A Brief Tour of Generative AI on Google Cloud

Google models & Vertex AI Studio

Foundation models

BERT [Oct 2018]: Pre-text task with ~340M transformer model

XLNet [June 2019]: Autoregressive Pretraining

GPT-3 [May 2020]: Chatbot model at extreme scales (175B)

CLIP [Jan 2021]: Image captioning using pre-training tricks inspired by BERT (63M)

DALL·E [Jan 2021]: Text-to-image generation with a "mini" GPT-3 (12B)

HAI: Human-centered Artificial Intelligence

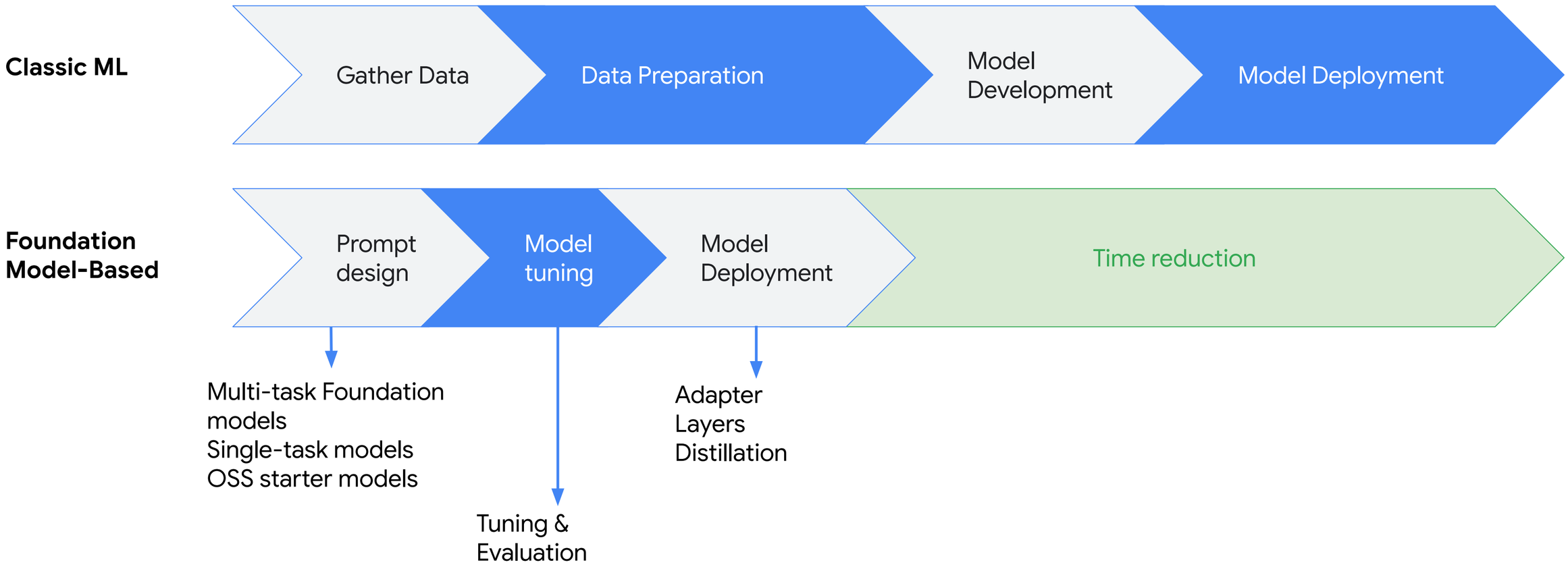
On the Opportunities and Risks of Foundation Models

Rishi Bommasani* Drew A. Hudson Ehsan Adeli Russ Altman Simran Arora
Sydney von Arx Michael S. Bernstein Jeannette Bohg Antoine Bosselut Emma Brunskill
Erik Brynjolfsson Shyamal Buch Dallas Card Rodrigo Castellon Niladri Chatterji
Annie Chen Kathleen Creel Jared Quincy Davis Dorottya Demszky Chris Donahue
Moussa Doumbouya Esin Durmus Stefano Ermon John Etchemendy Kawin Ethayarajh
Li Fei-Fei Chelsea Finn Trevor Gale Lauren Gillespie Karan Goel Noah Goodman
Shelby Grossman Neel Guha Tatsunori Hashimoto Peter Henderson John Hewitt
Daniel E. Ho Jenny Hong Kyle Hsu Jing Huang Thomas Icard Saahil Jain
Dan Jurafsky Pratyusha Kalluri Siddharth Karamcheti Geoff Keeling Fereshte Khani
Omar Khattab Pang Wei Koh Mark Krass Ranjay Krishna Rohith Kuditipudi
Ananya Kumar Faisal Ladhak Mina Lee Tony Lee Jure Leskovec Isabelle Levent
Xiang Lisa Li Xuechen Li Tengyu Ma Ali Malik Christopher D. Manning
Suvir Mirchandani Eric Mitchell Zanele Munyikwa Suraj Nair Avanika Narayan
Deepak Narayanan Ben Newman Allen Nie Juan Carlos Niebles Hamed Nilforoshan
Julian Nyarko Giray Ogut Laurel Orr Isabel Papadimitriou Joon Sung Park Chris Piech
Eva Portelance Christopher Potts Aditi Raghunathan Rob Reich Hongyu Ren
Frieda Rong Yusuf Roohani Camilo Ruiz Jack Ryan Christopher Ré Dorsa Sadigh
Shiori Sagawa Keshav Santhanam Andy Shih Krishnan Srinivasan Alex Tamkin
Rohan Taori Armin W. Thomas Florian Tramèr Rose E. Wang William Wang Bohan Wu
Jiajun Wu Yuhuai Wu Sang Michael Xie Michihiro Yasunaga Jiaxuan You Matei Zaharia
Michael Zhang Tianyi Zhang Xikun Zhang Yuhui Zhang Lucia Zheng Kaitlyn Zhou
Percy Liang*[1]

Center for Research on Foundation Models (CRFM)
Stanford Institute for Human-Centered Artificial Intelligence (HAI)
Stanford University

AI is undergoing a paradigm shift with the rise of models (e.g., BERT, DALL-E, GPT-3) trained on broad data (generally using self-supervision at scale) that can be adapted to a wide range of downstream tasks. We call these models foundation models *to underscore their critically central yet incomplete character. This report provides a thorough account of the opportunities and risks of foundation models, ranging from their capabilities (e.g., language, vision, robotic manipulation, reasoning, human interaction) and technical principles (e.g., model architectures, training procedures, data, systems, security, evaluation, theory) to their applications (e.g., law, healthcare, education) and societal impact (e.g., inequity, misuse, economic and environmental impact, legal and ethical considerations). Though foundation models are based on standard deep learning and transfer learning, their scale results in new emergent capabilities, and their effectiveness across so many tasks incentivizes homogenization. Homogenization provides powerful leverage but demands caution, as the defects of the foundation model are inherited by all the adapted models downstream. Despite the impending widespread deployment of foundation models, we currently lack a clear understanding of how they work, when they fail, and what they are even capable of due to their emergent properties. To tackle these questions, we believe much of the critical*

Foundation models accelerate time to model deployment



This revolution started at Google

2017

Transformers

- Pathbreaking Neural Network Architecture
- Open Sourced by Google in 2017
- Started the revolution in Language Models

2018

BERT

(Bidirectional Encoder Representations from Transformers)

- World's first Language Model
- Open Sourced by Google in 2018
- SOTA on number of language benchmarks

2019

T5

(Text-to-Text Transfer Transformer)

- Large Language Encoder-Decoder Model
- 10-billion parameter model
- Open Sourced by Google in 2019

2020

LaMDA

(Language Model for Dialog Applications)

- Model trained on dialogue data
- Model could talk about virtually anything
- Published by Google in 2020

2121

Diffusion Models

High Fidelity Image Generation Using Diffusion Models

PaLM

- (Pathways Language Model)
- Single model to generalize across domains
- 540-billion parameter, dense decoder model

2022

CALM

(Confident Adaptive Language Modeling)

- Accelerating the text generation of LMs

2023

Bard

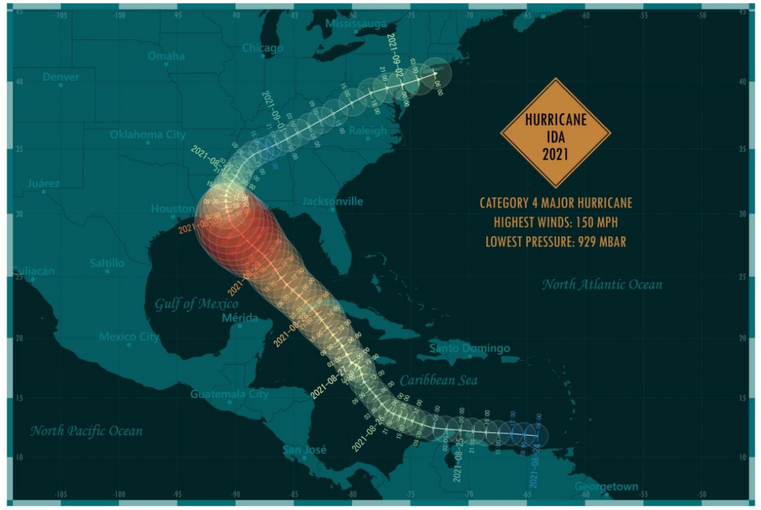
- A conversational AI service powered by LaMDA.

Gemini is a multimodal foundation model trained on text, images, video and audio

- Prompts can contain a combination of text, images, and video
- Capable of performing a wide range of text and vision-related tasks
  - Generate text
  - Extract text from images & video
  - Caption images, video, or audio
  - Understand and respond to questions about video, text and audio
- Multiple versions:
  - **Pro**: Most advanced
  - **Flash**: Nearly as good and faster & cheaper

Gemini's multimodal capabilities mean it can understand images, graphics & tables

Prompt

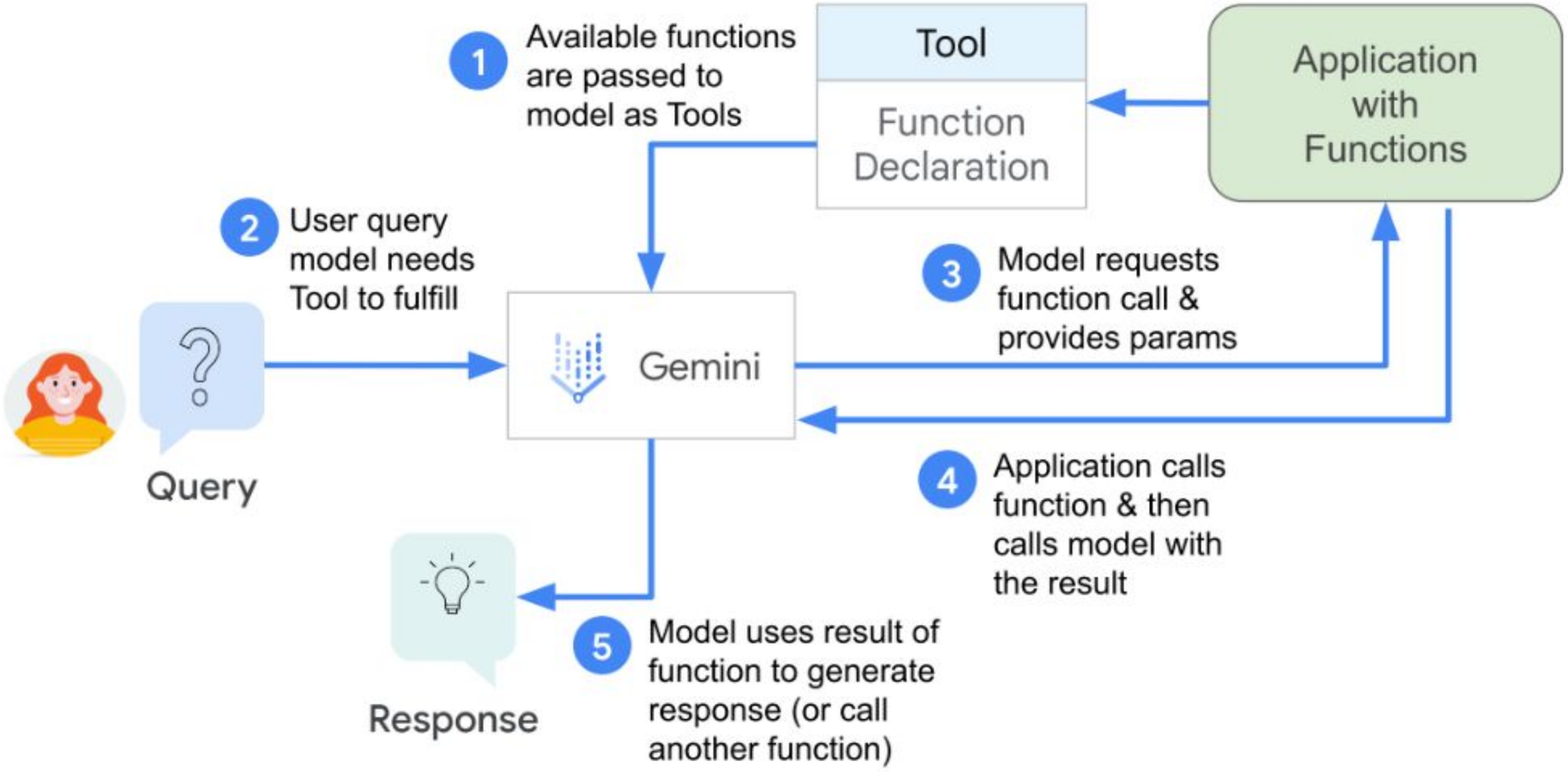


What major event is represented in the image? Which state did it have a severe impact on and when did it make landfall? Answer all questions in bullet points with just the answer, do not use complete sentences.

Response

- Hurricane Ida
- Louisiana
- August 29, 2021

Function Calling lets Gemini pass your system a request for a function to be called, then use the result you return



Gemini for Google Cloud Portfolio

Application Lifecycle

Efficiently manage cloud applications

Gemini Cloud Assist

Preview

Software Development

Accelerate software delivery

Gemini Code Assist

Security

Elevate security expertise

Gemini in Security

Data Analytics

Fast-track data analysis

Gemini in BigQuery

Business Intelligence

Automate data Insights

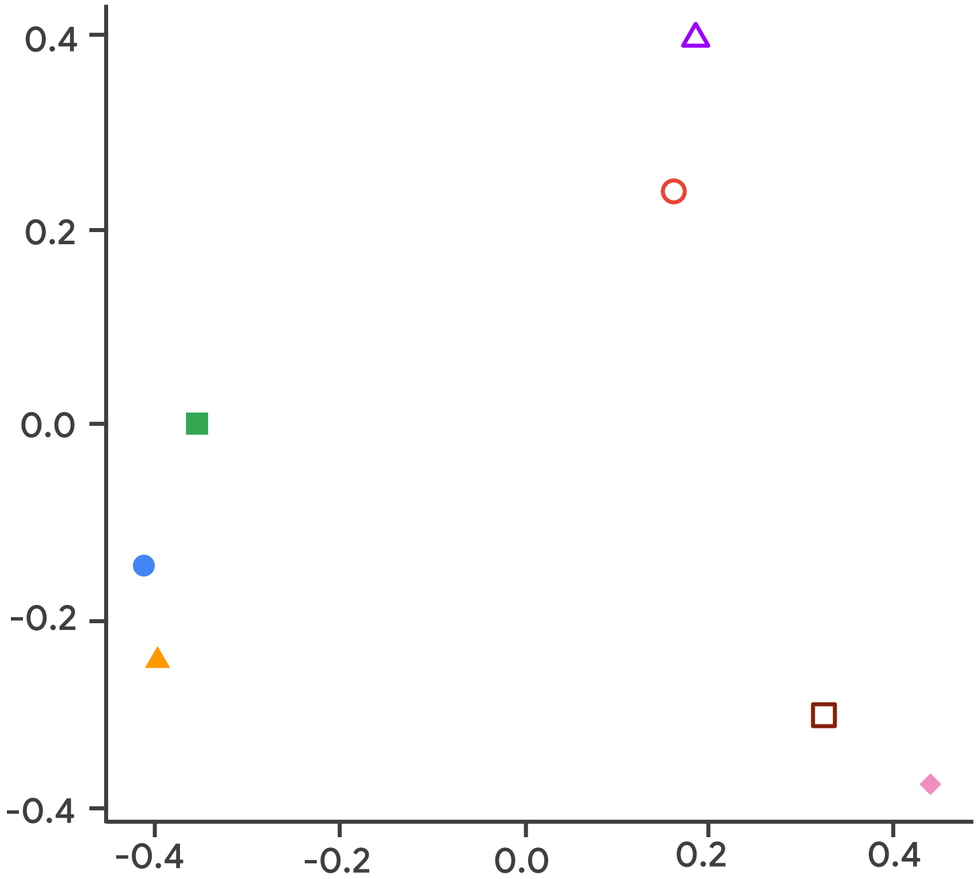
Gemini in Looker

Databases

Supercharge database development & management

Gemini in Databases

The Embeddings API returns embeddings for text, text-multilingual or multimodal prompts



sentences

- Missing flamingo discovered at swimming pool
- Sea otter potted on surfboard by beach
- Baby panda enjoys boat ride
- Breakfast themed food truck beloved by all!
- New curry restaurant aims to please!
- Python developers are wonderful people
- TypeScript, C++ or Jave? All are great!

Gemma is a family of lightweight, open models built from the same technology used to create the Gemini

- Small enough to run on mobile devices, desktop and laptop computers, and your own servers
- Comes in multiple flavors
  - **Gemma 2**: The latest text-only version
  - **PaliGemma**: Image + text as input, text as output
  - **CodeGemma**: Further trained on code & math
  - **RecurrentGemma**: A distinct model focusing on memory efficiency
- Can deploy using Vertex AI Model Registry and Model Endpoints

You can try out Gemma on your own computer

- Go to https://ollama.com/ and download and install the program
- Go to your terminal and type: `ollama run gemma`
- Enter a prompt to try it out

When should you deploy Gemma on a project?

- On systems that can’t connect to the Cloud for security or latency reasons
- On edge devices
- For experimenting with whole foundation-model tuning

MedLM is a HIPAA-compliant suite of medically tuned models and APIs powered by Google Research

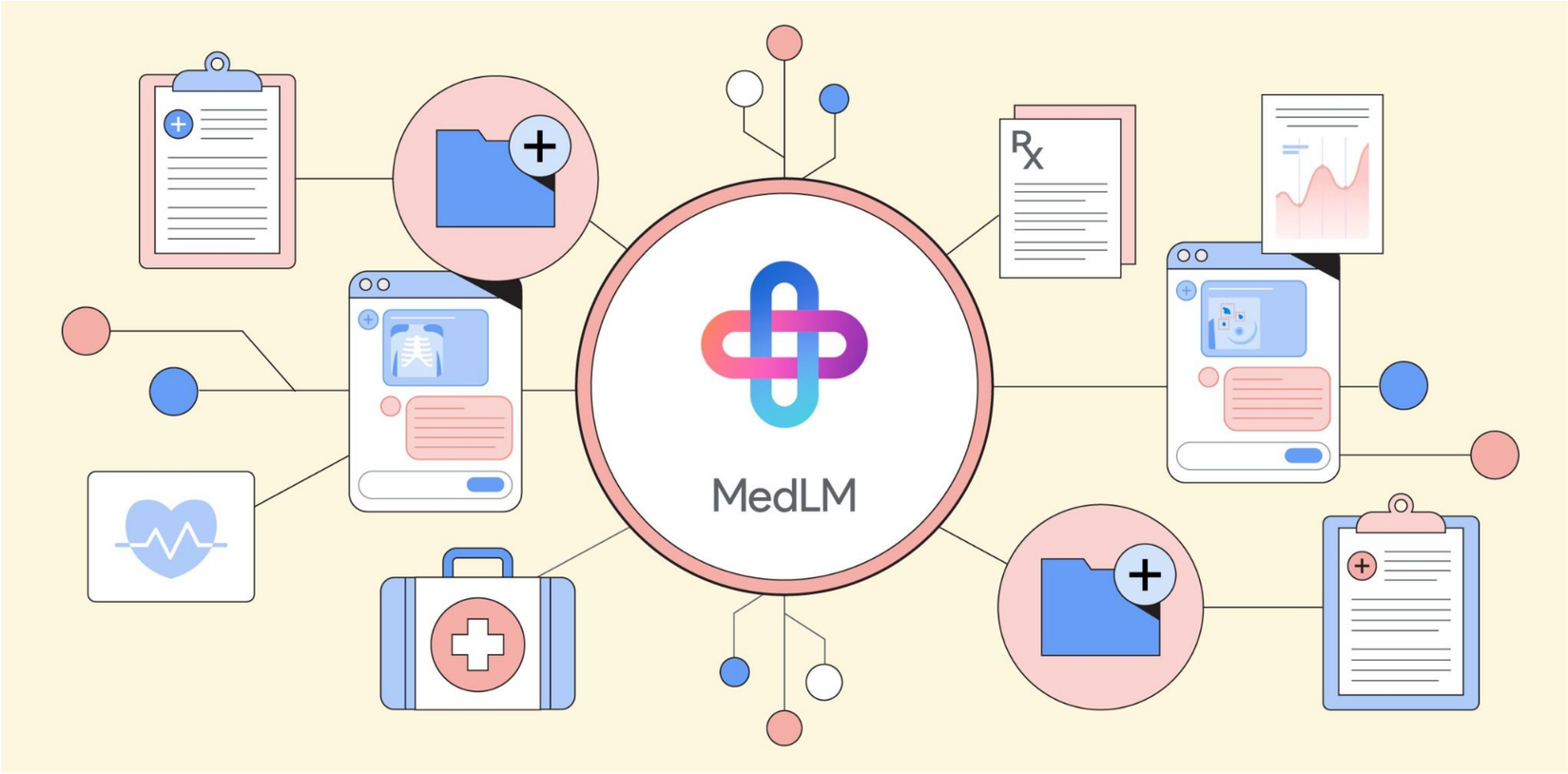
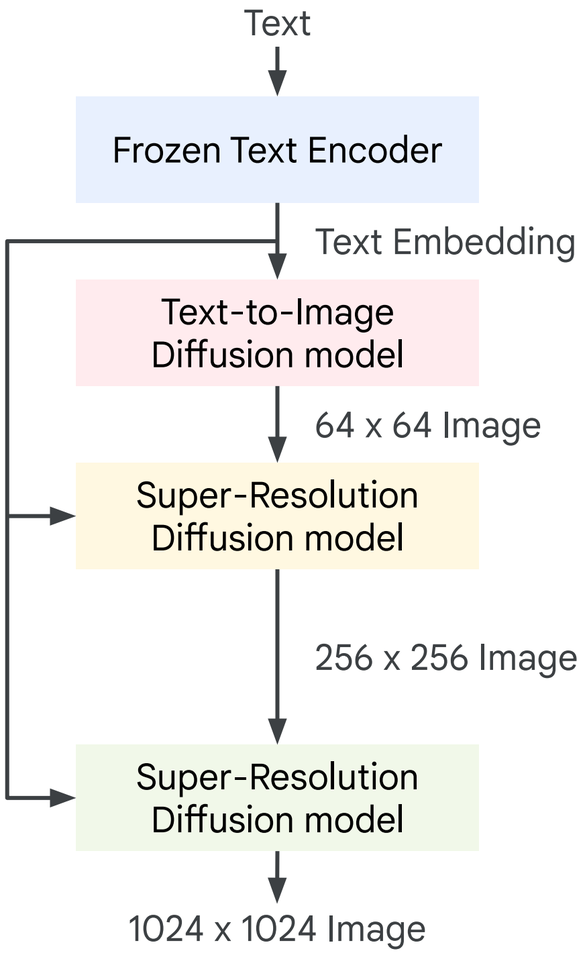


Imagen is Google's Foundation model for Vision

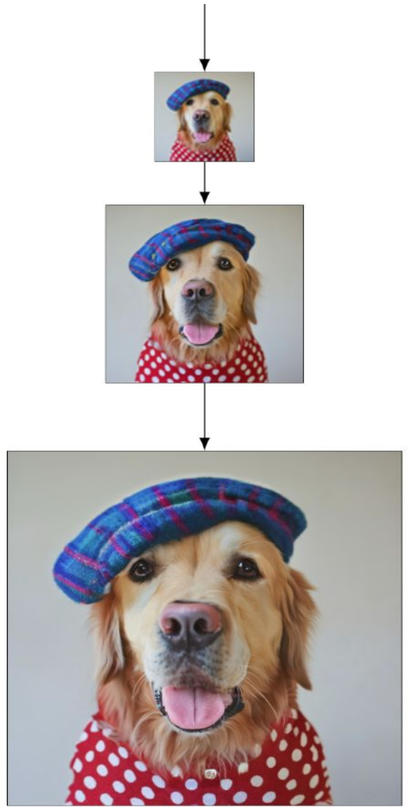
- Imagen is capable of performing a wide range of vision-related tasks
  - Generate an image
  - Edit a masked section of an image
  - Caption an image
  - Visual Q&A (Answer questions about an image)
  - The documentation shows a feature roadmap with more features planned

A dragon fruit wearing a karate belt in the snow.

Imagen uses diffusion-based techniques to generate images



"A Golden Retriever dog wearing a blue beret and red dotted turtleneck."



Diffusion models iteratively refine a noise-filled image to approximate the target image distribution.

Vertex AI allows for tuning of Google foundation models

Gemini

- Supervised fine-tuning is available for Gemini

Text embeddings models

- Use parameter-efficient tuning for customizing what the model considers similar or dissimilar

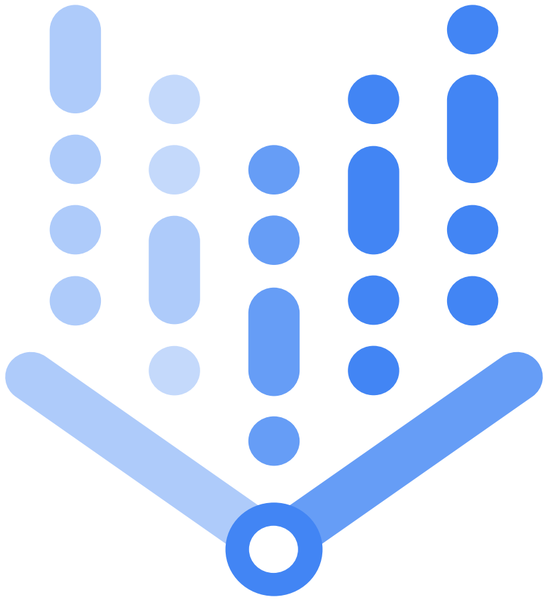
Imagen

- Style Fine-tuning
- Subject Fine-tuning

PaLM (previous generation of text models)

- Reinforcement learning through Human Feedback
- Knowledge distillation

Let’s take a look at Vertex AI Studio!



Even more models in Vertex AI Model Garden

Let’s check out some other models available to you in Model Garden!




You can call Anthropic's Claude natively on Vertex AI

```
from anthropic import AnthropicVertex

client = AnthropicVertex(region="us-east5", project_id=PROJECT_ID)
message = client.messages.create(
   max_tokens=1024,
   messages=[
       {
           "role": "user",
           "content": "Send me a recipe for banana bread.",
       }
   ],
   model="claude-3-5-sonnet@20240620")
print(message.model_dump_json(indent=2))
```

Key Docs:
Model Versions, Token Windows & Discontinuation Dates

Get familiar with the documentation!

Bookmark this Google models documentation to see:

- the latest model version identifiers
- capabilities (like eligibility for grounding & tuning)
- model specifications (like token windows)
- version discontinuation dates

Note the token windows for text & multimodal embeddings.

Check out useful guides in the docs as well.

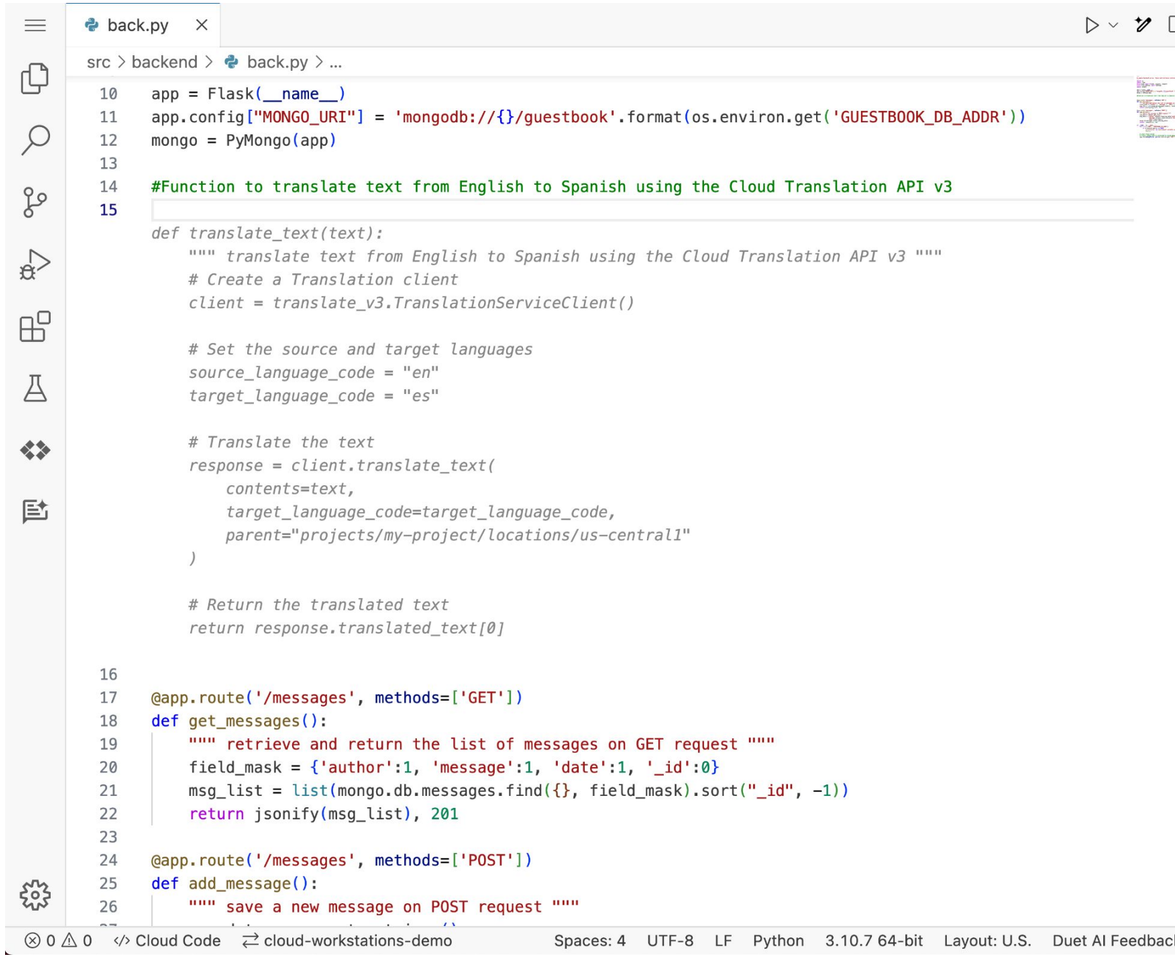
- Using a GenerativeModel for a free token count
- Prompt design strategies (we'll review this a little later)
- Deployment best practices for generative AI models
- Understand rate limits / quotas on Google generative AI models

Developer Acceleration with Gemini Code Assist

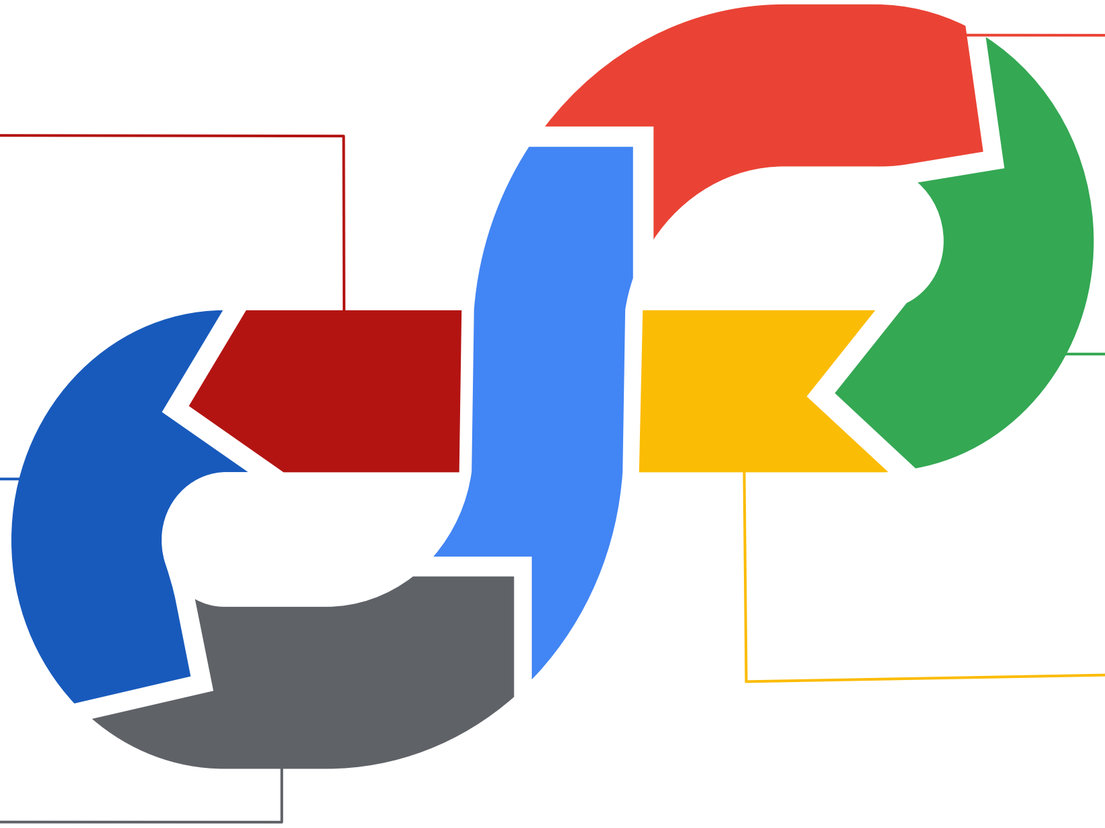
Gemini Code Assist is your AI developer collaborator

- Duet AI for Developers is now Gemini Code Assist
- In-IDE Code completion, Chat and Smart actions
- Powered by state-of-the-art Gemini models
- Enterprise ready governance, controls, and security

Since December 2023 launch, 5,000+ businesses actively use Gemini Code Assist



Gemini Code Assist improves each stage of the SDLC



01. Design

Design and prototype a applications with **in-IDE chat**

02. Build

Generate code or modify existing code with **code completions**

03. Test + Maintain

Generate unit tests and documentation via smart actions.

04. Deploy

Manage environments with **assistance for infrastructure code interfaces** (Terraform, KRM, gCloud)

05. Troubleshoot

AI-driven issue analysis in Crashlytics for Crashes, ANRs and Errors

06. Operate

Assistance in **understanding and modifying** existing applications

Case study: Gemini Code Assist helped Wayfair increase productivity and developer satisfaction

+55%

Faster environment setup time

+48%

Higher unit test coverage

+60%

Developers report being able to focus on more satisfying work

When AI helps improve developer productivity, it impacts the entire business

Generative AI in software engineering functions could increase **productivity by 20-45%**[1]

Code generation	Code refactoring
Code completion	Code explanation
Code documentation	Code testing

Business Impact

Scale

- Faster time to market with higher quality products
- Grow business with less communication overhead

Cost

- Reduce costs to train and upskill developers
- Decrease technical debt

Experience

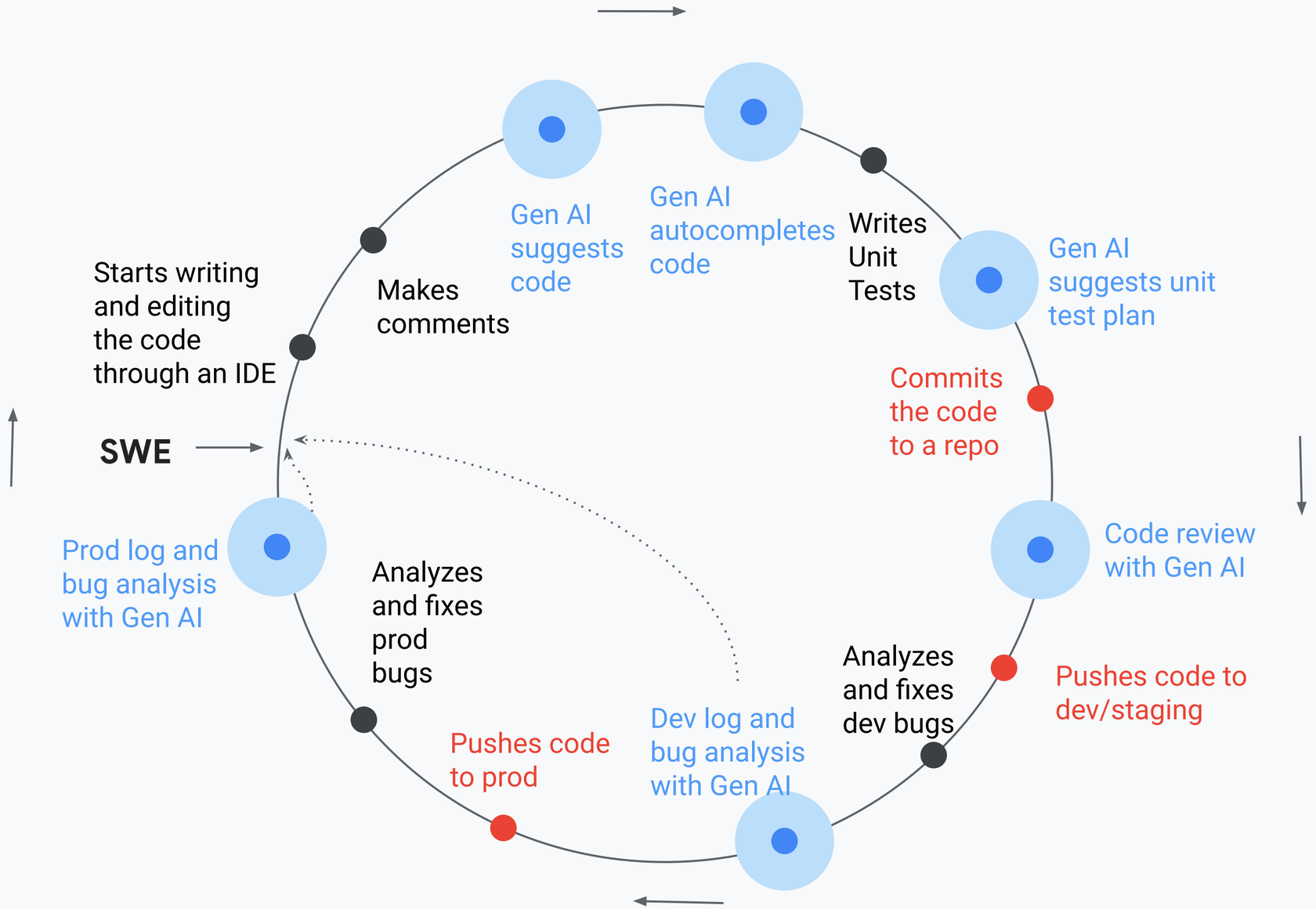
- Increase employee satisfaction & retention
- Reduce time spent on rote tasks
- Attract top engineering talent

Revenue

- Accelerate product innovation
- Monetize new IP
- Increase brand value

Source: [1] McKinsey & Company 2023

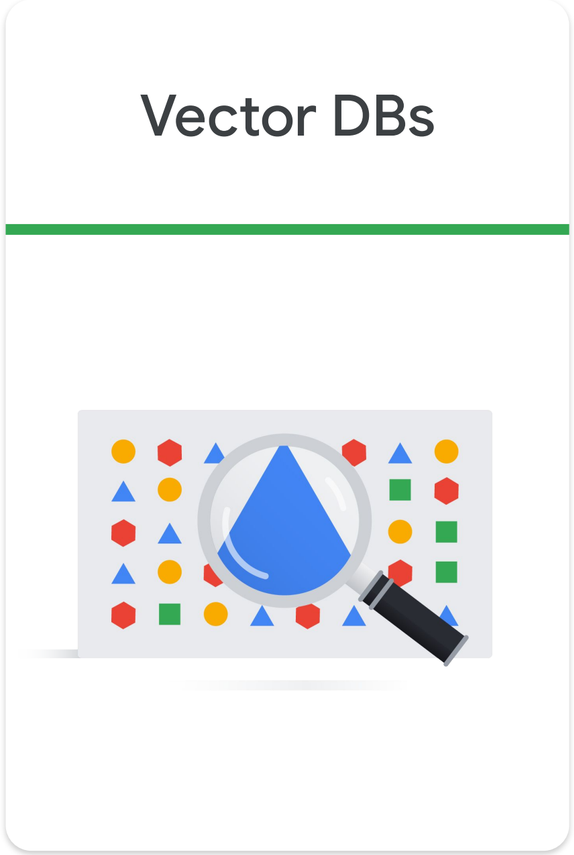
Workflow with Gemini Coding Assistant



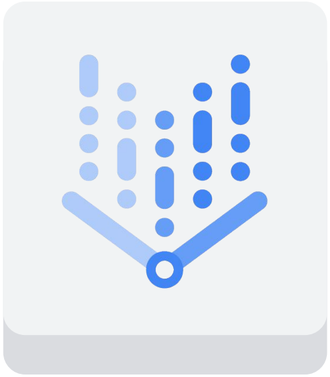
Embeddings & Vector Databases

Storing embeddings in vector databases

- Vector databases are optimized for fast and efficient vector storage and similarity search
- Vector database options include:
  - Vertex AI Vector Search
  - Chroma DB
  - Faiss (Facebook AI Similarity Search)
  - Pinecone
- Store the embedding with a key that links back to the original data
  - Firestore for example



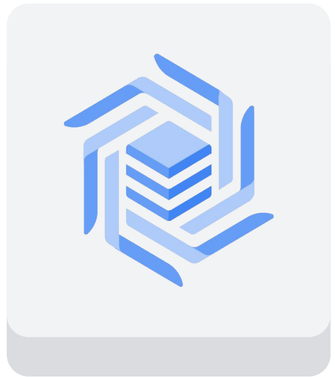
Vertex AI Vector Search



pgvector for Cloud SQL

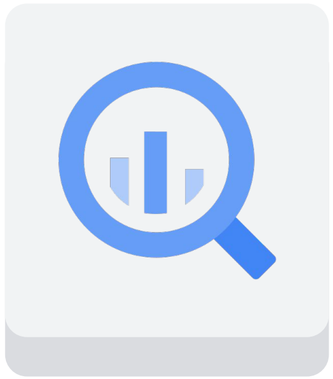
Cloud Spanner

Cloud Bigtable



AlloyDB and AlloyDB AI

BigQuery



Memorystore

Firestore

Choosing an embeddings storage solution

Vertex AI Vector Search

- Use in addition to a database
- Better for bigger datasets
- Result filtering with tags
- Supported index distance functions:
  - Dot product distance (default)
  - Euclidean L2 distance
  - Cosine distance
  - Manhattan L1 distance

Others

- Collocate data and embeddings
- Smaller datasets already in PostgreSQL
- Generate embeddings with textembedding-gecko from SQL
- Hybrid search with Postgres full-text search
- Supported index distance functions:
  - Dot product (or inner product)
  - Euclidean L2 distance
  - Cosine distance

Agent Builder

Agent Builder allows you to build four types of generative AI-powered apps in a few clicks

Select app type

Select the type of application you want to create

Search

Get quality results out-of-box and easily customize the engine

SELECT

Chat

Answer complex questions out-of-the-box

SELECT

Recommendations

Create a content recommendation engine

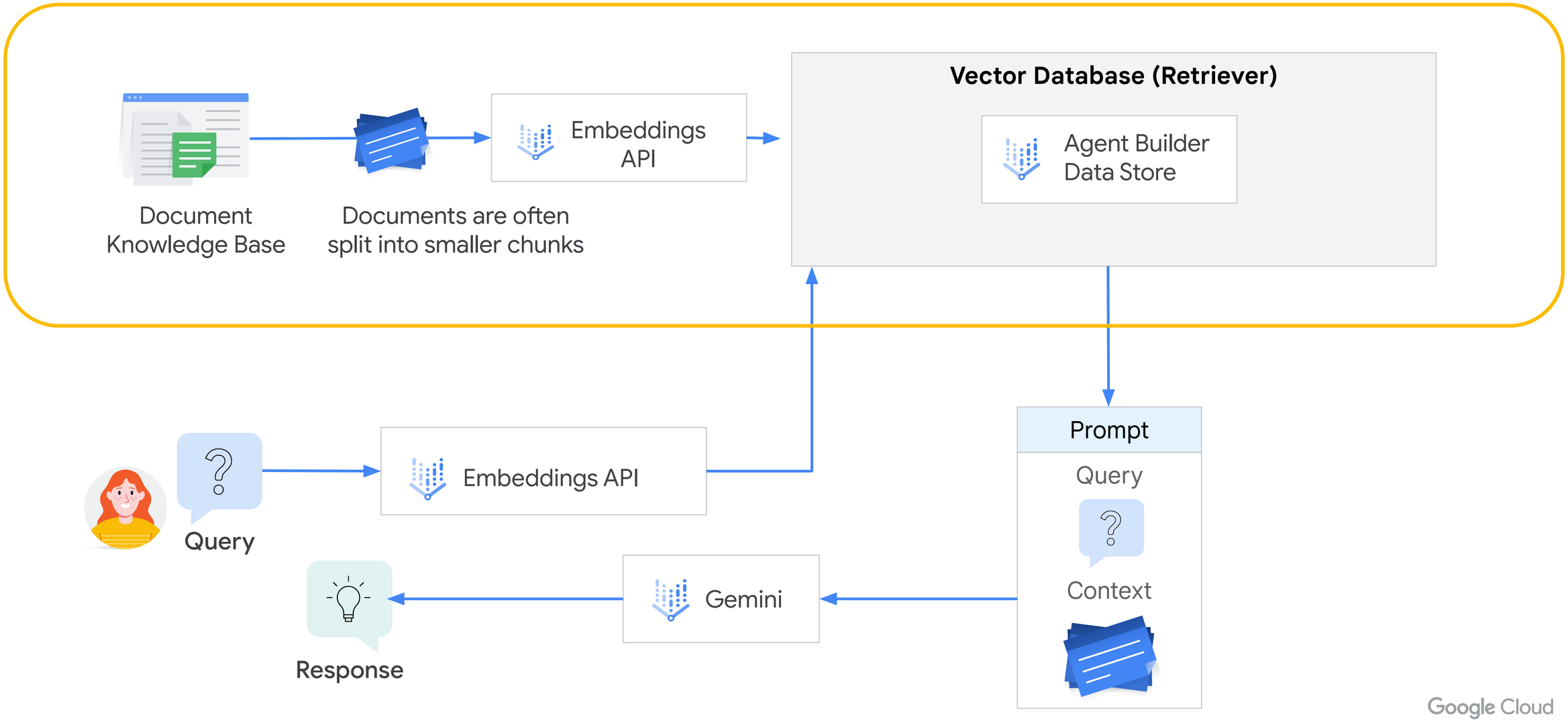
SELECT

Agent PREVIEW

Built using natural language, agents can answer questions from data, connect with business systems through tools, and more

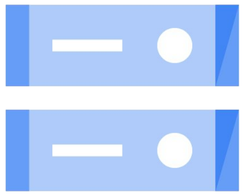
SELECT

Agent Builder Data Stores



Data stores allow your apps to respond based on content from many sources

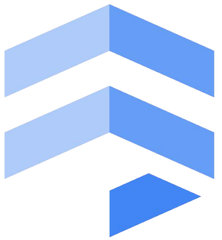
Cloud Storage



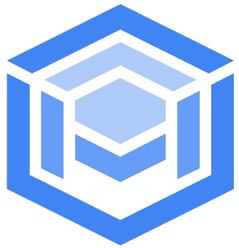
BigQuery

Website Content

Firestore

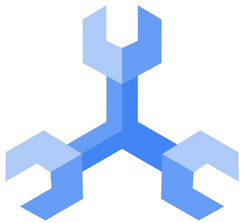


AlloyDB and AlloyDB AI



Cloud SQL

Cloud Spanner



Google Drive, APIs & more

LangChain and Google Cloud

Introducing LangChain

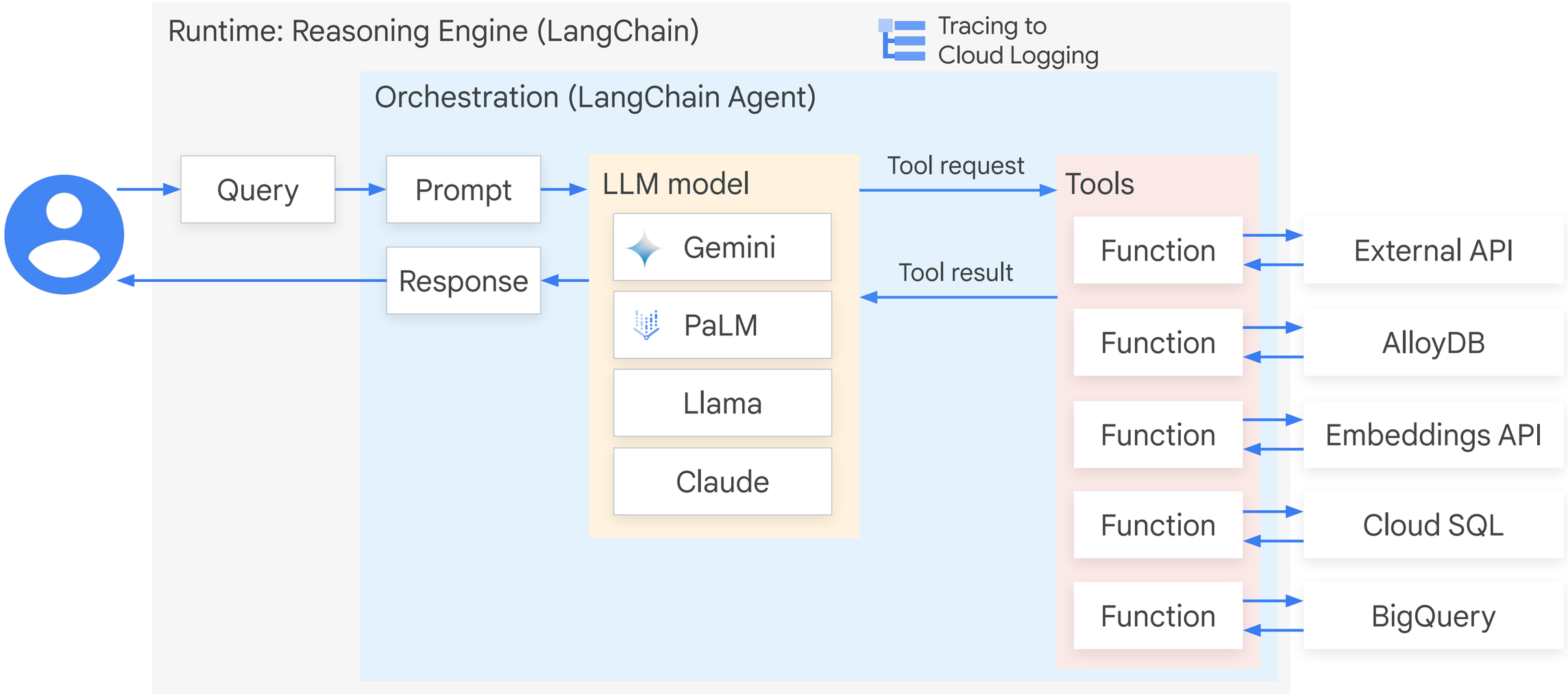
- An Open Source modular framework
  - Chatbots and virtual assistants
  - Text generation and summarization
  - Document question answering
- Relies on language models to reason
- Connects a LLM to sources of context
- Combines components into complex workflows

LangChain

LangChain Google Community components include integrations with various Google services

Let’s take a quick Look at the Google Integrations

Reasoning Engine is a managed runtime for LangChain Agents as APIs with Cloud Logging observability



Firebase Genkit

To use Vertex AI models in a web or mobile app, investigate using Firebase GenKit

- Designed for app developers to integrate generative AI models into Firebase web or mobile applications
- Currently supports JavaScript/Typescript (Node.js) with Go support in active development

Firebase Genkit

```
import { gemini15Flash } from '@genkit-ai/vertexai';
import { generate } from '@genkit-ai/ai';

const result = await generate({
 model: gemini15Flash,
 config: { temperature: 0.3, maxOutputTokens: 200 },
 prompt: 'What makes you the best LLM out there?',
});
console.log(result.text());
```

Introducing Firebase Genkit

- Open source modular framework
- Designed for developers to create apps that:
  - Generate custom content
  - Use semantic search
  - Handle unstructured inputs
  - Answer questions with your business data
  - Autonomously make decisions
  - Orchestrate tool calls
- Supports server-side development with Go support

Firebase Genkit

Different starting points for different personas

Vertex AI

Train predictive ML models, tune Gen AI models, and build apps from scratch.

AI/ML Engineers

Model training/tuning
Advanced ML skill
Some coding

Vertex AI Agent Builder

Build complete, deployable Gen AI apps following common patterns with no or little code.

Application Developers

Fast and easy setup
No ML skill required
No code / low code / code-first

Reasoning Engine / Genkit

Integrate off-the-shelf or tuned models into your custom, complex applications in Node.js or Go.

Enterprise Developers

Customization
Some ML skill
Coding required

Vertex AI
Generative AI Evaluation Service

Generative AI evaluation service:

A suite of tools with different approaches to evaluating GenAI models

Two Evaluation Paradigms

Pointwise evaluations

evaluate a single model on metrics you choose.

Pairwise evaluations

compare two models to select a preferred one.

Generative AI evaluation service:

A suite of tools with different approaches to evaluating GenAI models

Two Types of Metrics

Computation-based metrics

compare a model's output to ground truth.

Model-based metrics

use an autorater model to evaluate another model's output.

These paradigms and metrics combine into a few flavors of evaluation services:

API-based

Rapid evaluation

An API providing low-latency evaluations (computation-based or model-based) pointwise evaluations of a model's results. Designed for evaluating small batches of data while tweaking prompts, model parameters, etc.

Pipeline services. Larger batch evaluations run on Vertex AI Pipelines, of either:

Computation-based evaluations for pointwise evaluation on larger batches of responses.

AutoSXS (auto side-by-side) pairwise evaluations to compare two models with rationales for why they are chosen.

A variety of provided metrics can help evaluate summarization, text generation, tool use, and more

View the metric bundles. Some examples are below. Note that some require ground-truth input ("Reference"), and others do not (using model-based evaluation instead).

Metrics bundle name	Metric name	User input

`text_generation_similarity`	`exact_match`
	`bleu`
	`rouge`
		Prediction
		Reference

`tool_call_quality`	`tool_call_valid`
	`tool_name_match`
	`tool_parameter_key_match`
	`tool_parameter_kv_match`
		Prediction
		Reference

`text_generation_quality`	`coherence` `fluency`	Prediction

How are these metrics defined?

View the
Evaluation methods
& metrics documentation
for definitions & guidance.

General text generation

The following metrics help you to evaluate the model's ability to ensure the responses are useful, safe, and effective for your users.

`exact_match` `bleu` `rouge` `coherence` `fluency` `safety` `groundedness` `fulfillment`

The `coherence` metric describes the model's ability to provide a coherent response.

- **Pairwise support**: No
- **Token limit**: 4,096

Evaluation criteria

Evaluation criterion	Description
Follows logical flow	Ideas logically progress with clear transitions that are relevant to the main point.
Organized	Writing structure is clear, employing topic sentences where appropriate and effective transitions to guide the reader.
Cohesive	Word choices, sentence structures, pronouns, and figurative language reinforce connections between ideas.

Labs for Later

Recommended Labs

- Explore and Evaluate Models using Model Garden
- Prompt Design - Best Practices
- Introduction to Function Calling with Gemini
- Use Embeddings to Cluster Products Based on Descriptions
- Using Vertex AI Vector Search and Vertex AI Embeddings for Text for StackOverflow Questions
- Using BigQuery Embeddings in a RAG Architecture
- Multimodal Retrieval Augmented Generation (RAG) using the Vertex AI Gemini API
- Using Vertex AI Search as a RAG (Agent Builder)
- Introduction to LangChain with Vertex AI
- Document Question Answering with Vertex AI Search and LangChain
- Measure Gen AI performance with Rapid Evaluation
- Deploy and Secure a GenAI Web Application

Questions and answers


Google Cloud