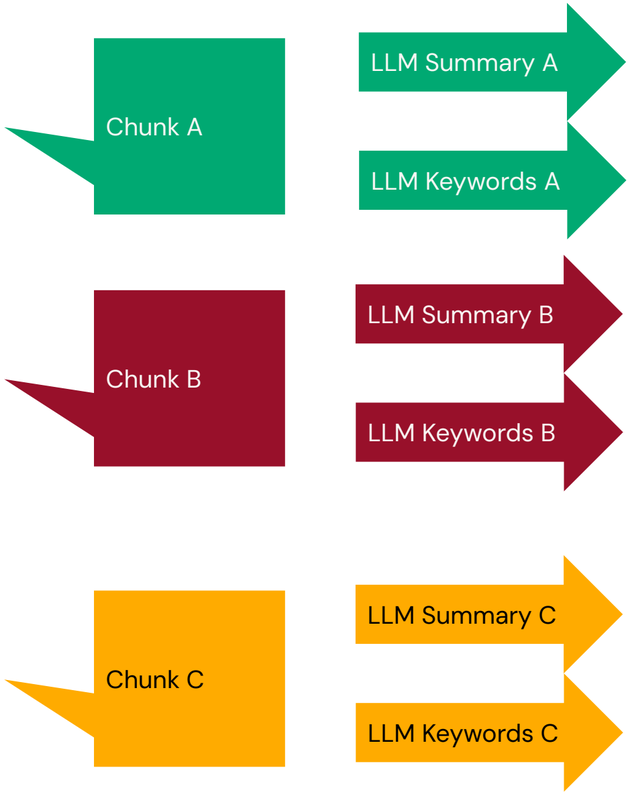
Neural network theory has served to identify better how the neurons in the brain function and provide the basis for efforts to create artificial intelligence.

History [edit]

The preliminary theoretical base for contemporary neural networks was independently proposed by Alexander Bain[4] (1873) and William James[5] (1890). In their work, both thoughts and body activity resulted from interactions among neurons within the brain.

For Bain,[4] every activity led to the firing of a certain set of neurons. When activities were repeated, the connections between those neurons strengthened. According to his theory, this repetition was what led to the formation of memory. The general scientific community at the time was skeptical of Bain's[4] theory because it required what appeared to be an inordinate number of neural connections within the brain. It is now apparent that the brain is exceedingly complex and that the same brain "wiring" can handle multiple problems and inputs.

James[5] theory was similar to Bain's,[4] however, he suggested that memories and actions resulted from electrical currents flowing among the neurons in the brain. His model, by focusing on the flow of electrical currents, did not require individual neural connections for each memory or action.



Extraction can be hard though

Real World Examples

Image

Text

Table

Price and disclaimer

Features:

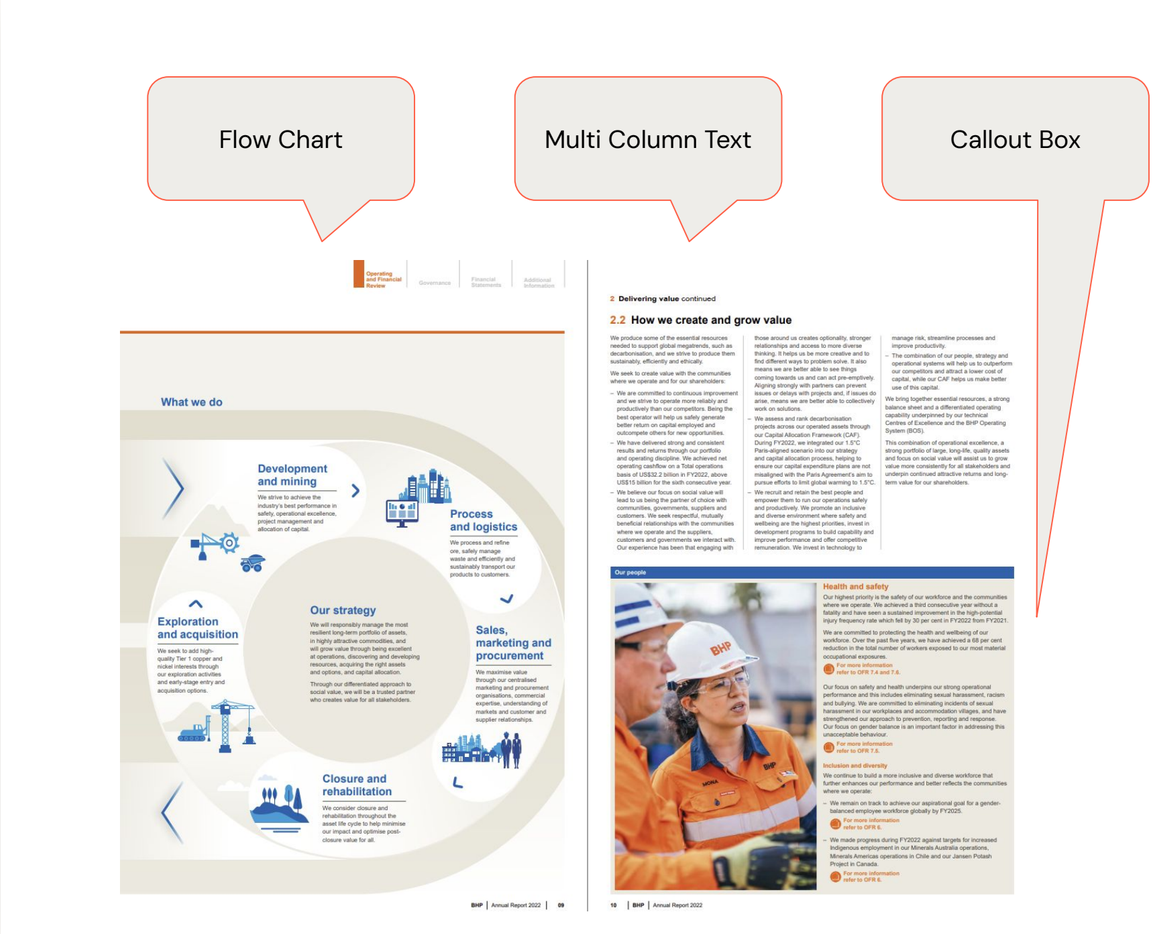
- Text mixed with image
- Irregular placement of text
- Special callouts with different colours

Extraction can be hard though

Real World Examples

Features:

- Parsers may not understand flow charts
- Multi-column text is usually a challenge
- Callout boxes don’t fit neatly in the arrangement



General Approaches

Ways we can solve

Traditional Approach

Libraries:

- PyMuPDF
- PyPDF

Features:

- Breaks down text to into raw constructs
- Very low level requires hardcoding rules

Use a layout model

Libraries:

- Huggingface
  - LayoutLMv3
- doctr
- Donut
- Unstructured

Features:

- Apply Deep learning models built to do text extraction and context extraction

Multi-Modal Models

Models:

- GPT-4
- OpenFlamingo Framework
- Idefics

Features:

- Multimodal LLMs intrinsically understand images but are still more experimental at this stage

Enhancing Retrieval

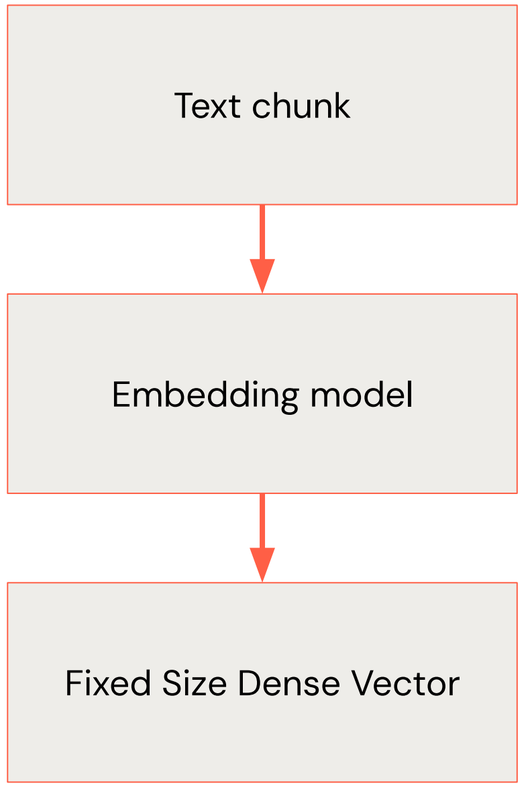
Understanding embeddings

HF Sentence Transformers library

For embedding we use Sentence Transformers:

- Take a whole chunk as input

- Produce a fixed size vector



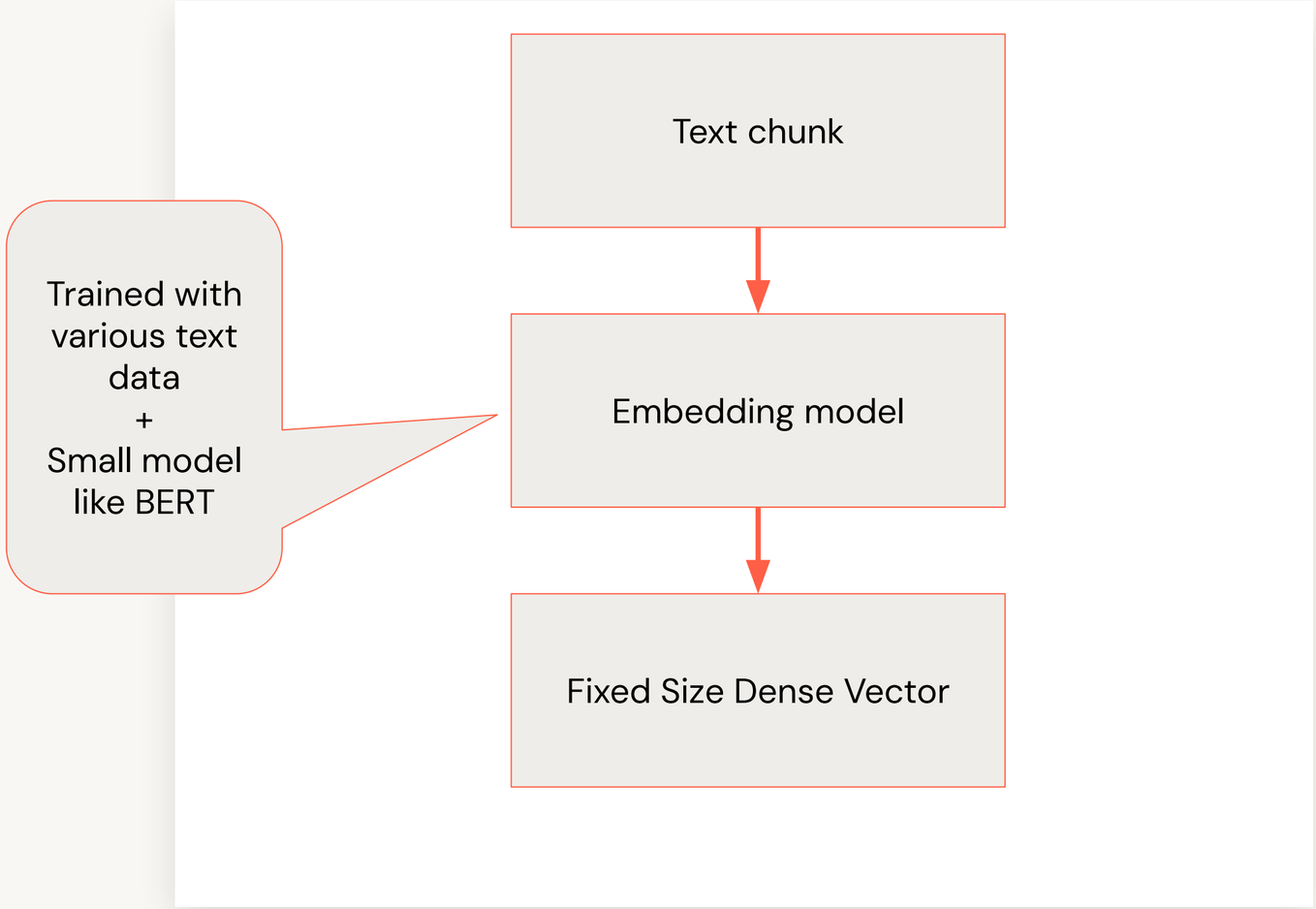
Understanding embeddings

HF Sentence Transformers library

There are various different models that you can use:

- See leaderboard: https://huggingface.co/spaces/mteb/leaderboard

- Embedding will retrieve similar context



Look beyond Vector DBs

Vector DBs are just another search tool

Consider Graph DBs?

Consider Metadata Filtering?

Maybe add a ReRanker?

1 Data preparation

Embedding serverless endpoint
Transform text to vector

doc

raw doc

Clean dataset
Doc chunks

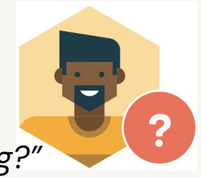
Databricks Vector Search index
automatically sycn your delta table

Download docs
doc.databricks.com

Parse and split doc
Split html documents and extract text

Real time Similarity search
Find similar docs as context to the prompt

2 Inferences



"How can I track Databricks billing?"

Databricks chatbot model endpoint

Engineer custom prompt with extended context

Knowing this context :
- <*Similar doc 1 from chroma*>
- <*Similar doc 2 from chroma*>..

Answer the question:
How to track Databricks billing?

Databricks AI Gateway
Proxy to models
route/credential/throughput/logging

MosaicML Endpoint (llama2)

"Here is how you can easily track your billing: ..."

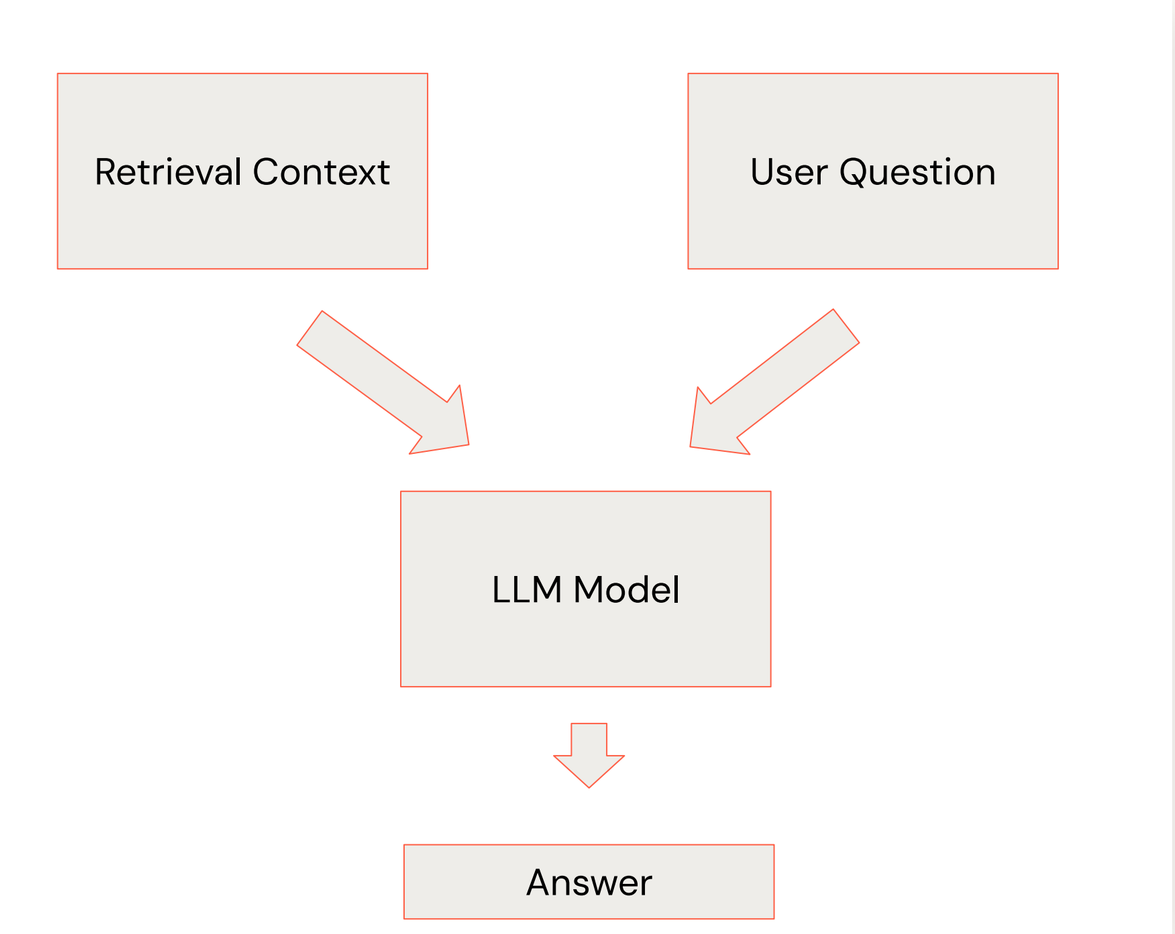
Enhancing Generation

What components are involved?

Some observations:

- Quantized Larger model >> Non-quantized smaller model

- Whilst getting the right context to model is important there are still dials we can tweak here.

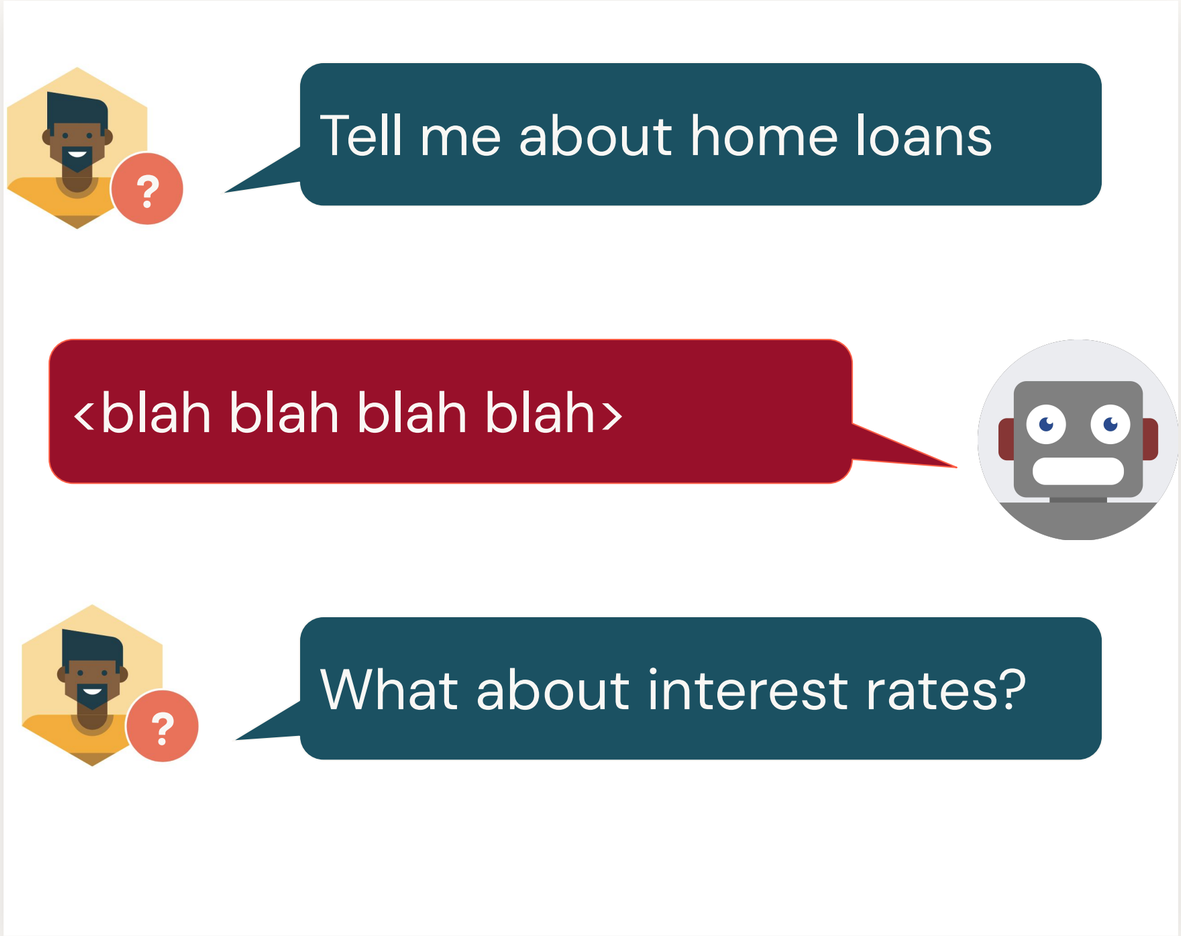


Adding Memory

Our Earlier bot:

- Would not remember the past

- Will not allow a user to progressively unpack their question



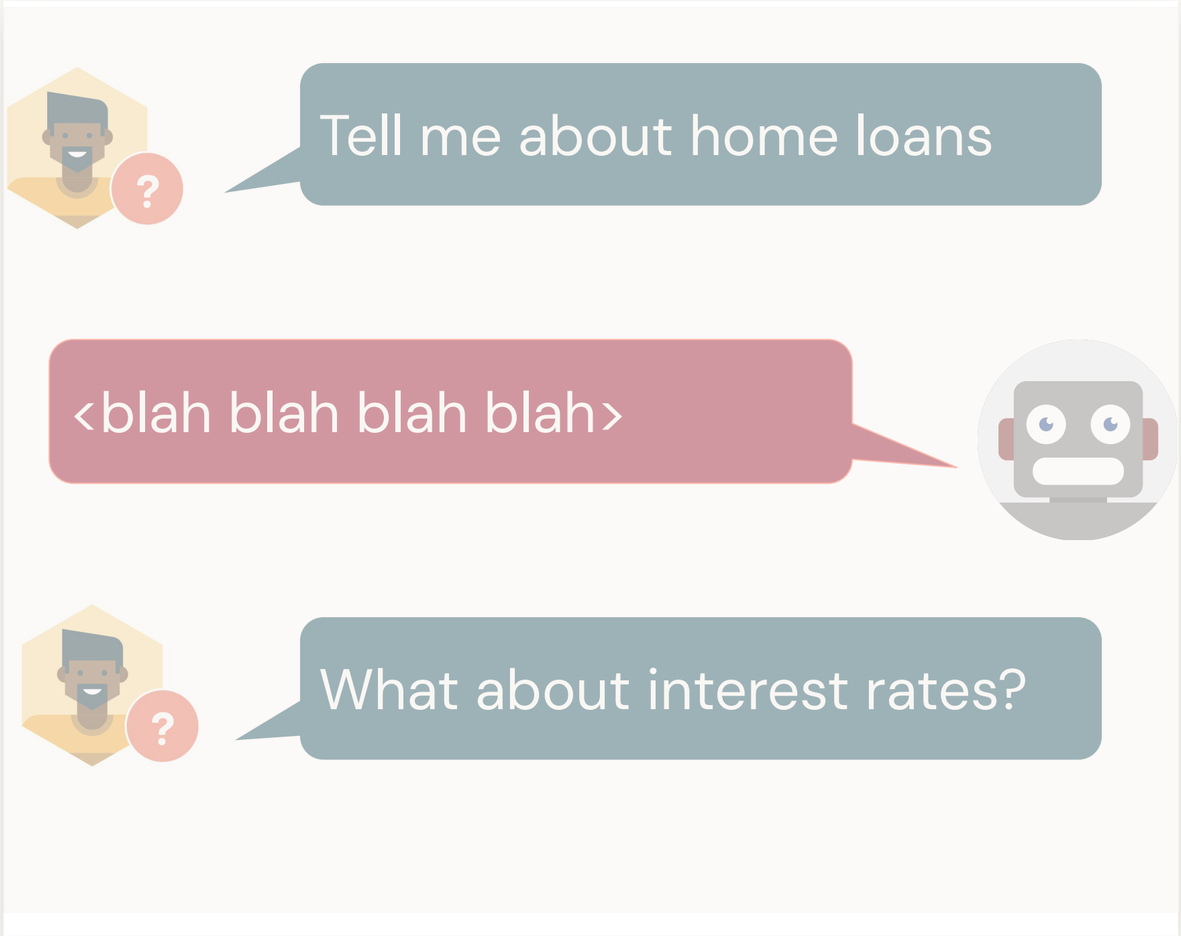
Adding Memory

We can concatenate the chat history in with the prompt.

Options:

- Raw History
  - May hit limit issues
- Summarised History
  - May lose nuisance
  -

(Also consider privacy issues)



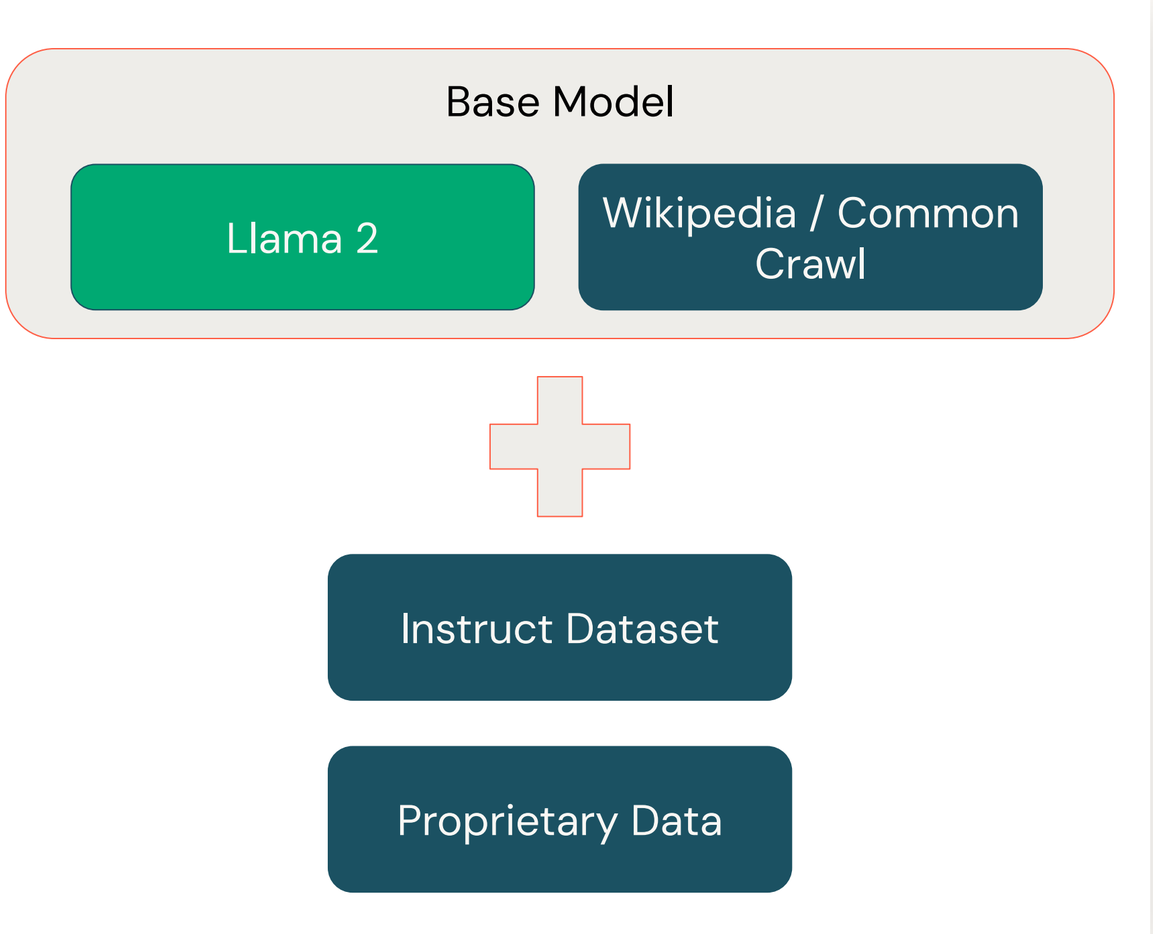
Adding capability to an existing model

Intro to finetuning

Model weights store information:

Ie GPT-3.5 / 4 has no RAG component

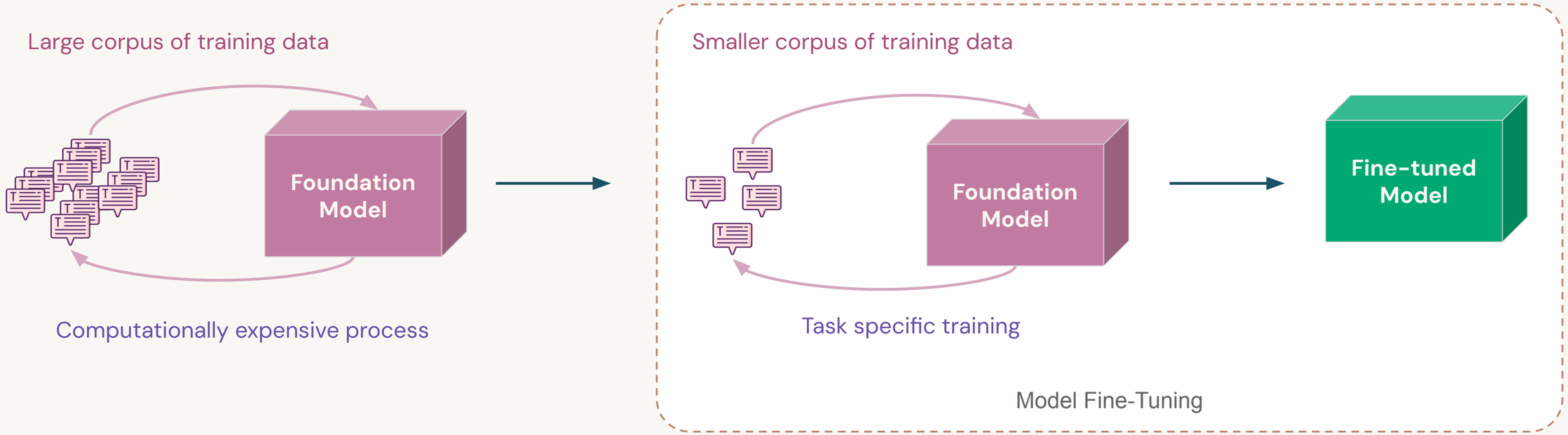
We can rerun the training to add knowledge



Fine Tuned Models

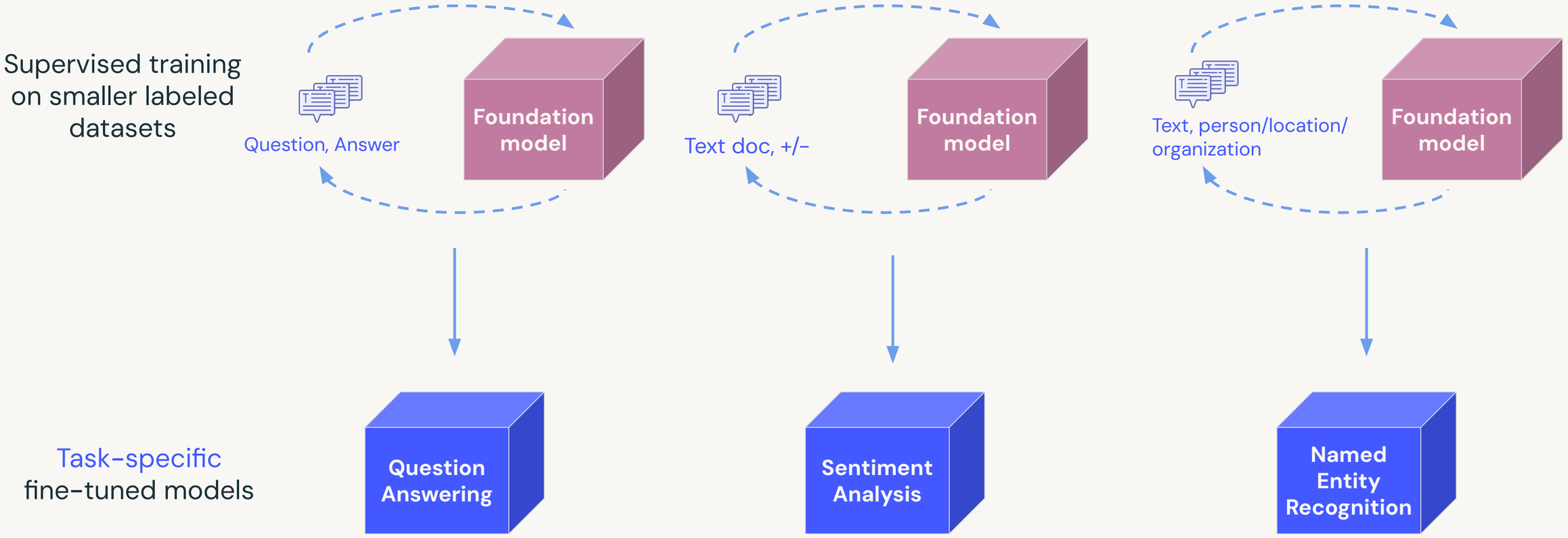
What is fine-tuning and how it works

Fine-tuning: The process of further training a pre-trained model on a specific task or dataset to adapt it for a particular application or domain.



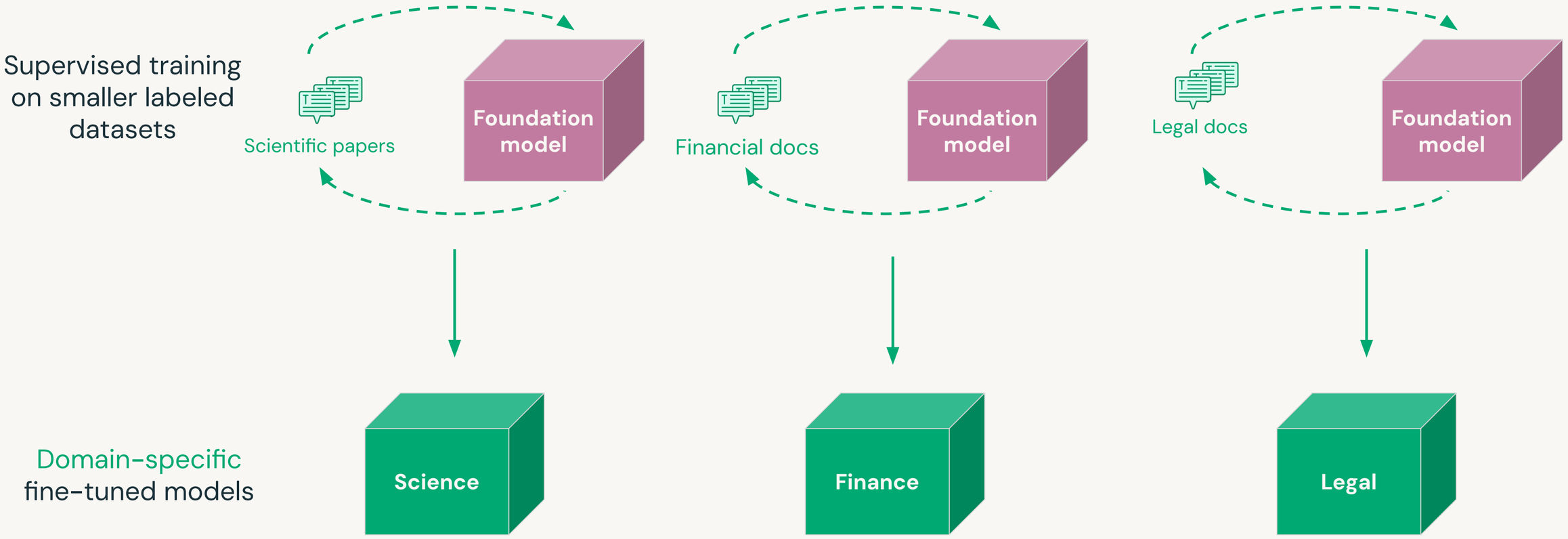
Fine-tuning models

Foundation models can be fine-tuned for **specific tasks**



Fine-tuning models

Foundation models can be fine-tuned for **domain adaptation**



Advanced Options

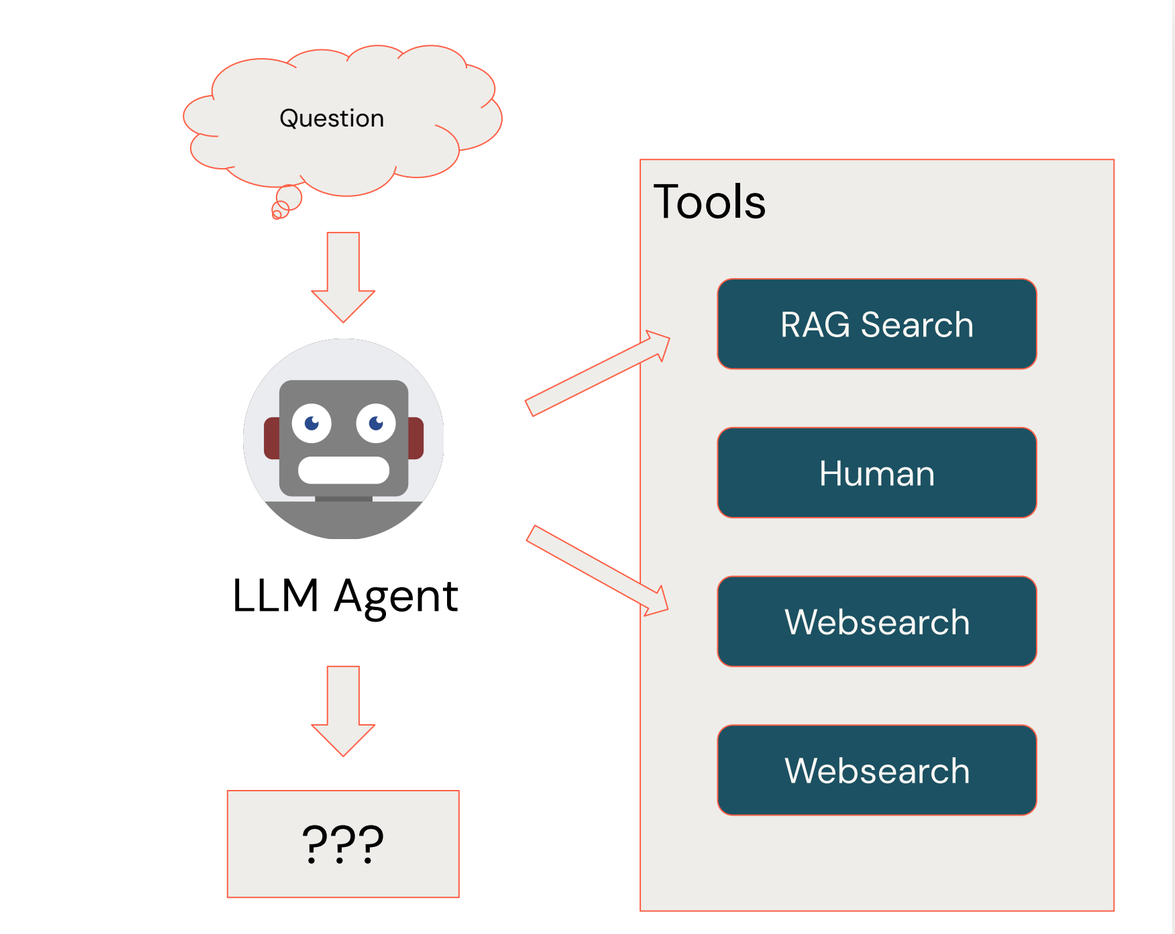
Leverage Agents

At the moment we still have a one hop architecture:

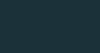
- Question -> Retrieval -> LLM -< Answer

Agents can add logic capabilities

- But adds an extra variability



From Dev to Prod



1 Data preparation

2 Inferences

Embedding serverless endpoint
Transform text to vector

"How can I track Databricks billing?"

doc

raw doc

Clean dataset
Doc chunks

0.1 0.2 1.1 0.8 0.3 1.20 1 ...

Databricks Vector Search index
automatically sycn your delta table

Databricks chatbot model endpoint

Engineer custom prompt with extended context

Knowing this context :
- <*Similar doc 1 from chroma*>
- <*Similar doc 2 from chroma*>..

Answer the question:
How to track Databricks billing?

Download docs
doc.databricks.com

Parse and split doc
Split html documents and extract text

Real time Similarity search
Find similar docs as context to the prompt

Databricks AI Gateway
Proxy to models
route/credential/throughput/logging

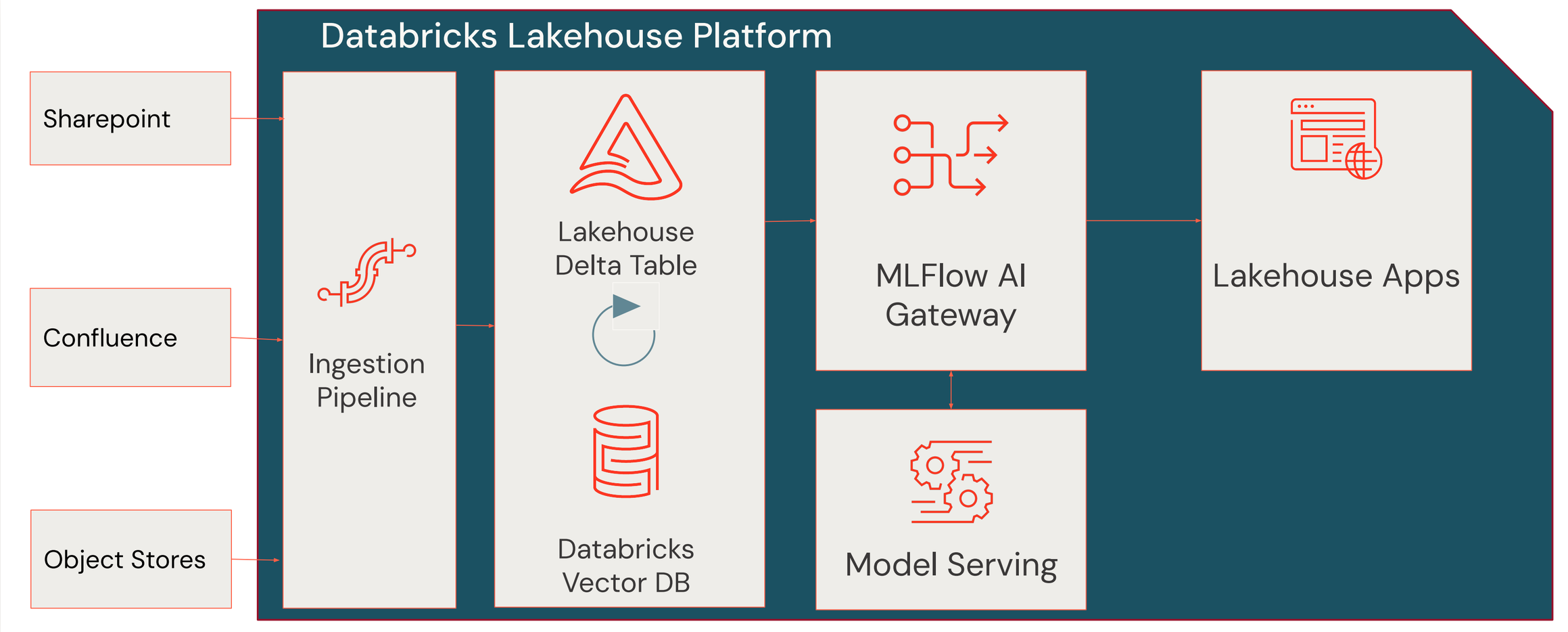
MosaicML Endpoint (llama2)

"Here is how you can easily track your billing: ..."

Moving to Production

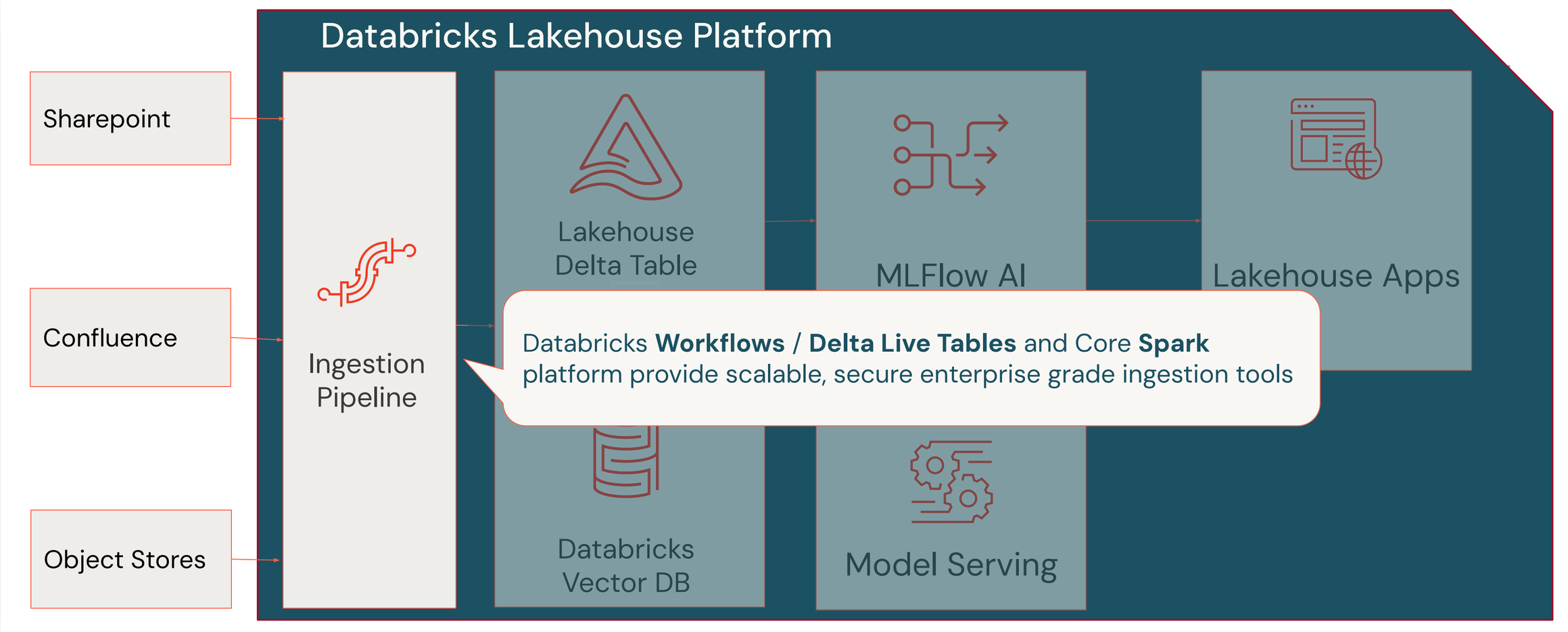
Databricks Lakehouse AI

Building the architecture with Databricks



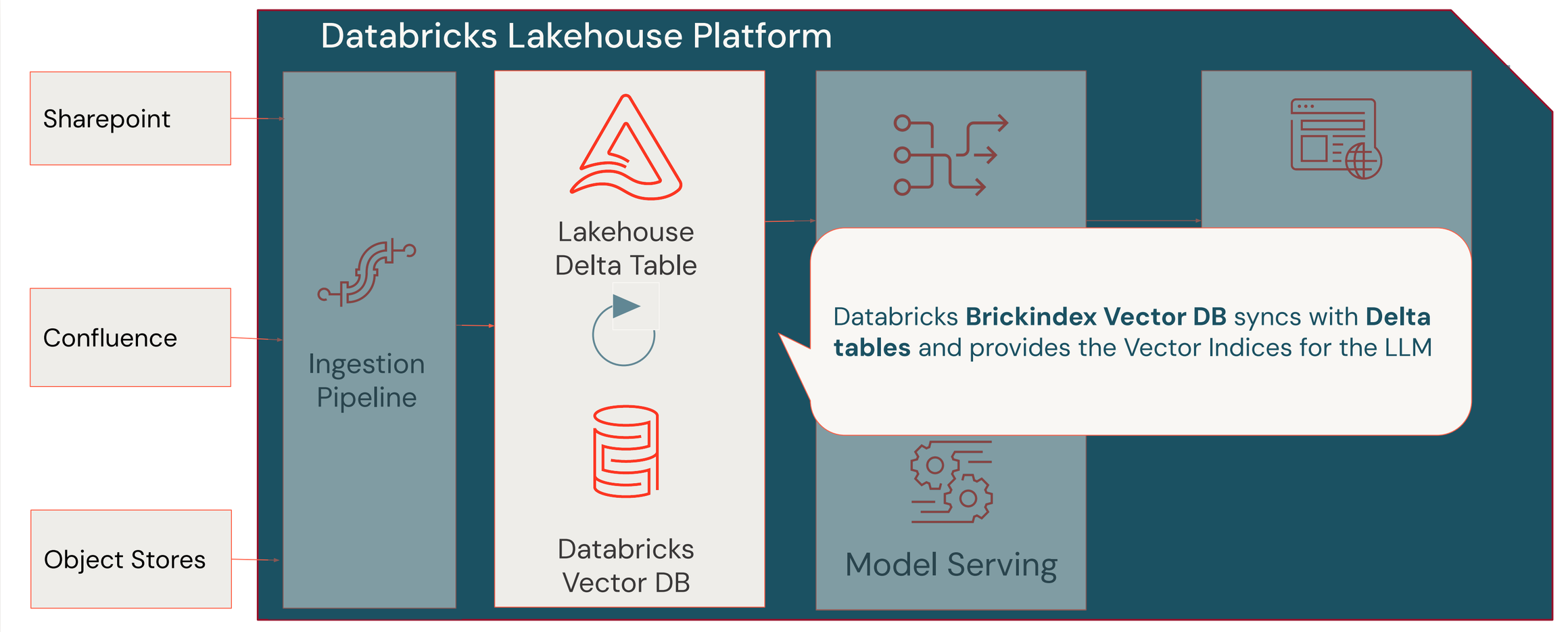
Databricks Lakehouse AI

Building the architecture with Databricks



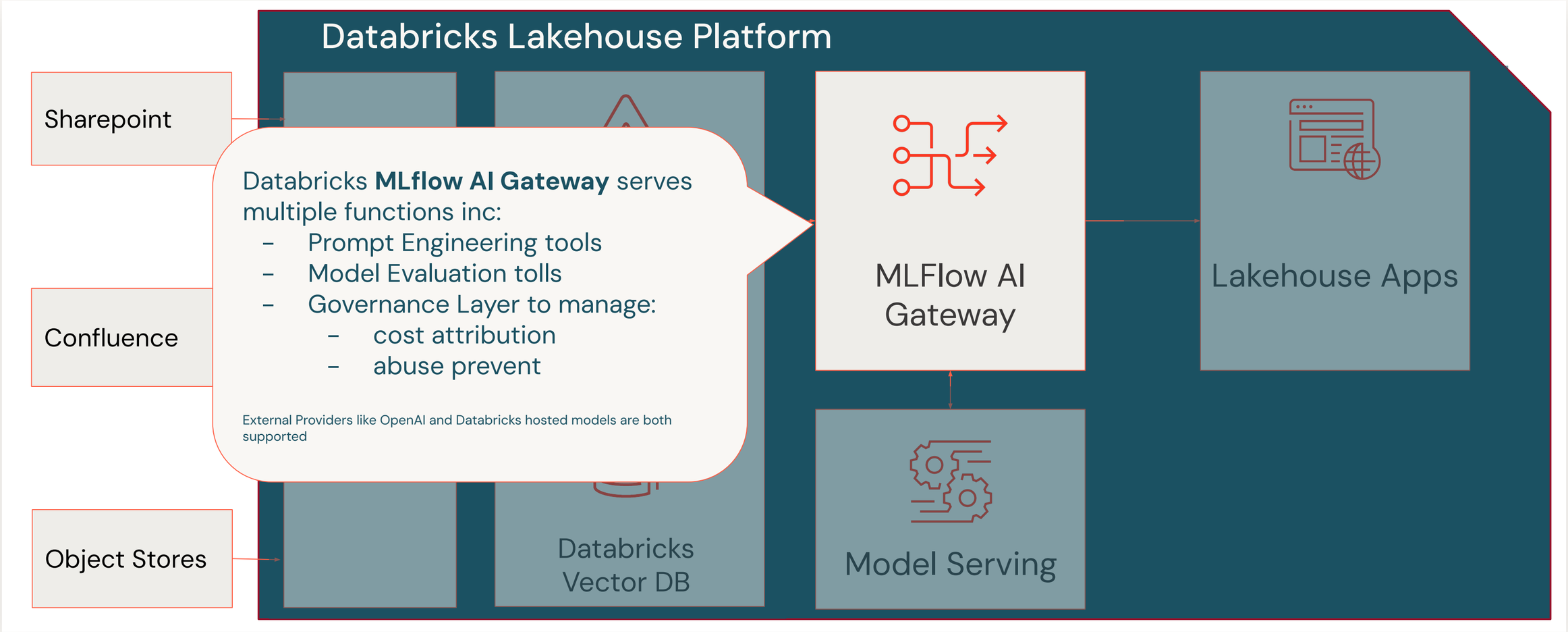
Databricks Lakehouse AI

Building the architecture with Databricks



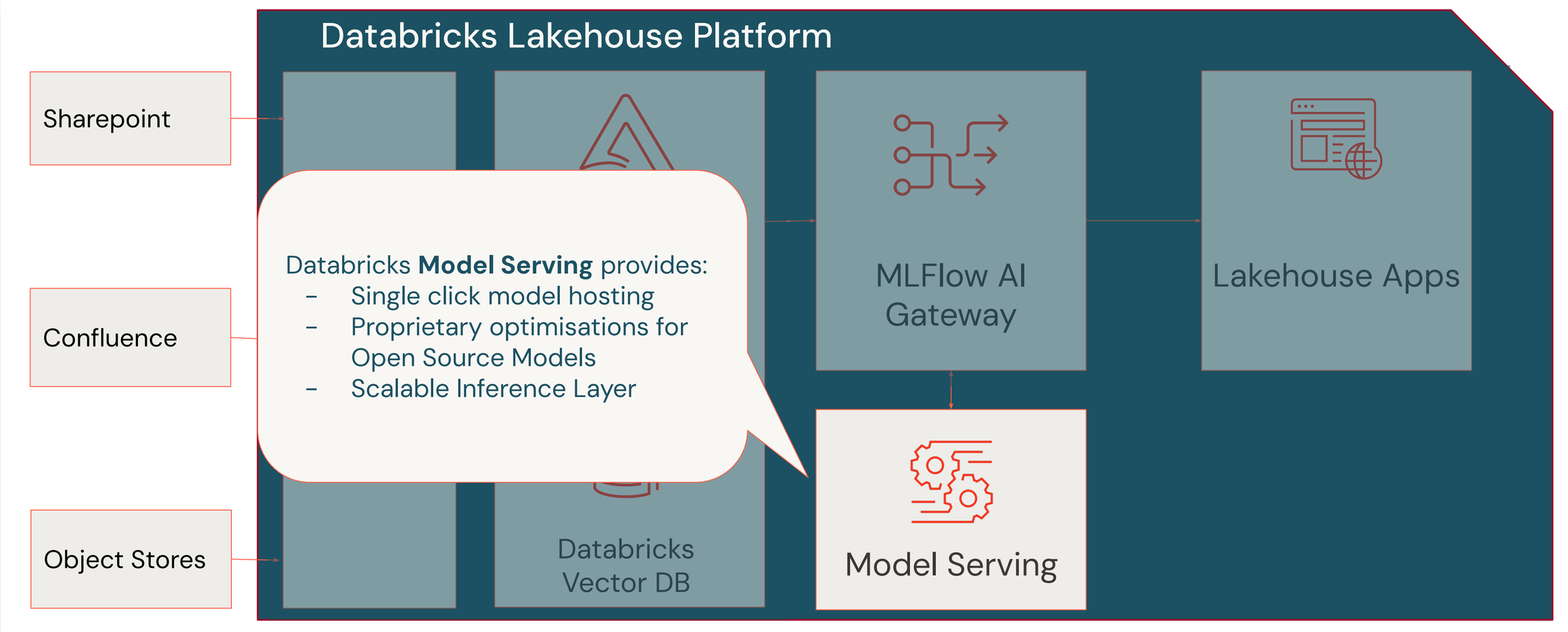
Databricks Lakehouse AI

Building the architecture with Databricks



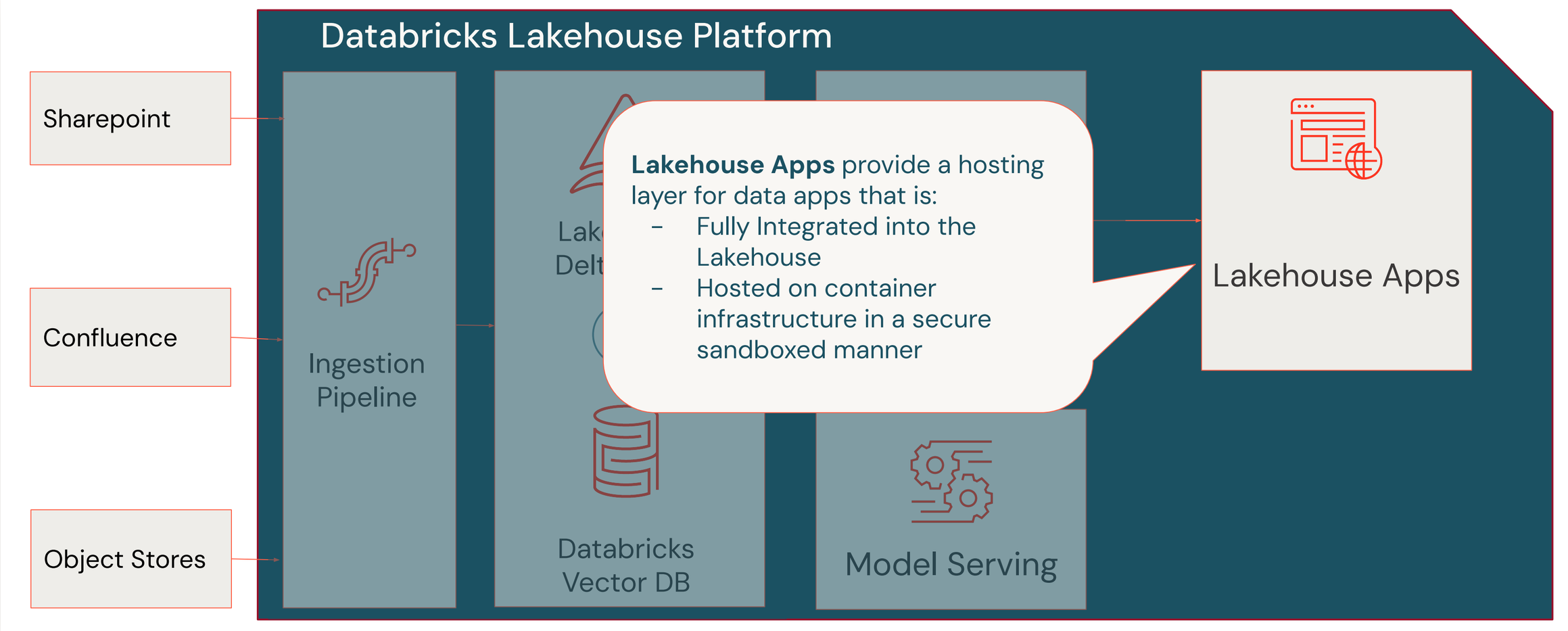
Databricks Lakehouse AI

Building the architecture with Databricks



Databricks Lakehouse AI

Building the architecture with Databricks



Walkthrough

Further Reading

LLM Links

Replit Model and Overview – https://fullstackdeeplearning.com/llm-bootcamp/spring-2023/shabani-train-your-own/

Databricks ML Examples – https://github.com/databricks/databricks-ml-examples

Building LLMs on your data talk – https://www.youtube.com/watch?v=37iUsgJiYos

De-risking Language Models talk – https://www.youtube.com/watch?v=Hofspg_6nvQ

Evaluating LLMs talk – https://www.youtube.com/watch?v=2CIIQ5KZWUM

AI Risks – https://www.youtube.com/watch?v=MK2qtRpjNbU

LLM Hallucinations and Internals – https://www.youtube.com/watch?v=3eXMpSNNQ-s

Further Reading

Databricks Courses and Blogs

Courses mentioned during training and useful resources:

- Databricks Edx Course – https://www.edx.org/course/large-language-models-application-through-production
- Deep dive into LLM Internals – https://www.youtube.com/watch?v=3eXMpSNNQ-s
- Finetuning Blog – https://www.databricks.com/blog/2023/03/20/fine-tuning-large-language-models-hugging-face-and-deepspeed.html
- QLoRa Blog - https://www.databricks.com/blog/efficient-fine-tuning-lora-guide-llms
- Deploy your Own RAG Solution Acceleratior - https://www.databricks.com/resources/demos/tutorials/data-science-and-ai/lakehouse-ai-deploy-your-llm-chatbot
-

Further Reading

Literature

Papers mentioned during training and useful resources:

- Textbooks Are All You Need
- Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor
- LIMA: Less Is More for Alignment
- Llama 2: Open Foundation and Fine-Tuned Chat Models
- Demonstrate-Search-Predict: Composing retrieval and language models for knowledge-intensive NLP

Thank You