# It’s  time to embrace Vector Databases

Megha

ABNASIA.ORG

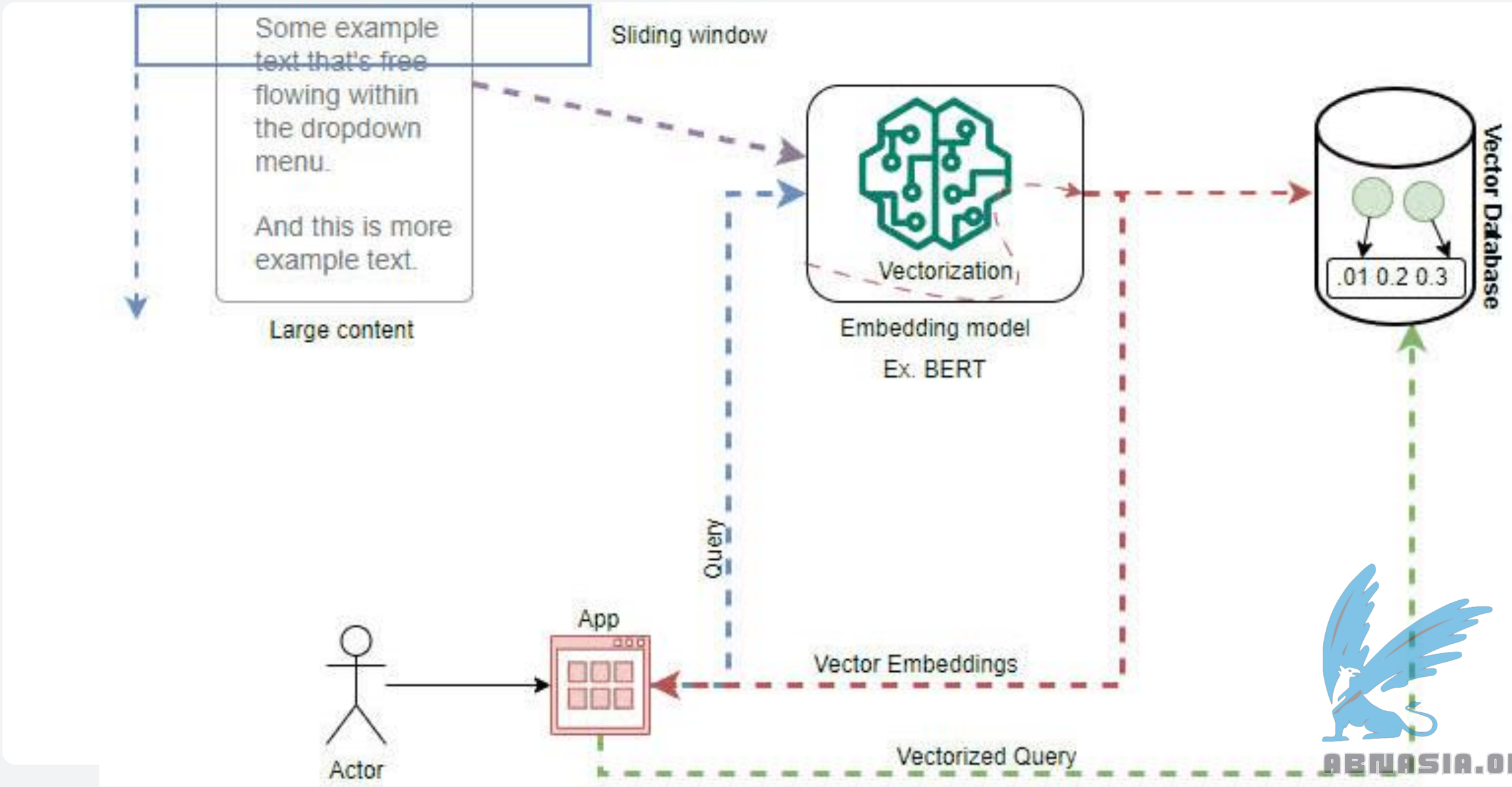
# What is a vector database?

A vector database is a collection of data stored as mathematical representations. Vector databases make it easier for machine learning models to remember previous inputs, allowing machine learning to be used to power search, recommendations, and text generation use-cases. Data can be identified based on similarity metrics instead of exact matches, making it possible for a computer model to understand data contextually.

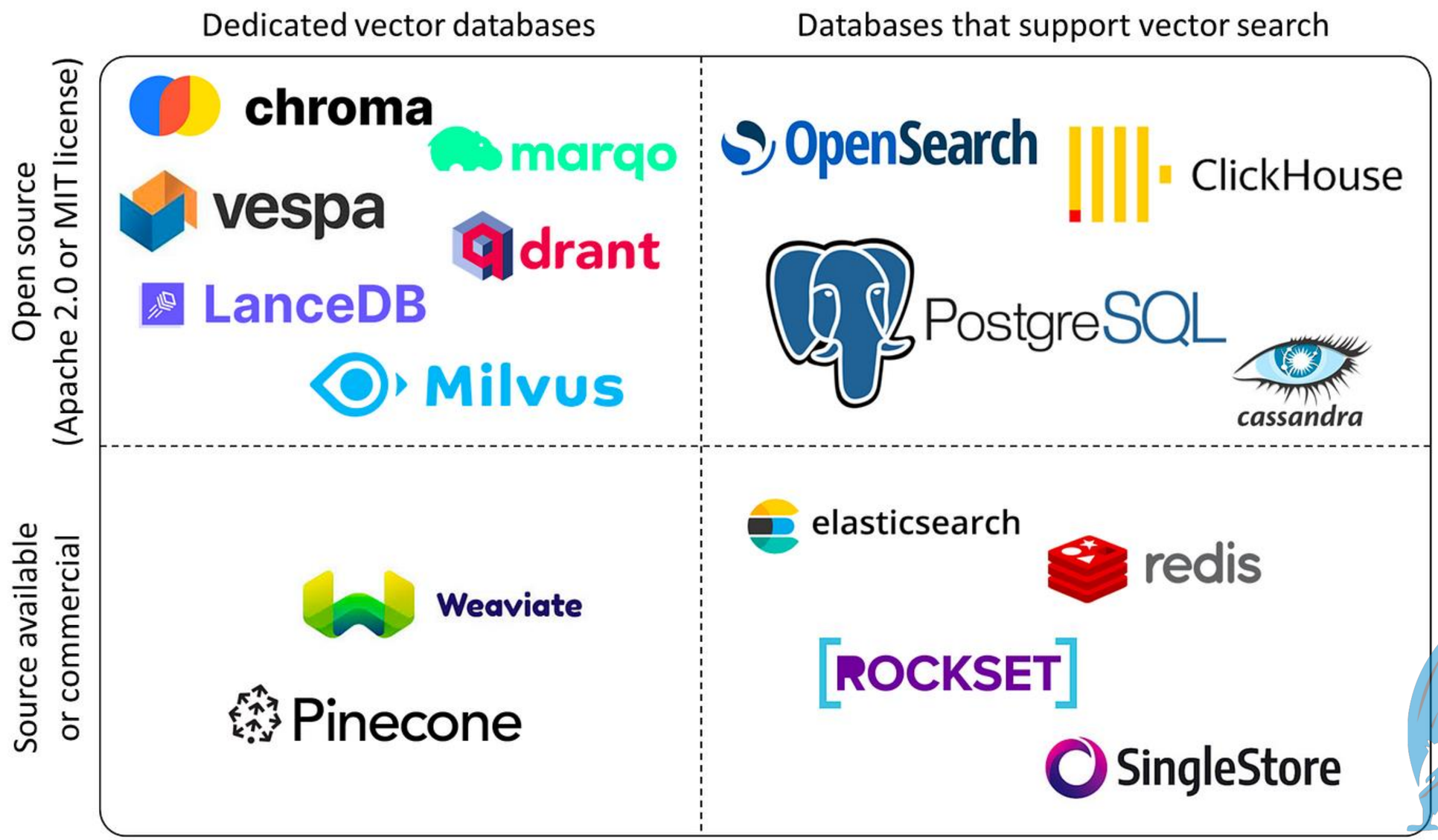
When one visits a shoe store, a salesperson may suggest shoes that are similar to the pair one prefers. Likewise, when shopping in an ecommerce store, the store may suggest similar items under a header like "Customers also bought..." Vector databases enable machine learning models to identify similar objects, just as the salesperson can find comparable shoes and the ecommerce store can suggest related products. (In fact, the ecommerce store may use such a machine learning model for doing so.)

To summarize, vector databases make it possible for computer programs to draw comparisons, identify relationships, and understand context. This enables the creation of advanced artificial intelligence (AI) programs like large language models (LLMs).

# LLM and Vector Databases



# Different Types of Vector Databases



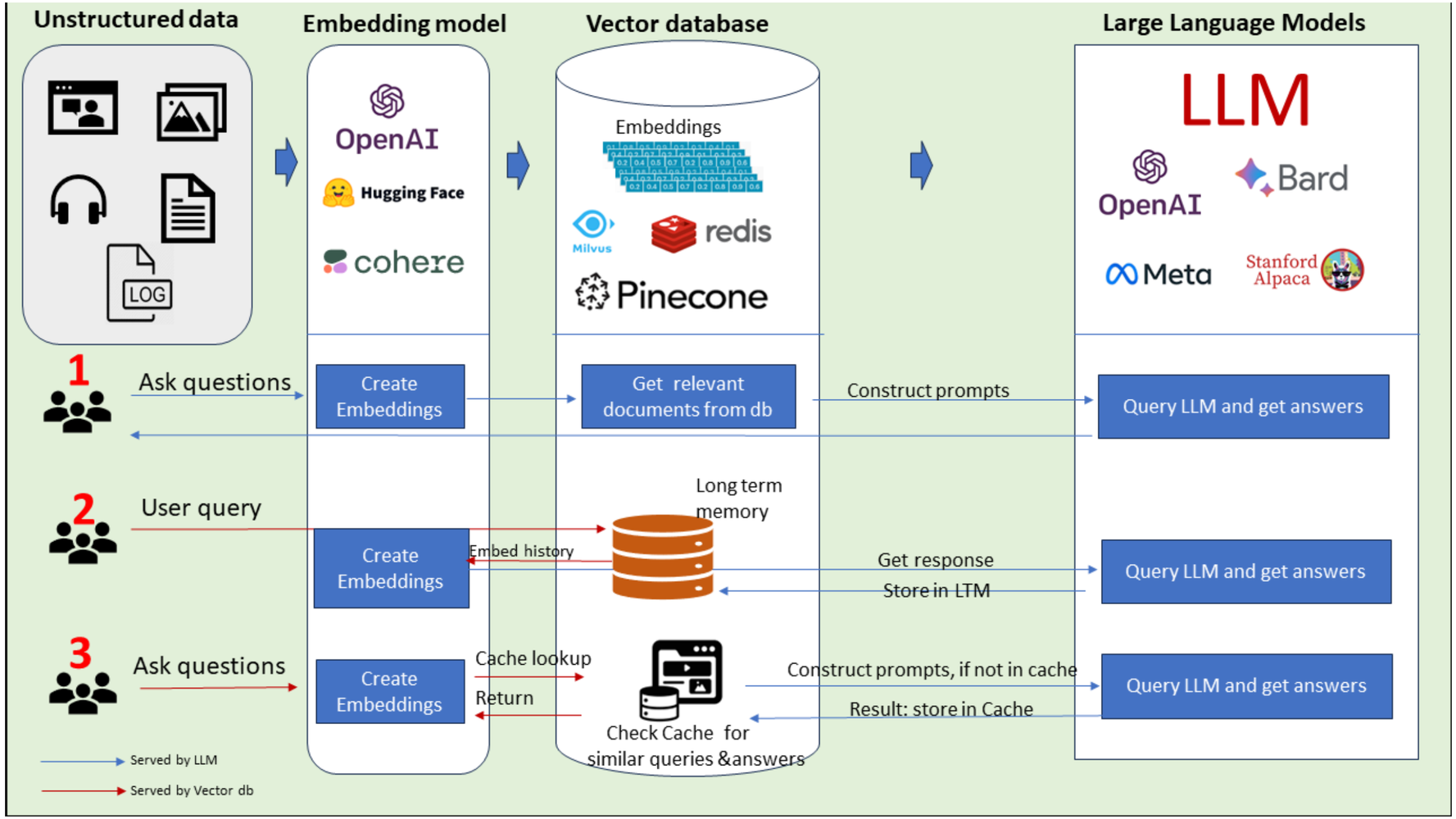
# Closed Source Vector Databases

**Closed-source vector databases** are specialized systems designed for storing and managing high-dimensional vector data, primarily used in applications involving machine learning and artificial intelligence. These databases utilize proprietary code and are typically offered as managed services, meaning that the underlying infrastructure is handled by the service provider, allowing users to focus on application development without worrying about database management.

## Key Features of Closed-Source Vector Databases :

- **Proprietary Technology**: Closed-source databases are built using proprietary technology, which means the source code is not publicly available. This can lead to enhanced performance optimizations and dedicated support from the vendor.
- **Managed Services**: Many closed-source vector databases are offered as managed services, simplifying deployment and scaling. Users can access the database via APIs without needing to manage the underlying infrastructure.
- **High Performance**: These databases are optimized for fast retrieval and similarity search of high-dimensional vectors, making them suitable for applications like recommendation systems, semantic search, and real-time analytics.
- **Integration Capabilities**: Closed-source vector databases often provide extensive integration options with other cloud services and machine learning platforms, facilitating seamless data workflows.
- **User-Friendly Interfaces**: Many closed-source solutions prioritize developer experience, offering intuitive interfaces and tools that simplify the process of building and deploying applications.

# Closed-Source Vector Databases



Pinecone

Pinecone is a cloud-based managed vector database that provides high-performance similarity search and is designed to make it easy for businesses and organizations to build and deploy large-scale machine learning applications. Unlike most popular vector databases, Pinecone uses closed-source code.

## Key Features of Pinecone :

- **User-friendly interface:** Pinecone provides a simple, intuitive interface that abstracts away the complexity of managing the underlying infrastructure, allowing developers to focus on application development.
- **High-dimensional vector support:** Pinecone is optimized for handling high-dimensional vector databases, making it suitable for various use cases, including similarity search, recommendation systems, personalization, and semantic search.
- **Single-stage filtering:** Pinecone supports single-stage filtering capability, which can be useful for applications like threat detection and monitoring against cyberattacks in the cybersecurity industry.
- **Real-time data analysis:** Pinecone's ability to analyze data in real-time is another key feature that makes it suitable for applications requiring immediate insights.
- **Extensive integrations:** Pinecone supports integrations with multiple systems and applications, including Google Cloud Platform, Amazon Web Services (AWS), OpenAI, GPT-3, GPT-3.5, GPT-4, ChatGPT Plus, Elasticsearch, and Haystack

Pinecone

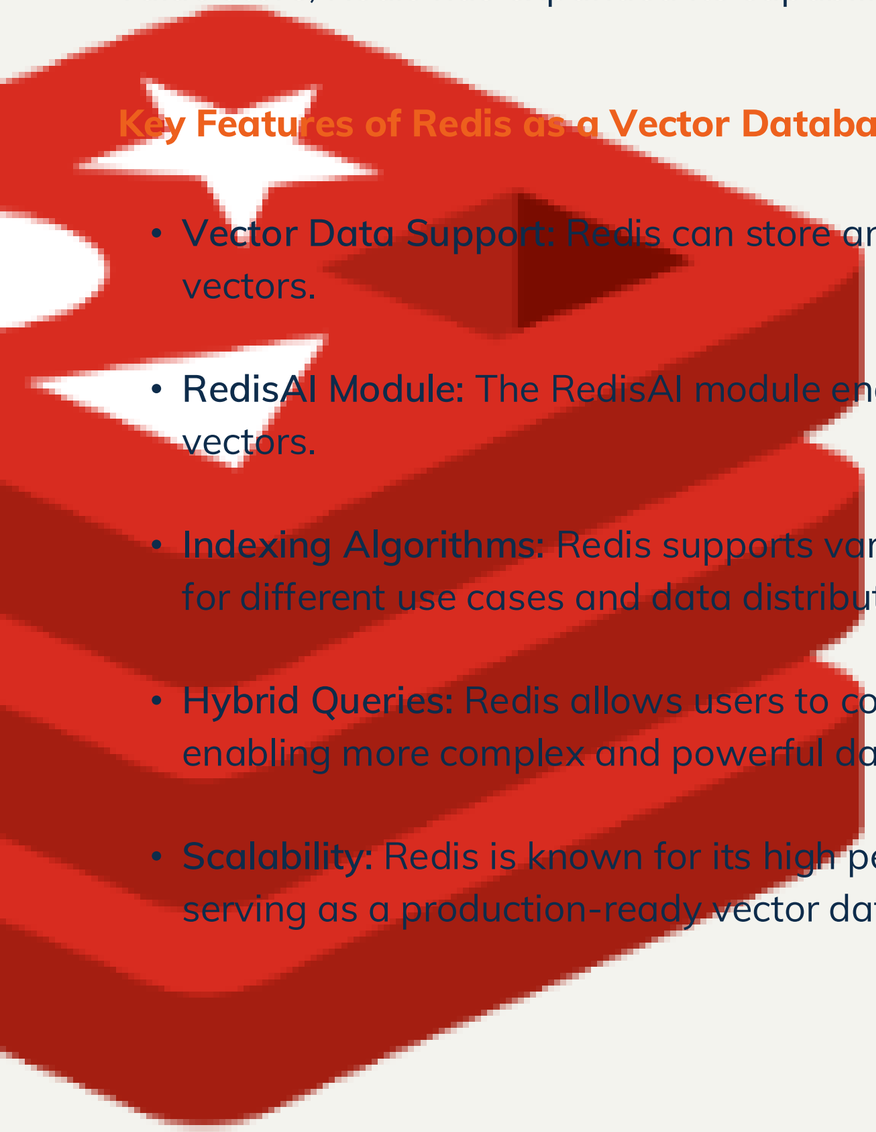
## Advantages of Pinecone :

- **Simplifies deployment and scaling:** As a fully managed service, Pinecone abstracts away the complexities of infrastructure management, making it easier for organizations to deploy and scale their machine learning applications.
- **Optimized for machine learning:** Pinecone is designed specifically for machine learning applications, providing high-performance similarity search and scalability.
- **Cost-effective:** Pinecone claims to offer 50x lower cost at any scale compared to traditional vector search solutions.

## Use Cases for Pinecone :

- **Recommendation systems:** Pinecone's vector search capabilities make it suitable for building recommendation systems that suggest similar products, content, or information based on user preferences.
- **Semantic search:** Pinecone can be used to implement semantic search functionality, allowing users to find relevant information based on the meaning of their queries rather than exact keyword matches.
- **Personalization:** By leveraging vector embeddings, Pinecone can help personalize content, products, or experiences for individual users based on their preferences and behavior.
- **Anomaly detection:** Pinecone's real-time data analysis capabilities can be useful for detecting anomalies or outliers in data for applications like fraud detection, network security monitoring, or quality control.

Redis a popular open-source in-memory data structure store, can also be used as a vector database. While primarily known as a key-value store, Redis has expanded its capabilities to include vector data handling through its RedisAI module.



**Key Features of Redis as a Vector Database:**

- **Vector Data Support:** Redis can store and manage vector data, allowing users to perform efficient similarity searches on high-dimensional vectors.
- **RedisAI Module:** The RedisAI module enables Redis to handle vector data types and provides commands for indexing and querying vectors.
- **Indexing Algorithms:** Redis supports various indexing algorithms for vector data, such as FLAT and HNSW, which optimize performance for different use cases and data distributions.
- **Hybrid Queries:** Redis allows users to combine vector search with other types of queries, such as text, numeric, and geospatial searches, enabling more complex and powerful data retrieval.
- **Scalability:** Redis is known for its high performance and scalability, making it suitable for handling large volumes of vector data and serving as a production-ready vector database.

## Use Cases for Redis as a Vector Database :

- **Recommendation Systems:** Redis can be used to build recommendation systems that suggest similar products, content, or information based on user preferences or item characteristics.

- **Semantic Search:** By representing text as vectors, Redis can enable semantic search capabilities, allowing users to find relevant information based on the meaning of their queries rather than exact keyword matches.

- **Image and Video Search:** Redis can store and search through vector embeddings of images and videos, enabling content-based retrieval and similarity-based recommendations.

- **Anomaly Detection**: Redis's vector search capabilities can be leveraged for anomaly detection tasks, where outliers in high-dimensional data need to be identified for applications like fraud detection or network security monitoring.

## Deployment Options : **Redis can be deployed in various ways to serve as a vector database**

- **Redis Stack:** Redis Stack is a distribution that includes Redis along with additional modules, such as RedisSearch and RedisAI, making it easy to set up a vector database out of the box.

- **Redis Enterprise:** Redis Enterprise is a commercial offering that provides a fully managed Redis service, including support for vector data and advanced querying capabilities.

- **Redis Cloud:** Redis Cloud is a managed Redis service offered by Redis, which includes support for vector data and can be easily provisioned for development and production use cases.

# KDB.AI

KDB.AI is a specialized vector database developed by KX, designed to support scalable, reliable, and real-time applications, particularly in the context of generative AI. It integrates advanced search capabilities, recommendation systems, and personalization features, making it suitable for a variety of AI applications.

## Key Features of of KDB.AI:

- **Time-Based Vector Database:** KDB.AI uniquely combines unstructured vector embeddings with structured time-series data, enabling hybrid use cases that leverage both data types for enhanced analytics.
- **Multiple Index Types:** It supports various indexing methods such as Flat, IVF, IVFPQ, and HNSW, allowing for optimized performance based on specific use cases.
- **Distance Metrics:** KDB.AI provides multiple distance metrics, including Euclidean, Inner-Product, and Cosine, to facilitate accurate similarity searches.
- **Advanced Search Capabilities:** The database allows for Top-N retrieval and metadata filtering, improving the efficiency and relevance of search results.
- **Integration with AI Frameworks:** KDB.AI integrates with popular AI frameworks like Langchain and LlamaIndex, enhancing its utility for generative AI applications.

KDB.AI

## Use Cases : KDB.AI is particularly well-suited for

- **Temporal Similarity Search:** Analyzing time-series data alongside unstructured data for insights and trends.
- **Recommendation Systems:** Building systems that suggest content or products based on user behavior and preferences.
- **Hybrid Search:** Combining dense and sparse search methods to improve accuracy in retrieving relevant data.
- **Multimodal Applications:** Handling various data types, including text, images, and audio, for comprehensive analysis.

## Deployment Options : KDB.AI offers two primary deployment options

- **KDB.AI Cloud**: A cloud-based solution suitable for experimenting with smaller generative AI projects.
- **KDB.AI Server**: An on-premises option for evaluating large-scale generative AI applications, providing more control over the deployment environment.


Azure AI Search is a cloud-based search service offered by Microsoft that integrates vector search capabilities for advanced retrieval and generation in AI applications. Here are the key points about Azure AI Search:

**Overview:**

- Azure AI Search is a platform-as-a-service that helps developers create cloud search solutions with sophisticated retrieval strategies.
- It provides an enterprise-ready vector database with built-in security, compliance, and responsible AI practices.
- Azure AI Search enables seamless platform and data integrations for AI models, frameworks, and data sources.

**Key Features:**

- Performs advanced search and retrieval, focusing on exponential growth with an enterprise-ready vector database.
- Provides sophisticated retrieval strategies backed by decades of research and customer validation.
- Supports hybrid retrieval with reranking to deliver higher quality results.
- Allows searching across various data types like text, images, PDFs, audio, and more based on conceptual similarity.


## Use Cases :

- Building enterprise knowledge bases to uncover insights from unstructured data using large language model (LLM) reasoning and contextual knowledge.
- Implementing AI assistants that provide relevant information to answer questions quickly.
- Enabling retrieval augmented generation (RAG) to ground generative AI applications with proprietary data.

## Pricing and Availability :

- Azure AI Search is available through search units to provide the requested capacity and throughput to set up and scale a search and retrieval engine.
- It is available in more than 30 countries and regions worldwide, with new ones being added regularly.

Azure AI Search simplifies the development of advanced search and retrieval capabilities in AI applications by providing an enterprise-grade vector database with hybrid search and reranking features. Its seamless integration with Azure services and support for various data types make it a compelling choice for building intelligent search solutions.

vectara

Vectara is a modern, API-first platform that provides Retrieval-Augmented Generation (RAG) as a service, enabling organizations to create intelligent applications and AI assistants grounded in their own data. Here are the key aspects of Vectara:

## Overview:

- **RAG-as-a-Service**: Vectara offers a comprehensive service that includes all components necessary for RAG, allowing developers to build applications that intelligently retrieve and generate responses based on data.

- **Text Extraction**: It can extract text from various file formats, including PDFs, PowerPoint presentations, and Word documents, simplifying the data ingestion process.

- **Boomerang Embeddings Model:** Vectara uses its proprietary Boomerang embeddings model to encode text chunks into vector embeddings, which are stored in its internal vector database.

## Key Features:

- Internal Vector Database: Vectara maintains its own vector database where text chunks and their corresponding embeddings are stored, facilitating efficient retrieval.

- Query Service: The platform includes a query service that automatically encodes user queries into embeddings and retrieves relevant text segments. It supports hybrid search and maximum marginal relevance (MMR) for improved result quality.

- Generative Summarization: Vectara can generate summaries based on the retrieved documents, providing context and citations for the information presented.

vectara

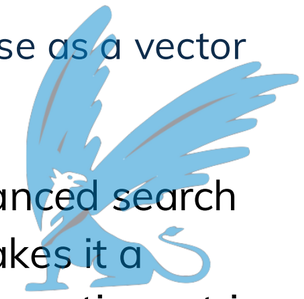
## Use Cases :

- **AI Assistants:** Organizations can create ChatGPT-like experiences that leverage their internal knowledge bases, allowing for more accurate and context-aware interactions.
- **Content Retrieval:** Vectara is suitable for applications requiring efficient retrieval of information from large datasets, such as customer support systems or knowledge management platforms.
- **Enhanced Search Capabilities:** By combining traditional search techniques with AI-driven retrieval, Vectara improves the relevance and accuracy of search results.

## Integration and Setup :

- Vectara can be easily integrated into applications using its API. Developers can sign up for an account, create a corpus, and obtain an API key to start using the service.
- It supports various programming environments, including integration with LangChain, allowing for seamless use as a vector store.

Vectara provides a robust platform for organizations looking to harness the power of generative AI and advanced search capabilities. Its focus on RAG, combined with an easy-to-use API and powerful data processing features, makes it a compelling choice for building intelligent applications that require contextual understanding and efficient information retrieval.



mongoDB Atlas

Mongo DB Atlas is a cloud-based database service that provides a fully managed platform for building and deploying applications. It includes a variety of features, with a significant focus on vector search capabilities. Here are the key aspects of MongoDB Atlas, particularly its vector search functionality:

## Overview:

- **Integrated Data Services:** MongoDB Atlas is designed to simplify the development process by integrating various data services, allowing developers to build applications faster and more efficiently.

- **Multi-Cloud Support:** It supports deployment across multiple cloud providers, including AWS, Azure, and Google Cloud, providing flexibility and scalability for applications.

## Key Features:

- **Approximate Nearest Neighbor (ANN) Search:** Atlas Vector Search utilizes algorithms like Hierarchical Navigable Small World (HNSW) for efficient indexing and querying of millions of vectors.

- **Hybrid Search**: Users can combine semantic search with traditional keyword search, allowing for more comprehensive query capabilities.

- **Integration with AI Frameworks**: The platform supports integration with popular AI models and frameworks, facilitating the development of AI-powered applications.

mongoDB. Atlas

## Use Cases :

- **Generative AI Applications:** Atlas Vector Search is particularly effective for applications that require retrieval-augmented generation (RAG), where relevant documents are retrieved to enhance the performance of generative models.

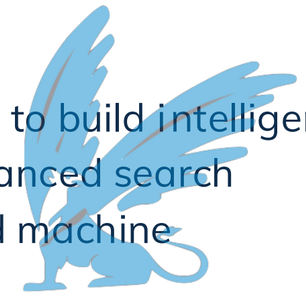
- **Recommendation Systems:** The ability to perform semantic searches makes it suitable for building recommendation engines that suggest relevant content based on user interactions.

## Integration and Setup :

- **Search Nodes**: Dedicated infrastructure for Atlas Search and Vector Search workloads allows for optimized resource usage and better performance at scale.

- **Low Latency:** MongoDB Atlas can provide low-latency responses for vector search queries, maintaining performance even with large datasets.

MongoDB Atlas, with its integrated vector search capabilities, provides a powerful solution for developers looking to build intelligent applications. Its ability to manage both structured and unstructured data in a single platform, combined with advanced search functionalities, makes it a compelling choice for modern application development, especially in the realm of AI and machine learning.



Neo4j is a popular open-source graph database that has introduced features for handling vector data, suitable for applications requiring both graph and vector capabilities.

## Overview:

- Neo4j is a native graph database designed to efficiently store, manage and query graph-structured data.
- It uses a graph data model with nodes, relationships, and properties to represent and connect data.
- Neo4j provides a declarative query language called Cypher for interacting with the database.

## Key Features:

- Native Graph Storage: Neo4j stores data in a graph format optimized for traversals and pattern matching.
- ACID Transactions: It supports ACID transactions to ensure data integrity and consistency.
- Scalability: Neo4j can scale to billions of nodes and relationships while maintaining high performance.
- Cypher Query Language: The Cypher language allows for expressive and efficient querying of graph data.

## Use Cases :

- **Recommendation Engines:** Neo4j's graph structure is well-suited for building recommendation systems that suggest related products, content or connections.
- **Fraud Detection:** Graphs can model complex relationships to identify suspicious patterns and fraudulent activities.
- **Master Data Management:** Neo4j provides a flexible data model to integrate data from multiple sources into a unified view.
- **Network and IT Operations:** Graphs can model IT infrastructure and dependencies to enable impact analysis and root cause identification.

## Vector Search Capabilities

- Neo4j 5.11 introduced vector index support, enabling similarity searches and complex analytical queries on high-dimensional vector embeddings.
- The vector index implementation uses the HNSW (Hierarchical Navigatable Small World) algorithm for efficient approximate nearest neighbor search.
- Vector indexes can be created on both nodes and relationships, allowing flexible modeling of vector data in the graph.
- Cypher queries can leverage vector indexes to find similar nodes based on their embeddings, useful for applications like recommendation systems and semantic search.

elasticsearch

Elastic Search provides a built-in vector database that allows storing and searching high-dimensional vector embeddings efficiently. It provides a robust vector search solution that can be used for a wide range of AI-powered search applications. Its combination of vector search, semantic search, and natural language processing capabilities makes it a compelling choice for building intelligent search experiences.

**It supports two main types of vector search:**

- Exact search using the script_score query for brute-force linear search
- Approximate nearest neighbor (ANN) search using the knn search option with indexing techniques like HNSW

## Setting up Vector Search in Elasticsearch:

- Setting up Vector Search in Elasticsearch
- To enable vector search, you define a dense_vector field in the index mapping, specifying the number of dimensions.
- Elasticsearch automatically indexes the vectors for efficient ANN search by default.
- You can configure the knn search with filters, expected similarity, and run multiple knn searches simultaneously.

elasticsearch

## Use Cases : Elasticsearch's vector search powers a variety of applications:

- Semantic search for finding relevant information based on meaning, not just keywords
- Similarity search for images, videos, and audio to find visually similar content
- Personalization by finding items similar to ones a user has shown interest in
- Natural language processing to enrich search experiences with Q&A, NER, and sentiment analysis
- Generative AI by providing high-relevance context windows from private data to improve language model outputs

## Semantic Search in Elasticsearch

- Semantic search in Elasticsearch combines vector search with the Learned Sparse Encoder (ELSER) model.
- ELSER is a proprietary model that generates embeddings for both documents and queries, allowing semantic search without needing to create separate embeddings externally.
- Semantic search is available as part of the Elastic Stack subscription, while basic vector search is available without a subscription.

Rockset is a cloud-based search and analytics database-as-a-service that specializes in real-time indexing and querying of dynamic, semi-structured data such as JSON, CSV, and TSV. Here are the key points about Rockset:

## Overview:

- Acquisition: Rockset was acquired by OpenAI in June 2024, with plans to cease its database service in September 2024 and delete all customer data.

- Dynamic Schema: Rockset supports a dynamic schema, allowing it to index entire datasets without needing to infer a schema from a sample. This means new data can be instantly queried without rejection.

## Setting up Vector Search in Elasticsearch:

- **Document-Based Structure:** Data is stored as documents, which contain fields and have unique IDs. Updates to documents are atomic, and multiple fields can be updated simultaneously.

- **Indexing Mechanism:** Rockset employs a combination of three indexes: Inverted Index, Columnar Index, and Document Index, optimized for various query types including key-value, time-series, document, search, aggregation, and graph queries.

- **Real-Time Indexing:** The database maintains live, real-time indexes updated from multiple data sources, ensuring that queries can be executed against the most current data.

- **Hybrid Storage Model:** Rockset uses a hybrid storage model that supports both columnar and document-based data, enhancing performance for dynamic datasets.

## Use Cases :

- **Real-Time Analytics:** Rockset is suitable for applications that require real-time data insights, such as dashboards and monitoring systems.
- **Search Applications:** Its capabilities make it ideal for search applications that need to index and query large volumes of semi-structured data quickly.
- **Data Transformation:** Users can perform SQL-based data transformations during ingestion, allowing for ETL processes within the database.

## SQuery Processing

- Rockset utilizes a bottom-up approach for query processing, employing both the iterator model (volcano model) and vectorized execution for scaling queries.
- The system is designed to handle complex queries efficiently, utilizing optimizations to ensure fast response times.

# Open-Source Vector Databases

**Open source vector databases** are databases specifically designed to store, manage, and retrieve high-dimensional vector data, which is essential for applications in machine learning, artificial intelligence, and data analysis. Unlike traditional databases that handle structured data, vector databases are optimized for similarity search and can efficiently process unstructured data like text, images, and audio.

## Key Features of Open-Source Vector Databases :

- **High-Dimensional Data Handling:** These databases can manage vectors with varying dimensions, often ranging from a few to thousands, depending on the complexity of the data.
- **Efficient Similarity Search:** Open-source vector databases allow for fast retrieval of data based on vector proximity, enabling semantic searches that go beyond exact matches.
- **Scalability:** Many open-source vector databases are designed to scale horizontally, accommodating large datasets that may contain millions or billions of vectors.
- **Community-Driven Development:** Being open-source, these databases benefit from contributions by a community of developers, which can lead to rapid improvements and a wide range of features.

# Categories of Vector Databases

chroma

Chroma is an open-source, AI-native embedding vector database that aims to simplify the process of creating LLM applications powered by natural language processing by making knowledge, facts, and skills pluggable for machine learning models at the scale of LLMs – as well as avoiding hallucinations. It also provides a JavaScript client and a Python API for interacting with the database.

## Overview :

- Chroma is an open-source vector database focused on developer productivity and ease of use.
- It is designed to be AI-native, with batteries included features like embeddings, vector search, document storage, full-text search, and metadata filtering.
- Chroma runs in various modes - in-memory, in-memory with persistence, and as a server running on a local machine or in the cloud

## Key Feature :

- Supports different underlying storage options like DuckDB for standalone or ClickHouse for scalability.
- Provides SDKs for Python and JavaScript/TypeScript.
- Focuses on simplicity, speed, and enabling analysis.
- Allows creating collections to store embeddings, documents, and metadata.
- Automatically converts text into embeddings using models like all-MiniLM-L6-v2 by default.
- Supports querying collections by text or embedding to retrieve similar documents, with optional metadata filtering.

Milvus

Milvus is another famous open source vector database that is designed for efficient similarity searches and vector embedding. Milvus is used to simplify the unstructured data search and also provides a better experience across multiple deployment environments. It is one of the most popular vector database used for applications such as chatbots, chemical structure search, and image search.

## Overview :

- **Purpose**: Milvus is built for high-performance vector similarity search, enabling applications to manage and query large-scale embedding vectors generated by machine learning models.
- **Community and Adoption**: Since its launch in 2019, Milvus has gained significant traction, with over 28,410 GitHub stars and a strong community of contributors.

## Key Feature :

- **Cloud-Native Architecture**: Milvus separates storage and compute layers, allowing for flexible scaling and efficient resource management.
- **High Performance**: It utilizes advanced indexing algorithms that provide a 10x performance boost in retrieval speed, making it suitable for processing billions of vectors.
- **Flexible Data Handling**: Supports various data types, including vectors, scalar, and structured data, facilitating diverse use cases.
- **Extensive SDK Support**: Milvus offers SDKs for popular programming languages like Python, Java, and Go, simplifying integration into existing applications.

Milvus

**Use Cases : Milvus is used across various applications, including:**

- **Semantic Text Search:** Enhances search capabilities by processing and querying text across multiple vectors.
- **AI Advertising:** Improves targeted advertising by indexing user behavior and interests as high-dimensional vectors.
- **Media Similarity Search:** Enables the discovery of similar videos, audio, and images through advanced similarity search options.
- **Recommendation Systems:** Powers product recommendation engines by analyzing user behavior and preferences.
- **Question Answering Systems:** Supports natural language processing applications for customer support and information retrieval.
- **AI Drug Discovery:** Stores and indexes vector representations of compounds to facilitate new drug discovery.

Milvus is a powerful tool for organizations looking to leverage vector databases for AI applications. Its robust architecture, high performance, and flexibility make it an ideal choice for building similarity search-based applications. With a growing community and active development, Milvus continues to evolve as a leading solution in the vector database space.

Weaviate

Weaviate is also another famous open source vector database. It is a cloud-native database that is resilient, quick, and scalable. This vector database tool is used to convert photos, text, and multiple data into a searchable vector database by using algorithms and machine learning models.

Software developers use this tool to vectorize their data during the import process which ultimately creates systems for question-and-answer extraction, categorization, and summarization.

## Overview :

- **Purpose:** Weaviate is designed to enable vector search and similarity-based queries on structured data.
- **Data Model:** It uses a graph-like data model, storing both objects and their vector embeddings.
- **Integrations:** Weaviate integrates with popular machine learning frameworks like PyTorch and TensorFlow for generating vector embeddings.

## Key Feature :

- **Hybrid Search:** Weaviate supports combining vector search with structured filtering, allowing for more precise and relevant results.
- **Scalability:** It is designed to be cloud-native and scale horizontally to handle large datasets.
- **Extensibility:** Weaviate can be extended with custom modules for tasks like image recognition and text generation.
- Weaviate consists of built-in modules for AI-powered searches, automated categorization, and combining LLMs and Q&A.
- This vector database is used to seamlessly transfer machine learning models to MLOps using the database.
- Weaviate operates perfectly on Kubernetes.

deeplake

Deep Lake is an AI database powered by a proprietary storage format designed specifically for deep-learning and LLM-based applications that leverage natural language processing. It helps engineers deploy enterprise-grade LLM-based products faster via vector storage and an array of features.

Deep Lake works with data of any size, is serverless, and allows you to store all data in a single location.
It also offers tool integrations to help streamline your deep learning operations. For example, using Deep Lake and Weights & Biases, you can track experiments and achieve full model repeatability. The integration delivers dataset-related information (URL, commit hash, view ID) to your W&B runs automatically.

**Key Feature :**

- Storage for all data types (embeddings, audio, text, videos, images, pdfs, annotations, and so on).
- Querying and vector search.
- Data streaming during training models at scale.
- Data versioning and lineage for workloads.
- Integrations with tools like LangChain, LlamaIndex, Weights & Biases, and many more.

Qdrant is an open-source vector similarity search engine and database. It offers a production-ready service with an easy-to-use API for storing, searching, and managing points-vectors and high dimensional vectors with an extra payload.

The tool was designed to provide extensive filtering support. Qdrant's versatility makes it a good pick for neural network or semantic-based matching, faceted search, and other applications.

**Key Feature :**

- JSON payloads can be connected with vectors, allowing for payload-based storage and filtering.
- Supports a wide range of data types and query criteria, such as text matching, numerical ranges, geo-locations, and others.
- The query planner makes use of cached payload information to improve query execution.
- Write-Ahead during power outages, with the update log recording all operations, allowing for easy reconstruction of the most recent database state.
- Qdrant functions independently of external databases or orchestration controllers, which simplifies configuration.

elasticsearch

**Elasticsearch** Elasticsearch is an open-source, distributed, and RESTful analytics engine that can handle textual, numerical, geographic, structured, and unstructured data. Based on Apache Lucene, it was initially published in 2010 by Elasticsearch N.V. (now Elastic). Elasticsearch is part of the Elastic Stack, a suite of free and open tools for data intake, enrichment, storage, analysis, and visualization.

Elasticsearch can handle a wide range of use cases – it centrally stores your data for lightning fast search, finetuned relevance, and sophisticated analytics that scale easily. It expands horizontally to accommodate billions of events per second while automatically controlling how indexes and queries are dispersed throughout the cluster for slick operations.

**Key Feature :**

- Clustering and high availability.
- Automatic node recovery and data rebalancing.
- Horizontal scalability.
- Cross-cluster and data center replication, which allows a secondary cluster to operate as a hot backup.
- Cross-datacenter replication.
- Elasticsearch identifies errors in order to keep clusters (and data) secure and accessible.
- Works in a distributed architecture that was built from the ground up to provide constant peace of mind.

# How to choose the right vector database for your project

When picking a vector database for your project, consider the following factors:

- Do you have an engineering team to host the database, or do you need a fully managed database?
- Do you have the vector embeddings, or do you need a vector database to generate them?
- Latency requirements, such as batch or online,
- Developer experience in the team,
- The learning curve of the given tool,
- Solution reliability,
- Implementation and maintenance costs,
- Security and compliance.