Vector Stores in Artificial Intelligence (AI)

You're probably aware of the buzz around artificial intelligence (AI), language learning (LL), and machine learning (ML), which impact everything from social media algorithms to self-driving cars, but you may not know that the real magic behind these technological advancements and their query performance is data management. Understanding this data dependence is crucial for grasping how these emerging technologies are able to produce intelligent decisions so quickly.

At their core, these sophisticated digital technologies rely heavily on data that has been collected and processed to ultimately provide intelligent answers from a specialized database called a vector database. The process starts by turning raw information like words, images, video or music, into vectors, then feeding them into a pre-trained machine language model (MLM) that

optimizes the original ones into embeddings, which are high dimensional vectors that can be searched in vector databases.

Table of Contents:

Raw data's transformation

This blog post will focus on how data is transformed into vectors and how they are used in vector databases, including some discussion around search, vector space, and approximate nearest neighbor.

Although important, this article will not dig deep into defining vector indexes, cosine similarity searches, cosine distance, euclidean distance, orthogonal vectors, or facebook ai similarity search.

What are vector stores?

Typically found in vector databases, a vector store is a digital storage system that holds vectors – the unique fingerprint for a piece of data, whether it's a sentence, an image, or a user's browsing habits. The vector store not only keeps these vectors together in a vector space, but also quickly and efficiently finds and places similar vectors or identical vectors near one another.

Similar vectors and increasingly dissimilar vectors are quickly identified so that only the requested vectors are included in search results. This is incredibly useful in applications like search engines, recommendation systems, and AI models, where understanding and matching similar pieces of data, and eliminating those that don't apply, is key to producing an accurate response.

Vector stores can also be used behind the scenes in facial recognition technologies, which helps compare facial features within vector databases for apps and other technologies that call for it. In addition, Google’s search algorithms use vector stores to process vast amounts of data from web pages, making search results more accurate and relevant.

But before diving too deeply into what vector stores and vector databases can do, let’s define what a vector is and how it is used in AI.

What is a vector in AI?

You've probably heard the word before. Perhaps from that 80s movie, where the main character, a pilot, says, "What's the vector, Victor?" Or maybe you remember the term from your high school physics or math class. No matter your previous history with the word, it's important to know how it's defined for data science.

In artificial intelligence, a vector is a mathematical point that represents data in a format that AI algorithms can understand. Vectors are arrays (or lists) of numbers, with each number representing a specific feature or attribute of the data. They are housed in vector stores and used in vector databases in AI applications.

For example, if we wanted to create a vector store focused on animals, each animal would be represented by a vector that identifies attributes such as their size, speed, and lifespan. A small, fast animal with a short lifespan might be represented by [1, 10, 3] (1 for small size, 10 for high speed, and 3 for short lifespan).

Another example could be a car. Imagine you're analyzing different cars based on certain features. Each type of car is a vector, where each number corresponds to one of these features: speed, fuel efficiency, and price.

A vector for a particular car might look like this: [180, 30, 25000].

This means:

- The car has a top speed of 180 mph.

- It has a fuel efficiency of 30 miles per gallon.
- It costs $25,000.

So, a vector in data science helps you understand and analyze an object (like a car) in a consistent, structured way. In simple terms, vectors are numerical representations of data. Essentially, all types of data, whether they are tables, text, images, videos, music, or sounds, can be transformed into multi-dimensional numerical arrays and fed into a ML model where they can be manipulated to be stored in multi-dimensional spaces like vector databases.

How are vectors stored so they can be used by AI?

Two phrases that often come up when discussing how vectors are stored are "vector store" and "vector storage." It’s important to point out that these terms are closely related but can be distinguished based on their usage and context in data science and AI.

Vector storage

"Vector storage" is a more general term that refers to the method or process of storing vector representations of data. It encompasses the hardware, software, and algorithms used for storing them, but not necessarily within a specific vector database. It’s more about the concept and practice of handling vector data so it can be efficiently used for various computational processes.

Example: In a broader data management system, vector storage might refer to how user profiles are stored across different servers or in the cloud.

Vector stores

A "vector store" refers to an actual system or platform that is designed to handle the complexities and specifics of vector data, often in association with a vector database, and is used in applications like machine learning, AI, and data analytics. Vector stores can sift through enormous amounts of data at lightning speed, making them powerful tools for finding, organizing, and suggesting content based on similarities when used in vector databases.

But they're not just digital storage spaces, they’re smart systems that understand, categorize, and connect pieces of digital content in a sophisticated and efficient way. They’re also scalable, being able to handle large volumes of vectors, which make them ideal for extensive vector databases of animals, cars, and a wide variety of other data.

MongoDB is an example of a vector database that enables you to store vectors with their metadata. Unlike dedicated standalone vector storage options, however, the platform enables searching across vectors alongside all your other application data for fast and easy retrieval of complex query combinations that are unique to your use case. This is important because vectors are not your

only data type, even within generative AI applications, and prevents the need to stitch together various tools.

Use case for vector stores

As stated earlier, vectors are lists of numbers where each number represents a specific feature or attribute of the data. Let's take a closer look at how that works with a real life scenario.

Example: When you look for a recipe with certain characteristics (like low-calorie sauce and gluten-free pasta), the vector store conducts a vector search in a vector database that compares these preferences with stored recipe data. It calculates which recipes are most similar to your preferences based on the numbers in the vectors. This is akin to flipping through the pages of a cookbook to find recipes that match your needs, but much quicker.

What makes vector stores and vector databases different from traditional data storage options?

When you hear about data management you might think about tidy organized tables. This kind of data is known as structured data because it can be contained in a tabular form. Traditional databases handle this type of data easily. With vector stores and vector databases however, almost 80% of the data is unstructured and needs a more complex storage system.

Examples of unstructured data are images, text (eg. : documents, tweets, or emails). As you have already learned, vector stores and vector databases are designed to handle complexity and fast processing, making them more suitable for AI applications.

What is the role of ML models, vector embeddings, and ANN in AI?

In the illustration at the top of this article, we showed how ML models and vector embeddings fit into the process of transforming raw data into vectors that can be used in vector databases.

Raw data comes first, followed by vectors, a machine learning model, and vector embeddings that are made available for use in vector databases. One additional vector-refinement process we didn't show in that illustration is the approximate nearest number (ANN), which occurs during the embedding step.

Let’s take a closer look to learn more about each step.

Inputting vectors into machine learning models

Through its unique architecture, MLMs process and train vectors. During training, the model adjusts its internal parameters to learn patterns and relationships between vectors.In the case of deep learning models, this process involves multiple layers of computation, each extracting and learning different features.

Generating embeddings

As the MLM processes the data, it transforms the initial, often simplistic, vector to produce sophisticated and informative vector representations, known as embeddings. In natural language processing, a word embedding vector captures not just the identity of a word, but also its relationship with other words, its context, and its usage.

Using embeddings

Once generated, a vector embedding can be used for a variety of tasks such as classification, prediction, recommendation, and more. They are particularly useful because they reduce the complexity while retaining the essential characteristics needed for these tasks.

Adding in ANN

Searching for a similar vector (like finding a word with a similar meaning or an image with similar content) in high-dimensional spaces strains systems, especially those with a vast amount of data -- this is where approximate nearest neighbor (ANN) comes into play.

- ANN algorithms are designed to quickly find "nearby" or "similar" data points in high-dimensional vectors, like those formed by embeddings.
- Instead of calculating the exact nearest neighbors, which is computationally intensive, ANN algorithms focus on finding a "good enough" solution.
- For example, in a recommendation system, user and item embeddings might represent users and products; the ANN algorithm can quickly identify

products that are close to a user's profile in the embedding space, thus providing personalized recommendations in vector databases.

Conclusion

In summary, vector stores and their integration in AI technologies represent a transformative leap in data management and processing. This is evident in a myriad of applications, from search engines to recommendation systems, and even in cutting-edge areas like facial recognition.

As we continue to advance in the realms of AI and ML, understanding and leveraging the capabilities of vector stores will be paramount in driving innovation and creating solutions that are both sophisticated and highly effective.

MongoDB is the leading developer data platform that stores and enables sophisticated search across various data types, including but not limited to vectors. Many leading organizations are building cutting edge generative AI applications on MongoDB – learn more about Atlas Vector Search.

Get Started With MongoDB Atlas

Try Free

MongoDB

English

About

Careers

Investor Relations

Legal Notices

Privacy Notices

Security Information

Trust Center

Support

Contact Us

Customer Portal

Atlas Status

Customer Support

Manage Cookies

Social

GitHub

Stack Overflow

LinkedIn

YouTube

X

Twitch

Facebook