JaguarDB

Vector Database

Frequently Asked Questions

1. What is a vector database?

   A vector database, also known as a vector index or vector search engine, is a type of database specifically designed to efficiently store, index, and retrieve high-dimensional vector data. This kind of database is well-suited for applications that involve similarity search, nearest neighbor search, and other tasks that require comparing vectors based on their distances or similarities.

   In a vector database, each data entry is represented as a vector in a high-dimensional space. These vectors can represent various types of data, such as images, audio features, text embeddings, molecular structures, and more. The goal of a vector database is to enable fast and accurate retrieval of vectors that are similar to a given query vector.

2. What is the difference between non-vector data and vector data?

   Non-vector data and vector data exhibit distinct characteristics that set them apart. Non-vector data is inherently inflexible, serving as representations of discrete attribute values associated with objects. Conversely, vector data arises from sources like advanced AI language models or embedding generators, carrying semantic significance and data intelligence.

   In addition to their semantic disparities, the structure of these two data types is also different. Vector data, existing in a high-dimensional space, contrasts with the conventional low-dimensional nature of non-vector data. The manipulation and analysis of these contrasting data types require disparate methodologies.

3. What are typical dimensions of vectors in AI?

   Typical dimensions of vector embeddings are 768, 1024, 1536, or higher.

4. Why processing vectors is hard?

   With large amounts of vectors, and each vector being high dimensional, vectors search such as K-nearest neighbor or similarity search can be extremely time consuming. Fast algorithms have to be invented and applied to such vector data processing.

5. Is JaguarDB a vector database library?

   No. It is not a library. It is a fully distributed vector database system.

6. Is JaguarDB a vector database?

   Yes, 100%. It is actually much more than a vector database. Not only can it handle vector data, but also data that are not vectors. The hybrid or multimodal capabilities of JaguarDB makes it much more powerful to handle real-world application scenarios.

7. Is JaguarDB an in-memory store for vectors?

   Yes and no. It has in-memory stores to perform fast search for vector data, as well as on-disk data store for persistence. It is a full-fledged distributed vector database system, that can handle multiple concurrent client connections, data sharding, fault tolerance, data replication, vector search, hybrid search, SQL-like full-spectrum data search and data aggregation. It also supports time-series, geospatial data for real-time AI such as autonomous driving and drone flights, general data indexing, and media storage in AI data lake systems.

8. Does JaguarDB manage vector data and non-vector data separately or in one place?

   Both vector data and non-vector data are very tightly integrated for fast search and lookup since in most use cases both types of data are closely related to each other. In JaguarDB, a table can have both non-vector columns and multiple vector columns.

9. What algorithm does JaguarDB use to handle vector data?

JaguarDB uses the state-of-the-art Hierarchical Navigable Small World (HNSW) graph data structure and algorithm to manage vector data. Yury Malkov is the author of the most adopted ANN algorithm HNSW.

10. Is JaguarDB relational or NoSQL database?

    Jaguar is NoSQL, meaning scalable with limited SQL features, focusing on scalability and performance.

11. What types of data does JaguarDB support?

    Vector data, feature vectors, embeddings, time series data (data that is associated with timestamps), geolocation or geospatial data, images, photos, videos, audios, files, and JSON data.

12. Does JaguarDB support indexes?

    Yes, it supports vector indexes, as well as general indexes. You can create as many indexes as you want for each table. In each index, you can duplicate some columns in the original table for fast data access.

13. Is JaguarDB distributed?

    Yes, data is distributed among multiple servers in the cluster, for elevated performance, with linear scalability in terms of performance and storage.

14. How many nodes can JaguarDB support?

    It can support up to 60,000 nodes as of release 3.3.5. This number will be further increased in future releases.

15. Is JaguarDB an in-memory database?

    No, JaguarDB is a persistent vector database. Memory is used for temporary caching and computing in Jaguar. Of course, JaguarDB runs faster if there is more memory on each node.

16. Is JaguarDB optimized for SSD?

Yes, Jaguar runs on HDD as well as on SSD. On SSD, it can use less DRAM in the system.

17. What language is used in JaguarDB?

    JaguarDB server is written with C++. Client API is supported with Java, Scala, NodeJS, PHP, Python, etc.

18. I have started Jaguar server, why jag has connection failures?

    When JaguarDB starts up, it may take a while for it to reload existing data. After a few seconds, jag should be able to connect properly.

19. I still get connection error on jag, what happened?

    Make sure the password has length of at least 12 characters.

20. I copied the commands from the Web or the document, why do I get errors?

    The Web pages or the document may contain special Unicode characters. You should make sure that the commands are in pure ASCII format.

21. When I run Java client programs, I get error of “Unable to load library”. Why?

    Make sure you have included the $JAGUAR_HOME/jaguar/lib in LD_LIBRARY_PATH so that the libJaguarClient.so file can be found by your Java program.

22. When I start and stop JaguarDB server, should I do this on each node?

    No. You can run the jaguarstart_on_all_hosts.sh script from just one host. The script jaguarstatus_on_all_hosts.sh checks the status of JagaurDB on all nodes in the system.

23. I have multiple network cards on my server, how can I connect to JaguarDB?

If you have multiple NIC cards and IP addresses, you must specify which IP address your JaguarDB should listen to. This is specified by LISTEN_IP in conf/server.conf. When your client connects to JaguarDB, it also must use the same IP address.

24. How many cores does JaguarDB require?

    JaguarDB does not have requirement on the number of cores. However, more cores will enable JaguarDB to accept more client connections and speed up its read performance.

25. What is the jagexportsql program?

    You can execute this script to pull all data records from all nodes and save data in a single file. You just need to run it from one node. The data can be imported to database with the jagimportsql program.

26. What is the jagexport program?

    This program reads all data from a table and writes the data to a temp file in the export directory. The table can be dropped, recreated with a new schema. The jagimport program can be executed to reload the data from the temp file to the table.

27. How can I keep snapshot of tables?

    There are local backup settings in the conf/server.conf file which can specify the interval for taking snapshots of all tables in JaguarDB.

28. Does one of JaguarDB nodes goes down, will JaguarDB continue working?

    Yes, JaguarDB cluster has fault-tolerance support in which if one node goes down or several hosts have hardware failure, the JaguarDB cluster will continue to work. Inserting new data and querying data will still work as usual.

29. How many replicates does JaguarDB offer for each data record?

JaguarDB can store a maximum of three copies for each data record. The replication factor is set by the REPLICATION parameter in conf/server.conf file.

30. What is JaguarDB in terms of CAP theory?

    Jaguar is an AP system. In case of hardware failure or network glitch, JaguarDB favors availability over consistency. It offers eventual consistency.

31. How long it will take to install JaguarDB on a cluster of 100 hosts?

    A few minutes. Suppose you have a user account on all the 100 hosts and the account password is the same on the hosts, then you just need to provide a file listing all the 100 hosts and execute one shell script, which will automatically install JaguarDB software on all the 100 hosts. Some ssh servers do not allow login with passwords, in such case you need root admin to enable "PasswordAuthentication yes" in the /etc/ssh/sshd_config file and restart sshd server.

32. I have Windows server, how can I run JaguarDB server?

    JaguarDB does not support Windows now.

33. I have JaguarDB installed on Centos 7 hosts, but I cannot connect to the JaguarDB server. What went wrong?

    Probably Centos 7 firewall is blocking connections to port 8888. You need to open the port 8888 for connections (run as root or sudo):

    ```
    # firewall-cmd --zone=public --add-port=8888/tcp --permanent
    # firewall-cmd --reload
    # systemctl restart firewalld.service
    ```

34. For time series data, how many time windows can I create?
    You create as many as you want. For example, you can setup 5-minute windows, 15- minute windows, hourly windows, daily windows, 7-day windows, etc. There is no limit.

35. What is a tick?

A tick represents a time window. For example, a 5-minute window is a tick. A 3-hour window is another tick. A tick is one of the segments along the timeline.

36. When can I query the data in the ticks?
    You can query the aggregation data in the tick as soon as the time series data is written to the base tables.

37. What timestamp is used in the tick tables?
    The timestamp in the tick tables is the starting time of the window. For example, an hourly tick has the start time of the hour, i.e., the minute and second of the timestamp are always zero.

38. Can I use the window function on the base tables?
    Yes, you can use the window function on the base as well as on the tick tables.

39. What are the metrics in the location data?

    These are metrics data at each location. Other databases might allow you to store only one metric value at each location. JaguarDB allows you to store unlimited metrics at each location. For example, at each location, you can store temperature, wind speed, air quality, population density, traffic speed, average traffic congestion index, zone info, municipal info, geopolitical info, crime data, school data, demographics info, income level, etc.

40. Can I combine location query and time query together?

    Yes, JaguarDB natively supports location data and time data, both data being stored in the same table in the same file. So naturally it allows for reading both types data in one query.

41. Can I combine vector search and non-vector search?

    Yes, the hybrid or multimodal search capabilities of JaguarDB allow you to perform search of different types of data.

42. Is time series data also distributed?

Yes.

43. Is location data distributed?
    Yes. Distribution of data allows high capacity and performance.

44. Is vector data distributed in JaguarDB?

    Yes, vector data, like other types of data, is also distributed with the ZeroMove hashing technique in JaguarDB.

45. How can I manage billions of vectors?

    Due to memory resource requirements to store the vectors in HNSW structure for fast search, it requires multiple nodes to manage billion vectors. JaguarDB, with its zeromove scaling technology, can effortlessly scale out to manage billons of vectors for AI applications.

46. How can I manage billions of IoT devices?
    You should create a device table that uses uuid for tracking devices:

```
create table mydevices
key: deviceid uuid, value:  name char(8), model char(4), …);
```

    where “deviceid” will uniquely represent your devices and other value fields will be properties of your devices. Using uuid as the key will be extremely efficient in searching your devices. Please refer to the IoT example PDF for such large scale IoT applications.

47. How can I manage billions of photos or videos?

    It is recommended that you use the zeromove uuid as the first key, such as photoid, videoid, fileid as uuid keys. Searching such items will be extremely fast in JaguarDB.

48. During insert of new records, how can I get the zeromove UUID value of a record?

Immediately after you have executed the insert statement, you can call the getLastUuid() method on the client handle to get the value of zeromove uuid.

49. JaguarDB has geometric shapes like square, circle, rectangle, cube, etc. What are those for?

    In autonomous driving and real-time AI applications, representing and recognizing objects are essential functions. For geometric shapes in real world, you can always use multiple points on the shape to represent a polygon corresponding to a shape. However, that is not cost effective. If a shape is a square, you can use only three data elements to represent a square: location of its center with (x,y) coordinates, and the length of its side. That is, JaguarDB stores only three data elements for squares, instead of multiple points on the perimeter of the square as a polygon. The same technique with circles, ellipses, cubes, etc.

50. How can I use vectors, files, keywords to manage and search AI data?

    There is an example program in the JaguarDB github project directory: examples/ ai_diagnosis.sql which delineates the structure for managing patient visits, symptoms, ECG diagrams, Xray images, MRI images, CAT scan images, and their embedding vectors for similarity search and keyword search. It allows users to search similar symptoms with predefined qualifiers.