ACCELERATING OPERATIONAL EXCELLENCE WITH GENERATIVE AI

Peter Landis & Gen Li
6/10/2024

Speakers

Peter Landis
Principal Engineer

Gen Li
Lead Engineer

Northwestern Mutual

About Northwestern Mutual

Unsurpassed Financial Strength[1]
with total company assets of
$359 billion

Aaa	A++	AAA	AA+
HIGHEST	HIGHEST	HIGHEST	SECOND HIGHEST
Moody's Investors Service	A.M. Best Company	Fitch Ratings	S&P Global Ratings

97%
of policyowners stay year after year[2]

U.S. Independent Broker-Dealer[3]
TOP 5
Measured by 2022 revenue

Wealth Management
$281 billion[4]
retail investment client assets held or managed by Northwestern Mutual

Largest direct provider of individual life insurance in the U.S.[5]

Total clients
5.1+ million

Industry leader in total dividend payout
$7.3 billion[6,7]

Recognized for[8]
"Social Responsibility," "Quality of Management," "Financial Soundness," and "Quality of Products/Services."

1,2,3,4,5,6,7,8: see references at end of presentation

CUSTOMER SERVICE IS EASY

IF THERE AREN'T ANY CUSTOMERS

Business Use-case Process

1 Customer Service Representatives answering customer questions

"What are the benefits from Whole Life One?"

Send question to retrieve context

Send question to generate response

2 Relevant information to answer questions is retrieved from knowledge base

Knowledge Base Articles

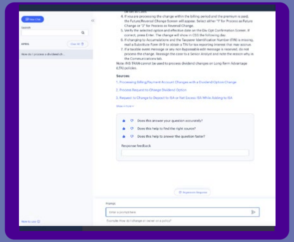
Send context to generate answer

3 LLM summarizes relevant context and generates an answer

Large Language Model

Display answer to Service Representatives

4 Service representatives get answer with citation to the source document



Iteratively improve performance human-in-the-loop

5 SMEs validates answer from GenAI tool by reviewing source documents, and provide feedbacks to improve the tool

Fast iterations

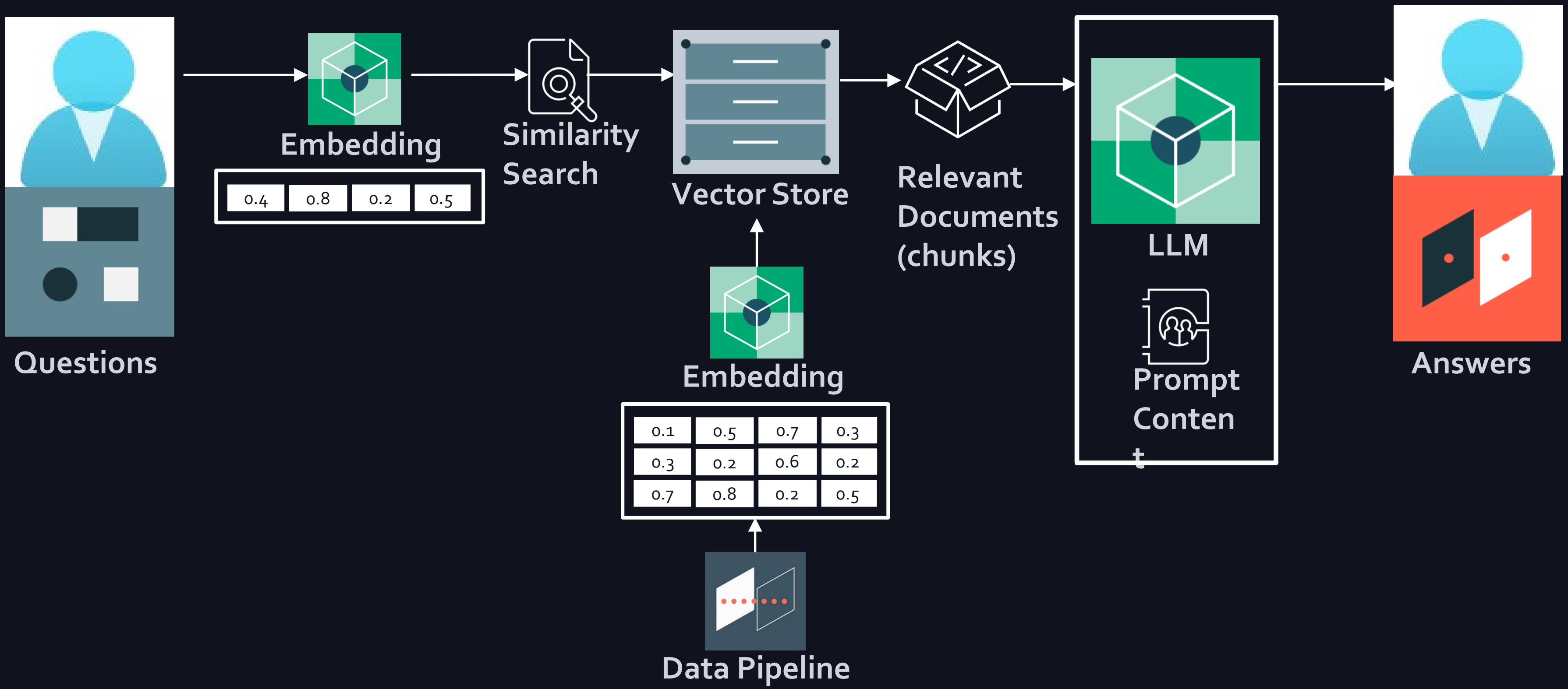
Track performance in live scenarios

6 Improve call efficiency and effectiveness

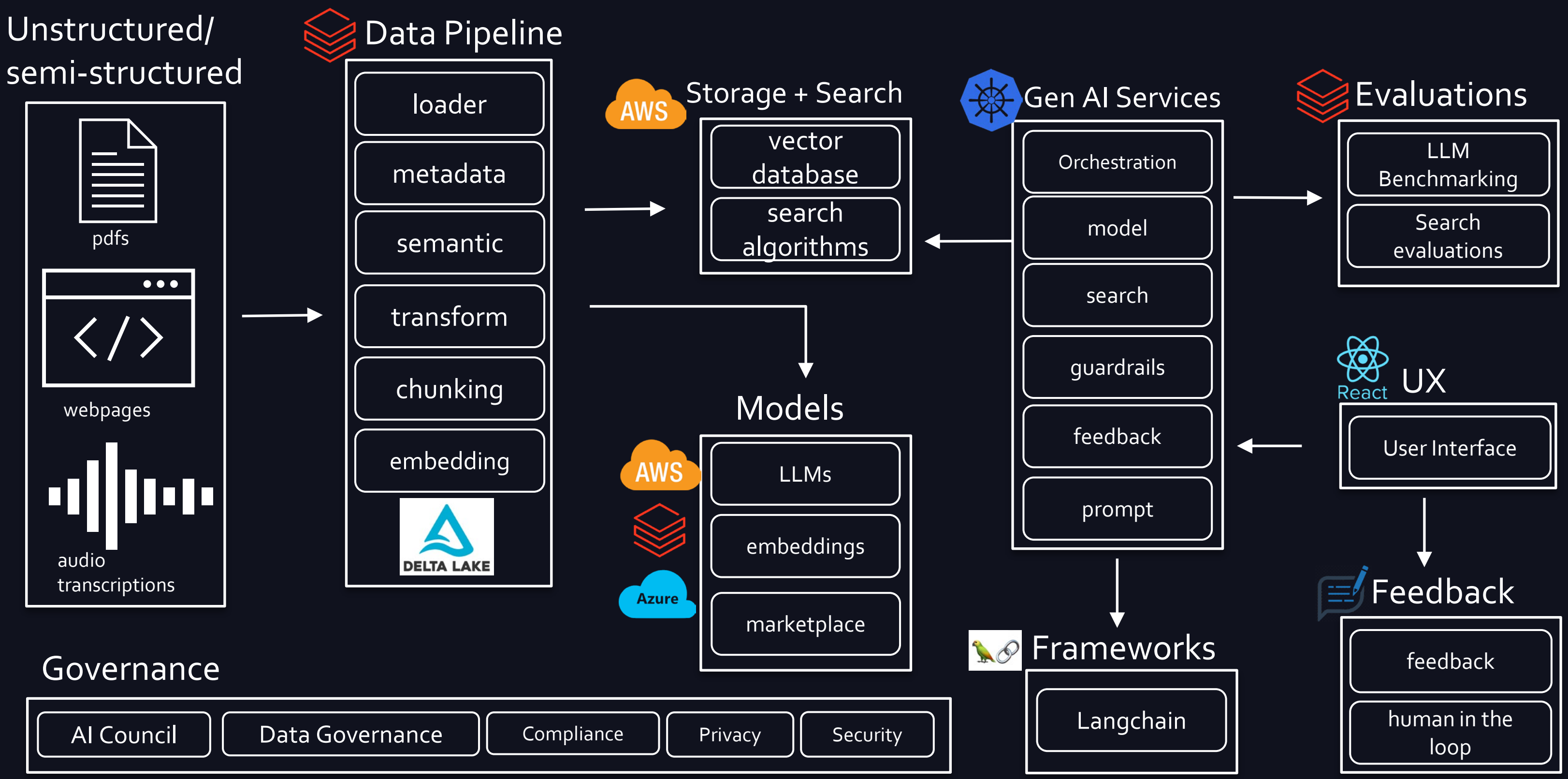
Outcomes:

- ✓ Shorter handle time
- ✓ Higher first call resolution
- ✓ Reduced call length

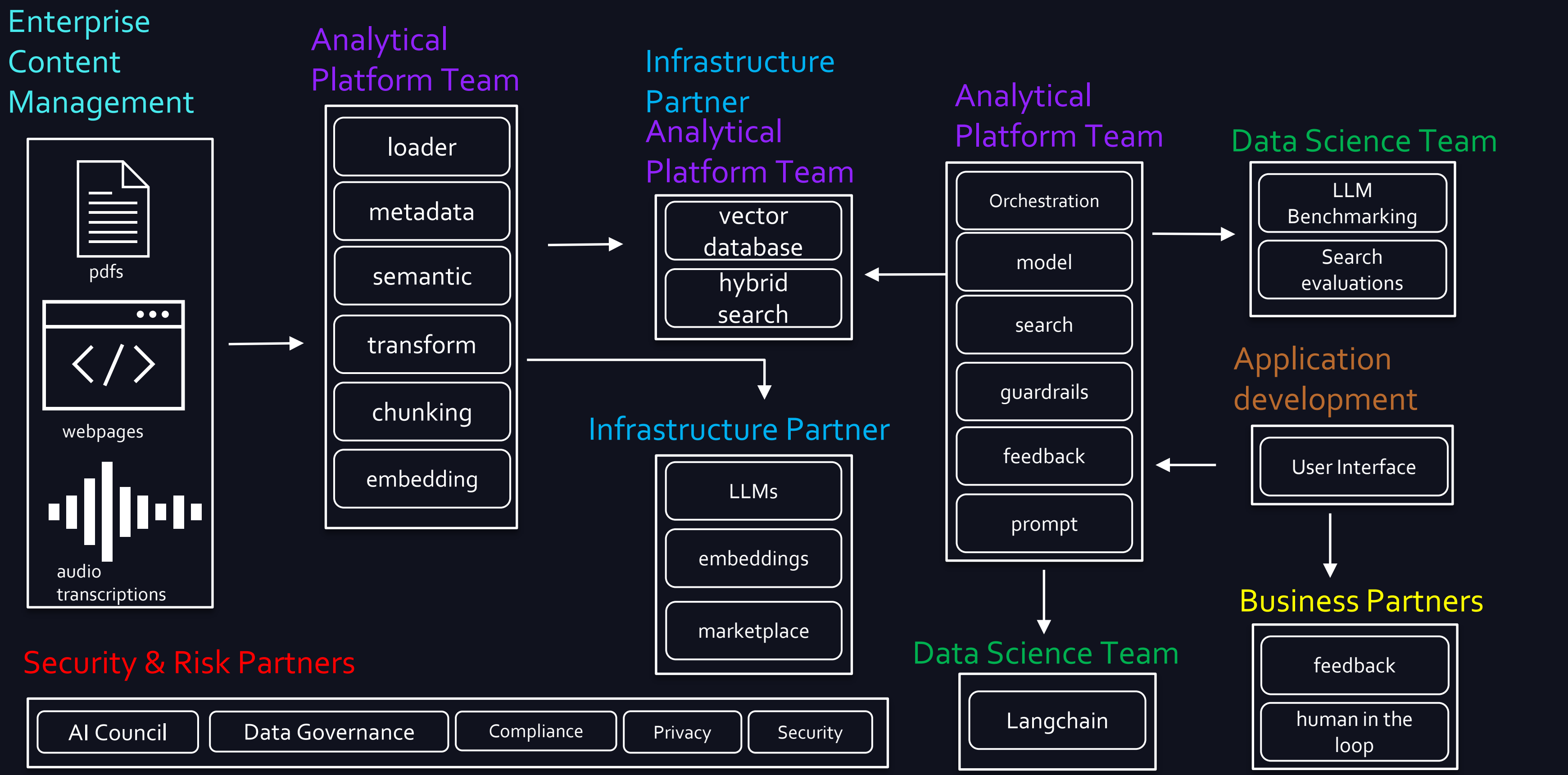
Retrieval Augmented Generation



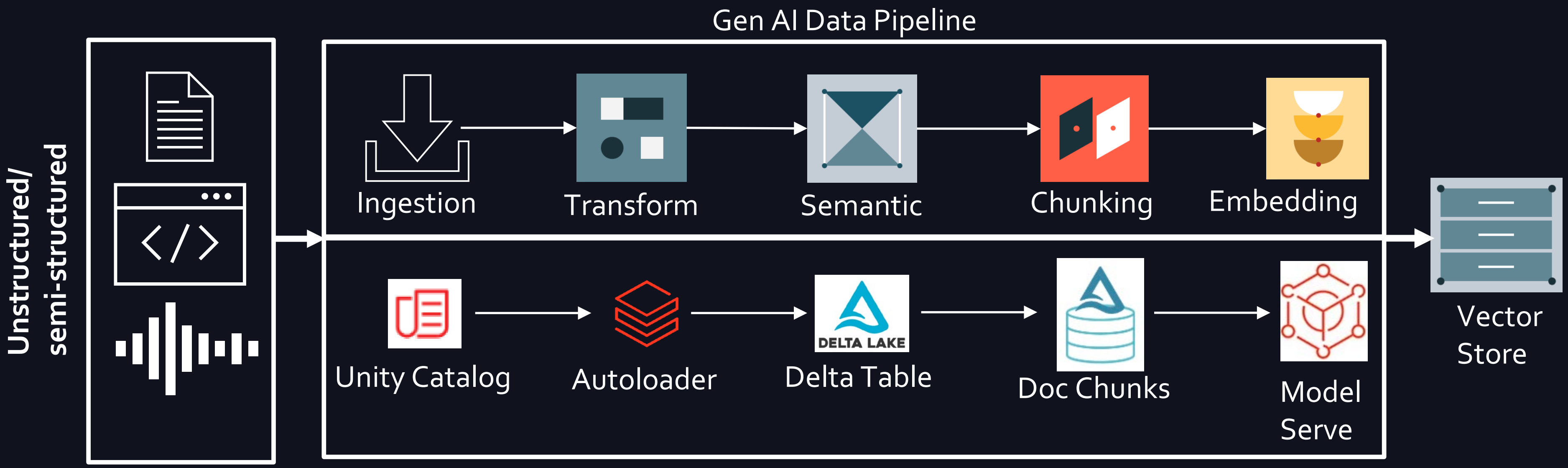
High-level Architecture



Cross Team Collaboration



Data Pipeline



Considerations:

- Unity Catalog Volumes
- Delta: Change Data Capture
- Metadata Enrichment
- Chunking strategies
- Embedding Table

Metadata

Considerations:

- Embeddings for section, headings, title
- Roles, AD groups
- Categories
- Keywords
- System data: last modified dtm, created dtm
- Content type
- Hash of content + metadata
- Data labeling

Vector Database

id	embedding	metadata	...

1	[0.7, 0.2, -0.3, ..., 0.1]	{
		"context_hash": "...",
		"keywords": ["...", "..."],
		"domain": ["...", "..."],
		"ac_group": ["...", "..."],
		"content": "...",
		"parent_doc_id": "...",
		"previous_content_id": "...",
		"next_content_id": "...",
		...
		}
			...

...	...	...	...

Chunking Strategies

Fixed Chunking

- Determine the number of tokens, token overlap, and separator

Recursive Chunking

- Looks at the structure of text and infer chunk sizes using a series of separators (\n\n, \n, " ", ",") based on chunk size

Document Specific Chunking

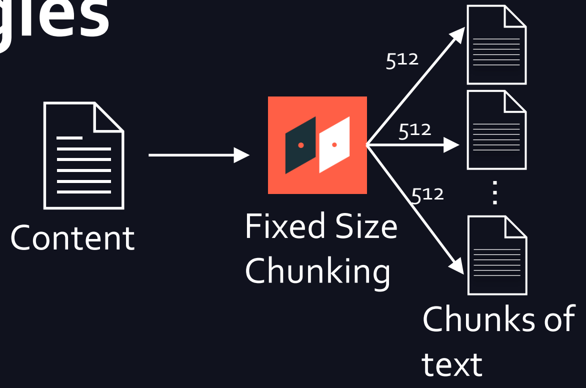
- Chunk based on the structure of the document. Markdown, PythonCode, XML, Documents with tables and images.

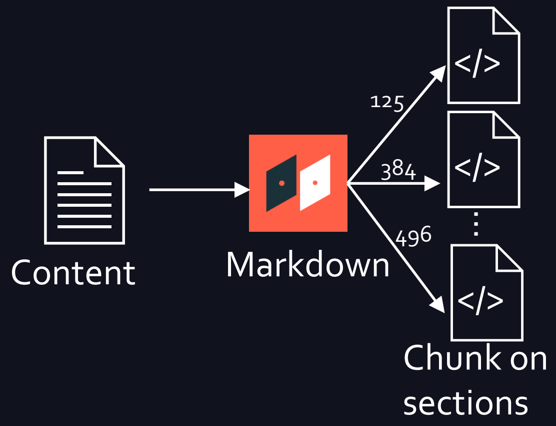
Semantic Chunking:

- Chunk based on the relationship within the text dividing the text into meaningful, semantically complete chunks

Agentic Chunking:

- Chunks into paragraphs extracting propositions from each paragraph using a LLM and then summarizing it and relating each proposition together to form a chunk.

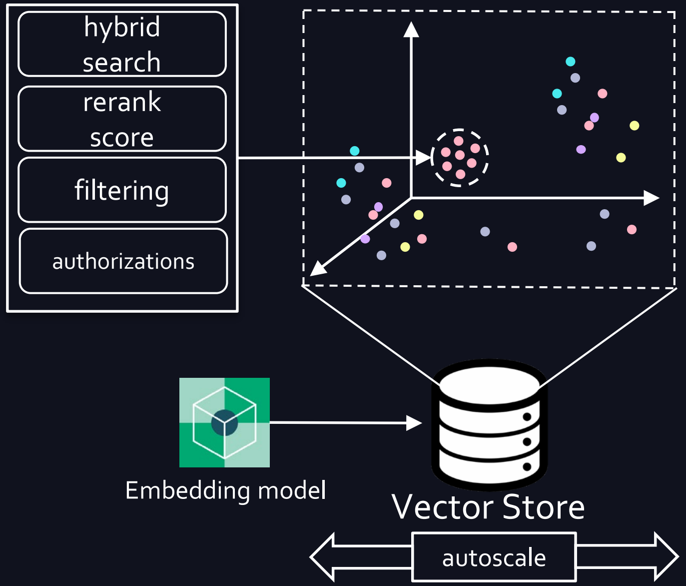




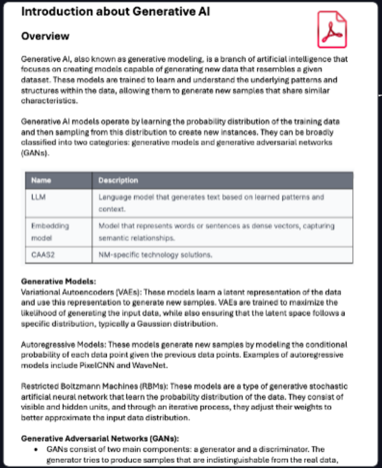
Storage and Search – Vector DB

Considerations:

- Hybrid Search and Algorithms
- Filtering Capabilities: Pre, Post, Inline
- Performance Scaling
- Index Algorithms (HNSW, IVFFlat, PGA, ..)
- Serverless
- Security
- Varity vs Volume
- Multi Model Support
- Embedding Models
- SDK/Rest API vs Frameworks



PDF use case example



Introduction about Generative AI

Overview

Generative AI, also known as generative modeling, is a branch of artificial intelligence that focuses on creating models capable of generating new data that resembles a given dataset. These models are trained to learn and understand the underlying patterns and structures within the data, allowing them to generate new samples that share similar characteristics.

Generative AI models operate by learning the probability distribution of the training data and then sampling from this distribution to create new instances. They can be broadly classified into two categories: generative models and generative adversarial networks (GANs).

Name	Description
LLM	Language model that generates text based on learned patterns and context.
Embedding model	Model that represents words or sentences as dense vectors, capturing semantic relationships.
CAAS2	NM-specific technology solutions.

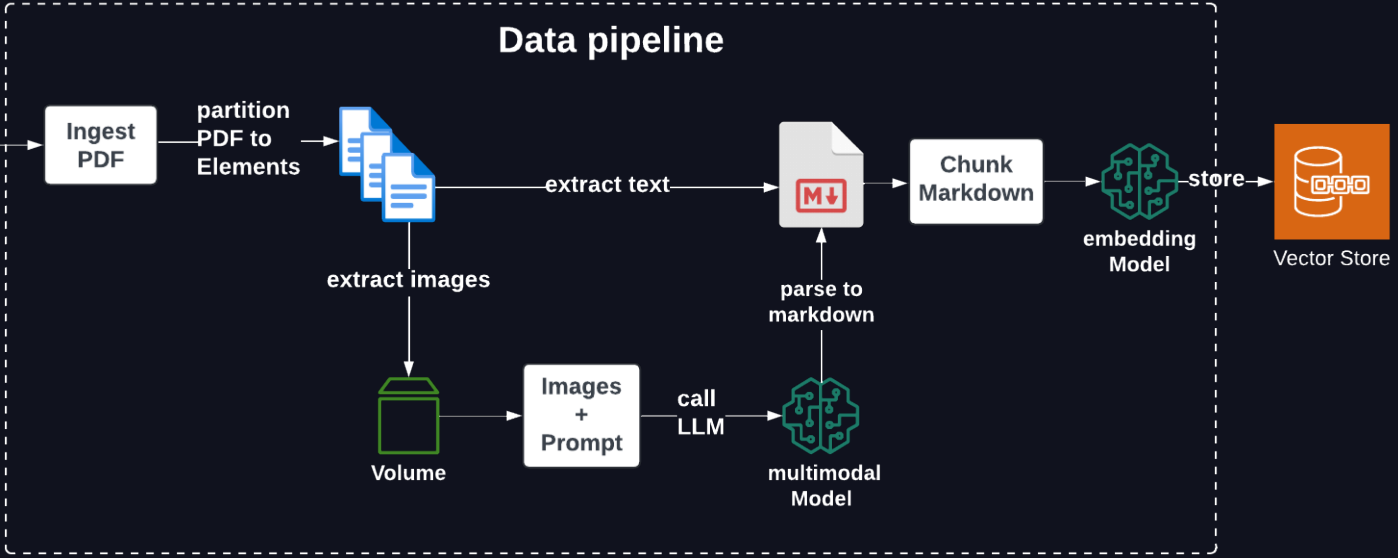
Generative Models:
Variational Autoencoders (VAEs): These models learn a latent representation of the data and use this representation to generate new samples. VAEs are trained to maximize the likelihood of generating the input data, while also ensuring that the latent space follows a specific distribution, typically a Gaussian distribution.

Autoregressive Models: These models generate new samples by modeling the conditional probability of each data point given the previous data points. Examples of autoregressive models include PixelCNN and WaveNet.

Restricted Boltzmann Machines (RBMs): These models are a type of generative stochastic artificial neural network that learn the probability distribution of the data. They consist of visible and hidden units, and through an iterative process, they adjust their weights to better approximate the input data distribution.

Generative Adversarial Networks (GANs):

- GANs consist of two main components: a generator and a discriminator. The generator tries to produce samples that are indistinguishable from the real data,



Considerations:

- Unstructured, LlamaIndex, or LangChain
- Multi-Modal LLM converting images to text

PDF parse example

1. Unstructured library to parse PDF file to elements

PYTHON

```python
from unstructured.partition.pdf import partition_pdf


filename = "./data/files/genai.pdf"
elements = partition_pdf(filename=filename,
      strategy='hi_res',
      extract_images_in_pdf=True,
      extract_image_block_output_dir = "./data/images",
)
```


```json
{
  "type": "Image",
  "element_id": "33abd0ffdaf1950da675d75705dcd117",
  "text": "Description Language model that generates text based on learned patterns and context. Embedding Model that represents words or sentences as dense vectors, capturing model semantic relationships. |caas2 NM-specific technology solutions.",
  "metadata": {
    "coordinates": {
      "points": [
        [200.0, 844.2599999999999],
        [200.0, 1222.4544444444443],
        [1500.0, 1222.4544444444443],
        [1500.0, 844.2599999999999]
      ],
      "system": "PixelSpace",
      "layout_width": 1700,
      "layout_height": 2200
    },
    "last_modified": "2024-05-24T17:27:01",
    "filetype": "application/pdf",
    "languages": ["eng"],
    "page_number": 1,
    "image_path": "./data/images/figure-1-1.jpg",
    "file_directory": "./data/files",
    "filename": "genai.pdf"
  }
}
```

PDF parse example

2. Create functions to accept image and parse to markdown format

PYTHON (convert image to base64)	PYTHON (convert image to markdown via LLM)

PYTHON (convert image to base64)

```python
def get_image_base64(image_path: str):
   with open(image_path, "rb") as image_file:
      image_bytes = image_file.read()
   return base64.b64encode(image_bytes).decode("utf-8")
```

PYTHON (convert image to markdown via LLM)

```python
def invoke_claude_3_with_image(image_base64: str, prompt: str, profile_name: str, model_id: str =
        "anthropic.claude-3-sonnet-20240229-v1:0",anthropic_version: str = "bedrock-2023-05-
        31",max_tokens: int = 5000):

  boto3.setup_default_session(profile_name=profile_name)
  client = boto3.client(service_name="bedrock-runtime",region_name="us-east-1")
  response = client.invoke_model(modelId=model_id,body=json.dumps(
    {"anthropic_version": anthropic_version,
          "max_tokens": max_tokens,
     "messages": [
        {"role": "user", "content": [
        {"type": "image", "source": {"type": "base64", "media_type": "image/jpeg", "data":
       image_base64}},
        {"type": "text", "text": prompt}
       ]}
      ]}
  ))
  result = json.loads(response.get("body").read())
  return result
```

PDF parse example

3. Get all elements converted and write to markdown

PYTHON

```python
prompt = "Look carefully at the image, convert the image to a table with markdown format."
profile_name = "YOUR_PROFILE"

output = []
for e in elements:
  items = e.to_dict()
  if items['type'] != 'Image':
    if items['type'] == 'Title':
      output.append("## " + items['text'])
    else:
      output.append(items['text'])
  else:
    image_path = items['metadata']['image_path']
    image_base64 = get_image_base64(image_path)
    response = invoke_claude_3_with_image(image_base64, prompt, profile_name)
    output.append(response['content'][0]['text'])

markdown_content = "\n\n".join(output)
file_path = 'data/output/output.md'
with open(file_path, 'w') as file:
  file.write(markdown_content)
```

Introduction about Generative AI

Overview

Generative AI, also known as generative modeling, is a branch of artificial intelligence that focuses on creating models capable of generating new data that resembles a given dataset. These models are trained to learn and understand the underlying patterns and structures within the data, allowing them to generate new samples that share similar characteristics.

Generative AI models operate by learning the probability distribution of the training data and then sampling from this distribution to create new instances. They can be broadly classified into two categories: generative models and generative adversarial networks (GANs).

Name	Description
LLM	Language model that generates text based on learned patterns and context.
Embedding model	Model that represents words or sentences as dense vectors, capturing semantic relationships.
CAAS2	NM-specific technology solutions.

PDF parse example

4. Chunking and Retrieval

PYTHON (convert to nodes/chunks)

```python
from llama_index.readers.file import MarkdownReader
from llama_index.core import VectorStoreIndex, SimpleDirectoryReader


parser = MarkdownReader()
file_extractor = {".md": parser}
documents = SimpleDirectoryReader("data/output", file_extractor=file_extractor
).load_data()

index = VectorStoreIndex.from_documents(documents)
md_node_parser = MarkdownElementNodeParser(include_metadata=True)
md_nodes = md_node_parser.get_nodes_from_documents(documents=documents)
```

PYTHON (query the chunks)

```python
index = VectorStoreIndex(md_nodes)
query_engine = index.as_query_engine(similarity_top_k=3)
response = query_engine.query("Only use the context you have currently, what is CAAS2?")
```

Response

```
Based on the provided context,
CAAS2 is a technology solution
specific to NM. The exact nature or
function of CAAS2 is unknown from
this text alone.
```

Name	Description
LLM	Language model that generates text based on learned patterns and context.
Embedding model	Model that represents words or sentences as dense vectors, capturing semantic relationships.
CAAS2	NM-specific technology solutions.

Advanced Search – Multi-Query Retrieval

User Question

How does a 529 affect financial aid?

LLM prompt

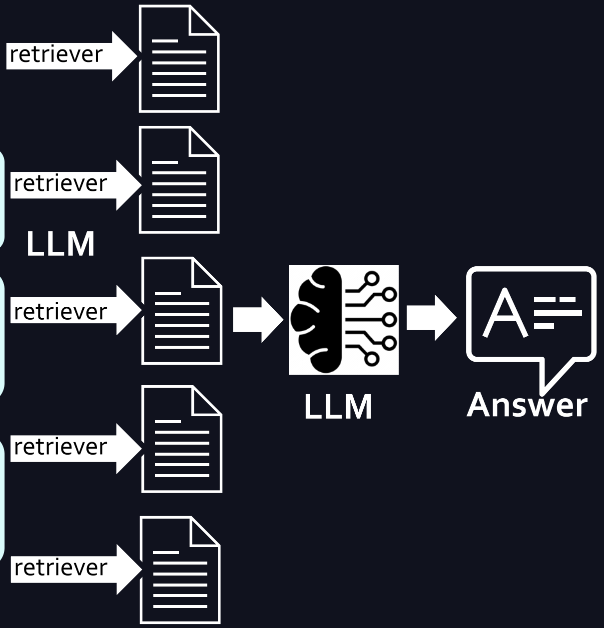
You are an AI language model assistant. Your task is to generate 3 different search queries that aim to answer the user question from multiple perspectives. Each query MUST tackle the question from a different viewpoint, we want to get a variety of RELEVANT search results. Provide these alternatives questions separated by newlines. Original question: {question}

Reframing Question

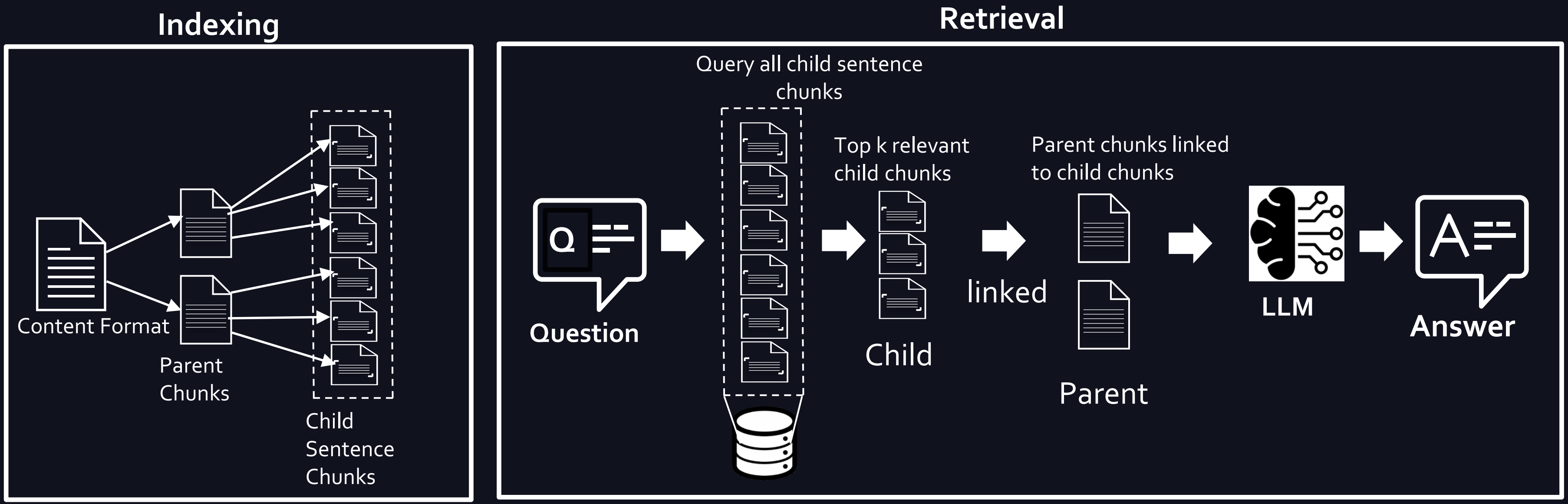
What are the implications of a 529 plan on college financial assistance?

In what ways does a 529 plan impact eligibility for student financial aid?

How does the presence of a 529 account influence the availability of government aid for education?



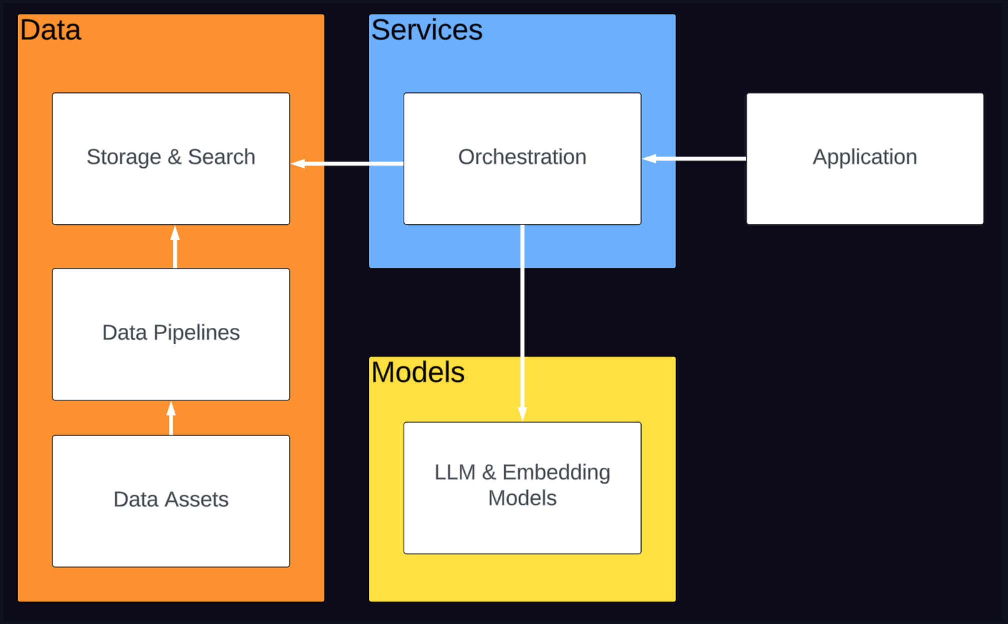
Advanced Search – Small-to-Big Retrieval



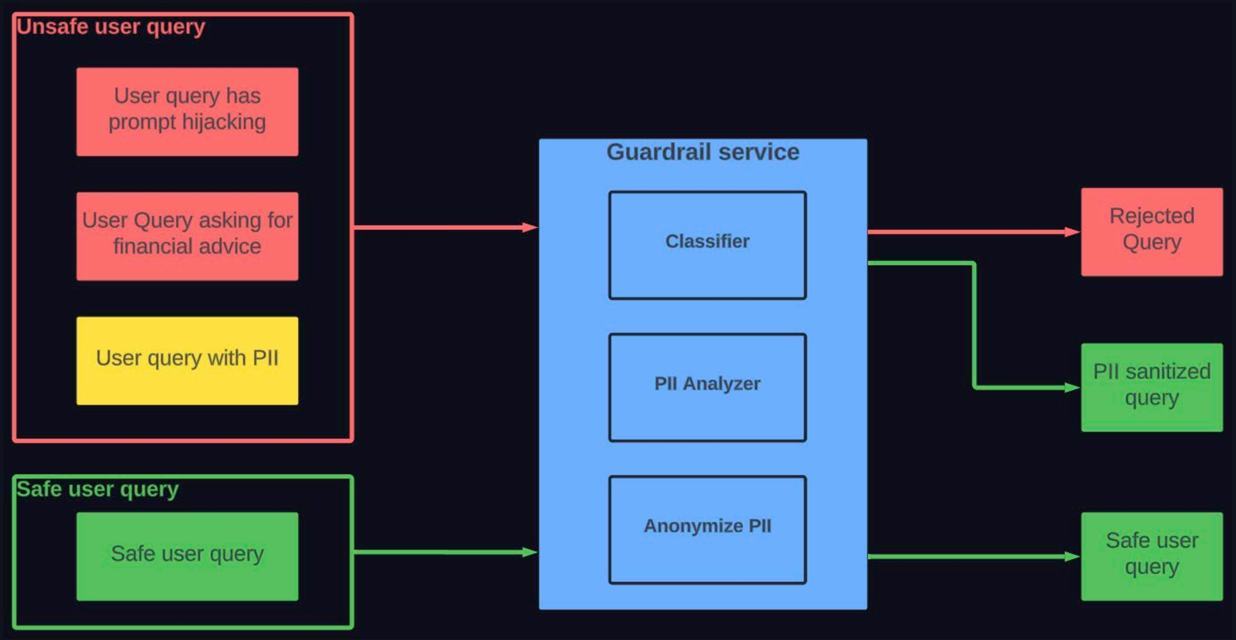
Orchestration

Considerations

- Loosely couple the application to various Gen AI services
- Glue that brings all Gen AI services together which accelerate and standardize implementations
- Offers inversion of control through configuration driven execution
- Provides scalability, extendibility, observability by leveraging standard NM infrastructure and services



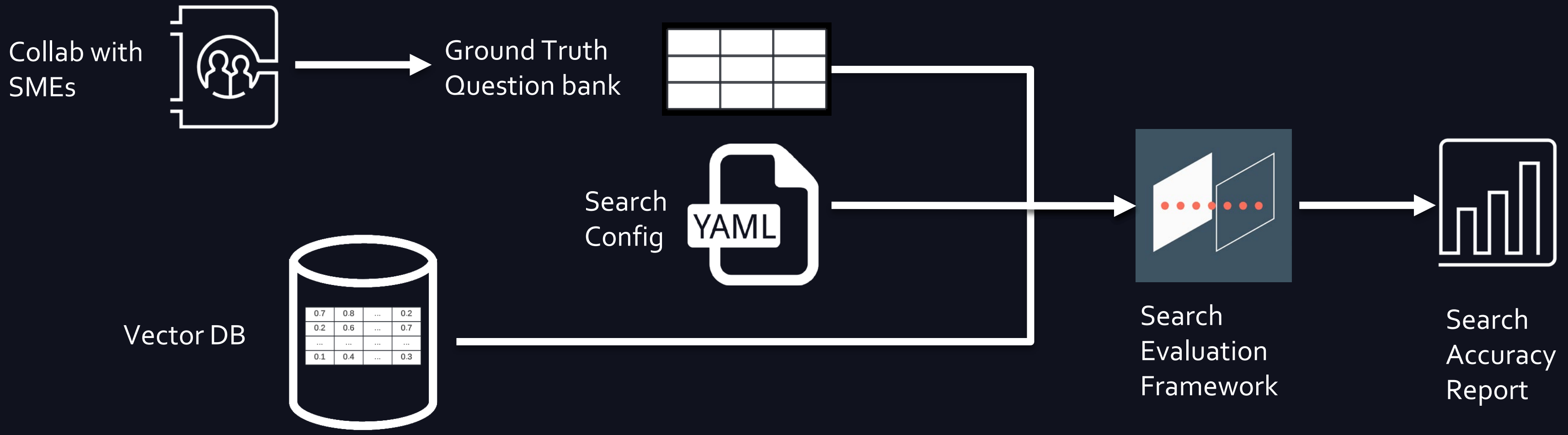
Guardrail Service



Considerations:

- PII (Personally Identifiable Information) guardrail
- Prompt hijacking guardrail
- Advice guardrail

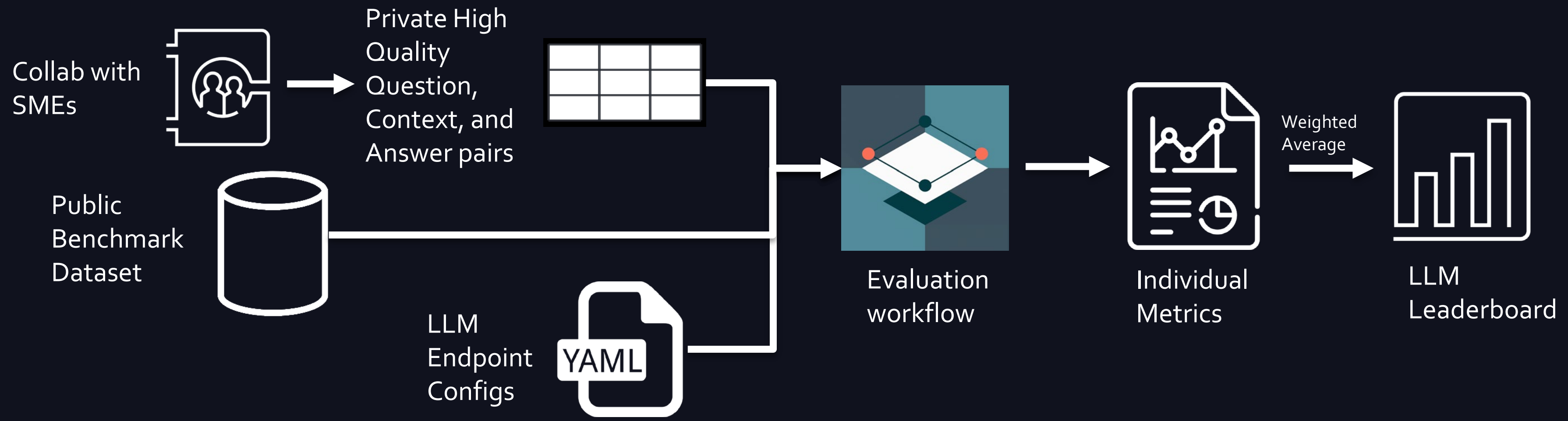
Search Evaluation Framework



Considerations:

- Reusable evaluation for use cases
- Ground truth question bank
- Chunking strategies
- Search parameter tuning
- Integrate into CICD

LLM Benchmark

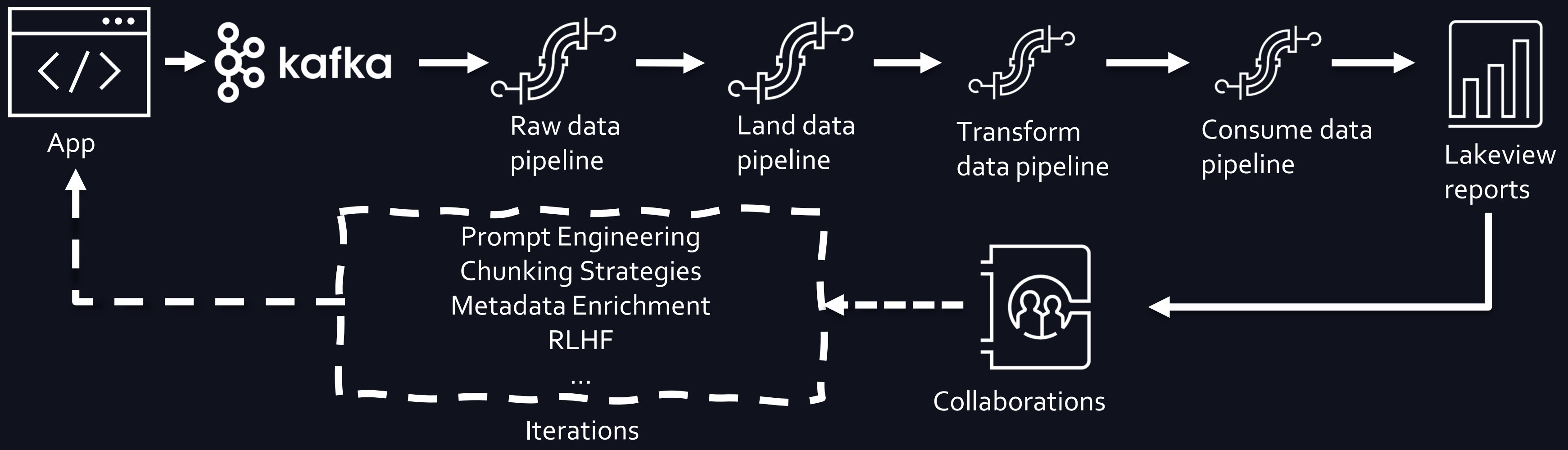


Considerations:

- Performance on both public and private data
- Various metrics to reflect effectiveness of the model output
- Rank based on metrics

Feedback Service

Enable user to comment on application performance.



Considerations:

- Thumbs up/down (relevance, accuracy, and efficiency)
- Free text (any additional inputs)

User Testimonials

"I feel like every time I use the Gen AI Tool it is a positive experience. It is very good about showing the answer right away and it is nice to be able to ask the tool a question."

"I have been able to find everything very quickly. Easy to search!"

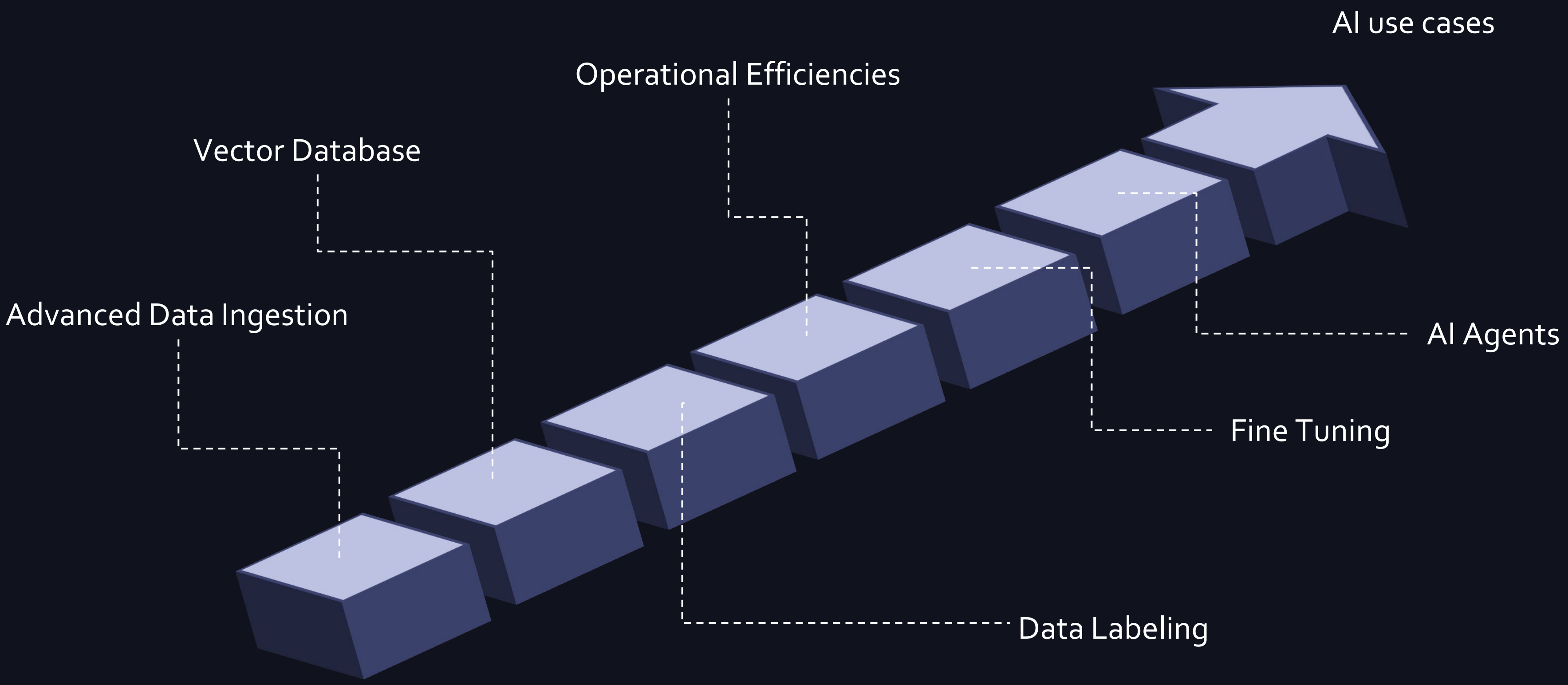
" Love how we can just ask the question without having to enter certain 'key words' to get results."

" The tool made it very easy to find the correct page for a question and I didn't have to put the caller on hold because it pulls up the information so quickly."

"The quickness of the tool is so wonderful; it makes it easier to answer questions for myself without putting it in chat or setting up a case question. This has helped me save me lots of time in case time."

" I have had a positive experience so far, and I have used it to learn more about topics relevant to my role."

Next Steps


AI use cases
Operational Efficiencies
Vector Database
Advanced Data Ingestion
AI Agents
Fine Tuning
Data Labeling

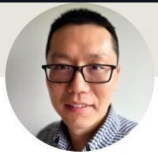
Team Member Acknowledgement

- Zack Harper – Lead Data Scientist
- Nancy Huang – Lead Data Scientist
- Ali Nemati – Sr. Data Scientist
- Anthony Randall - Sr. Director of Data Science
- Anju Gupta – VP of Data Science
- Jeff Parkinson – VP of Core Data Engineering
- Zach Taher – VP of Data Engineering
- Lewie Snyder – Sr. Data Engineer
- Jonathan Tagupa – Sr. Data Architect
- Adam Fine – Sr. Analytics Engineer
- Cindi Reynolds - Principal Cybersecurity Engineer

Contact Information

Peter Landis

Principal Engineer | AI | Generative AI | Machine Learning | Data Engineering | Data Lake House | Data Analytics | Data I...



Gen Li

Generative AI | Machine Learning | Solution Design | Data Engineering

References for Northwestern Mutual Facts

Figures as of or for the year ended December 31, 2023, unless otherwise noted. 1 Among U.S. life insurers. Ratings are for Th e Northwestern Mutual Life Insurance Company and Northwestern Long Term Care Insurance Company, as of the most recent review and report by each rating agency. Ratings as of: 11/23 (Moody's Investors Service), 08/23 (A.M. Best Company), 08/23 (Fitch Ratings), 05/23 (S&P Global Ratings). Ratings are subject to change. 2 Loyalty is based on Northwestern Mutual client data. 3 Ranking for Northwestern Mutual Investment Services, LLC (NMIS) based on total 2022 AUM, which includes figures that combine NMIS brokerage account activity and AUM with account activity and AUM of investment advisory account of NMIS's affiliate Northwestern Mutual Wealth Management Company (NMWMC), which are held through NMIS. Source: InvestmentNews, April 2023. 4 Combined client assets of Northwestern Mutual Investment Services, LLC (NMIS) and Northwestern Mutual Wealth Management Company (NMWMC). The advisory programs offered by NMWMC are in conjunction with brokerage services from NMWMC's affiliate, NMIS. NMIS is a wholly owned subsidiary of Northwestern Mutual. 19-0016 (REV 0124) 5 Latest U.S. rank as of 2022 based on direct premiums written. Source: S&P Capital IQ Pro. Prepared and calculated by Northwestern Mutual. 6 Decisions with respect to the determination and allocation of divisible surplus are left to the discretion and sound business judgment of the company's Board of Trustees. There is no guaranteed specific method or formula for the determination or allocation of divisible surplus. Accordingly, the company's approach is subject to change. Neither the existence nor the amount of a dividend is guaranteed on any policy in any given policy year. 7 Expected 2024 total dividend payout. 8 To determine FORTUNE 2024 World's Most Admired Companies® in more than 50 industries, FORTUNE asked executives, directors, and analysts to rate enterprises in their own industry on nine criteria. Details at fortune.com.

https://thedc.nml.com/globalassets/topic/strategic-communications/northwestern-mutual-fact-sheet.pdf