Google Developer Group
GDG Bali

Building Scalable Infrastructure for RAG-Enabled Generative AI with Vertex AI

Ananda Dwi Rahmawati

Google Developer Expert - Cloud

{ Build with AI }

Google Developer Group
GDG Bali

Ananda Dwi Rahmawati

- ❏ Cloud & DevOps Engineer, Singapore
- ❏ Google Developer Expert Cloud - Modern Architecture
- ❏ Master of Computer Science - University of Texas at Austin
- ❏ https://linktr.ee/misskecupbung

{ Build with AI }

Chapter One

LLM and RAG-Based Chat Applications

Overview

"The future of AI isn't just in generating human-like responses; it's in augmenting those responses with knowledge from the real world."

What is an LLM?

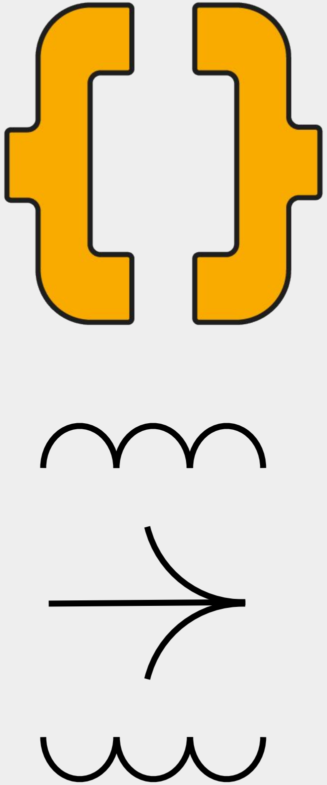
- Stands for **Large Language Model**
- Trained on vast text data to understand and generate human language
- Powers tools like ChatGPT, Google Gemini, and Meta LLaMA
- Can:
  - Answer questions
  - Summarize and translate text
  - Write code and assist with content creation

Gemini

What is RAG

- RAG = **Retrieval-Augmented Generation**
- Combines **external knowledge** retrieval with LLM-based text generation
- Improves **accuracy** by pulling in relevant and up-to-date external data at query time
- Especially useful when the LLM's training data is **outdated** or **incomplete**
- **Avoids costly** model retraining by using live, trusted, authoritative data sources

Why is Retrieval-Augmented Generation important?



Known challenges of LLMs include:

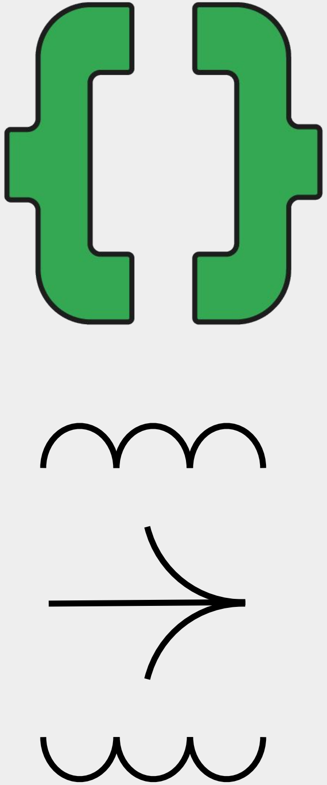
- Presenting **false** information when it does not have the answer.
- Presenting **out-of-date** or generic information when the user expects a specific, current response.
- Creating a response from **non-authoritative sources.**
- Creating **inaccurate responses** due to terminology confusion, wherein different training sources use the same terminology to talk about different things.

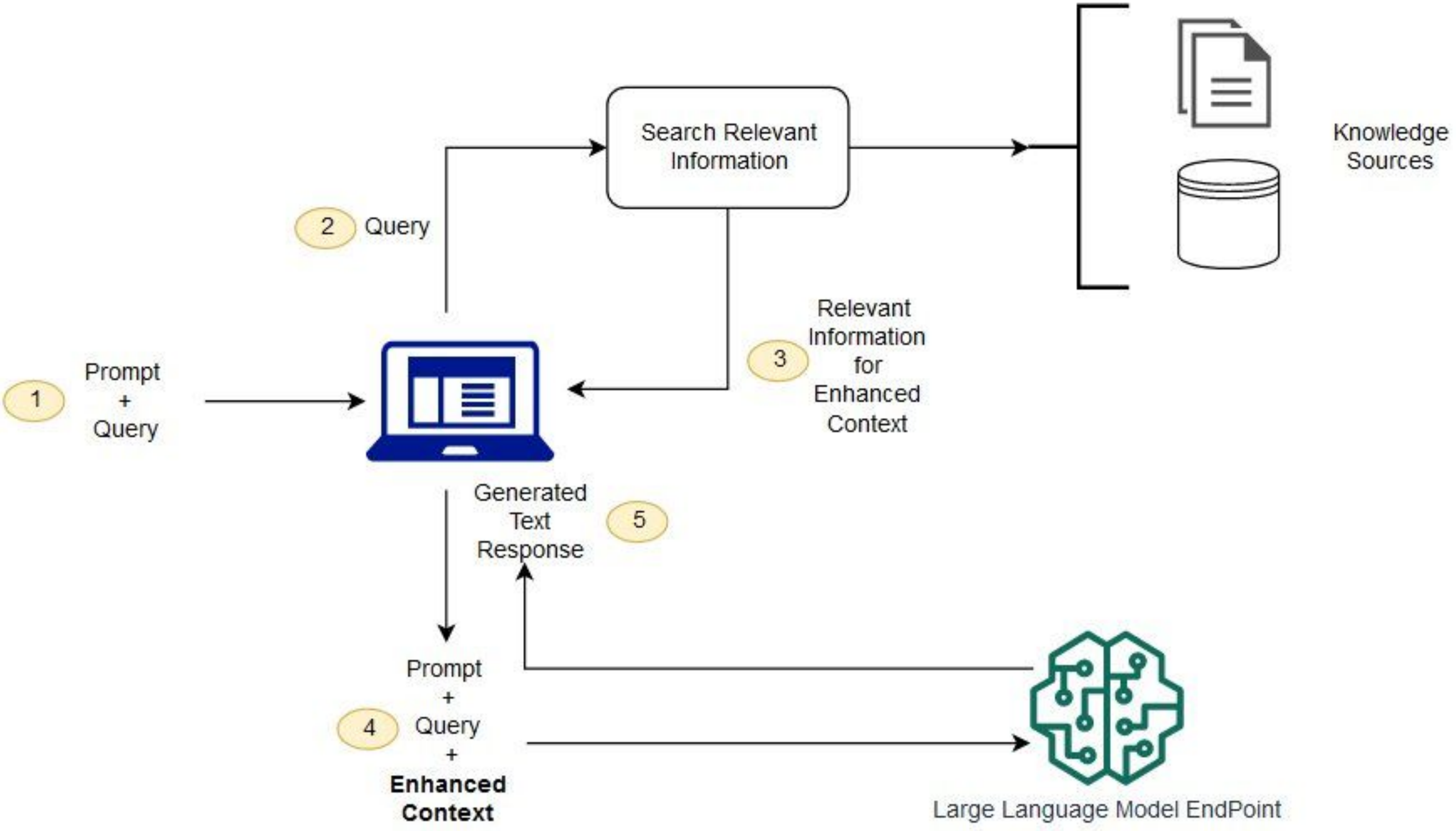
Key Benefits of Using RAG

- **Cost-efficient implementation strategy:** Skip expensive retraining
- **Real-time access to information**: Always current and relevant
- **Source-aware outputs**: Includes references or citations for trust
- **Greater developer control**: Manage data, behavior, and access securely

How RAG Works

- Receive **user query** from the chatbot or application
- Retrieve highly relevant documents via **vector-based search**
- Augment the LLM prompt with contextual, **retrieved information**
- **Generate accurate responses** using both training and new data
- Update external knowledge sources regularly for freshness





1 Prompt + Query

2 Query

Search Relevant Information

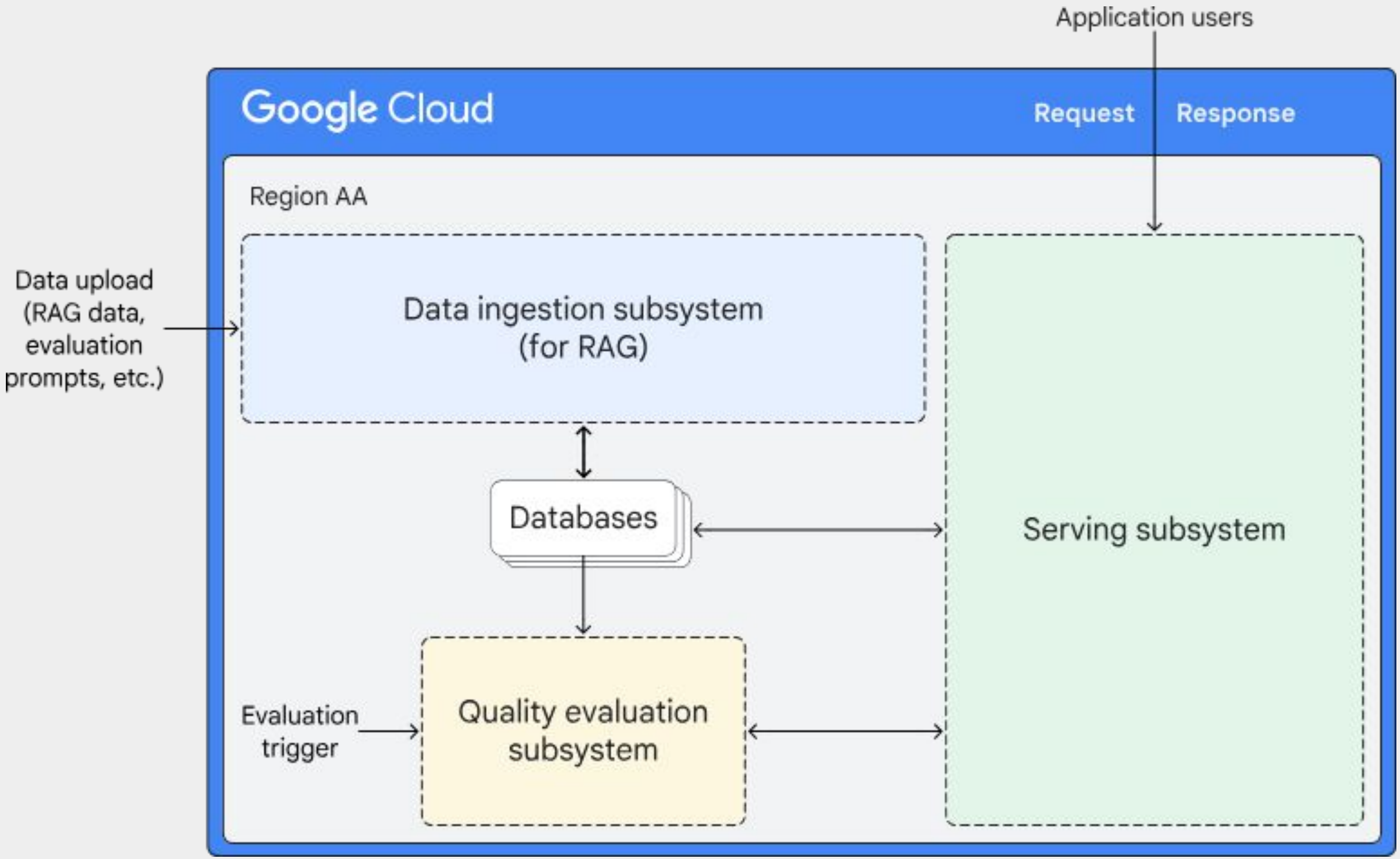
Knowledge Sources

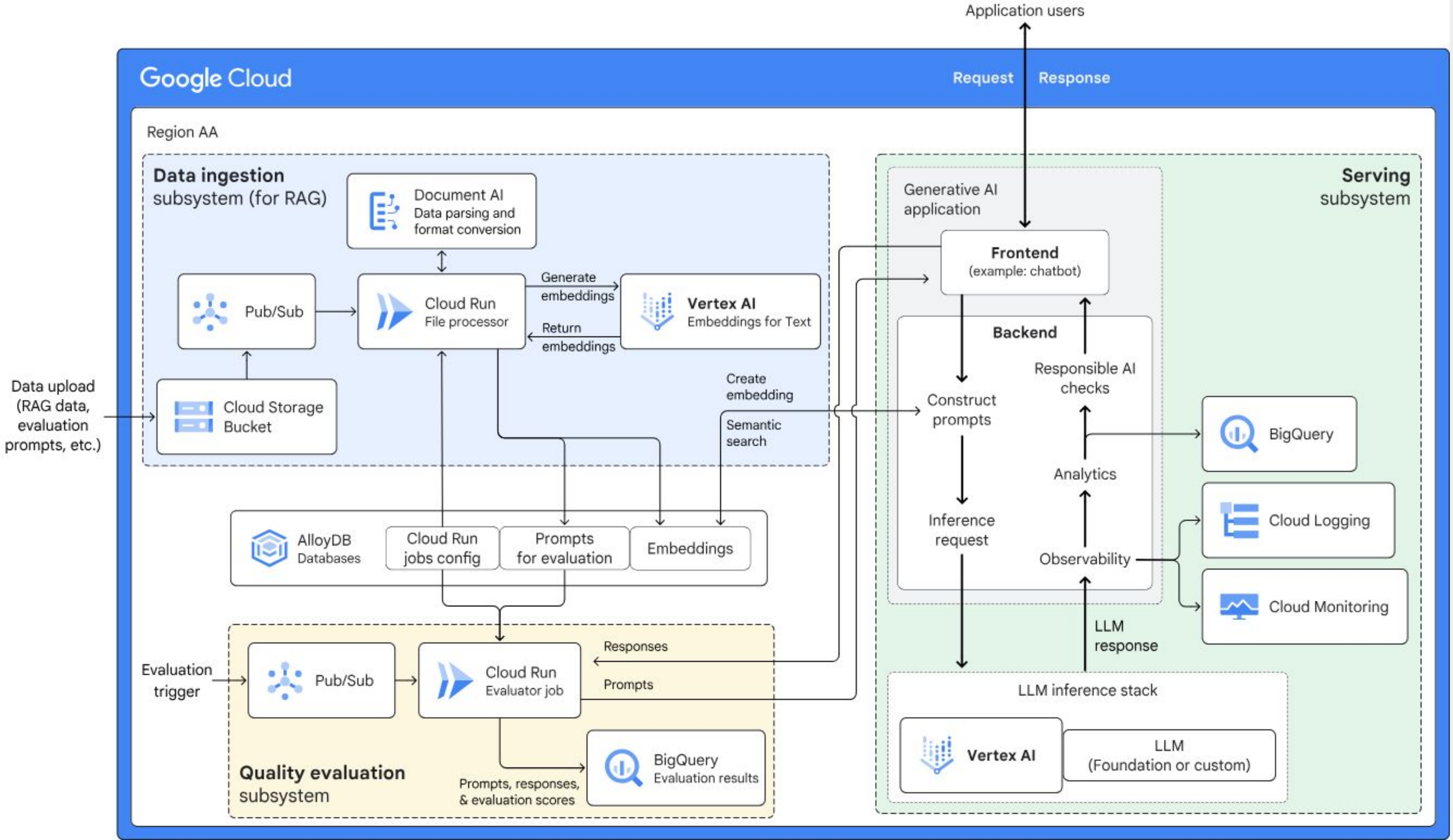
3 Relevant Information for Enhanced Context

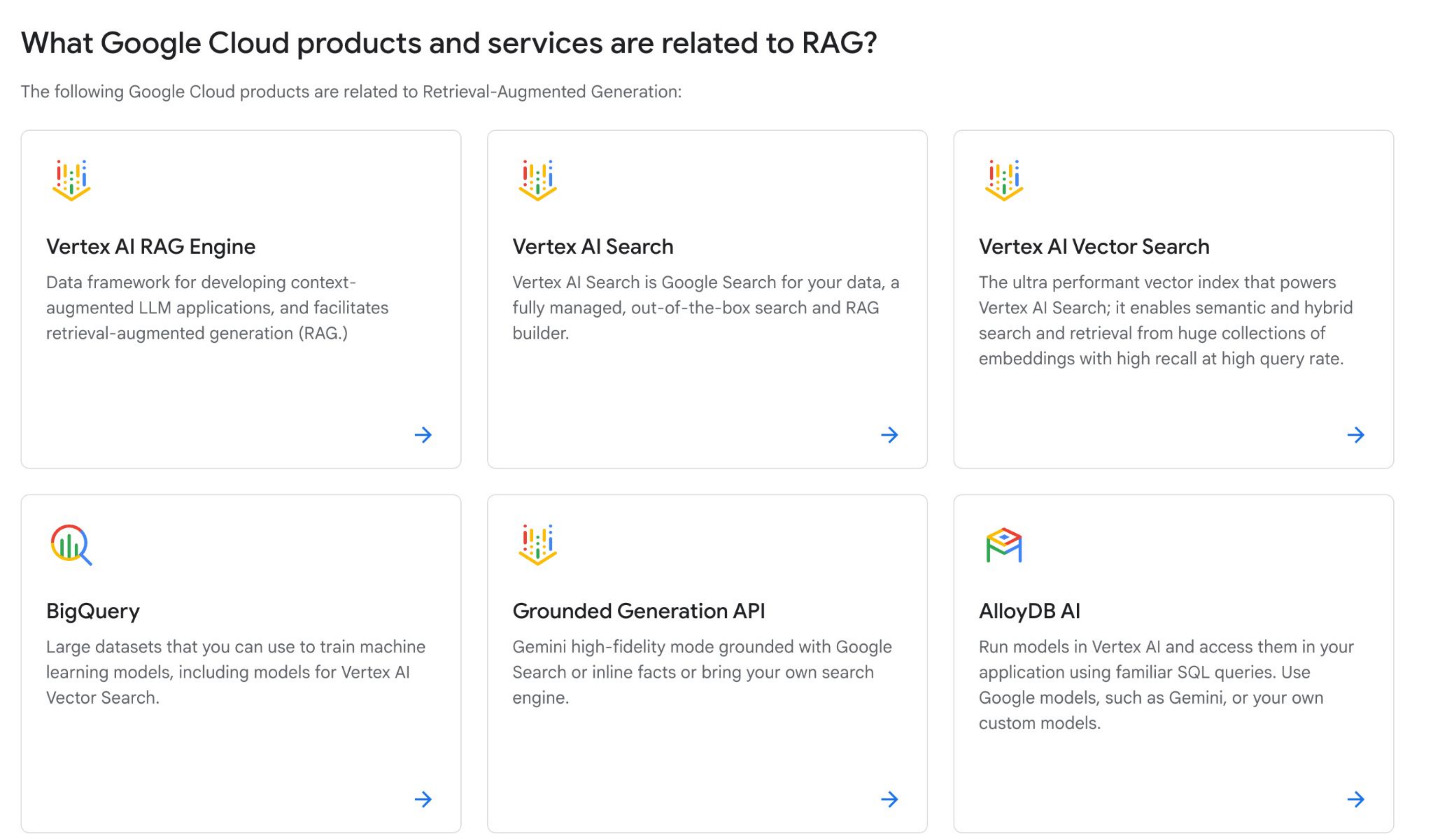
4 Prompt + Query + **Enhanced Context**

Large Language Model EndPoint

5 Generated Text Response


Application users
Google Cloud
Request
Response
Region AA
Data upload (RAG data, evaluation prompts, etc.)
Data ingestion subsystem (for RAG)
Databases
Serving subsystem
Evaluation trigger
Quality evaluation subsystem


Application users
Google Cloud
Request
Response
Region AA
Data ingestion subsystem (for RAG)
Document AI
Data parsing and format conversion
Pub/Sub
Cloud Run
File processor
Generate embeddings
Return embeddings
Vertex AI
Embeddings for Text
Create embedding
Semantic search
Data upload (RAG data, evaluation prompts, etc.)
Cloud Storage Bucket
AlloyDB
Databases
Cloud Run jobs config
Prompts for evaluation
Embeddings
Evaluation trigger
Pub/Sub
Cloud Run
Evaluator job
Responses
Prompts
BigQuery
Evaluation results
Prompts, responses, & evaluation scores
Quality evaluation subsystem
Serving subsystem
Generative AI application
Frontend
(example: chatbot)
Backend
Construct prompts
Inference request
Responsible AI checks
Analytics
Observability
BigQuery
Cloud Logging
Cloud Monitoring
LLM response
LLM inference stack
Vertex AI
LLM
(Foundation or custom)



What Google Cloud products and services are related to RAG?

The following Google Cloud products are related to Retrieval-Augmented Generation:

Vertex AI RAG Engine

Data framework for developing context-augmented LLM applications, and facilitates retrieval-augmented generation (RAG.)

→

Vertex AI Search

Vertex AI Search is Google Search for your data, a fully managed, out-of-the-box search and RAG builder.

→

Vertex AI Vector Search

The ultra performant vector index that powers Vertex AI Search; it enables semantic and hybrid search and retrieval from huge collections of embeddings with high recall at high query rate.

→

BigQuery

Large datasets that you can use to train machine learning models, including models for Vertex AI Vector Search.

→

Grounded Generation API

Gemini high-fidelity mode grounded with Google Search or inline facts or bring your own search engine.

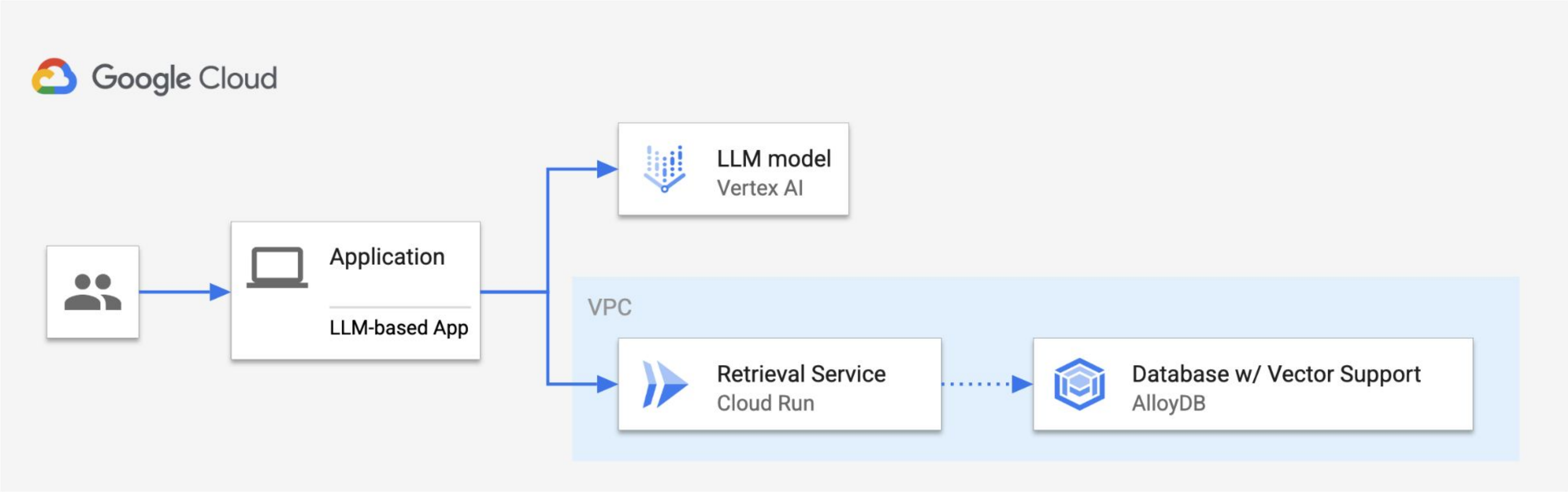
→

AlloyDB AI

Run models in Vertex AI and access them in your application using familiar SQL queries. Use Google models, such as Gemini, or your own custom models.

→

GenAI Database Retrieval



Google Cloud

LLM model
Vertex AI

Application

LLM-based App

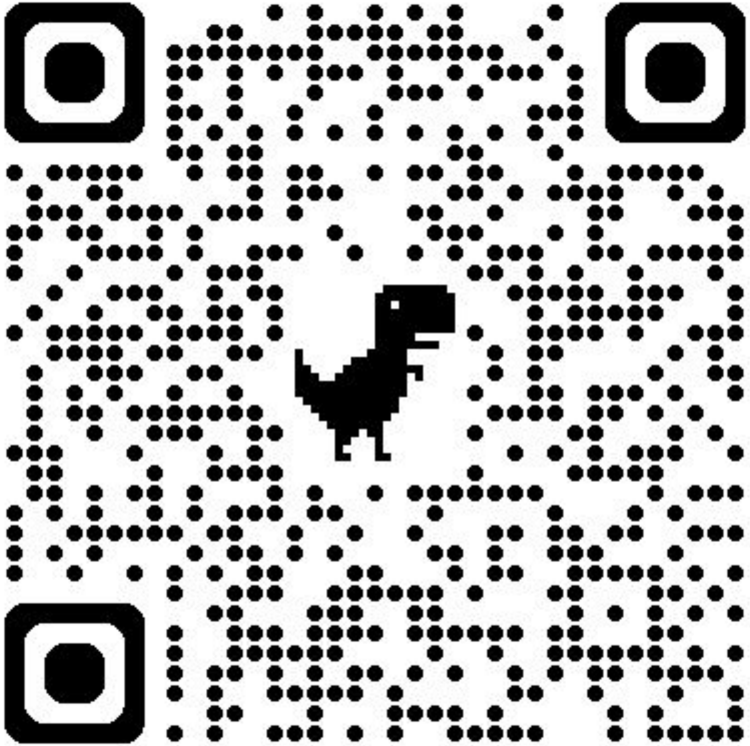
VPC

Retrieval Service
Cloud Run

Database w/ Vector Support
AlloyDB

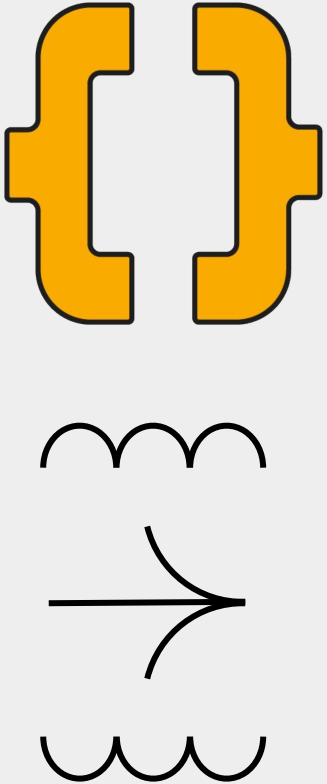
Key Components of the System

- **AlloyDB:** Managed, scalable database for storing large datasets (e.g., knowledge base, documents).
- **Vertex AI:** Provides access to Google's powerful language models for natural language understanding and generation.
- **Retrieval-Augmented Generation:** Combines LLMs with a search-based system to fetch relevant data, enhancing responses.
- **Chat Interface:** A front-end application where users can interact with the system.



AlloyDB for Data Storage and Retrieval

- **What is AlloyDB?**
  - Fully managed PostgreSQL-based database for scalable, high-performance querying.
  - Optimized for analytical workloads and real-time retrieval.
- **How AlloyDB Fits In:**
  - Stores large sets of documents, user interactions, or knowledge base for easy retrieval.
  - Acts as the primary data source for the retrieval step in RAG-based systems.
- **Use Case Example:**
  - Querying a database for customer information or knowledge articles to augment the chat conversation.



Leveraging Vertex AI for Language Understanding and Generation

- **What is Vertex AI?**
  - Google's AI platform for building and deploying machine learning models.
  - Includes pre-trained models for language understanding, generation, and specialized NLP tasks.
- **How Vertex AI Enhances Chat Application:**
  - Provides access to GPT-like models to handle natural language understanding.
  - Can be fine-tuned for specific tasks (e.g., customer service, FAQ generation).
- **RAG in Vertex AI:**
  - The retrieval part of RAG fetches relevant data from AlloyDB, then Vertex AI generates context-aware responses.

Integration and Workflow of the Chat Application

- **Workflow:**
  - **User Input:** User submits a query through the chat interface.
  - Knowledge Retrieval (RAG): The system queries AlloyDB to find relevant documents or data.
  - **Response Generation:** Vertex AI processes the retrieved data to generate a response that is contextually aware.
  - **User Response:** The generated response is sent back to the user.
- **Key Benefits:**
  - **Scalability:** AlloyDB handles large datasets, ensuring fast retrieval.
  - **Accuracy:** RAG improves response quality by supplementing LLM-generated content with external knowledge.
  - **Efficiency:** Vertex AI provides the language model backend, optimized for quick deployment and fine-tuning.

References

- https://github.com/GoogleCloudPlatform/genai-databases-retrieval-app/blob/main/docs/datastore/alloydb.md
- https://www.cloudskillsboost.google/focuses/97381?parent=catalog
- https://cloud.google.com/alloydb/docs/overview
- https://github.com/GoogleCloudPlatform/vertex-ai-samples