VectorDB and LLM

Semantic information retrieval, long-term memory, and more

Camilo Pestana

Image created with Midjourney:
“Large Language Models and Artificial Intelligence. Machines conversing with humans”

What’s a Vector Database?

“We’re in the midst of the AI revolution. It’s upending any industry it touches, promising great innovations - but it also introduces new challenges. Efficient data processing has become more crucial than ever for applications that involve **large language models, generative AI, and semantic search**.”

“All of these new applications rely on **vector embeddings**, a type of data representation that carries within it semantic information that’s critical for the AI to gain understanding and maintain a long-term memory they can draw upon when executing complex tasks.”

- Pinecone

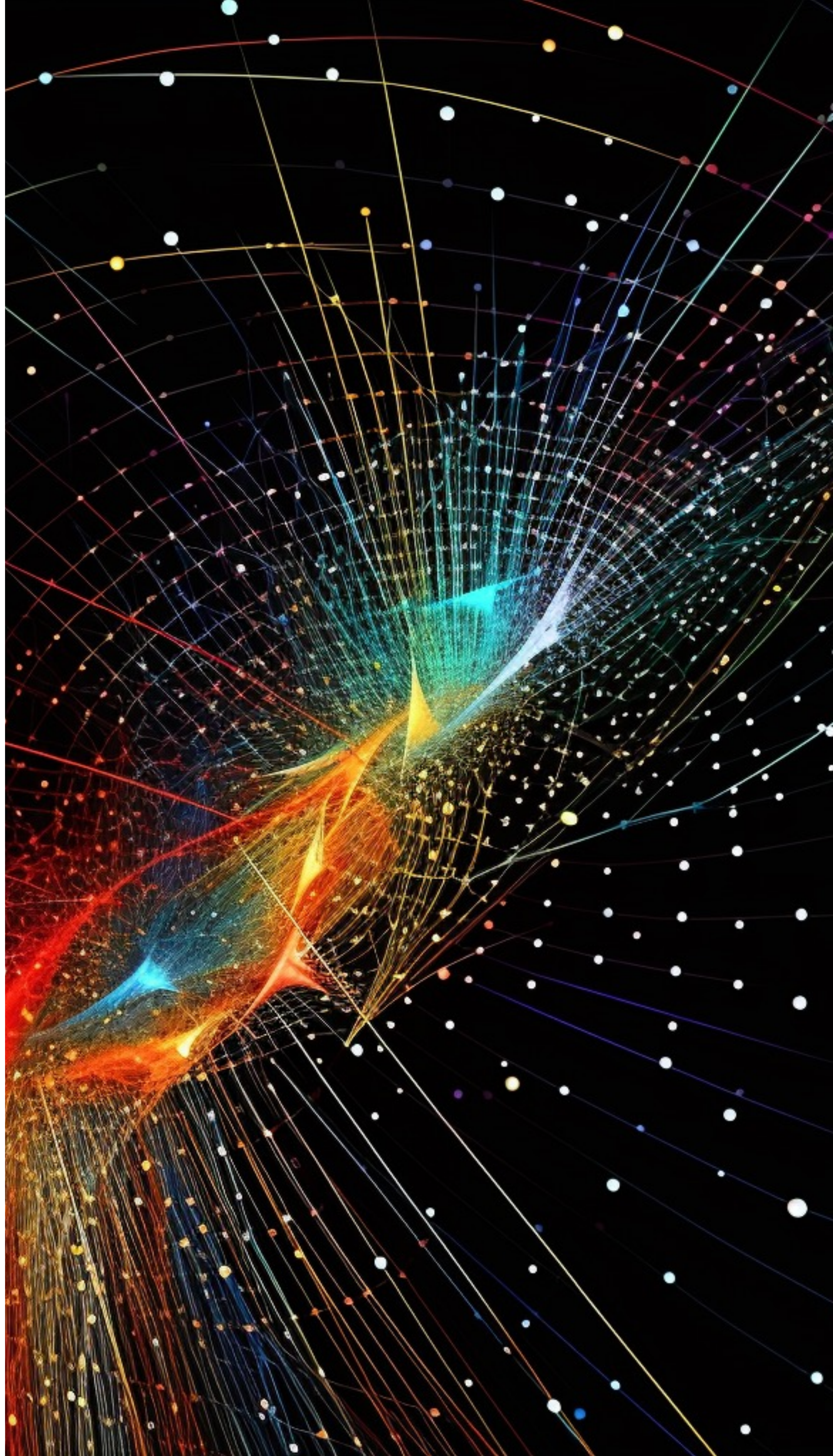
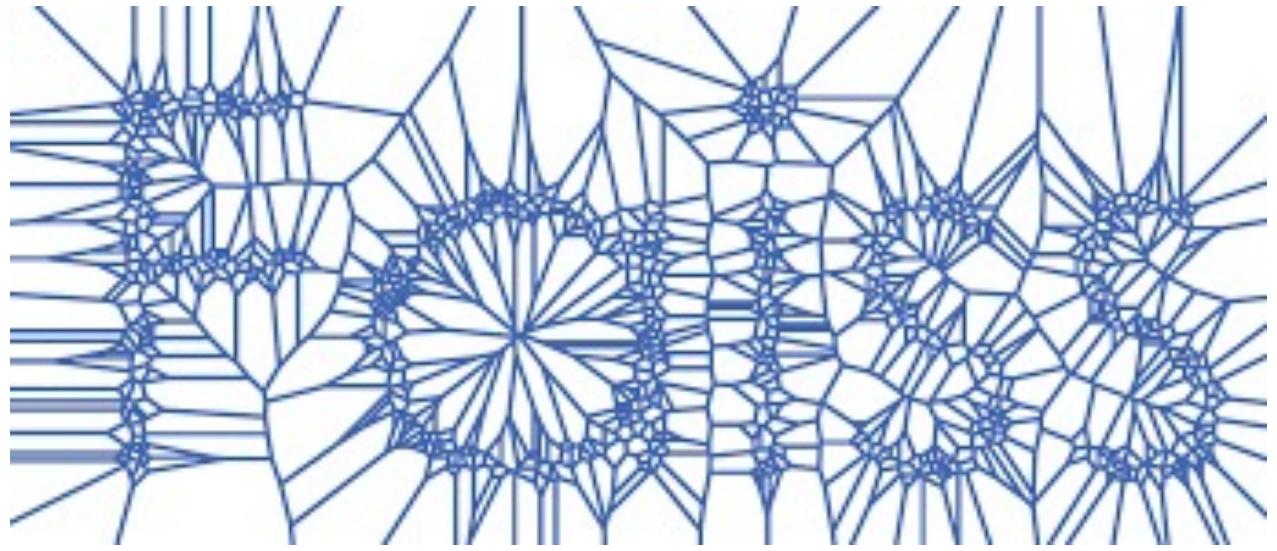


Image created with Midjourney: “vectors in a hyperspace. similarity search and cosine similarity”

VectorDBs and Search Engines



How do VectorDBs work?

1. We use the **embedding model** to create **vector embeddings** for the **content** we want to index.

2. The **vector embedding** is inserted into the **vector database**, with some reference to the original **content** the embedding was created from.

3. When the **application** issues a query, we use the same **embedding model** to create embeddings for the query, and use those embeddings to query the **database** for *similar* vector embeddings. And as mentioned before, those similar embeddings are associated with the original **content** that was used to create them

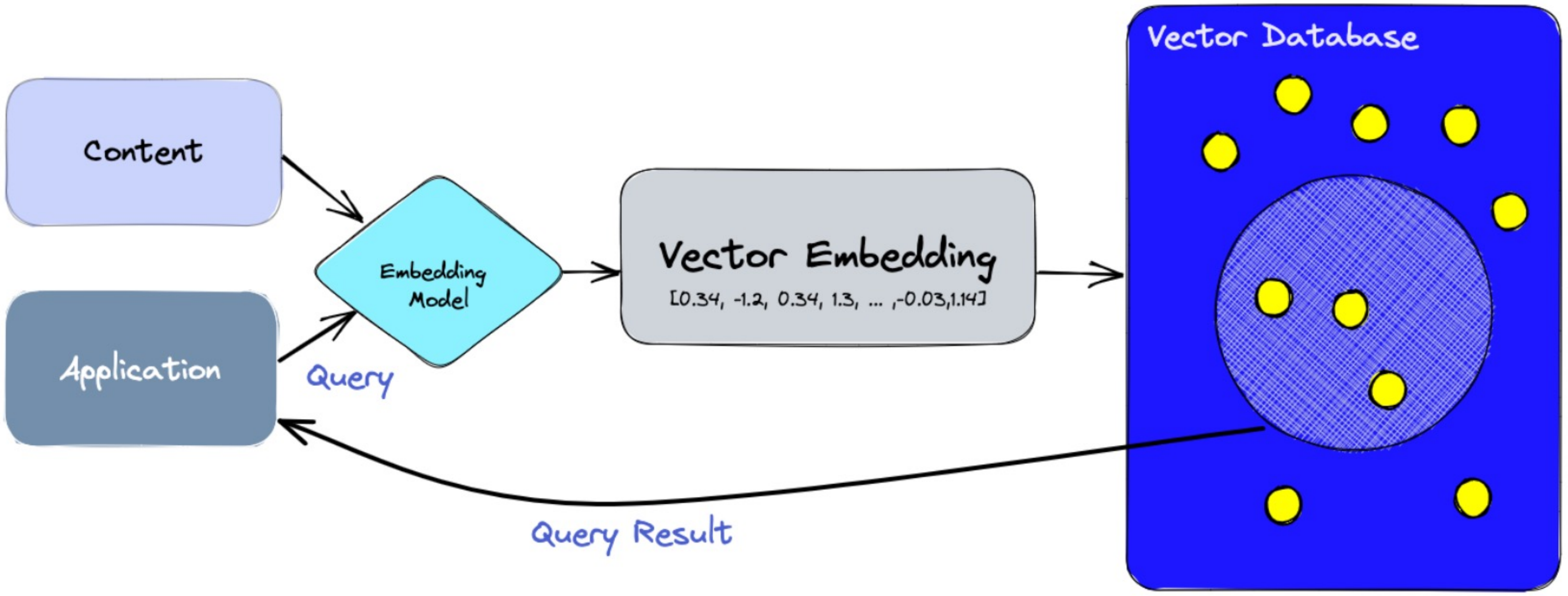


Image from https://www.pinecone.io/learn/vector-database

Vectors in Language Models

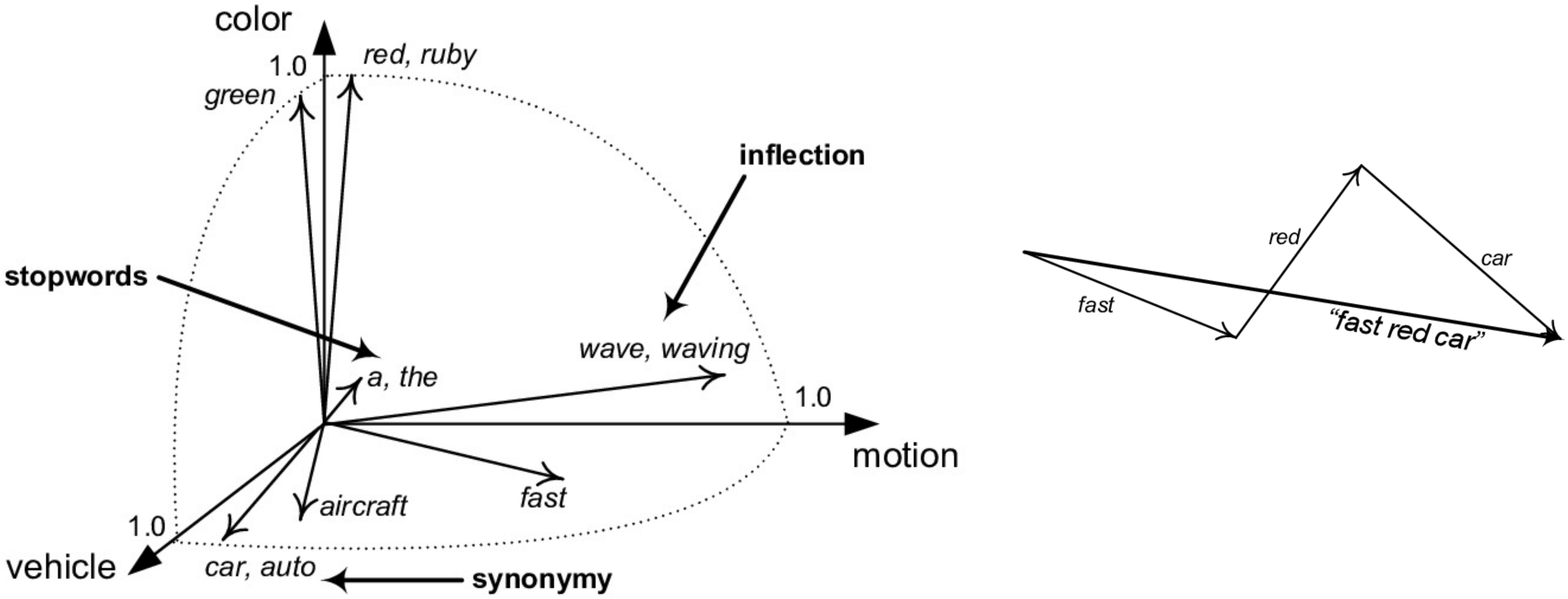
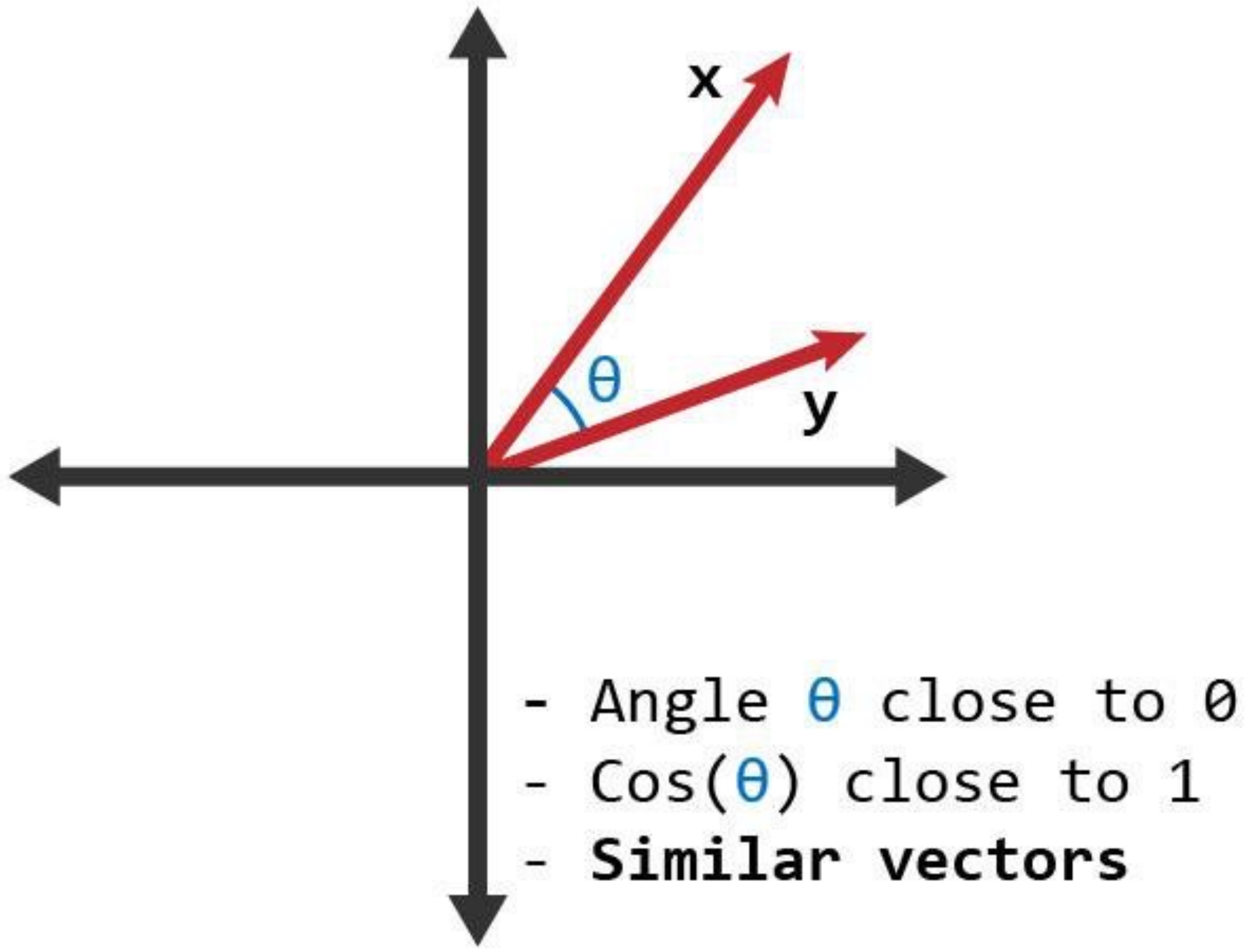


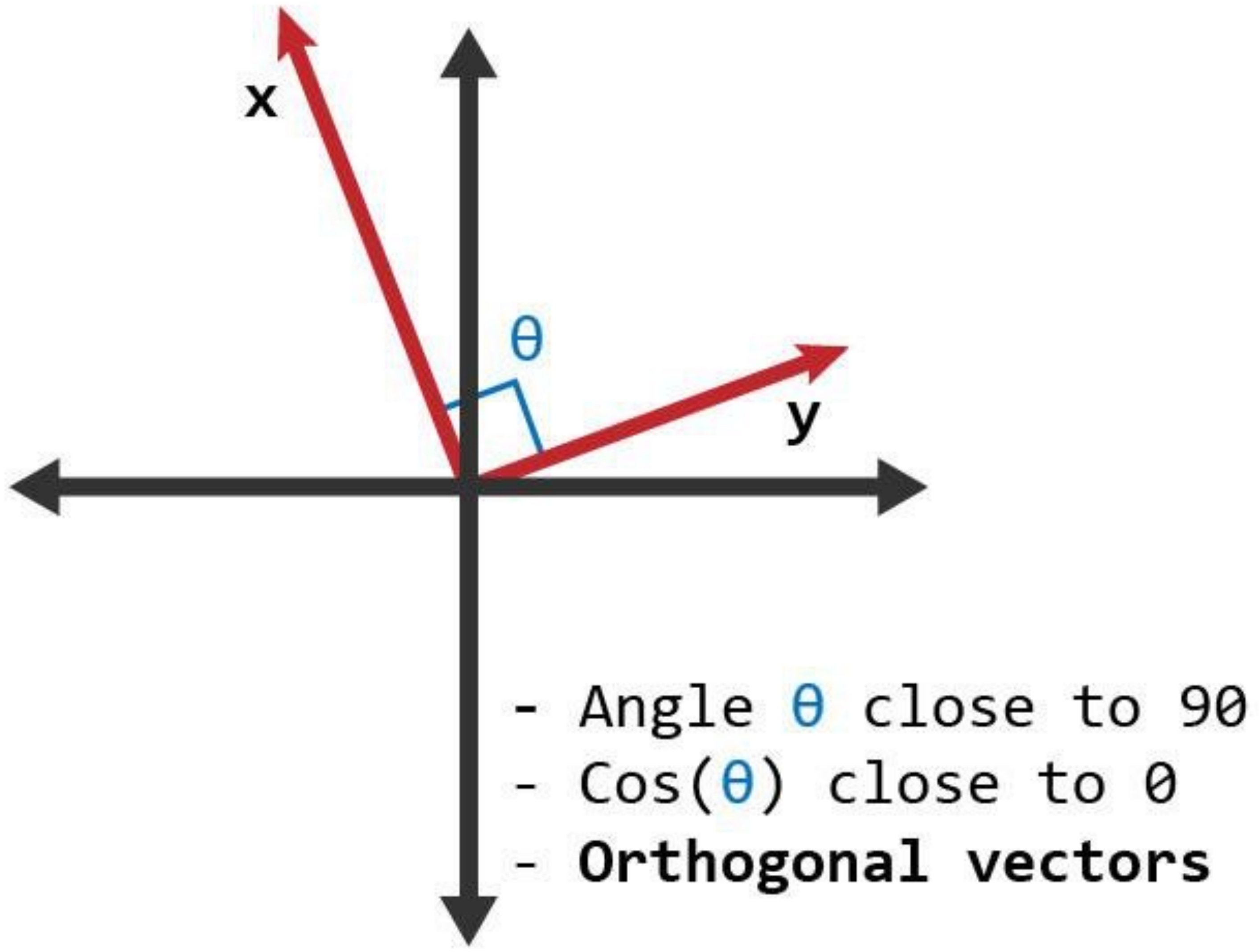
Image from https://www.researchgate.net/figure/TVSM-document-model-visualization_fig1_298215705

Similarity Metrics

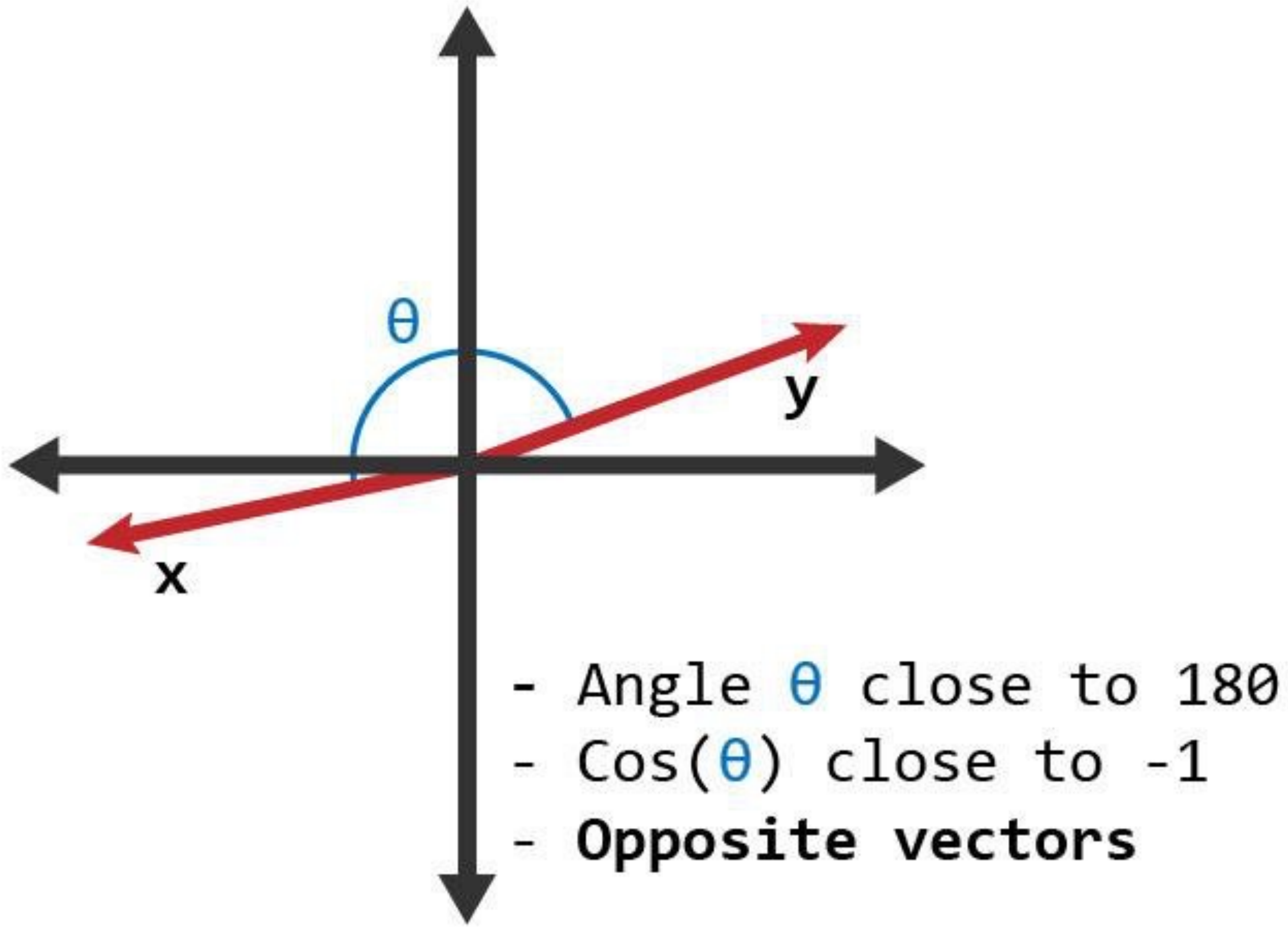
Cosine Similarity



- Angle θ close to 0
- Cos(θ) close to 1
- **Similar vectors**

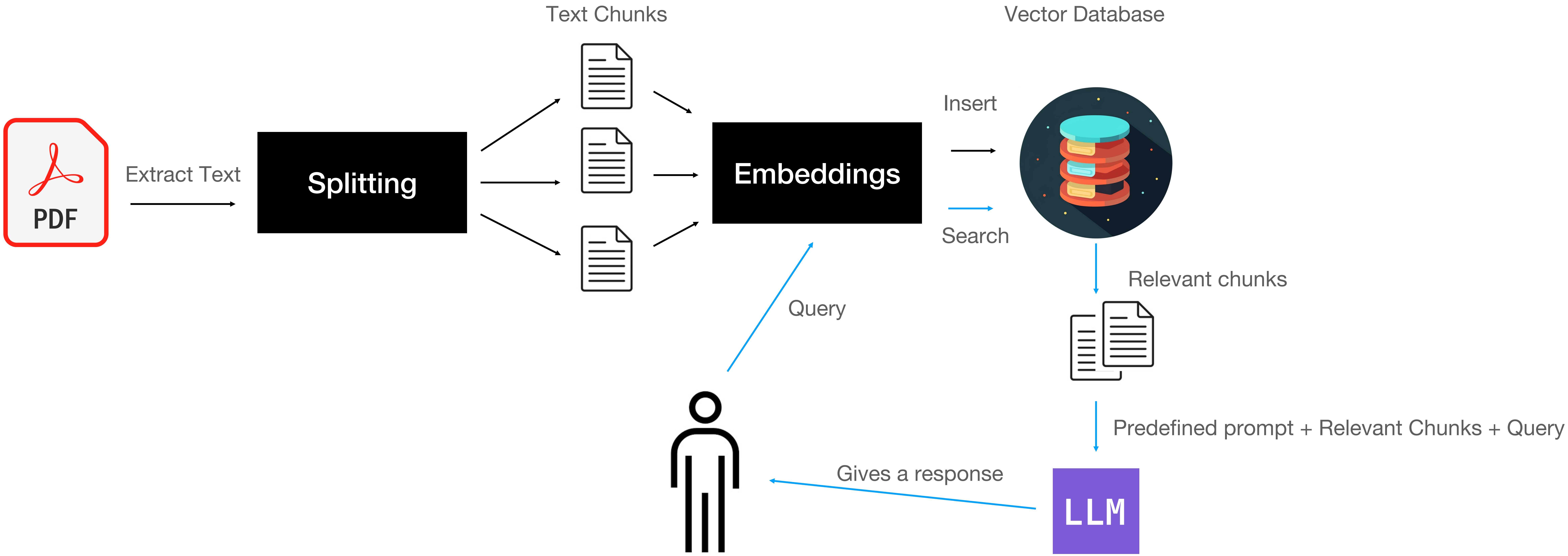


- Angle θ close to 90
- Cos(θ) close to 0
- **Orthogonal vectors**



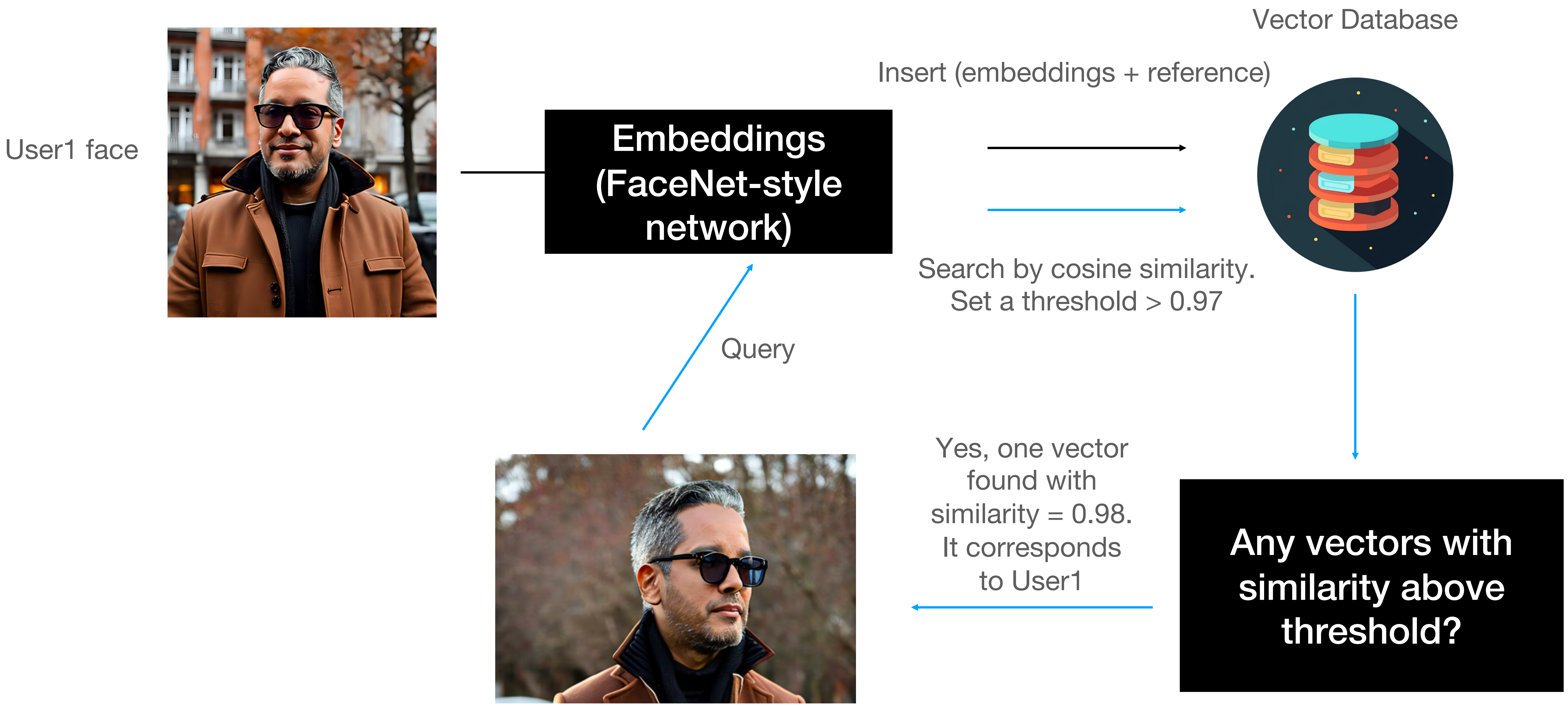
- Angle θ close to 180
- Cos(θ) close to -1
- **Opposite vectors**

Use Case: Document(s) Querying



Other use cases:

Facial Recognition



User1 face

Embeddings (FaceNet-style network)

Vector Database

Insert (embeddings + reference)

Search by cosine similarity.
Set a threshold > 0.97

Query

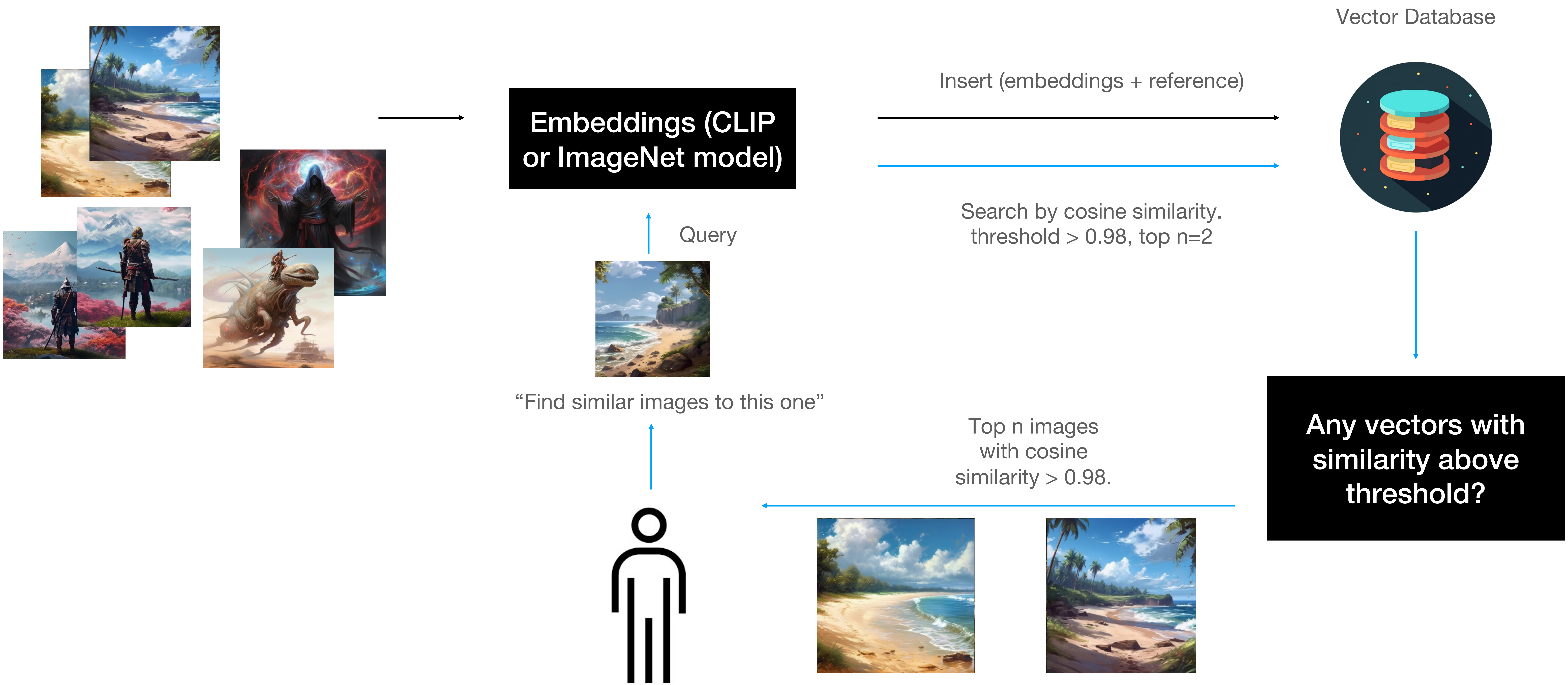
Any vectors with similarity above threshold?

Yes, one vector found with similarity = 0.98. It corresponds to User1

Image created with Stable Diffusion + ControlNet: “a man with fashionable glasses”

Other use cases:

Image Search



Embeddings (CLIP or ImageNet model)

Insert (embeddings + reference)

Vector Database

Search by cosine similarity.
threshold > 0.98, top n=2

Query

“Find similar images to this one”

Any vectors with similarity above threshold?

Top n images with cosine similarity > 0.98.

All Images created with Midjourney

Thanks

GitHub https://github.com/elcronos/ChatDocuments

in https://www.linkedin.com/in/camilopestana/