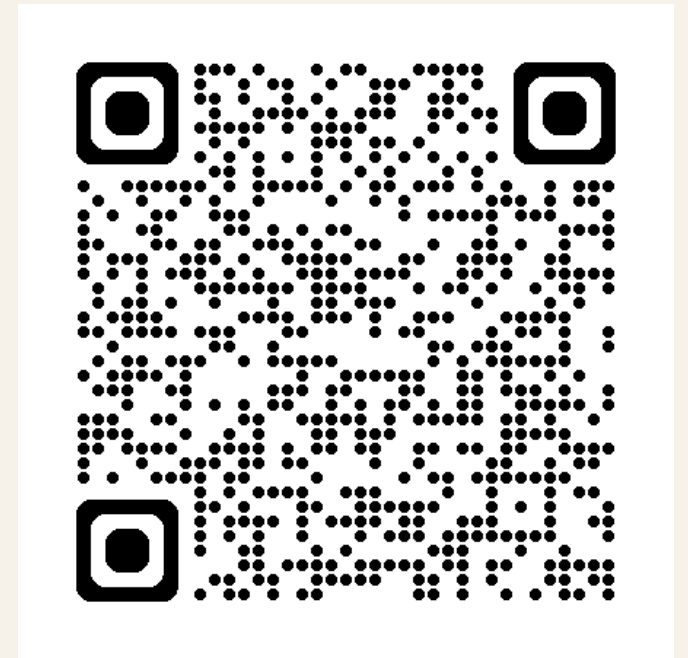


Hybrid Semantic Web with MongoDB and Vector Search using LangChain.js and Phi-3

Piti Champeethong (Fyi - รุตฟี)
MongoDB User Group Leader



MongoDB Thailand User Group

Agenda

1. Semantic Web
2. MongoDB Atlas Vector Search
3. Demo

Semantic Web

Smart Search

Example: When you type **jaguar** into a search engine, the Semantic Web helps the computer determine if you mean the **animal** or the **car brand** based on the **context**, so it shows you the correct information.

Connected Related Ideas

Example: If you're reading about **space**, the Semantic Web can suggest links to **planets**, **astronauts**, or **rockets** because it knows these topics are connected.

Understanding Meanings

Example: When you ask a voice assistant, **Show me pictures of bats**, the Semantic Web helps it **understand** whether you mean the **flying animal** or the **baseball equipment**, so it displays the correct images.

MongoDB Atlas Vector Search

MongoDB Atlas Vector Search allows you to perform efficient **similarity searches** by storing and querying data as **vectors** within your MongoDB database. In our image search example, it lets users find visually similar images without relying on keywords or tags, enhancing the functionality and user experience of your app.

```
{
  "_id": "image1.jpg",
  "imageData": "...",
  "embeddings": [0.12, -0.07, 0.85, ...]
}
```

```
db.images.aggregate({
  $vectorSearch: {
    index: "vector_index",
    queryVector: queryEmbeddings,
    path: "embeddings",
    exact: true,
    limit: 5
  }
});
```

Vector Search Score

Cosine Similarity

Formula: $\text{score} = (1 + \cos(\text{v1}, \text{v2})) / 2$

Use case: Text similarity, recommendation systems

Dot Product

Formula : $\text{score} = (1 + \text{dotProduct}(\text{v1}, \text{v2})) / 2$

Use case: Weighted scoring, collaborative filtering

Euclidean

Formula : $\text{score} = 1 / (1 + \text{euclidean}(\text{v1}, \text{v2}))$

Use case: Clustering, anomaly detection, geospatial data

Vector Search Queries

Approximate Nearest Neighbor (ANN)

Formula: $\text{score} = (1 + \cos(\text{v1}, \text{v2})) / 2$

Use case: Text similarity, recommendation systems

Exact Nearest Neighbor (ENN)

Use case:

- Query less than 10,000 documents without having to tune the number of nearest neighbors to consider.
- You want to include selective pre-filters in your query against collections where less than 5% of your data meets the given pre-filter.

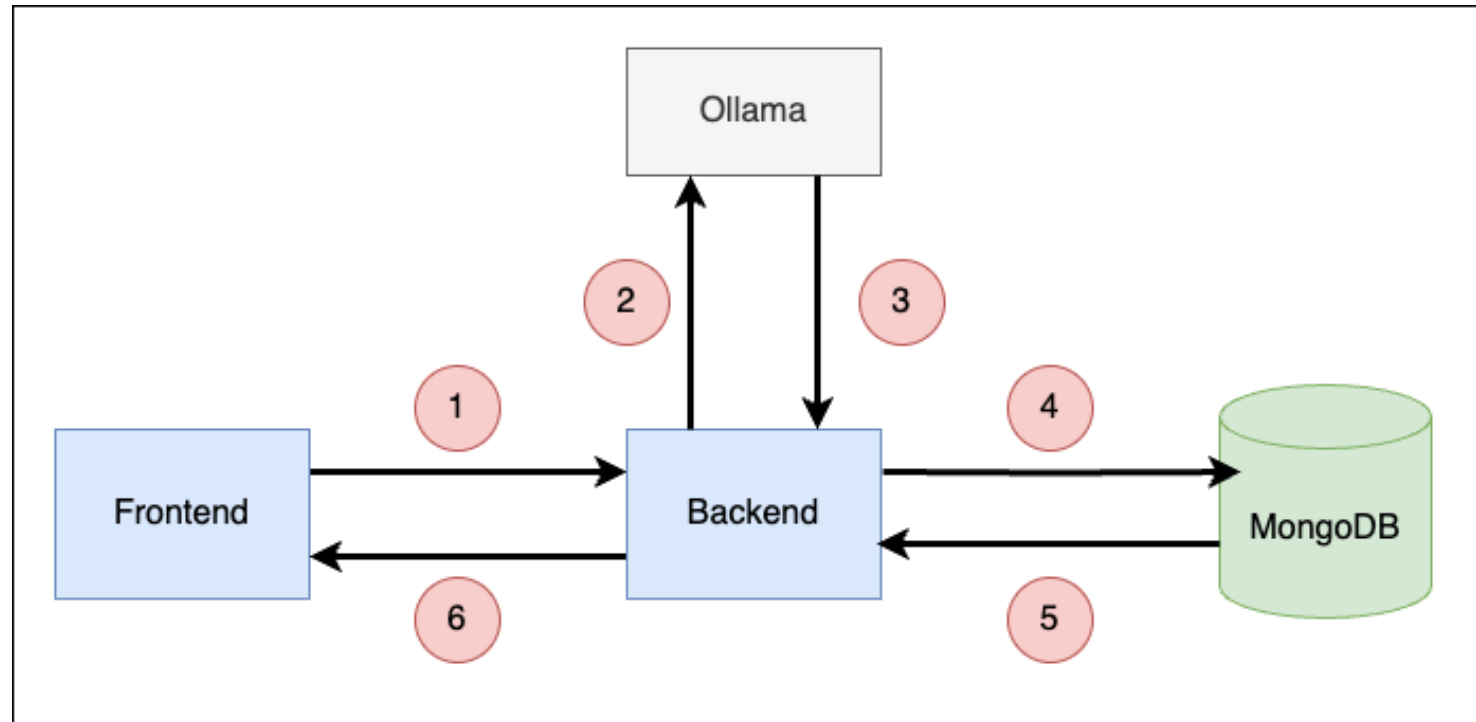
Reciprocal Rank Fusion (RRF)

Formula : $\text{reciprocal_rank} = 1 / (r + \text{rank_constant})$

$\text{weighted_reciprocal_rank} = w \times \text{reciprocal_rank}$

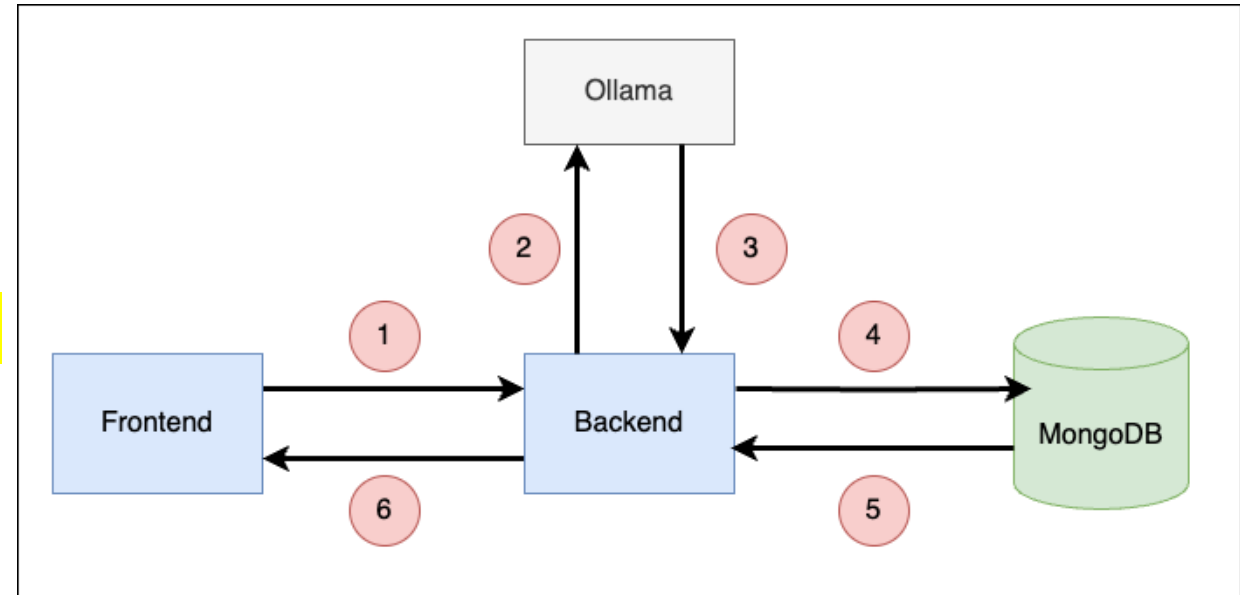
Demo

1. Based on MacOS
2. Atlas CLI: <https://www.mongodb.com/docs/atlas/cli/current/atlas-cli-local-cloud/>
3. Ollama: <https://ollama.com/download/mac>



Demo (contd.)

1. Frontend send a query to Backend.
2. Backend send a query to Ollama for embedding.
3. Ollama send a embed query to Backend.
4. Backend send embed query to MongoDB for Vector search query.
5. MongoDB send a raw result to Backend.
6. Backend send a formatted result to Frontend.



Resources

- https://en.wikipedia.org/wiki/Semantic_search
- <https://www.w3.org/2001/sw/>
- <https://www.scientificamerican.com/article/the-semantic-web/>
- <https://www.mongodb.com/docs/atlas/atlas-vector-search/tutorials/reciprocal-rank-fusion/#run-a-combined-semantic-search-and-full-text-search-query>

Thank you and Happy Coding!