

Mg. Ing. Ezequiel Guinsburg

ezequiel.guinsburg@gmail.com

Class of 5
chatbot

CLAS

Vector Databases vect and chatbot

creating vector database so can connect Rag:
u build a vector rag database

5 CLAS

A In vector database

Rag / Vector index generation
Rag database

RAG database
rag database

rag code generation
connected database

B database

vector database
database

Ass of vector to create
chatting chatbot

Problems

Performance, accuracy
Performance

Performance, accuracy
Performance

Performance, accuracy
Performance

Performance

Performance

Performance

Performance
Performance

Referencias:

- Paper “IMAGEBIND: One Embedding Space To Bind Them All”
- Paper “Retrieval-Augmented Generation for Knowledge Intensive NLP Tasks”

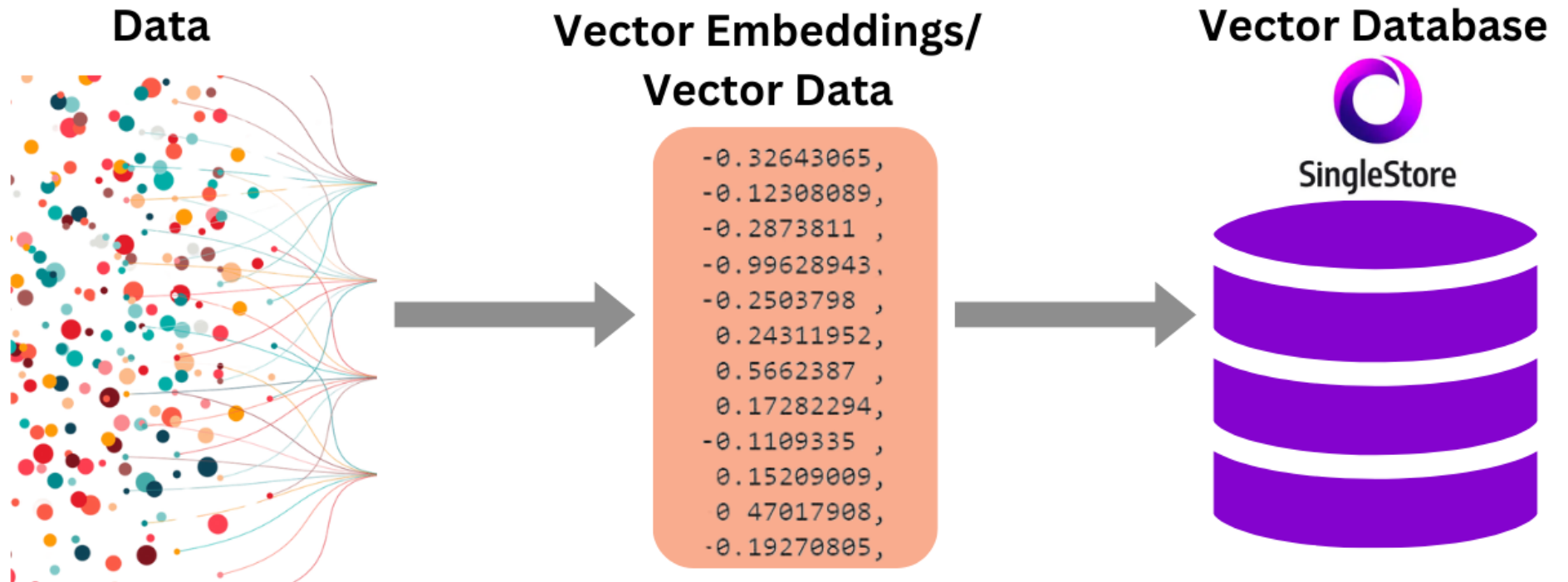
[Link REPO](#)

Temas:

- Bases de datos vectoriales (en contexto RAG)
- Retrieval Augmented Generation (RAG)
- RAG Multimodal
- Chatbots

Bases de datos vectoriales:

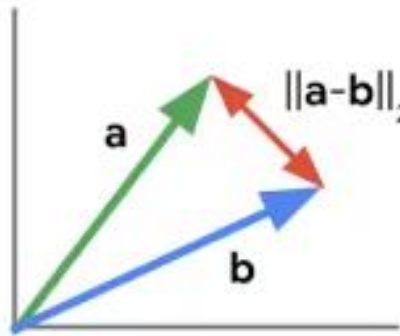
- Características



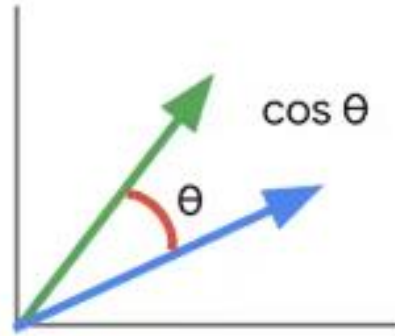
Bases de datos vectoriales:

- Búsqueda por similitud

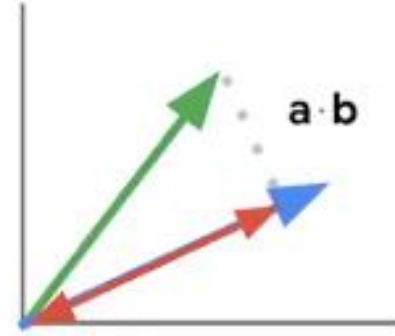
Distancia de Hamming(A, B) = $\sum (A_i \neq B_i)$



L2 distance



cosine similarity



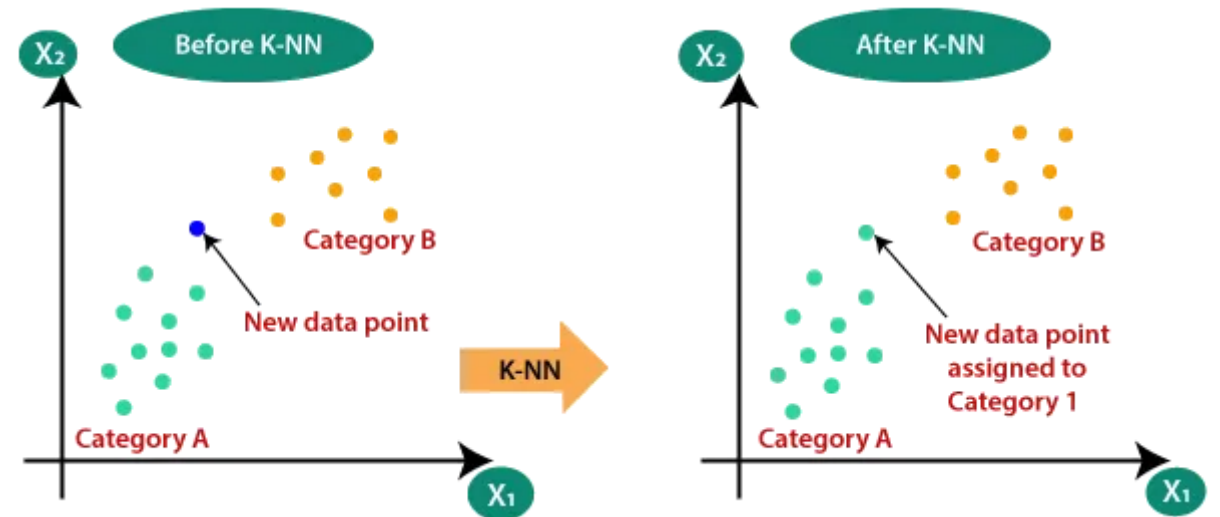
inner product

Bases de datos vectoriales:

- Algoritmos de ordenamiento para búsquedas eficientes:
 - k-dimensional tree
 - Locality Sensitive Hashing (LSH)
 - Faiss (Facebook AI Similarity Search) ([link](#))

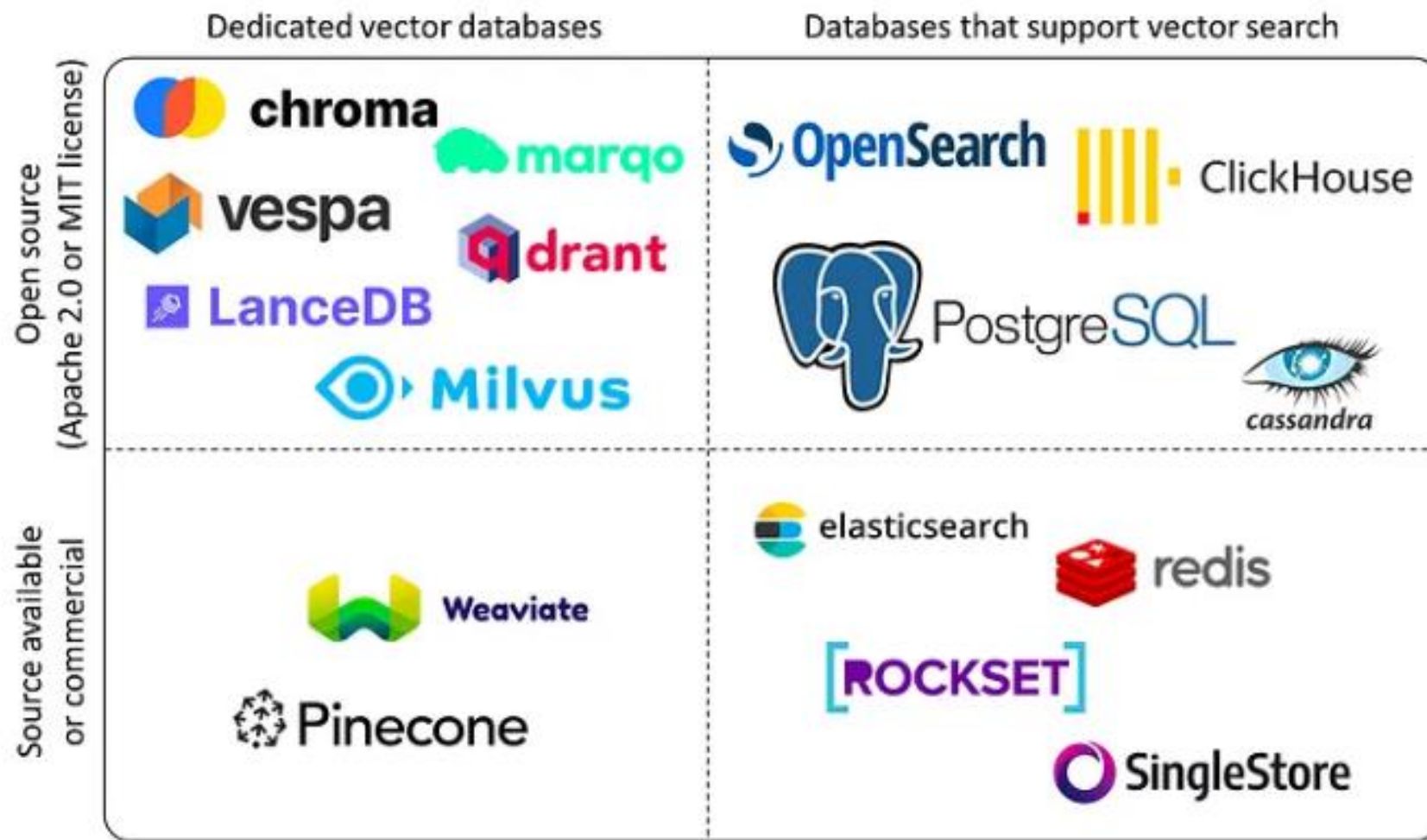


Source: [Medium](#)



Bases de datos vectoriales:

- Panorama

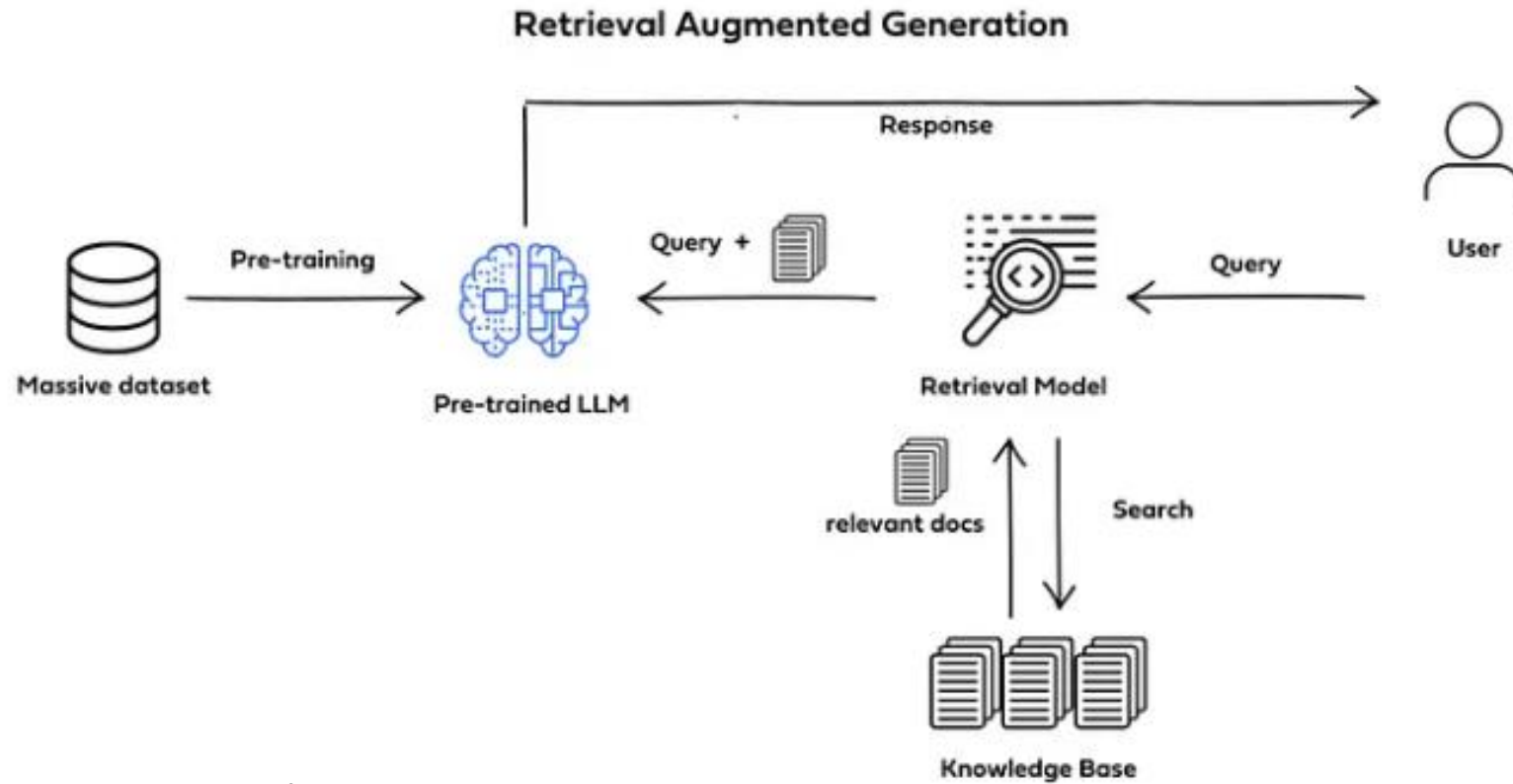


Bases de datos vectoriales:

- Ejemplo

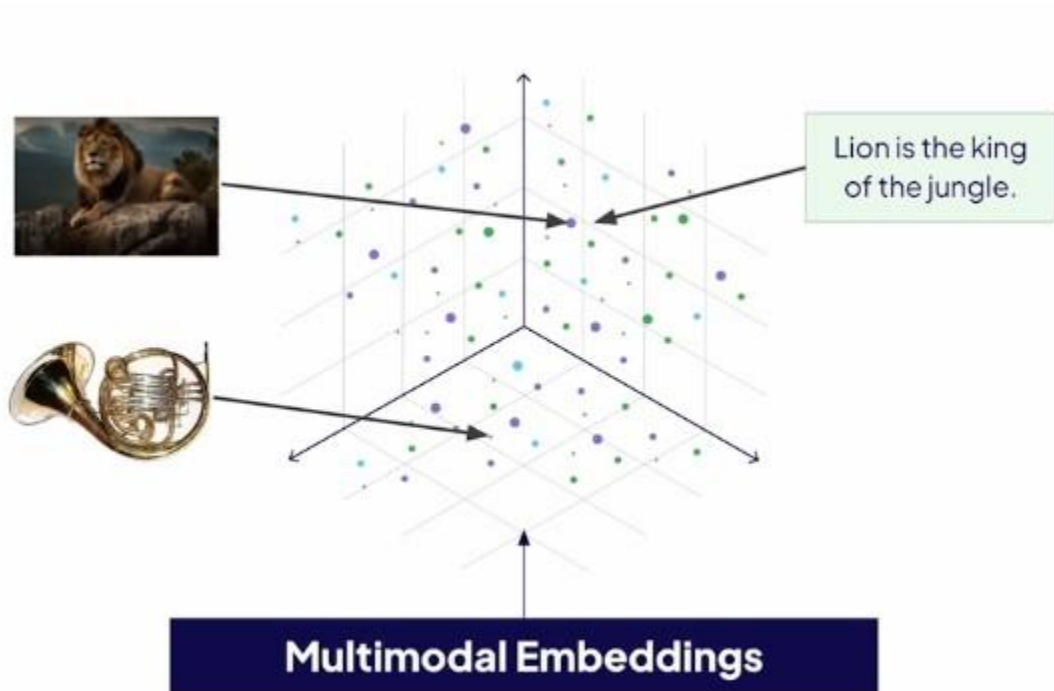
https://colab.research.google.com/drive/1MOMrO3Q_bV53W31BJphLxfhDpCIWKmh9?usp=sharing

Retrieval Augmented Generation (RAG):



Fuente: medium

RAG Multimodal:



The lion is the king
of the savannas.



Text Encoder

$[-0.10, 0.02, 0.14, \dots, 0.48]$



Image Encoder

$[0.66, 0.73, -0.42, \dots, 0.52]$



Audio Encoder

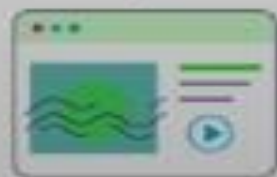
$[-0.61, 0.31, 0.88, \dots, 0.22]$



Video Encoder

$[-0.21, 0.15, 0.27, \dots, -0.59]$

Step 1



Customized
Response

Step 2

RAG
Multimodal:

Chatbots

- Simulación del contexto
- Memoria persistente gestionada programáticamente (Langchain)

Vemos el ejemplo de la implementación de un chatbot paso a paso!

Ejercicio en clase:

- Implementar un sistema de generación de texto (chatbot) que utilice la técnica de **Retrieval-Augmented Generation (RAG)**. En este ejercicio, el chatbot será capaz de recuperar información de una base de datos (o un conjunto de documentos) y usarla para generar respuestas más completas, mejorando la calidad de las respuestas generadas.

Ejercicio en clase :

Pasos

1. Preparación del entorno de trabajo: contar con IDE, cuenta de Pinecone (Starter), cuenta de Groq.
2. Cargar los CVs de los miembros del equipo y obtener los vectores de embeddings (utilizando algún modelo de embeddings de Groq).
3. Cargar los vectores a Pinecone.
4. Probar hacer una pregunta y, por medio de una comparación coseno, obtener el vector más cercano.
5. (BONUS) implementar un simple chatbot para obtener respuestas sobre el documento cargado.