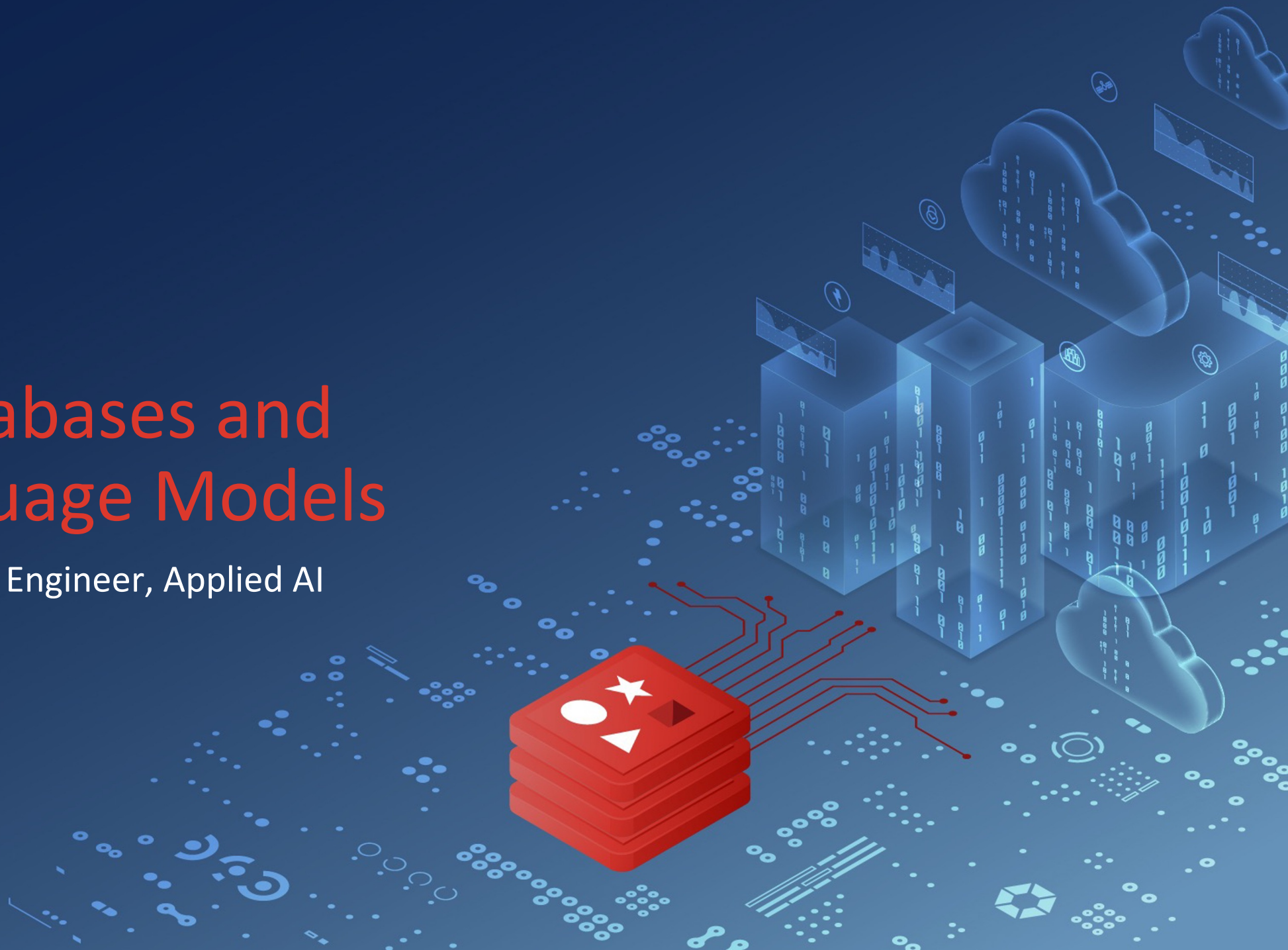
Vector Databases and Large Language Models

Sam Partee – Principal Engineer, Applied AI



Large Language Models

What are they and what do they do?

Application Development

LLM Providers

- LLMs are massive, general purpose neural networks, pre-trained on large amounts of text.
- Specifically focused on language understanding and generation (GPT, BERT, LLAMA).
- Commonly utilize **vector similarity search** to retrieve information from external databases
- Use Cases:
  - Translation
  - Sentiment Analysis
  - Content Generation/Summarization
  - Question Answering

Vector Embeddings

What are vector embeddings and how are they created?

Vectors

- Commonly represent unstructured data
  - Audio, text, images, etc
- Usually of high-dimension in the form of a **dense** embedding.
- Packed with information
- Easy to use API to create

Vector Embedding Creation

- Simple creation APIs
- Example with HuggingFace Sentence Transformer

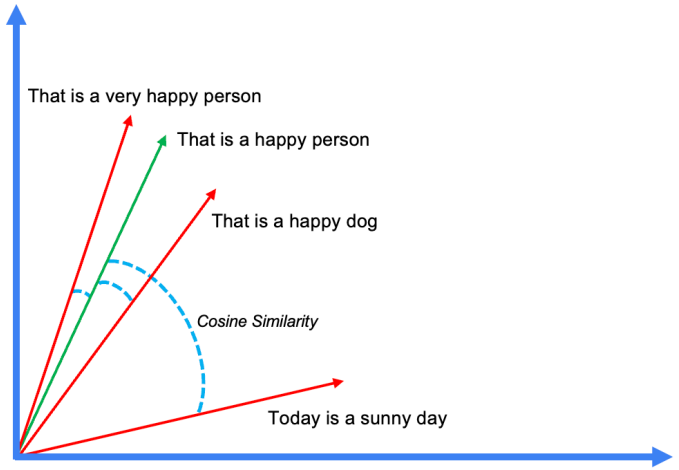
```python
from sentence_transformers import SentenceTransformer
model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2')

sentences = [
  "That is a very happy Person",
  "That is a Happy Dog",
  "Today is a sunny day"
]
embeddings = model.encode(sentences)
```

Vector Similarity Search

How are vector embeddings used for similarity search?

- 3 semantic vectors = **Search Space**
  - “today is a sunny day”
  - “that is a very happy person”
  - “that is a very happy dog”
- 1 Semantic vector = **Query**
  - “That is a happy person”

Goal: Find most similar vector to the query



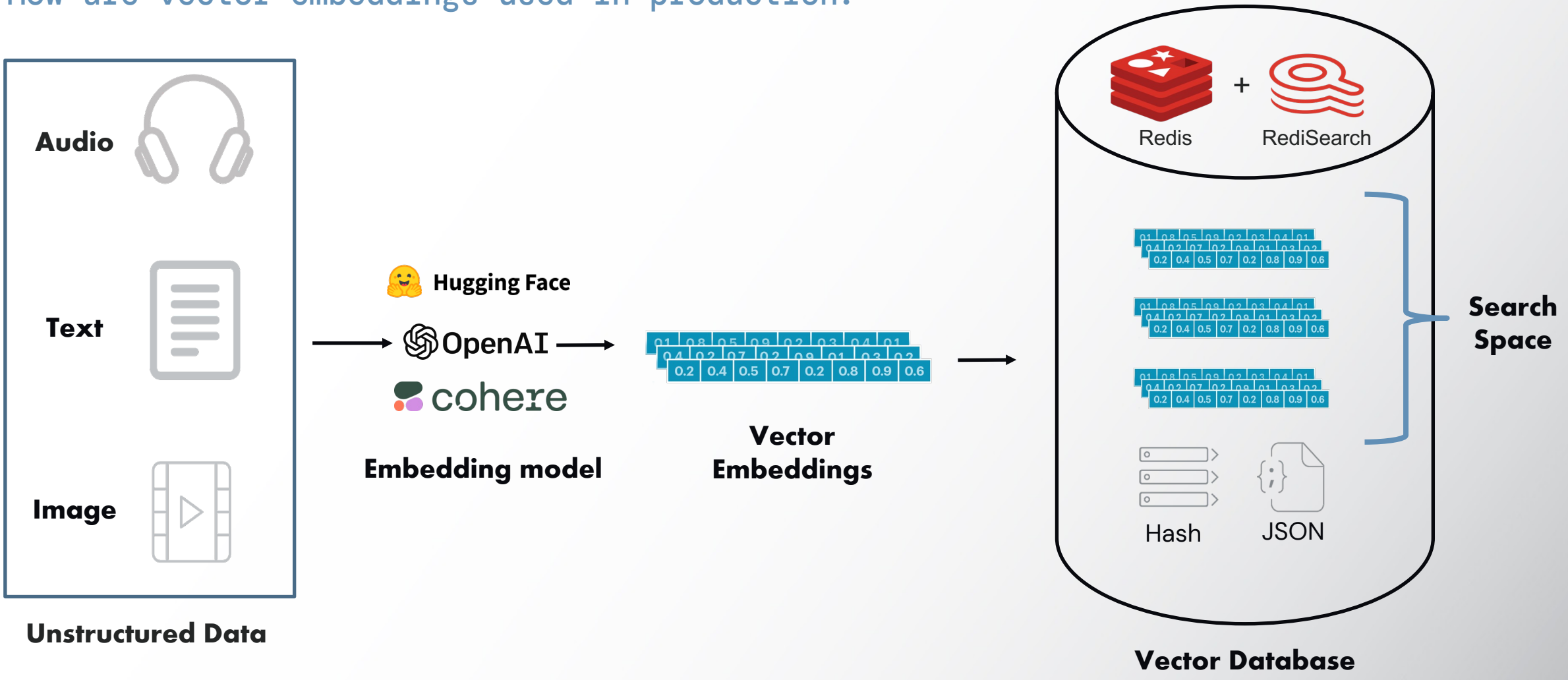
How? Calculate the distance (ex. Cosine Similarity)

That is a happy dog	0.695
That is a very happy person	0.943
Today is a sunny day	0.257

https://mlops.community/vector-similarity-search-from-basics-to-production/

Vector Database

How are vector embeddings used in production?



Redis – Vector Database

Redis + RediSearch

Integrations

OpenAI Retrieval Plugin

LangChain

metal.

Relevance AI

- **Redis**: Low-latency, scalable, in-memory database
- Indexing methods
  - HSNW (ANN)
  - Flat (KNN)
- Distance metrics
  - L2, Cosine, internal product
- Support for hybrid queries
  - Vector search + filtering by text, geo, etc.
- Store vectors in JSON (new in 2.6)

Design Patterns

For using Large Language Models with Vector Databases

Vector Database + LLM

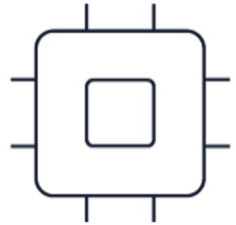
Design Patterns

Context Retrieval

- Search for relevant sources of text from the “knowledge base”
- Provide as “context” to LLM

LLM “Memory”

- Persist embedded conversation history
- Search for relevant conversation pieces as context for LLM

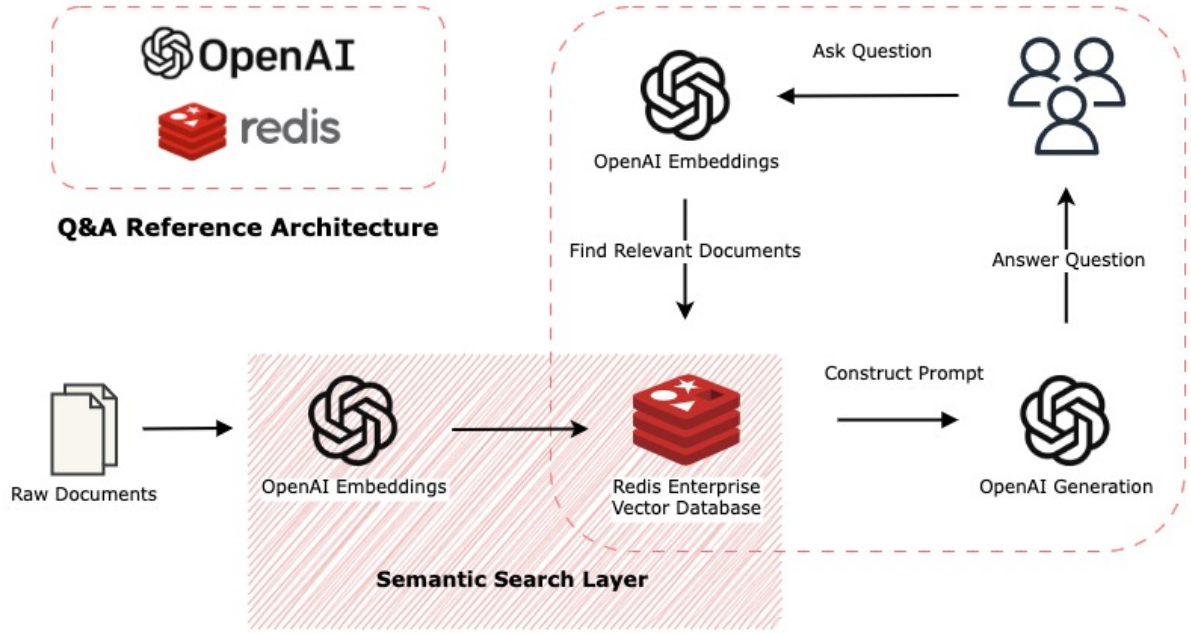


LLM Cache

- Search for semantically similar LLM prompts (inputs)
- Return cached responses

Context Retrieval

Finding relevant information for LLM queries



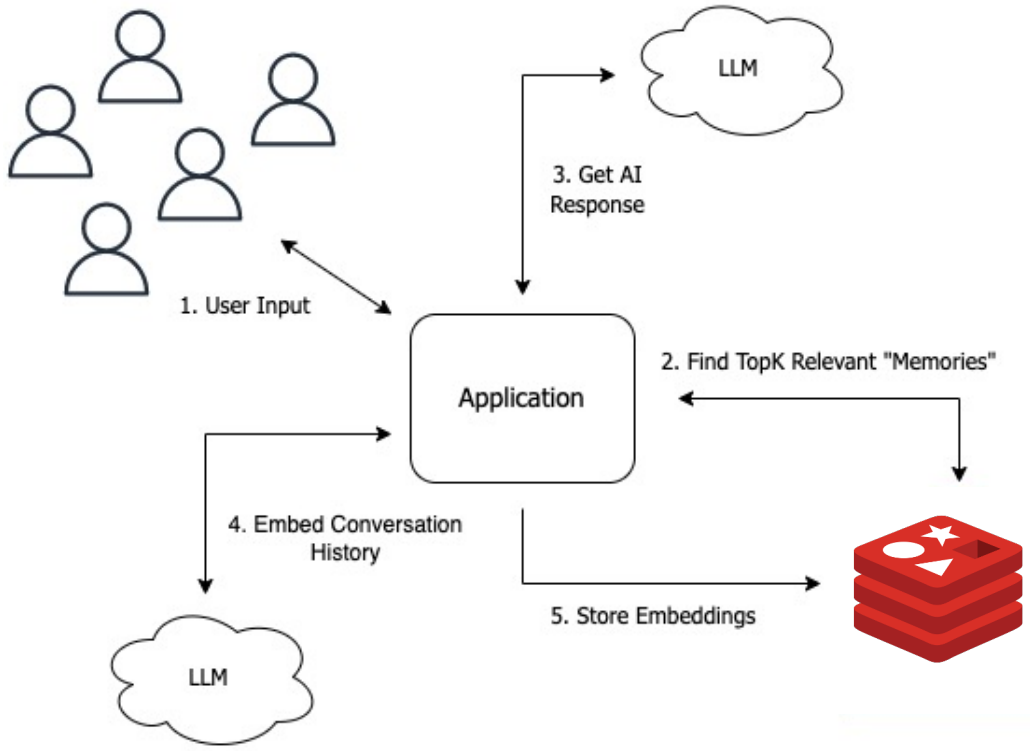
- Description
  - Vector database is used as an external knowledge base for the large language model.
  - Queries are used to detect similar information (context) within the knowledge base
- Benefits
  - **Cheaper and faster** than fine-tuning
  - **Real-time updates** to knowledge base
  - **Sensitive data** doesn't need to be used in model training or fine tuning
- Use Cases
  - Document discovery and analysis
  - Chatbots

Document QnA Example: https://github.com/RedisVentures/redis-openai-qna

Chatbot Example w/ Langchain: https://github.com/RedisVentures/redis-langchain-chatbot

Long-Term Memory

Increasing available context to LLMs

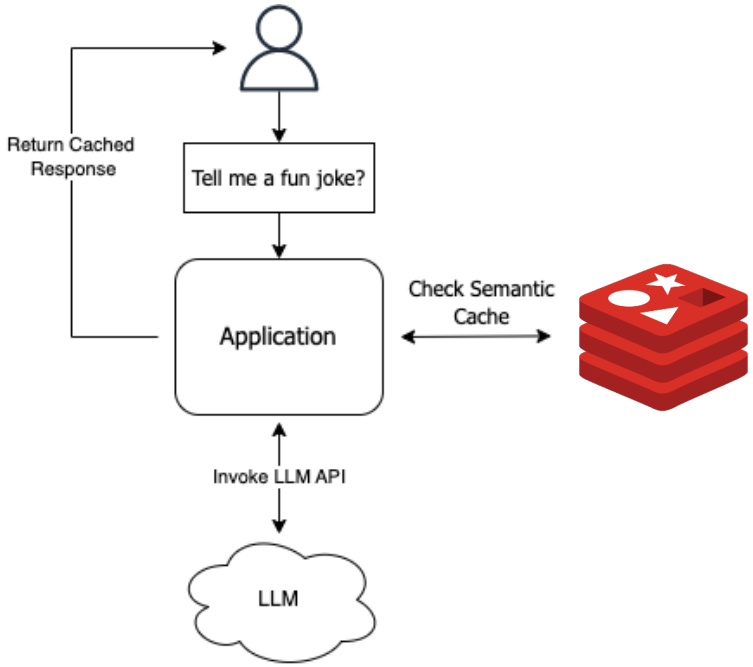


- Description
  - Theoretically infinite, contextual memory that encompasses multiple simultaneous sessions
  - Retrieves only last K messages relevant to the current message in the entire history.
- Benefits
  - Provides **solution to context length limitations** of large language models
  - Capable of **addressing topic changes** in conversation without context overflow
- Use Cases
  - Chatbots
  - Information retrieval

Repository: https://github.com/continuum-llms/chatgpt-memory

LLM Caching

Reducing cost and improving QPS of LLMs



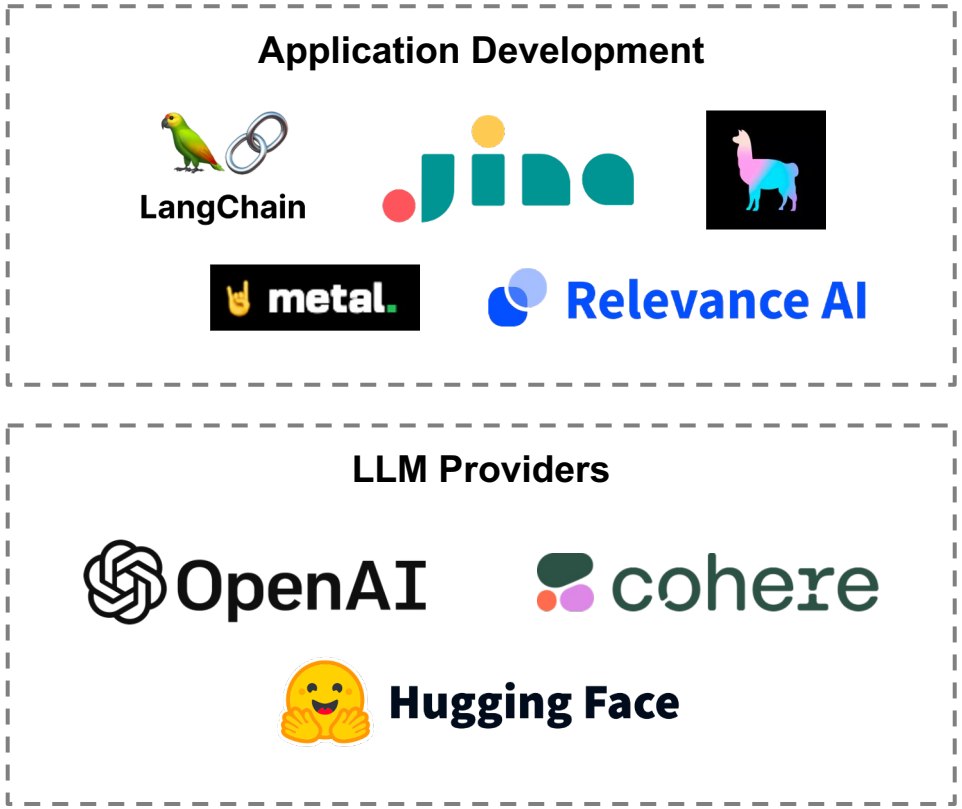
- Description
  - Vector database used to cache similar queries and answers
  - Queries embedded and used as a cache lookup prior to LLM invocation
- Benefits
  - **Saves on computational and monetary cost** of calling LLM models.
  - Can **speed up applications** (LLMs are slow)
- Use Cases
  - Every single use case we've talked about that uses an LLM.

Architecture

Considerations for LLM + Vector database designs

Providers

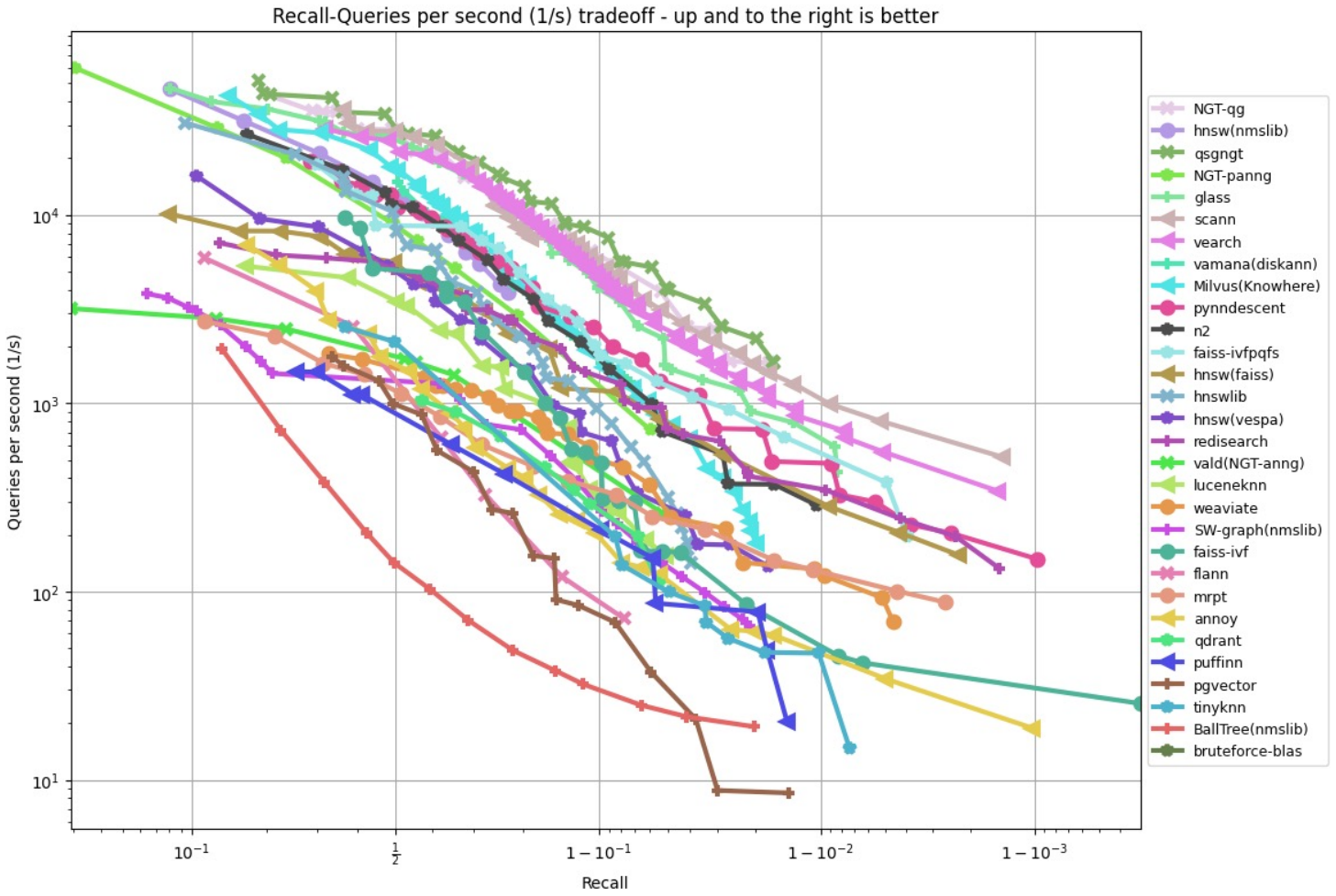
Embedding, integration, model providers



- Questions
  - Open Source vs Closed Source
  - Platform vs roll your own
  - Enterprise readiness
  - Cost
  - Scale and performance
- Integration providers
  - Ex. Langchain vs Relevance
- Model Providers
  - Ex. Huggingface vs OpenAI or Cohere
  - Infra cost vs model cost? Dev time?
- Embeddings Providers
  - Model/embedding manipulation? Fine-tuning?

Vector Database Performance

Queries per Second (QPS) to Recall



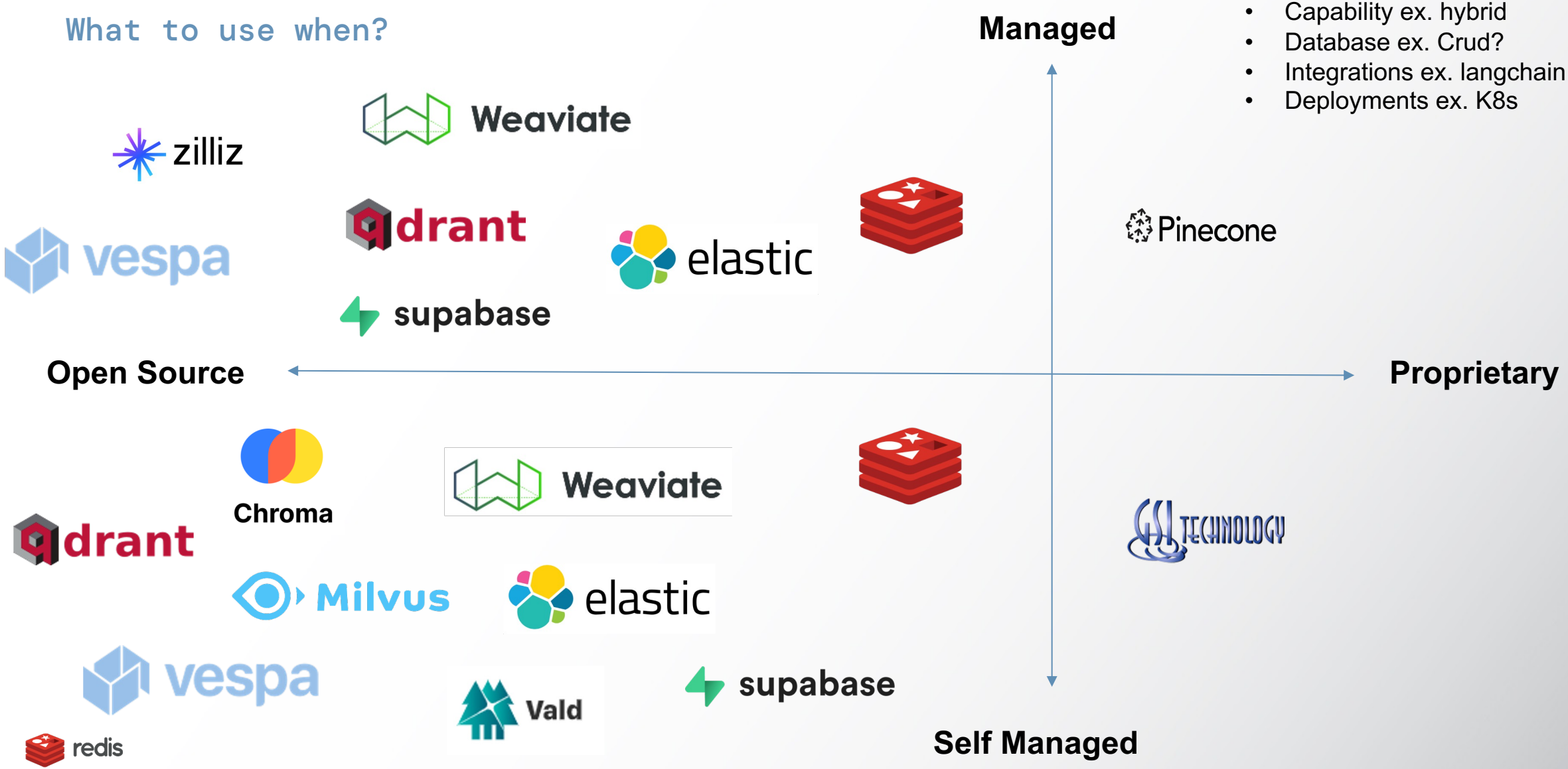
- Performance for ANN benchmarks commonly calculated by QPS/recall
- Importance of this varies by use case
  - Rec-Sys? **Important**
  - Chatbots? Less Important
- Higher in performance usually leads to having less features
  - PQ for scale, but then maintain cookbook
  - Faiss re-training
  - Hybrid search
  - Database featuress

https://github.com/erikbern/ann-benchmarks

Vector Algo? Vector Database?

What to use when?

Considerations
- Team size, ability, budget
- Performance
- Capability ex. hybrid
- Database ex. Crud?
- Integrations ex. langchain
- Deployments ex. K8s

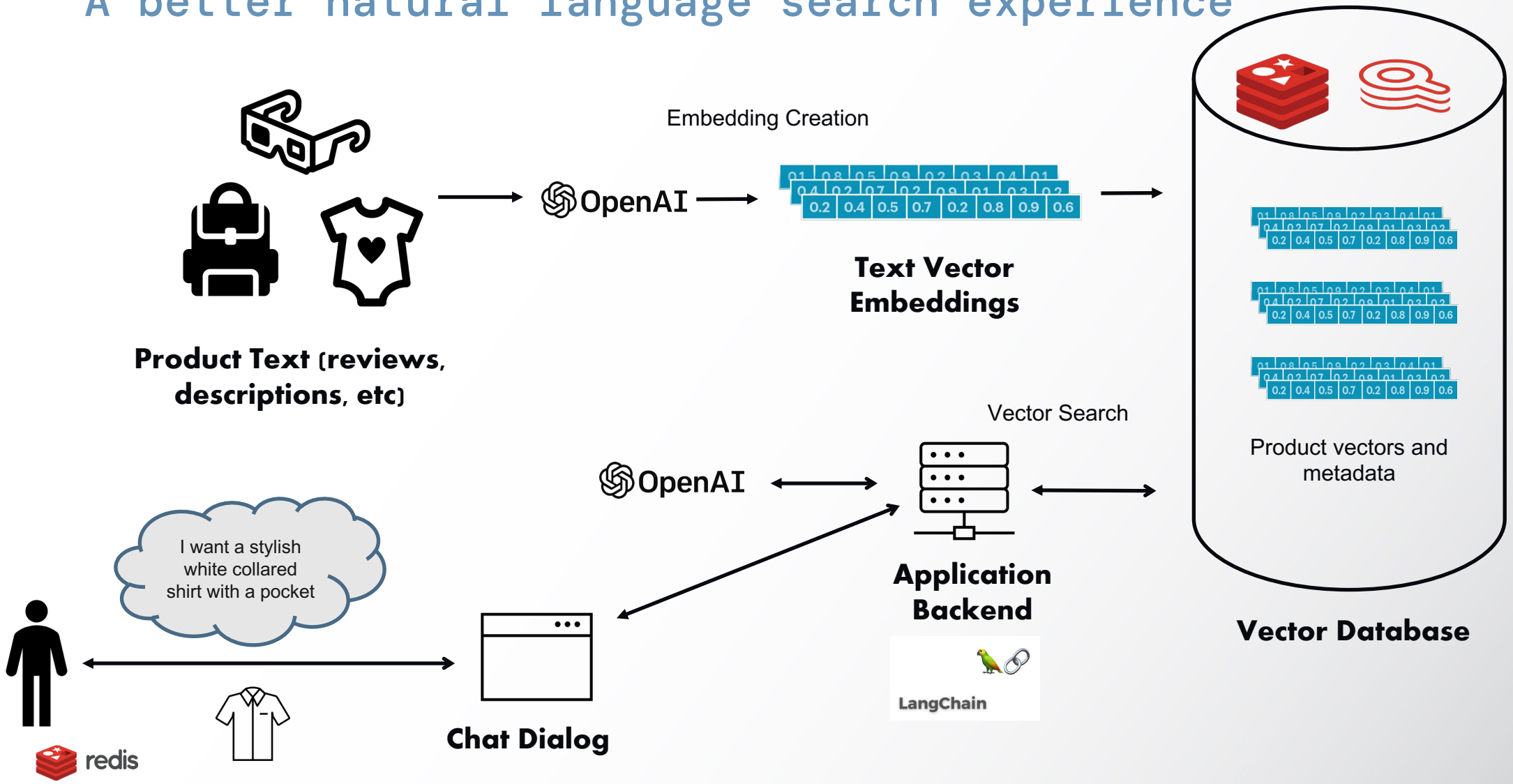


Example Use cases

For joint Vector Database + LLM architectures

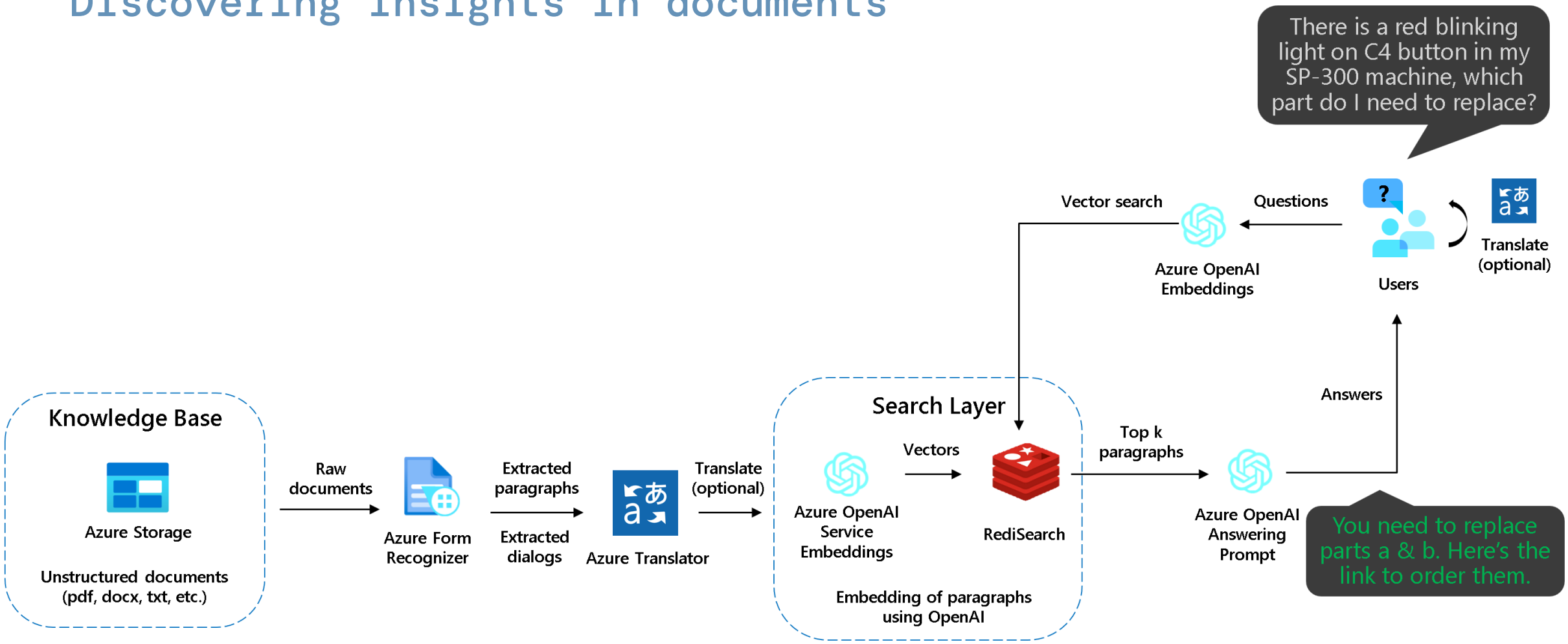
E-commerce Sales Assistant

A better natural language search experience



Document Intelligence + retrieval

Discovering insights in documents



Codebase: https://github.com/ruoccofabrizio/azure-open-ai-embeddings-qna

How to get started?

`Where do I go next?`

- Getting Started Resources
  - OpenAI Cookbook: https://github.com/openai/openai-cookbook/tree/main/examples/vector_databases/redis
  - Redis Ventures: https://github.com/redisventures
- Examples
  - **LLM Document Chat:** https://github.com/RedisVentures/LLM-Document-Chat
  - **OpenAI QnA:** https://github.com/RedisVentures/redis-openai-qna
- More information
  - Follow me @sampartee or add me on linkedin
  - Come to the LLM in Prod part 2 talk June 16th