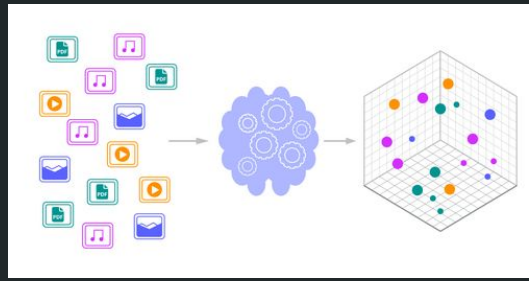
Vector Databases

Mirbaan Balagamwala, Lydia Lazor, Robert McDonald, Mahibah Saleh, Nidhi Tarpara

Image Sources - https://www.datastax.com/guides/what-is-a-vector-database

Introduction

- **Problem:** Growing Need for Efficient Similarity Search
  - Traditional databases struggle with high-dimensional data (e.g., images, text embeddings, audio)
  - Exact keyword matching is insufficient for applications like recommendation systems and semantic search



- When Was the Idea Discovered?
  - Early research on vector search algorithms dates back to the 1990s
  - FAISS by Facebook, Annoy by Spotify, and Milvus emerged in the late 2010s to early 2020s

- **Solution:** Vector Databases
  - Optimized for storing and searching high-dimensional vector data
  - Use Approximate Nearest Neighbor (ANN) search for fast retrieval
  - Enable AI-powered applications

Image Sources - https://airbyte.com/data-engineering-resources/vector-database-vs-graph-database

Introduction

- Vector Database - type of database designed to store, index, and retrieve vector embeddings efficiently
  - HNSW (Hierarchical Navigable Small World), IVF (Inverted File Index), and PQ (Product Quantization) for fast search
  - search engines, recommendation systems, fraud detection, computer vision applications
- Vector Embeddings – represent data points (text, images, audio) in high-dimensional space
- Similarity Search – finds the most similar items based on cosine similarity, Euclidean distance, or dot product

- Bridges the gap between structured data storage (SQL/NoSQL) and unstructured AI-driven search

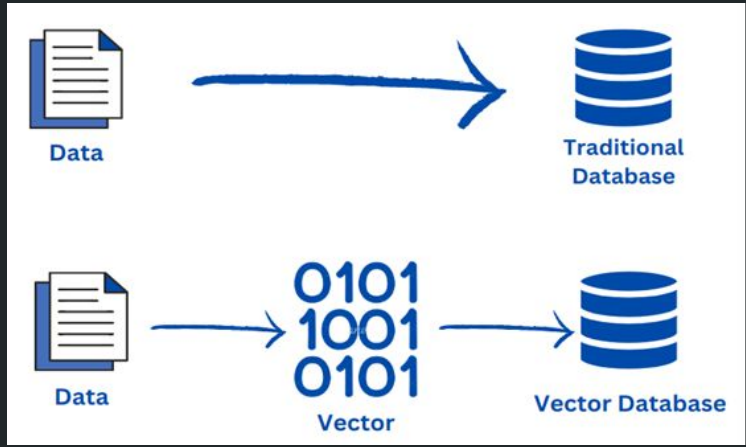


Image Sources - https://www.packtpub.com/en-us/learning/how-to-tutorials/everything-you-need-to-know-about-pinecone-a-vector-database

Product Overview

Key Features & Functionalities

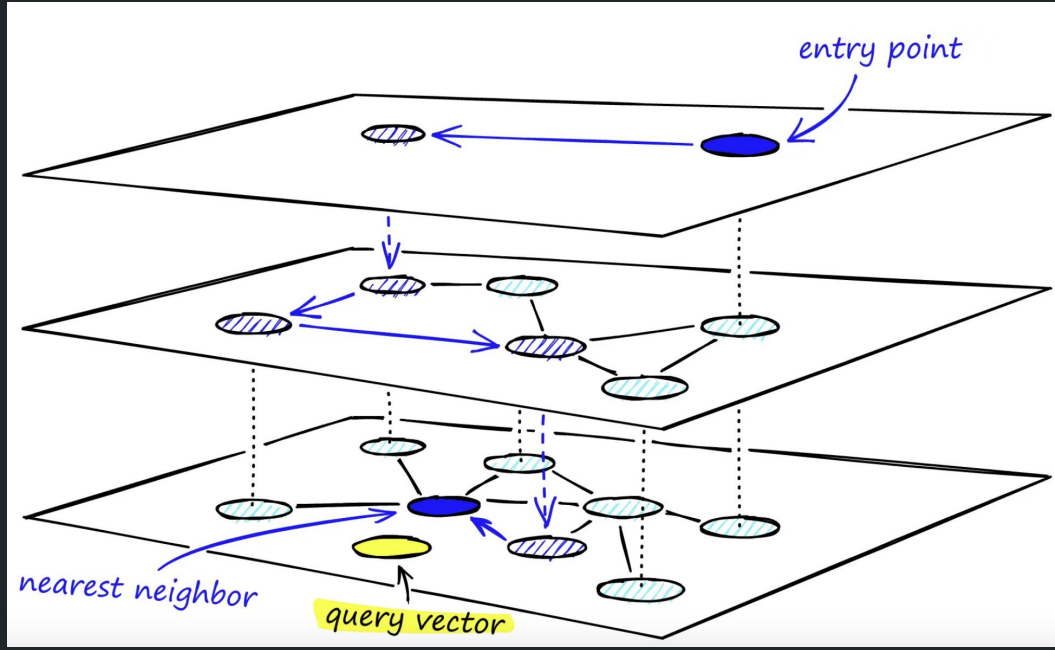
- **Scalability:** Vector DBs are designed to scale with growing data volumes and user demands, providing better support for distributed and parallel processing.
- **AI/ML Integration**: Easy integration with embedding models and tools like TensorFlow, PyTorch, and Hugging Face.
- **Data Management**: Offers features for inserting, deleting, and updating data.
- **Metadata storage and filtering**: Vector DBs can store metadata associated with each vector entry- users can then query the database using additional metadata filters for finer-grained queries.
- **Real-Time Updates**: Support real-time data updates for dynamically changing data.
- **Searching**: Use ANN (Approximate Nearest Neighbor) search to search vectors by meaning.
- **Indexing**: Use indexing algorithms like HNSW (Hierarchical Navigable Small World) and IVF (Inverted File) to optimize retrieval of nearest neighbor vectors

Product Overview - Comparative Chart

	Pinecone	Weaviate	Milvus	Chroma
Deployment	Managed Cloud	Local, Self-Hosted Cloud, & On Premise	Local, Self-Hosted Cloud, & On Premise	Local, Self-Hosted Cloud, & On Premise
Scalability	Supports Horizontal & Vertical Scaling	Supports Horizontal & Vertical Scaling	Supports Horizontal & Vertical Scaling	Supports Horizontal & Vertical Scaling
Metadata Filtering	Yes	Yes	Yes	Yes
Open Source	Proprietary	Core is Open Source, with available enterprise add-ons	Open Source	Open Source
Real Time Indexing	Yes	Yes	Yes	Yes
Primary Use Cases	Managed Vector Database for ML	Scalable Vector Storage, Semantic Search, and Multimodal Data Retrieval	Computer Vision, NLP Applications, Large Scale Similarity Search	LLM Apps Development

Technical Details - Pinecone

- Architecture -
  - Built for similarity search at scale
  - Uses Proprietary Indexing algorithms like Hierarchical Navigable Small World for nearest neighbor search
  - Compression
    - Quantization - reduce into smaller bits
    - Dimensionality Reduction - remove redundant features in vectors
    - Index storage - only active vectors kept in memory



https://docs.pinecone.io/reference/architecture/serverless-architecture

Image Sources - https://www.pinecone.io/learn/series/faiss/hnsw/

Technical Details - Pinecone

- Use Cases -
  - User preferences - matching with item embeddings for complicated cases (Netflix, Spotify)
  - Semantic Search - context aware search for unstructured data (documents, images)
  - Anomaly detection - identifying outliers in high-dimensional data (financial fraud)
- Key Difference -
  - Fully managed and serverless
    - No infrastructure required
    - Automatic scaling based on load
    - Minimal Setup time

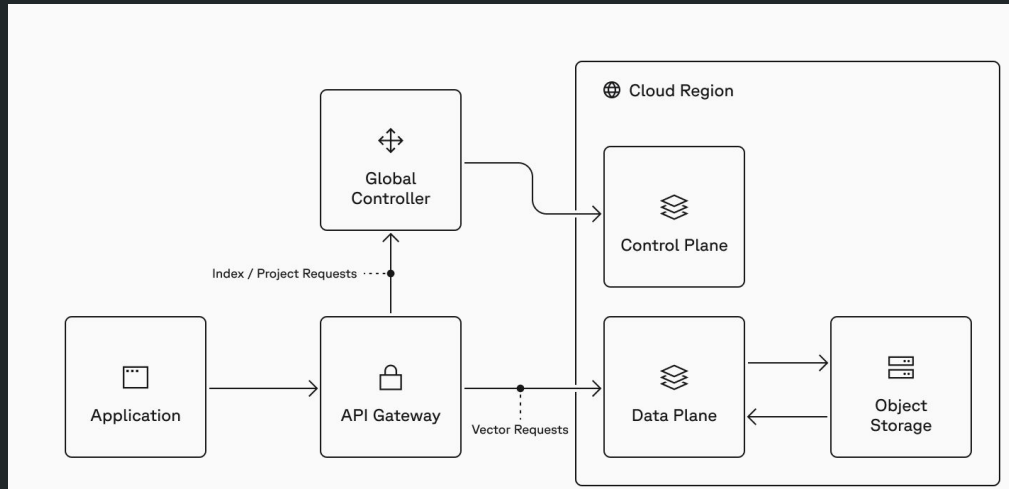


Image Sources - https://docs.pinecone.io/reference/architecture/serverless-architecture

Technical Details - Weaviate

- Architecture -
  - Hybrid Vector database
    - Vector search with structured data filtering
    - Store both vector and raw data
  - Open Source - self managed
  - HNSW + full text search, graph relationships
  - Requires users to determine a schema (unlike Pinecone)
  - Built-in integrations for Storage (PostgreSQL, S3) + Ml models (OpenAI, HuggingFace)

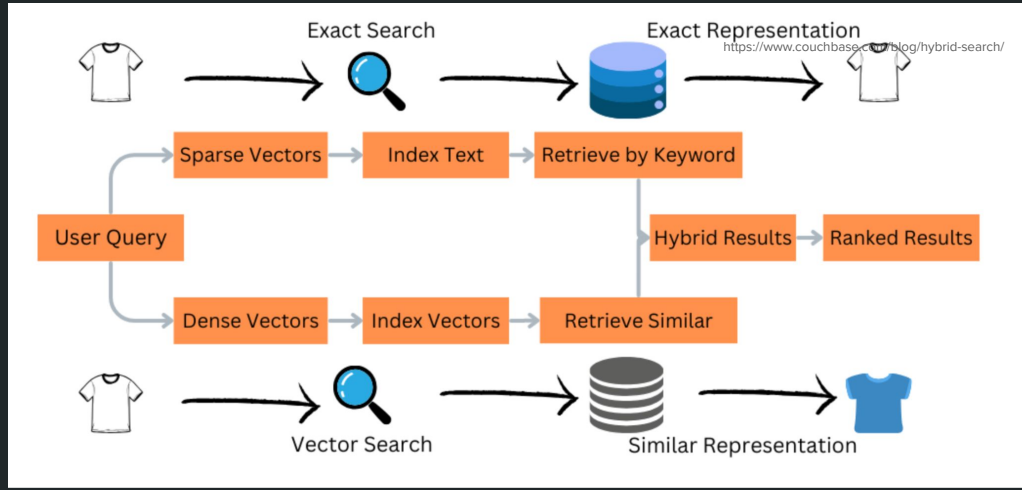


Image Sources - https://weaviate.io/ https://www.couchbase.com/blog/hybrid-search/

Technical Details - Weaviate

- Use Cases -
  - Hybrid Search - Combining vector embeddings with structured filters (Airbnb searches)
  - Multimodal search - upload an image and search for similar images
  - AI application integration - Retrieval Augmented Generation (RAG) for AI applications
- Key Difference
  - Open Source - Implement yourself, give yourself more control/flexibility, but more setup time
  - Hybrid Search - unique among vector databases,
    - Offers pre-built models

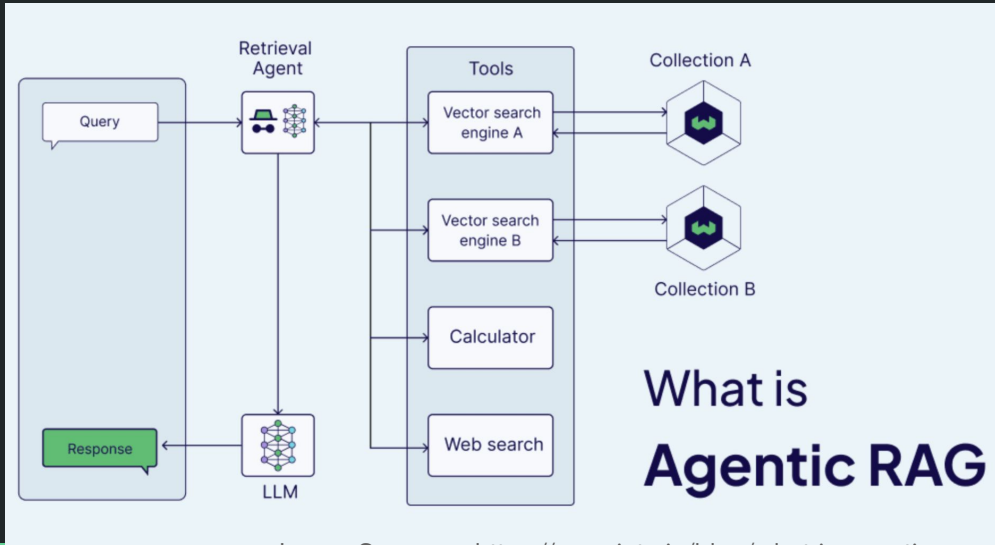


Image Sources - https://weaviate.io/blog/what-is-agentic-rag

Sample Applications - Frontier Medicines with Pinecone

- Overview of Frontier Medicines
  - A precision medicine company focused on developing groundbreaking treatments for genetically-defined patient populations
  - Uses the Frontier Platform, a chemoproteomics-powered engine that combines machine learning and covalent chemistry to target difficult disease-causing proteins
- Challenges Faced
  - Massive Data Generation:
    - The platform generates terabytes of data daily, this resulted in hundreds of millions of covalent molecule-proteome interaction data points over the past five years
  - Complex Similarity Searches:
    - Identifying potential drug candidates requires efficient similarity searches over billions of vectorized molecular structures, demanding high-performance computational resources
  - Scalability and Cost Efficiency:
    - Handling large-scale molecular datasets requires a high-performance, scalable, and cost-effective solution

FRONTIER MEDICINES

Image Sources - https://www.frontiermeds.com/newsroom#media-center

Sample Applications - Frontier Medicines with Pinecone

- Integration of Pinecone
  - Store and manage billions of vector embeddings representing molecular structures
  - Enable fast, high-precision semantic search to detect molecule similarities and improve drug discovery
- Benefits of Pinecone for Semantic Search
  - Drastically Improved Search Speed:
    - Searches are done in real-time and low-latency—even across billions of vectors
  - Better Molecular Match Accuracy:
    - Pinecone's high-dimensional indexing ensures precise similarity results
  - Seamless Scaling & Cost Efficiency:
    - Pinecone's serverless infrastructure scales automatically as new data is added
    - Eliminated the need for expensive, complex infrastructure management

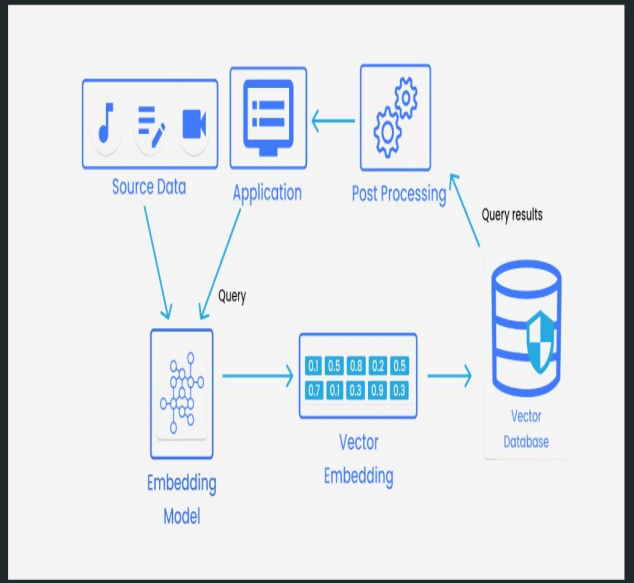


Image Source - https://www.trantorinc.com/blog/pinecone-ai-guide

Sample Applications - Unbody with Weaviate

- What is Unbody?
  - In 2016 began as a content management system (CSM) for Google drive
  - Evolved into the first AI-native backend, enabling developers to create data-driven websites, apps, and solutions
  - Their objective is to simplify AI integration for developers, making advanced AI accessible without requiring deep expertise

- Migration to weaviate
  - In 2022, Unbody transitioned to Weaviate.
  - Became the first headless CMS built entirely on a vector database

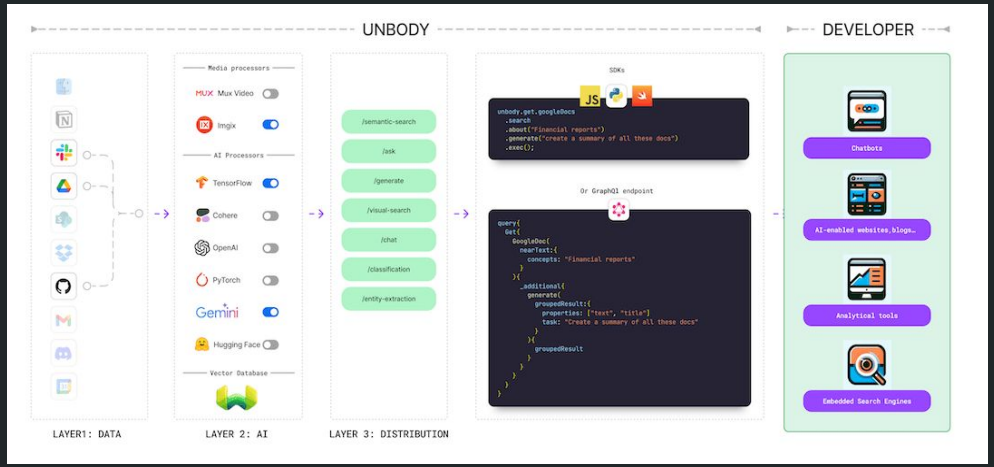


Image Sources - https://weaviate.io/blog/unbody-weaviate

Sample Applications - Unbody with Weaviate

- Benefits of using Weaviate
  - Simplified AI Data Processing
    - Weaviate's architecture allowed seamless application of embeddings and vectorizers, facilitating efficient handling of diverse content types, including text, images, and multimodal data
  - Enhanced Data Delivery
    - Leveraging Weaviate's GraphQL interface, Unbody provides tailored APIs, ensuring sophisticated yet user-friendly data interactions
  - Accelerated Development

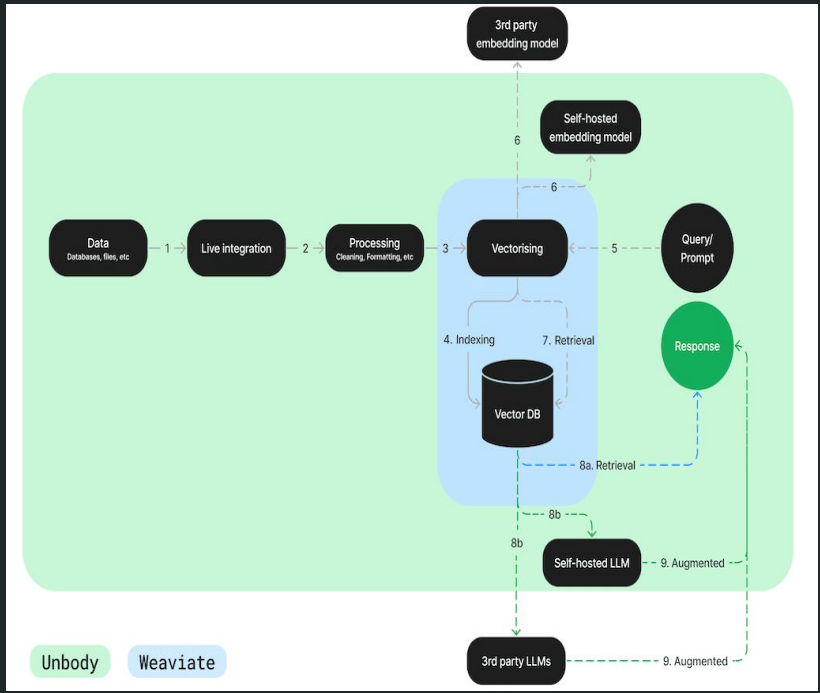


Image Sources - https://weaviate.io/blog/unbody-weaviate

Future Trends - AI & Generative Models

- Integration with LLM-Powered Applications
  - Vector databases are becoming essential for Retrieval-Augmented Generation (RAG) in AI chatbots and assistants
  - Examples: OpenAI, LangChain, and Hugging Face leveraging vector DBs for context-aware responses
- Expansion of Multi-Modal Search
  - Moving beyond text embedding to incorporate image, video, and audio vectors
  - Examples: Spotify's music similarity search, Pinterest's visual search, YouTube's video content recommendation
- AI-Driven Index Optimization
  - Auto-tuning vector indexes using machine learning to optimize search speed and accuracy dynamically

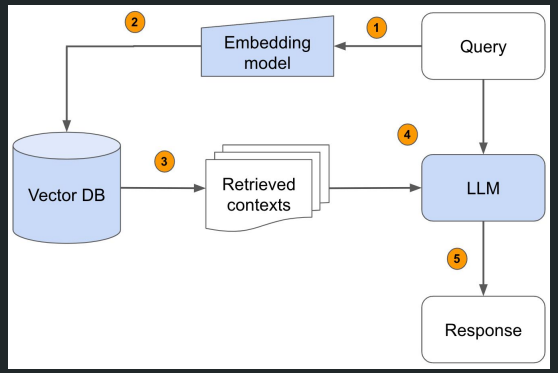


Image Sources - https://medium.com/@bijit211987/rag-vs-vectordb-2c8cb3e0ee52

Future Trends - Hybrid Search

- Bridging SQL and Vector Search
  - Future databases will support hybrid queries, combining keyword-based and vector-based search
  - Example: Weaviate’s hybrid search combining text-based filtering with vector embeddings
- Graph-Based & Semantic Search Advancements
  - Graph-enhanced vector search to improve context-based recommendations
  - Better semantic search with embeddings enriched by knowledge graphs
- Cross-Database Interoperability
  - More seamless integrations between vector databases and existing SQL/NoSQL systems
  - Standardization efforts for common vector embedding formats

Future Trends - Performance & Scalability

- GPU & TPU Acceleration for Billion-Scale Search
  - Faster ANN search leveraging NVIDIA TensorRT, TPUs, and AI-accelerated hardware
  - Example: Milvus and FAISS utilizing GPUs for high-speed vector similarity search
- Cloud-Native Vector Databases
  - More serverless, auto-scaling solutions (like Pinecone) to support real-time updates and large datasets
  - Federated vector search across multiple distributed clusters
- Edge AI & On-Device Vector Search
  - Efficient embedding storage and search for smartphones, IoT, and embedded AI systems
  - Example: Apple Neural Engine optimizing on-device similarity search

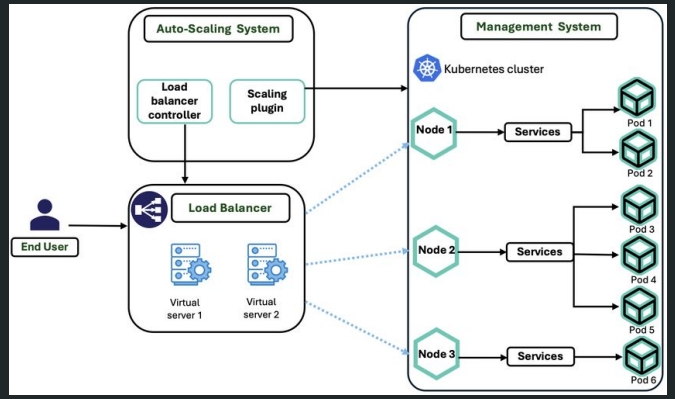


Image Sources - https://www.researchgate.net/figure/Auto-scaling-system-architecture_fig4_383489546

Research Challenges & Future Directions

- Efficient Storage & Compression Techniques
  - Advanced quantization (PQ, OPQ, SQ) and adaptive indexing to reduce memory footprint
  - Research in lossless compression for preserving high-accuracy embeddings
- Addressing bias & explainability in vector search
  - Concern: embeddings may inherit bias from training data (e.g., gender, racial, cultural bias)
  - Research in interpretable embeddings and bias-mitigation techniques
- Privacy-preserving vector search
  - Encrypted embeddings using homomorphic encryption and secure multiparty computation
  - Federated learning to reduce data centralization risks
- Standardization & interoperability efforts
  - Industry push for common APIs, embedding formats, and cross-database integration
  - Example: Vector Search Query Language (VSQ) proposals for unified querying across platforms

References

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

https://weaviate.io/blog/unbody-weaviate

https://www.pinecone.io/customers/frontier-medicines/