Fr. Conceicao Rodrigues College of Engineering

RETRIEVAL AUGMENTED GENERATION

Presented by
Rudalph Gonsalves

December 2025

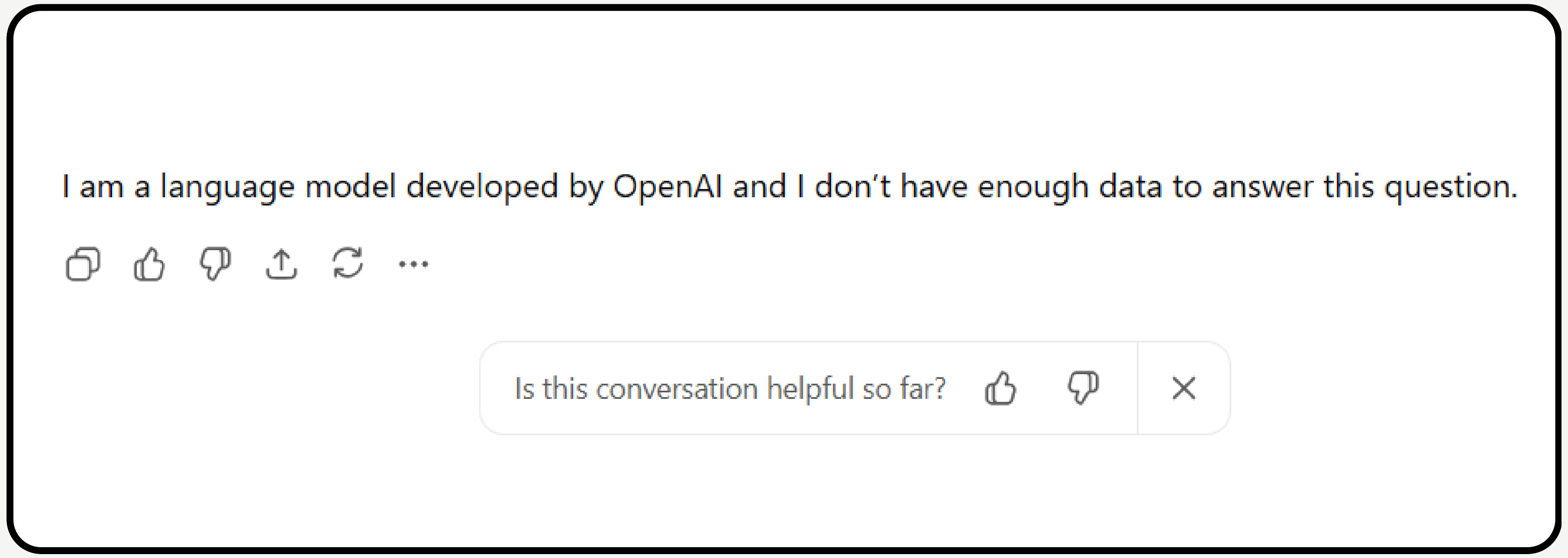
LLM’s Limitations

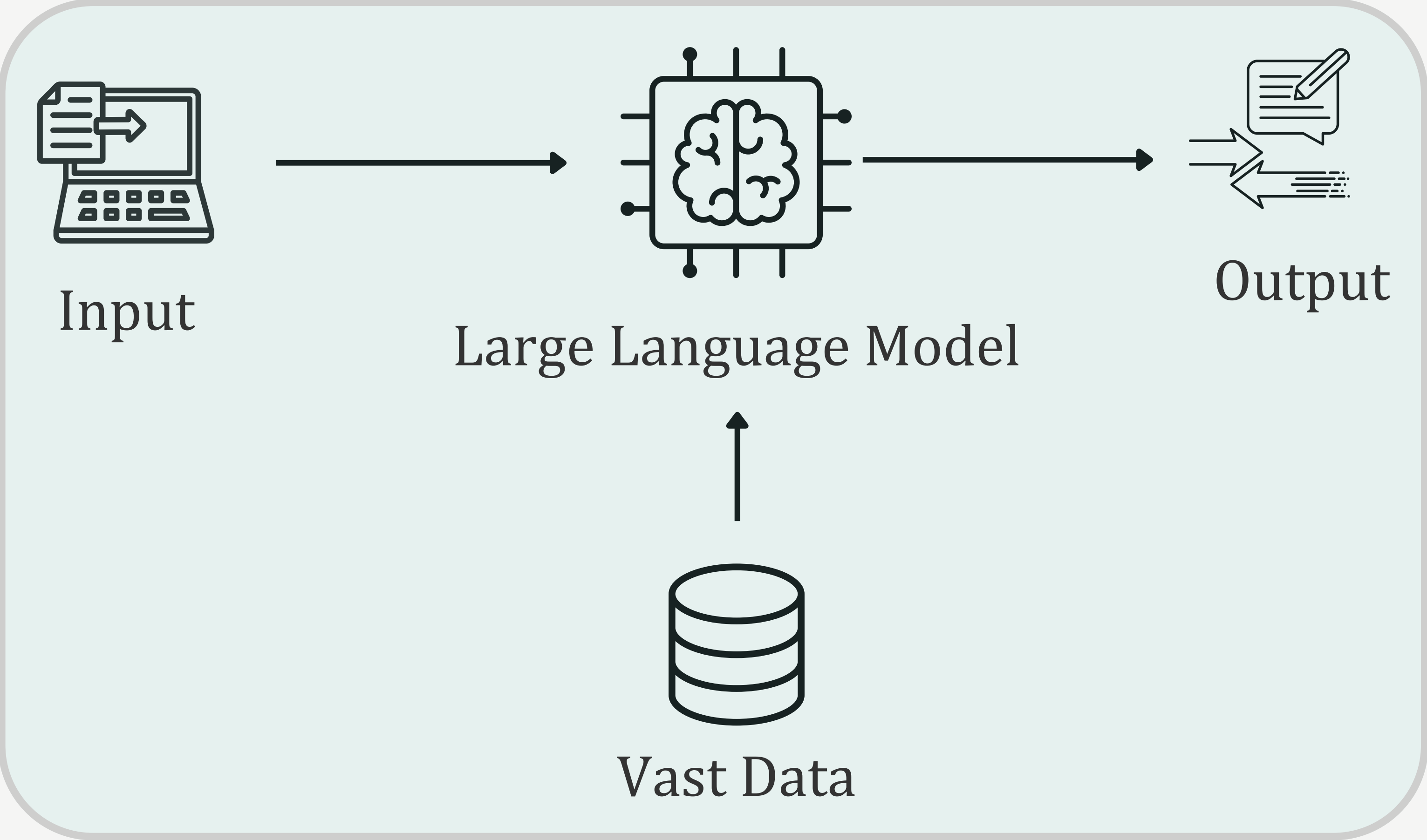
Why Retrieval Augmented Generation (RAG)?

- Large Language Model is trained on vast data
- LLM gets query from client it processes the query and answer based on trained data

What’s the limitation of Large Language Models then?

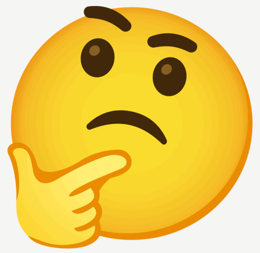
- Lack of ***CONTEXT*** → ***HALLUCINATIONS***
- Outdated knowledge
- Inability to access private data





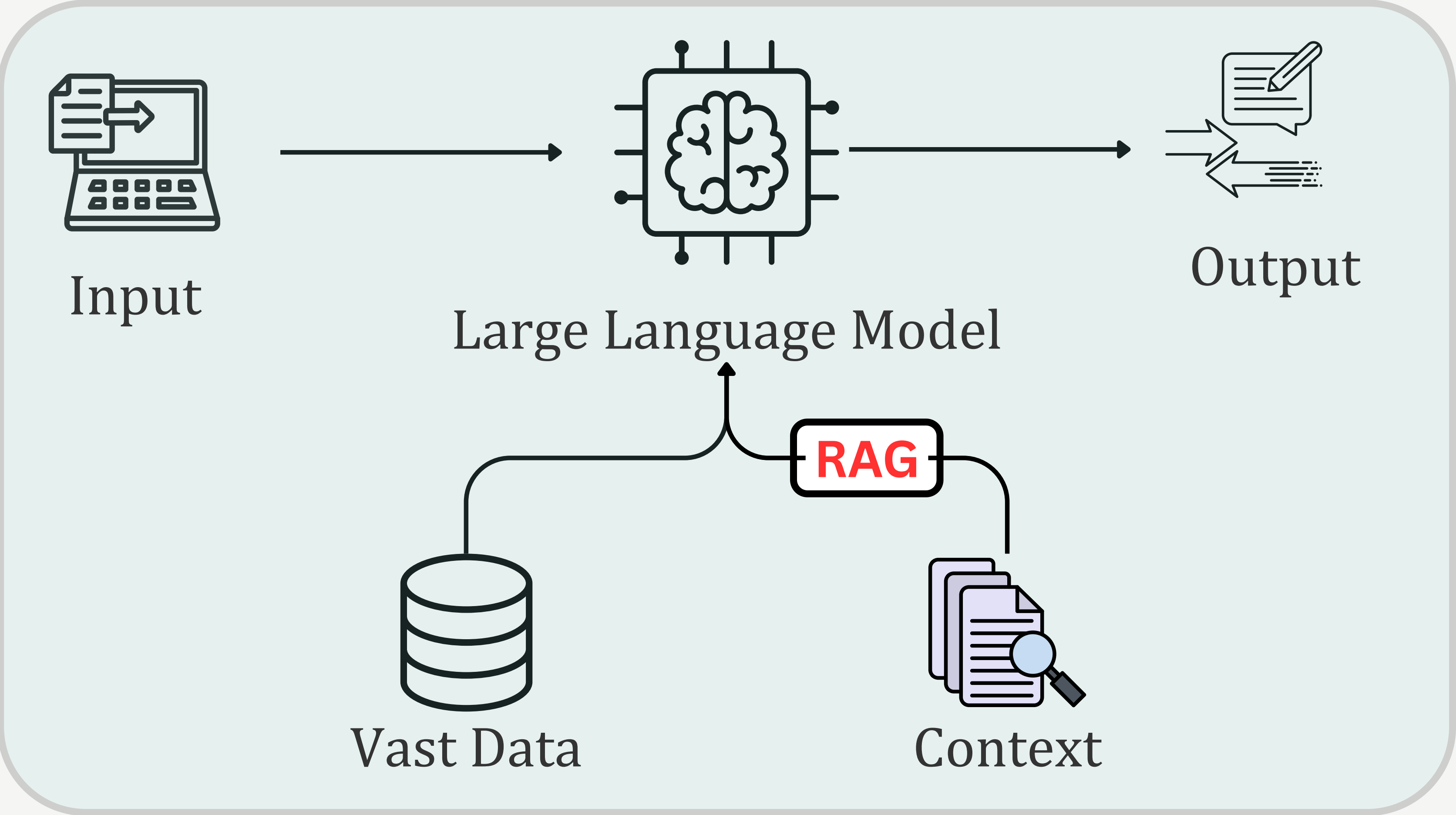
How RAG solves these problems?

CONTEXT



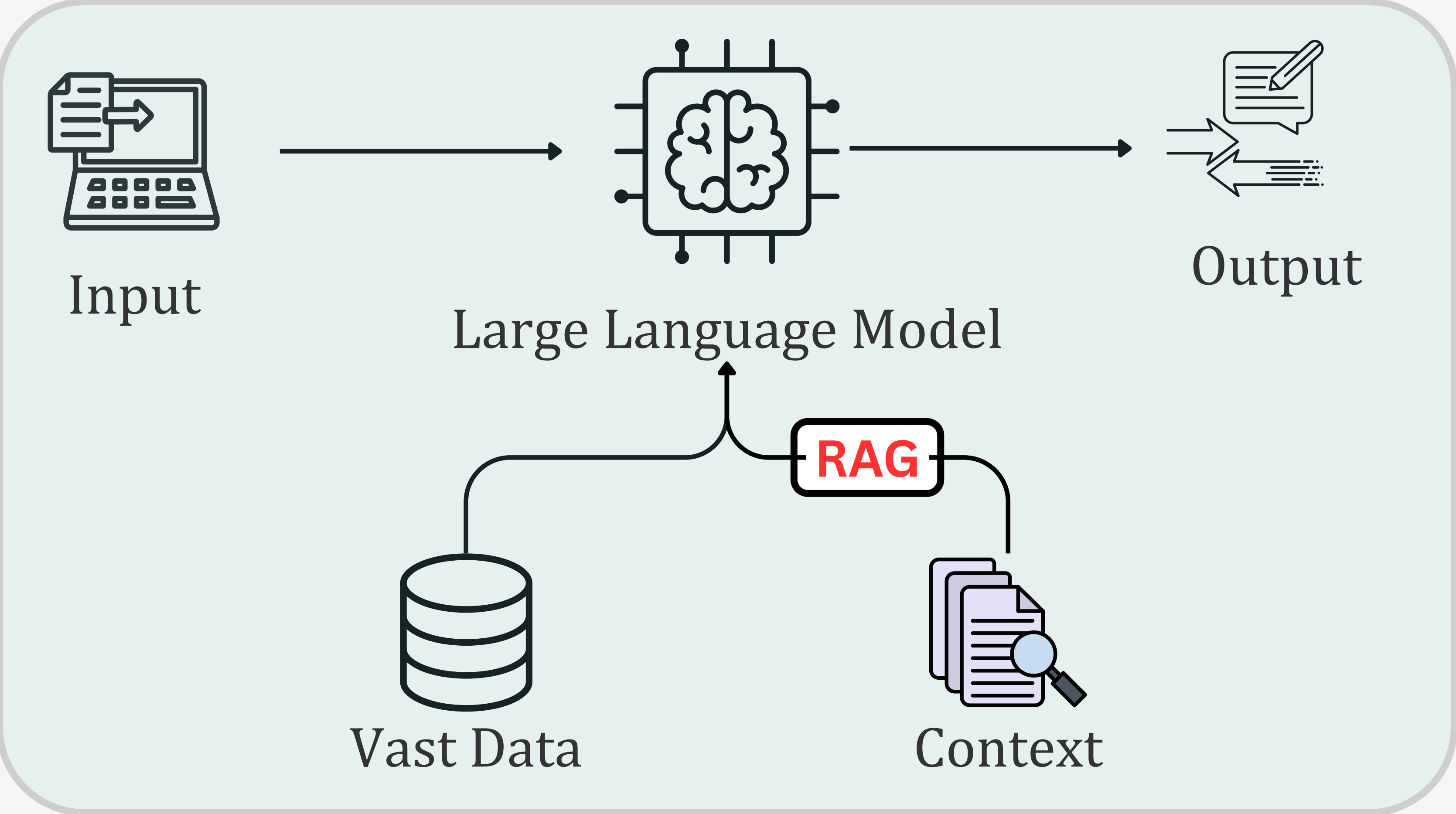
RAG (Retrieval-Augmented Generation) is a technique used to provide relevant external context to a Large Language Model by retrieving information from documents or databases before generating an answer.

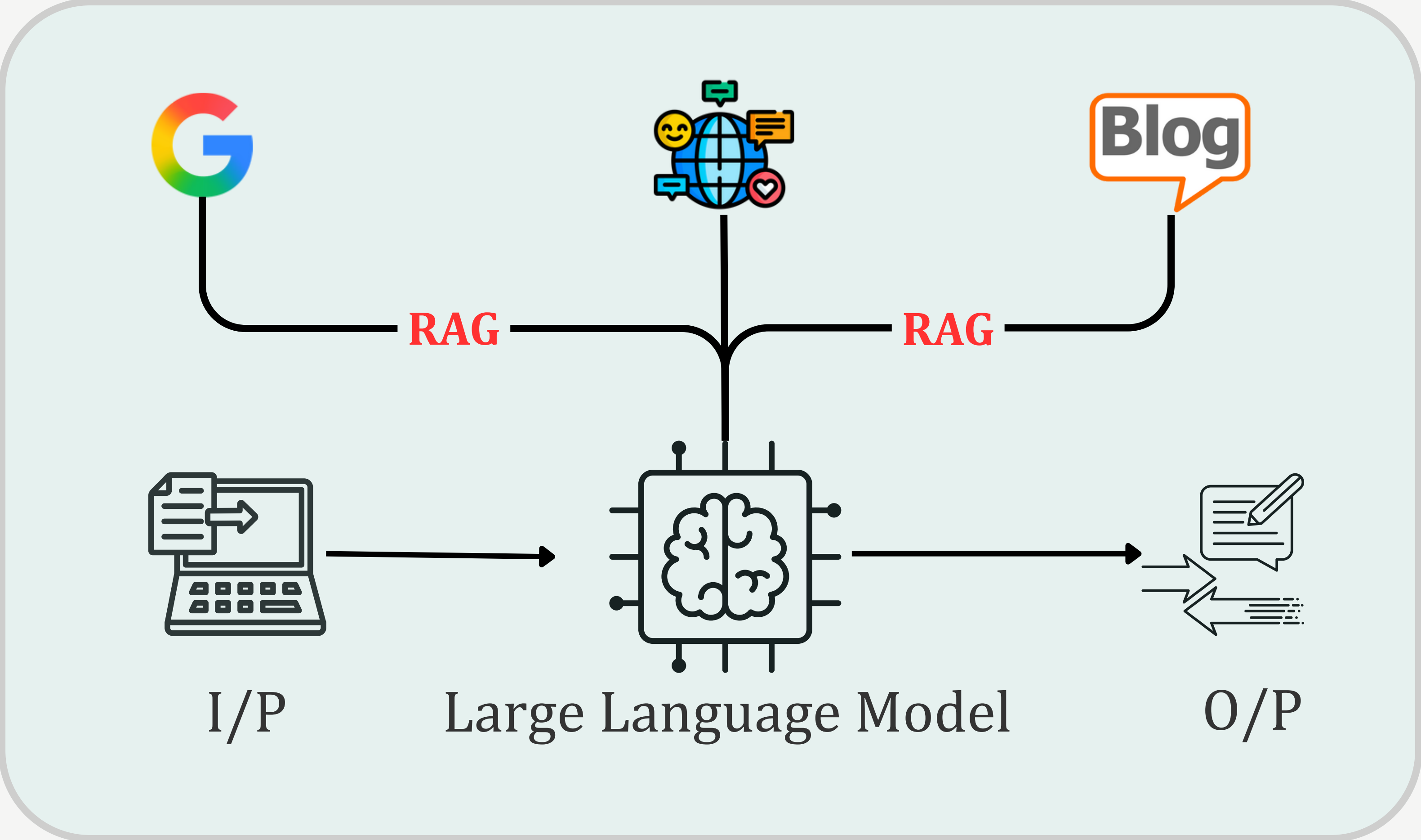
- **Retrieve:** The AI first searches for the most relevant information from documents, websites, or databases.
- **Augment:** It adds that real information to what it already knows.
- **Generate:** Then it creates a more accurate and factual answer using both—the searched info and its own knowledge.



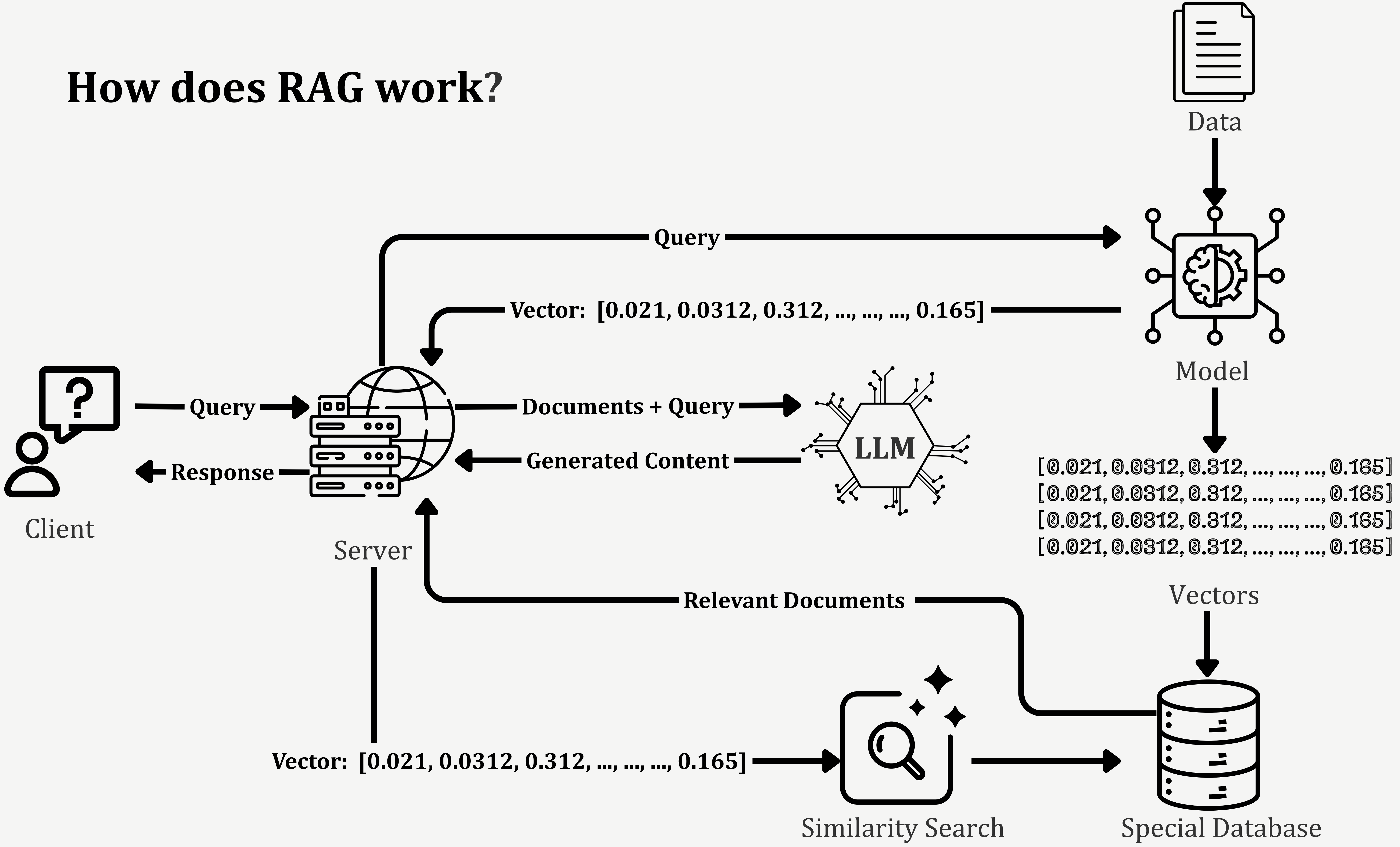
What is Retrieval Augmented Generation (RAG)?

- **Retrieve:** The AI first searches for the most relevant information from documents, websites, or databases.
- **Augment:** It adds that real information to what it already knows.
- **Generate:** Then it creates a more accurate and factual answer using both—the searched info and its own knowledge.

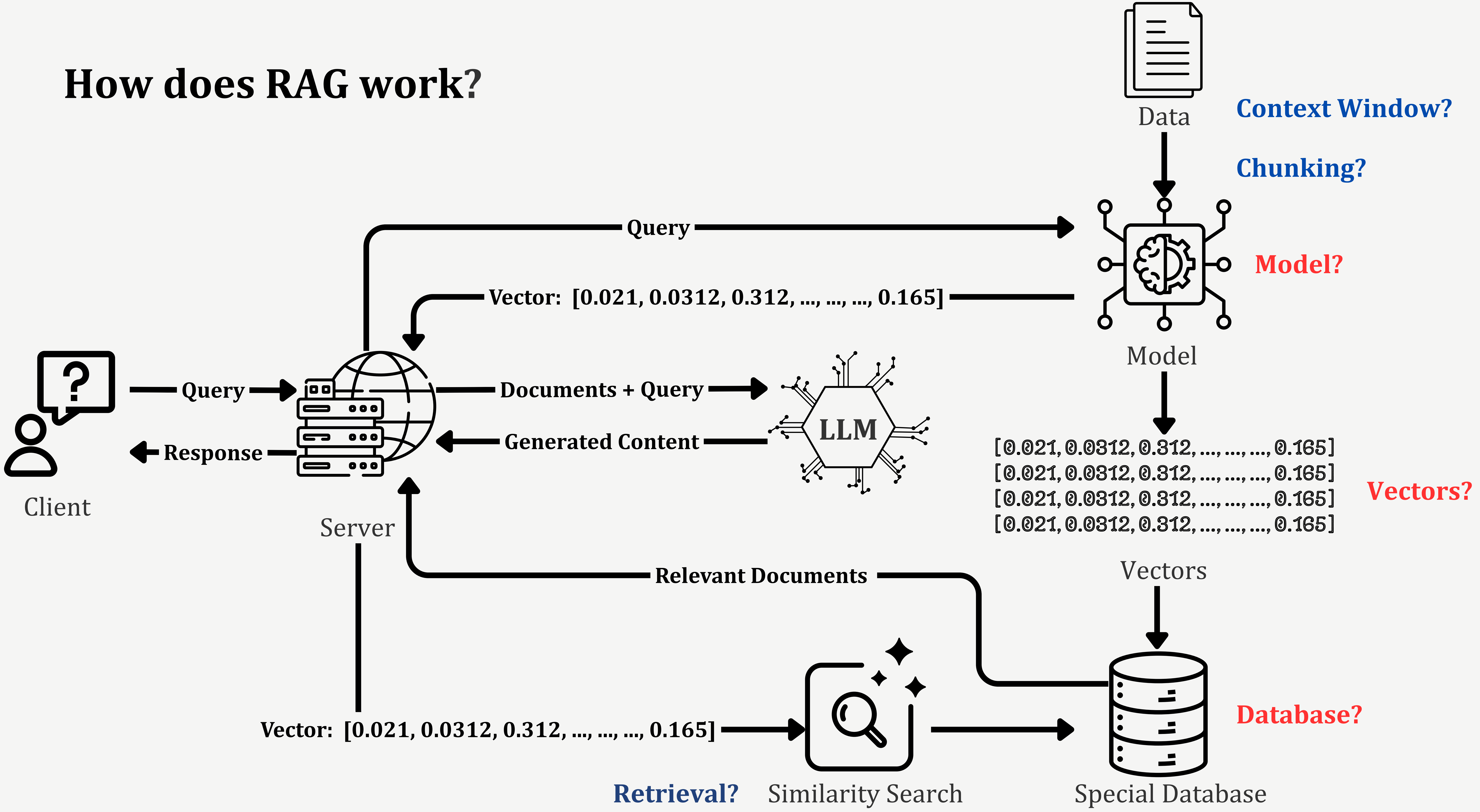




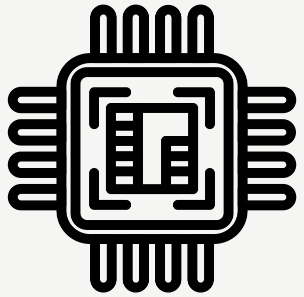
How does RAG work?



How does RAG work?

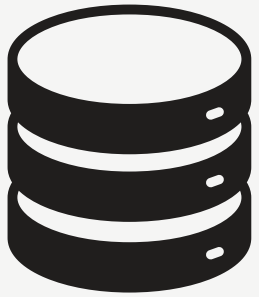


Important terminologies to understand — RAG



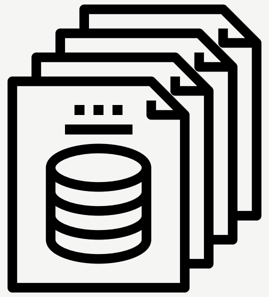
Embeddings

Numerical representations of text that capture meaning so similar texts have similar vectors.



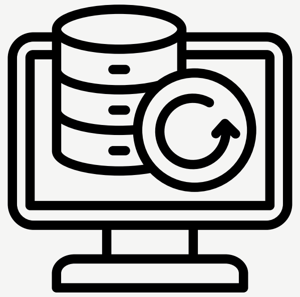
Vector Database

A specialized database that stores and searches embeddings to find the most similar chunks.



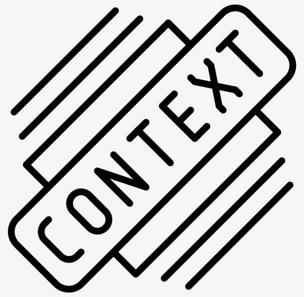
Chunking

Breaking long documents into smaller, meaningful pieces that fit within the model's context window.



Retrieval

The process of finding and returning the most relevant chunks from the vector database for a query.



Context Window

The maximum amount of text a model can read and use at one time.

Embeddings & Vector Database

- **Embedding:** A numerical representation of text, images, or data in a high-dimensional vector space
- **Vector:** The actual array of numbers (e.g., [0.12, 0.88, ...]) representing the embedding.
- **Purpose in RAG:** Allows similarity search between a query and documents to retrieve relevant context.

Advantages

- Captures semantic meaning, not just keywords.
- Enables fast similarity search using vector databases.
- Scales to large datasets efficiently.
- Makes LLMs dynamic and context-aware without fine-tuning.

King → [0.8, 1.0, 0.6]
Male → [0.9, 0.8, 0.4]
Queen → [0.7, 1.1, 0.5]
Female → [0.6, 1.0, 0.3]

Cosine similarity between King & Male is high, King & Queen is moderately high

[number-1, number-2, number-3, ..., number -n]

- Each number is called a dimension/component of the vector.
- Together, these numbers encode the semantic meaning of an item
- Pattern across all numbers captures meaning.
- Length of vector = number of dimensions

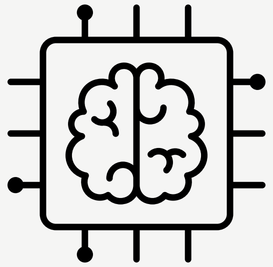
ChromaDB **pgVector** **MongoDB**

Context Window and Chunking

The context window is the maximum number of tokens the LLM can handle in one pass.

In RAG there are 2 phases:

1. Chunk document, Convert to Vectors and Store in Vector DB
2. Convert user's query into embedding, Retrieve relevant docs, pass query and docs to LLM



LLM

Context Window

1. Chunks exceed context window length
2. Query Embedding + Retrieved docs exceed context window length

2 important points to note

1. Always know the context window of LLM used in RAG pipeline

Size of chunks < context window

2. Reserve the Space for user's query while retrieving docs and passing query + retrieved docs to LLM

User query embeddings

Retrieved docs from Vector Database

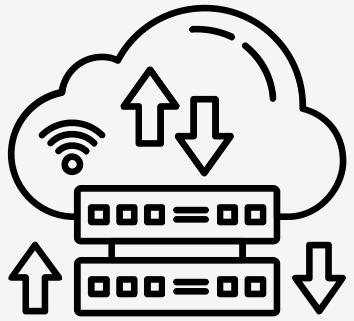
RAG vs Normal LLMs

Feature	Normal LLM	RAG (LLM + External Knowledge)
Source of information	Only trained on pre-existing data	Retrieves relevant documents
Accuracy	Can hallucinate or be outdated	Answers based on real sources
Knowledge update	Needs retraining	Instantly uses updated sources
Context limitation	Limited to model's training	Can handle larger context
Use-case	General questions	Domain-specific

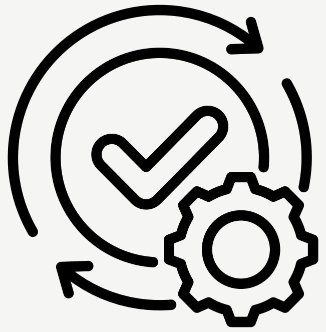
Why not just fine-tune?

- **Cost-efficient:** Fine-tuning large models is expensive and resource-intensive.
- **Faster deployment:** RAG works immediately on existing models with new data.
- **Dynamic updates:** No retraining needed; just update the document index.
- **Scalable:** Works for multiple domains without maintaining separate fine-tuned models.
- **Safer experimentation:** Easier to test new data sources without affecting the core model.

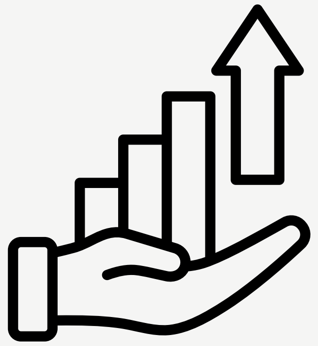
Cost-efficient



Faster deployment

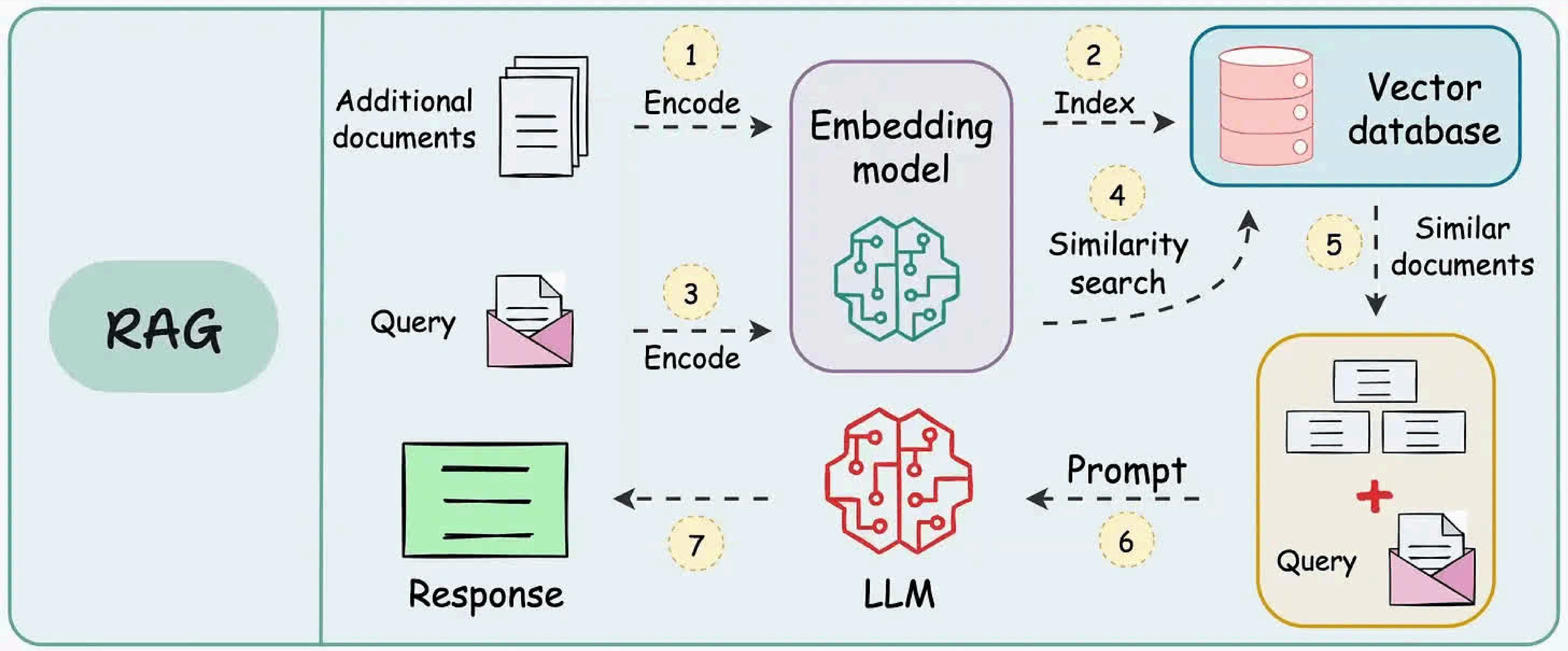


Dynamic updates



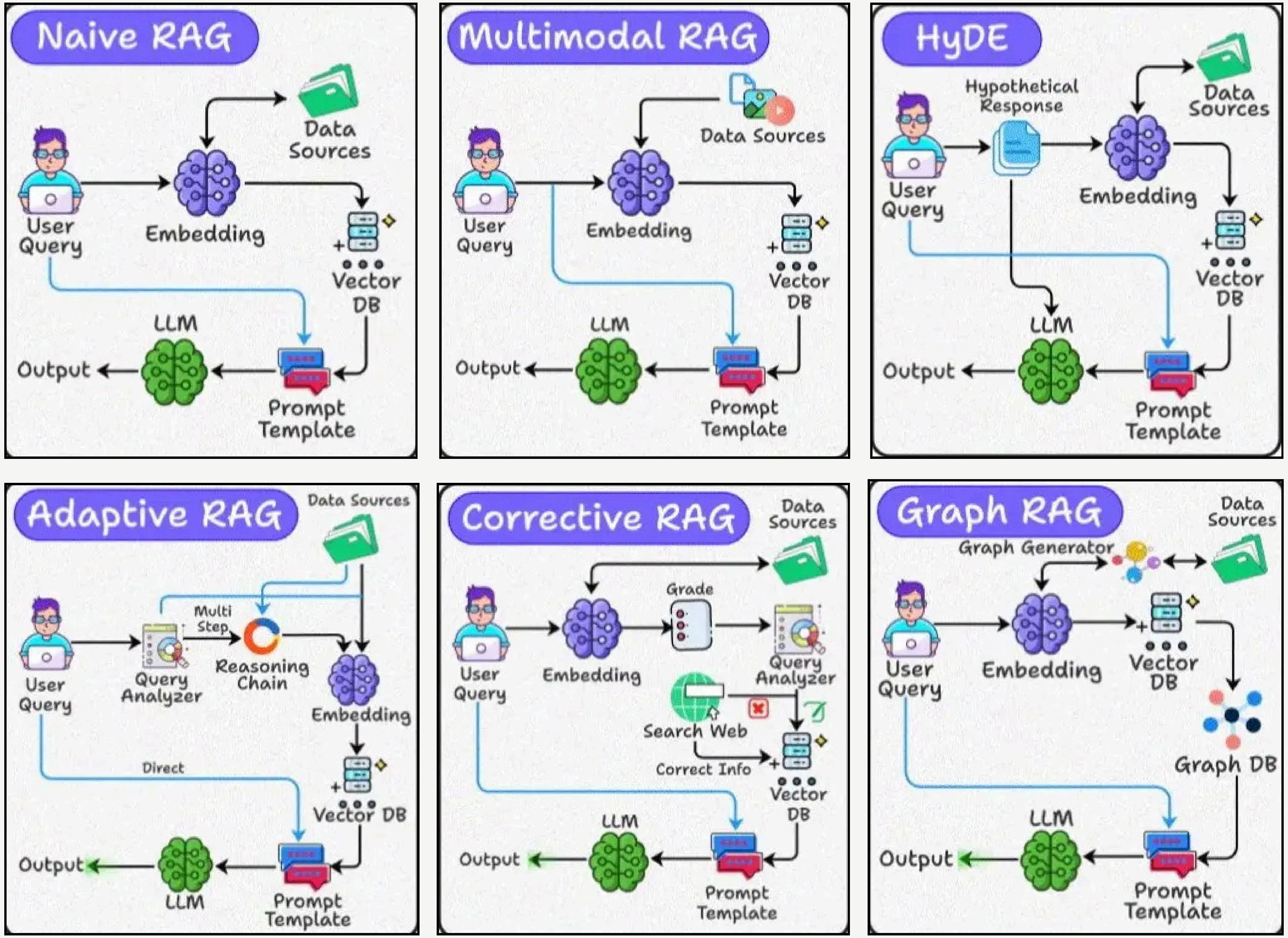
Scalable & Safer experimentation

Retrieval Augmented Generation — Architecture



Type of Retrieval Augmented Generation — RAG

1. Naive RAG
2. Multimodal RAG
3. Hypothetical Doc Embedding RAG
4. Adaptive RAG
5. Corrective RAG
6. Graph RAG



Technologies used for development — RAG Pipeline

Programming Language — *Python, Javascript*

Embedding Model — *Gemini, Llama, Opensource*

AI Framework — *Langchain, LlamaIndex*

UI Development — *Nextjs, TailwindCSS*

Vector Database — *ChromaDB, PineCone, pgVector*

Version Control — *GitHub, Gitlab*

Large Language Model — *Gemini, Gemma, Llama*

Testing Framework — *Playwright, Selenium*

AI Platforms — *Hugging Face, Groq AI*

Integration — *Axios, Fetch API*

Deployment — *AWS, GCP, Render, Vercel*

Where is RAG used?

Customer Support & Chatbots

Provides accurate, context-aware answers by retrieving FAQs, tickets, and product docs.

Finance & Banking

Retrieves regulatory rules, account info, and market data for accurate financial guidance.

Enterprise & Internal Search

Helps employees find internal documents and summaries quickly using company data.

Legal & Compliance

Summarizes contracts, statutes, and case law for faster legal research and drafting.

Healthcare & Life Sciences

Supports clinicians with patient history, research, and guidelines for evidence-based decisions.

Content Creation & Marketing

Generates marketing copy and content grounded in brand guidelines and documents.

Thank You!

https://linkedin.com/in/rudalphgonsalves

https://github.com/Rudalph

https://rudalph.vercel.app/

https://leetcode.com/u/gonsalvesrudalph

Any Questions?