



Cloud Computing as an Enabler for Generative AI

Martin Lavoie, FSA

Marcela Granados, FCAS, CSPA

Agenda

1 The Cloud

2 Generative AI

3 How to get started

Why Now: Cloud & GenAI?

1



Large Datasets

- Availability of large and diverse datasets
- AI models learn patterns, correlations, and characteristics of large datasets
- Pre-trained state-of-the-art models

2



Computational Power

- Advancements in hardware; GPUs
- Access to cloud computing
- Open-source software, Hugging Face

3



Innovative DL Models

- Generative Adversarial Networks (GANs)
- Transformers Architecture
- Reinforcement learning from human feedback (RLHF)

Cloud : What it is and its Advantages

What it is

Cloud computing is the delivery of computing services (e.g. servers, storage, databases, software and analytics) over the internet (“the cloud”)*

*source: <https://azure.microsoft.com>



Advantages

- **Scalability:** Allows insurers to expand/reduce resources based on demand
- **Cost Efficiency:** Pay-as-you model □ Avoids high initial costs for:
Setting up & maintaining on-prem infrastructures
- **Flexibility & Agility:** Experiment with new Analytical tools & Tech.
Fostering innovation & rapid development
- **Disaster recovery & Business Continuity:** Insurers can store backup copies of their data in multiple geographically diverse locations

Cloud : Disadvantages and Challenges

Disadvantages

- **Internet dependency:** Need reliable internet connection
If internet is down → No access to data and software
Downloading is an option but may insure cost. Making copies is not ideal
- **Security and privacy vulnerability:**
Companies are cautious → Data becomes accessible via the internet
Privacy concern → Data exist with someone else
Security sophistication → Still vulnerable to hacking and malicious attacks
However, “On-Premise” doesn’t mean less risk

Challenges

- **Data volume & complexity:**
Ensure efficient data transfer, storage and accessibility
- **Data governance:** Data ownership, access control and usage policies to prevent data misuse, security breaches and compliance
- **Data security:** Industry standards and legal requirements during and after migration → maintain data integrity & confidentiality

Demystifying Generative AI

Artificial Intelligence (AI)

Generative AI

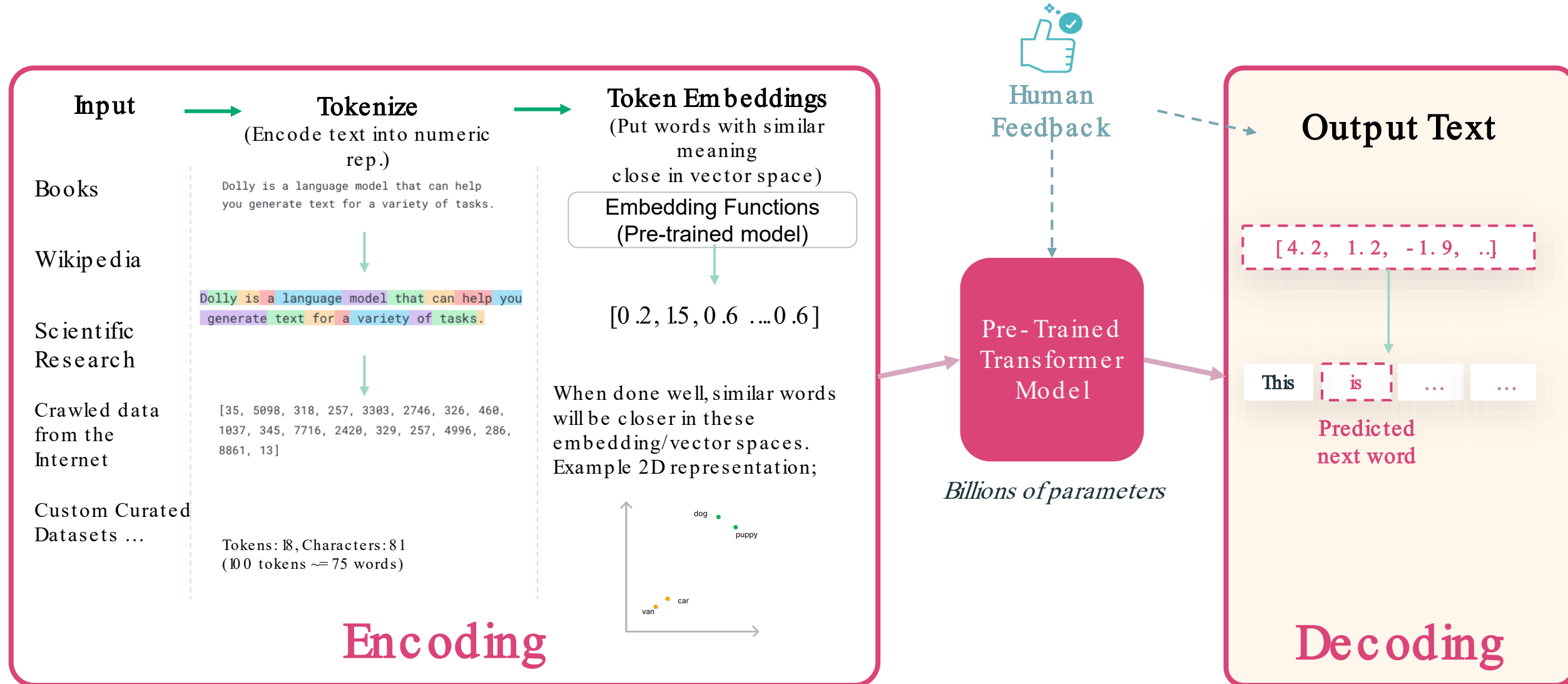
Foundation Models
(GPT-4, BART, MPT-7B etc.)

Large Language Models (LLMs)

- **Generative AI:** Focus on generating content
- **Foundation Models:** Transformers models that are pre-train. Cover all unstructured data
- **LLMs:** Models trained using billions of parameters but focus on text and code as the input and output

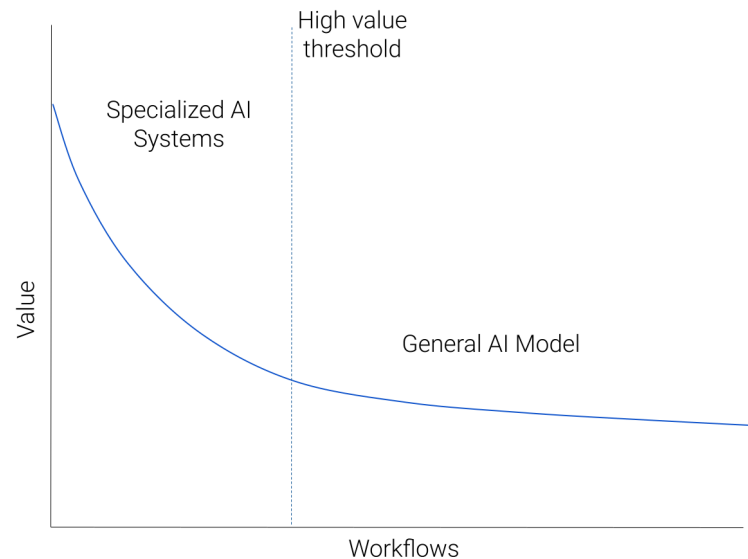
How Do LLMs Work?

A simplified version of LLM training process



Generative AI Approach Considerations

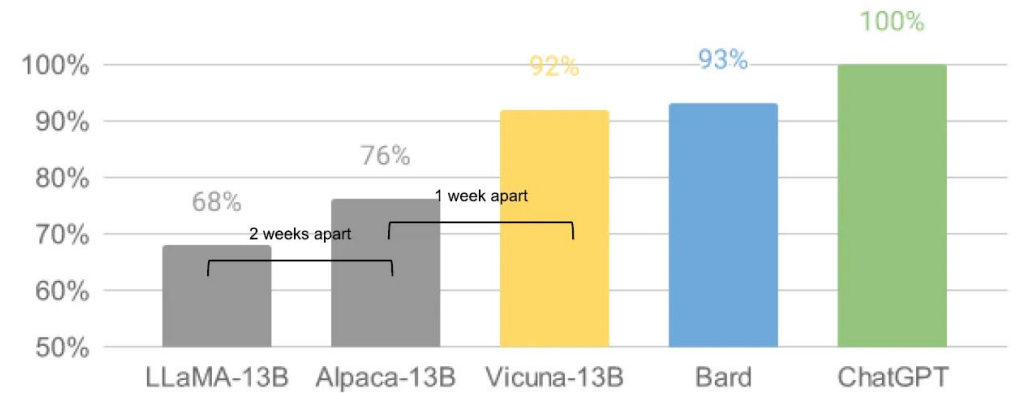
Small/Specific vs. General



“Well defined, high- value workflows will primarily be addressed by specialized AI systems not general purpose AI models.”

—Narthur Raghur (Samaya AI), Matej Zeman (Databricks), Eric Schmidt (Schmidt Futures)

Open Source vs. Proprietary



*GPT-4 grades LLM outputs. Source: <https://vicuna.lmsys.org/>

Open-source models are faster, more customizable, more private, and pound-for-pound more capable.”

—Leaked Google paper

Generative AI (GenAI) Maturity

M
A
T
U
R
I
T
Y

Foundational model as
SaaS with Prompts

- Ready to be plugged to your applications
- You don't control data inside the model knowledge base
- You cannot add your own data

Retrieval Augmented
Generation
(RAG)

- Augment knowledge of a GenAI model with your own data
- You need to find "your best prompt"
- You can add filters to prompts

Fine-tune foundational
model on your data

- You can update certain "parts" of the model with a few labels
- Requires computational resources and technical skills

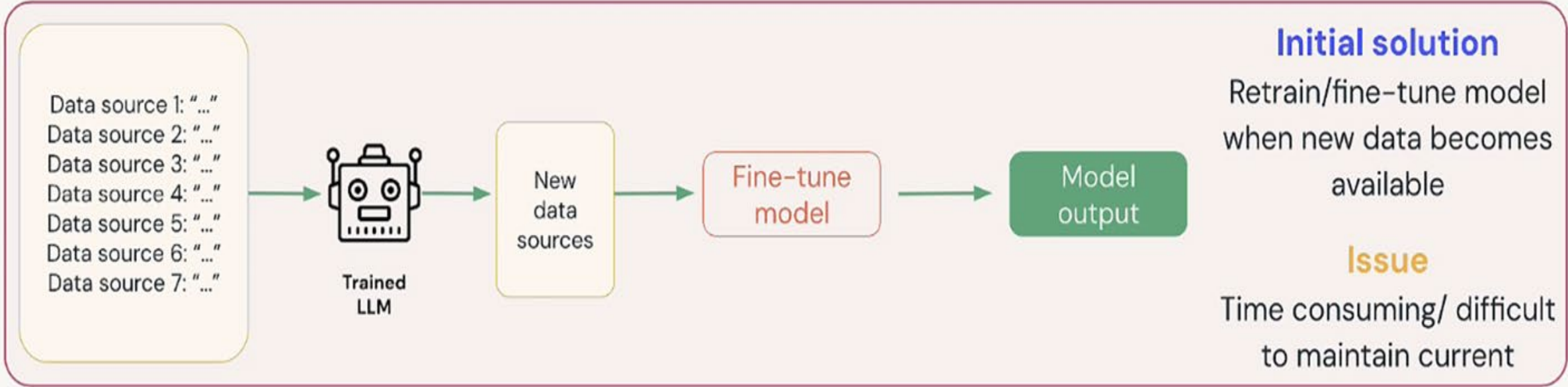
Fully retrain foundational
models (your own
"GPT")

- Owner of your mini GPT style
- Requires a lot of data or labels
- Requires resources both technical and computational

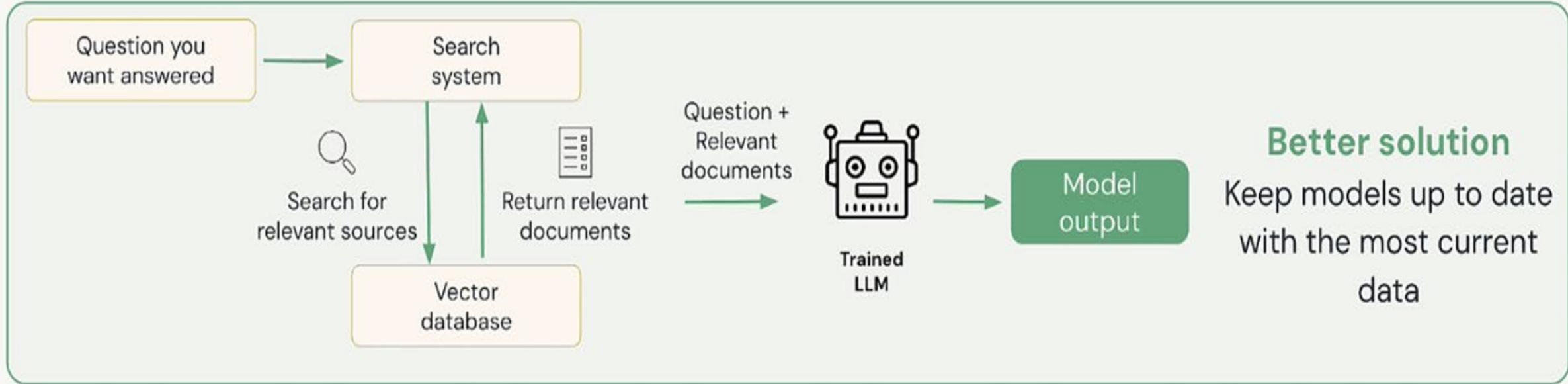


How RAG Reduces Hallucinations?

✗



✓



Why build your own model?

- Quality ————— Ensure accuracy and reliability
- Control ————— Own your own IP to differentiate
- Privacy ————— Maintain full privacy of your data
- Cost ————— Pay less to train and use your own model

We are seeing 4 Main Applications

How Insurers are thinking about LLMs

Internal Employee & advocacy

Providing employees with LLM powered IT service (user manual, FAQs, debug, for tools in your actuarial ecosystem)

Regulatory compliance

Searching for specific regulatory and compliance documents. This is bidirectional, Where using *Retrieval Augmented Generation* to populate Actuarial Memos, Quarterly review of reserves.

Market intelligence

Key executive summary from complex documents, same as above using *langchain / Vector Database*

Ancient code migration

Code snippets (best practices) or complex migration from legacy languages (SQL to Python) or English to Python (e.g. do a clustering analysis for ICD10 diagnoses for group benefits)



LLM Assistant

- Actuarial Qs on:
 - Portfolio metrics (Duration, asset yields, spreads)
 - Summarize results by business segment (RIS, Group Benefits, Retail Life), Product.
- Uses **context** and **usage** info to return more accurate responses
 - Unity Catalog
 - Popularity
 - Lineage
- **LLM-powered Assistant** integrated into the notebook, file and SQL editors:
 - Create SQL Queries
 - Fix code issues
 - Explain documentation

The screenshot shows an LLM Assistant window titled "Assistant" with a close button (X) and a settings gear icon. The context is "Command 1". The user's message is "Using df, plot the number of trips per day." The assistant's response provides a list of three steps to achieve this goal and includes a code block for implementation.

Assistant:

To plot the number of trips per day, you could:

1. Convert the pickup date-time column to a date-only column using the pandas dt accessor.
2. Group the data by date and count the number of trips in each group.
3. Plot the resulting counts using any plotting library of your choice.

Here's an example implementation using pandas and matplotlib:

```
python
import matplotlib.pyplot as plt

# Convert datetime column to date-only column
df['date'] = df['pickup_datetime'].dt.date

# Group trips by date and count number of trips in each group
trips_per_day = df.groupby('date').size()

# Plot trips per day
trips_per_day.plot()

# Set plot title and axis labels
```

Enter your message

Cmd 1

```
df = spark.sql('select ...')
(1) Spark Jobs
Command took 2.42 seconds -- b
michael.piatek@databricks.com'
```

Shift+Enter to run
Shift+Ctrl+Enter to run select

LLMs Business Use Cases

Code generation & developer productivity

- Code completion, boilerplate code generation
- Error detection and debugging
- Convert code between languages
- Write code documentation
- Automated testing
- Natural language to code generation
- Virtual code assistant for learning to code

```
sentiments.ts
1 #!/usr/bin/env ts-node
2
3 import { fetch } from "fetch-h2";
4
5 // Determine whether the sentiment of text is positive
6 // Use a web service
7 async function isPositive(text: string): Promise<boolean> {
8   const response = await fetch(`http://text-processing.com/api/sentiment/`, {
9     method: "POST",
10    body: `text=${text}`,
11    headers: {
12      "Content-Type": "application/x-www-form-urlencoded",
13    },
14  });
15  const json = await response.json();
16  return json.label === "pos";
17 }
```

Copilot

```
max_sum_slice.py
1 def max_sum_slice(xs):
2   if not xs:
3     return 0
4
5   max_ending = max_slice = 0
6   for x in xs:
7     max_ending = max(0, max_ending + x)
8     max_slice = max(max_slice, max_ending)
9   return max_slice
```

Copilot

LLMs Business Use Cases

Content Creation

- Technical writing: Documentation, user manuals, simplifying content etc.
- Where using Retrieval Augmented Generation to populate Actuarial Memos, Quarterly review of reserves.
- Summarize information from the market (S&P) and bring it into stress testing
- Automatically populate Model Risk Management templates

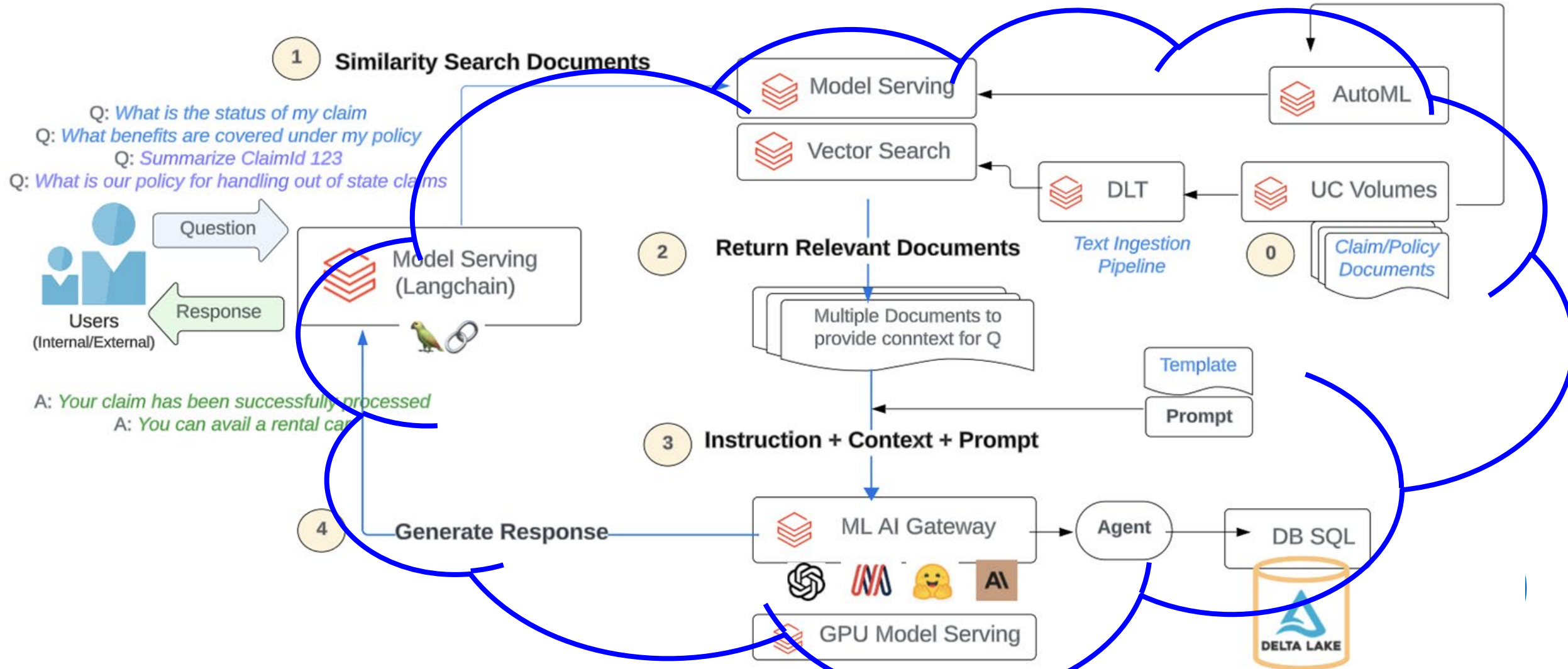
“ChatGPT launches boom in AI-written e-books on Amazon”
REUTERS

“AI is already writing books, websites and online recipes”
The Washington Post

“Will ChatGPT supplant us as writers, thinkers?”
The Harvard Gazette

LLM RAG Architecture

Transform documents into a Knowledge Engine for Q&A



The Cloud enables Generative AI

One platform to support multiple personas



BI & Data
Warehousing



Data Engineering



Data
Streaming



Data
Science & ML

Cloud computing turns the “data volume & complexity” challenge into opportunity

Simplest form of Generative AI (RAG) requires large volume of data (unstructured)

- 10's of thousands of words

All elements of GenAI require efficient cloud compute and cost-effective storage

- Content generation, summarization, data enrichment

Your Data must be AI ready & accessible first!

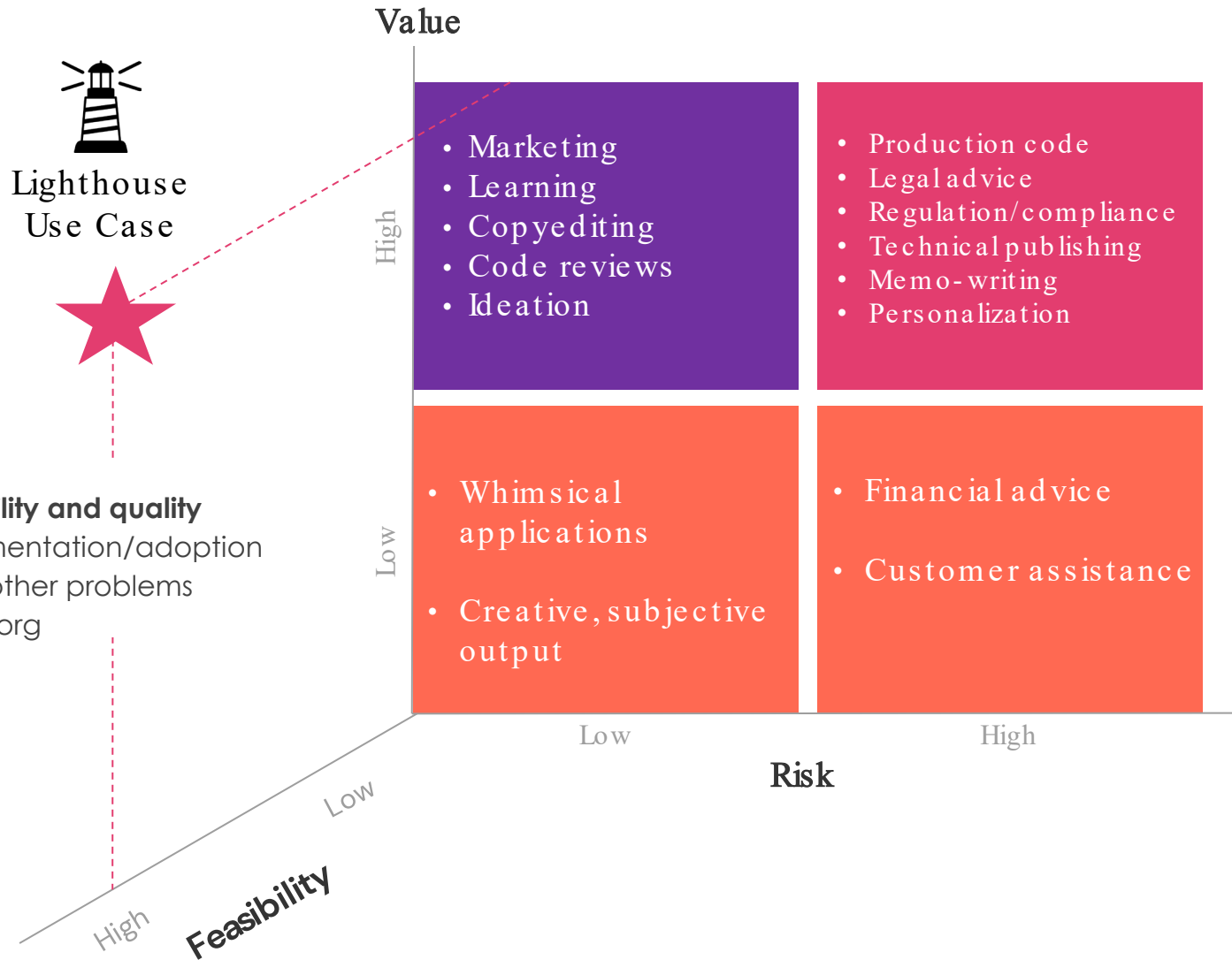


Cloud Lakehouse


All Raw Data
(Logs, Texts, Audio, Video, Images)



Use Case Selection Criteria



- ↪ Data accessibility and quality
- ↪ Ease of implementation/adoption
- ↪ Reusability to other problems
- ↪ Expertise of AI org

 **Lighthouse Use Case Desired Outcomes**

- ✓ Learning
- ✓ Testing
- ✓ Templating
- ✓ Piloting
- ✓ Creating excitement
- ✓ Educating employees

Questions?

Happy to connect



Marcela Granados

Global Insurance Leader at Databricks
| Ex-EY, Ex-AIG, Ex-Liberty Mutual | Organi...



Martin Lavoie

Director and Actuary at MetLife

