DATA+AI SUMMIT BY databricks

MOSAIC AI VECTOR SEARCH: WHAT, WHY AND HOW

Sonali Guleria, Ankit Vij
06/13/24

“Ever tried finding a slack message but couldn't remember the exact terms?“

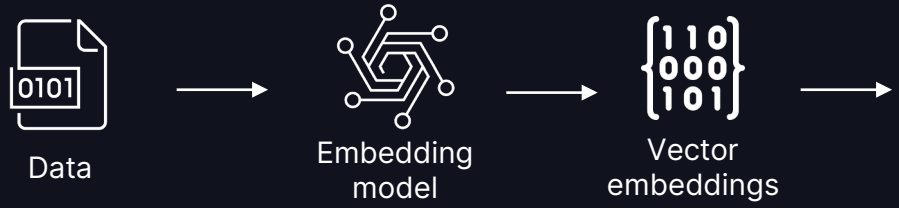
Agenda

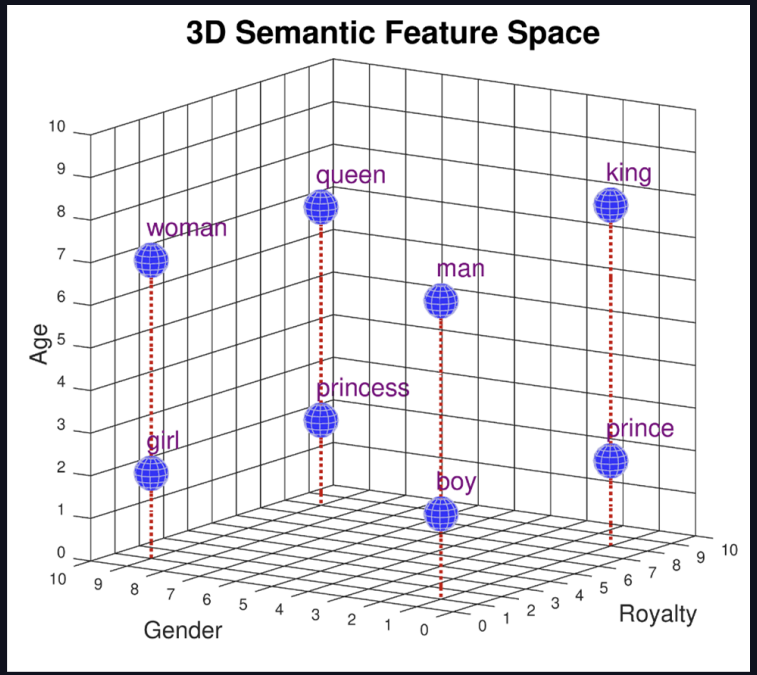
- Introduction to Vector Search
- When not to use?
- What is Mosaic AI Vector Search
- Best Practices for Path to Production
  - Data Ingestion
  - Vector Search Retrieval
- Conclusion
- Q/A

Vector Search

Search with understanding

Search based on semantic closeness or inherent characteristics using embeddings





Source: https://www.cs.cmu.edu/~dst/WordEmbeddingDemo/tutorial.html

Vector Search

Use Cases

- RAG systems like chatbots
- Recommendation systems
- Anomaly Detection
- Image and Video Recognition
- Drug Discovery
- “Classic” use-cases such as Personalization, product search
- and so on…

What it is not

Not an exact match

- Standard Database Look-up
- Conventional keyword search
  - Metrics
  - Aggregations
  - Other Observability…

Main Components for Vector Search

Ingestion

- Fast and Fault Tolerant
- Real-time Updates and Change Detection
- Effective Data Preprocessing
- Highly Scalable

Retrieval

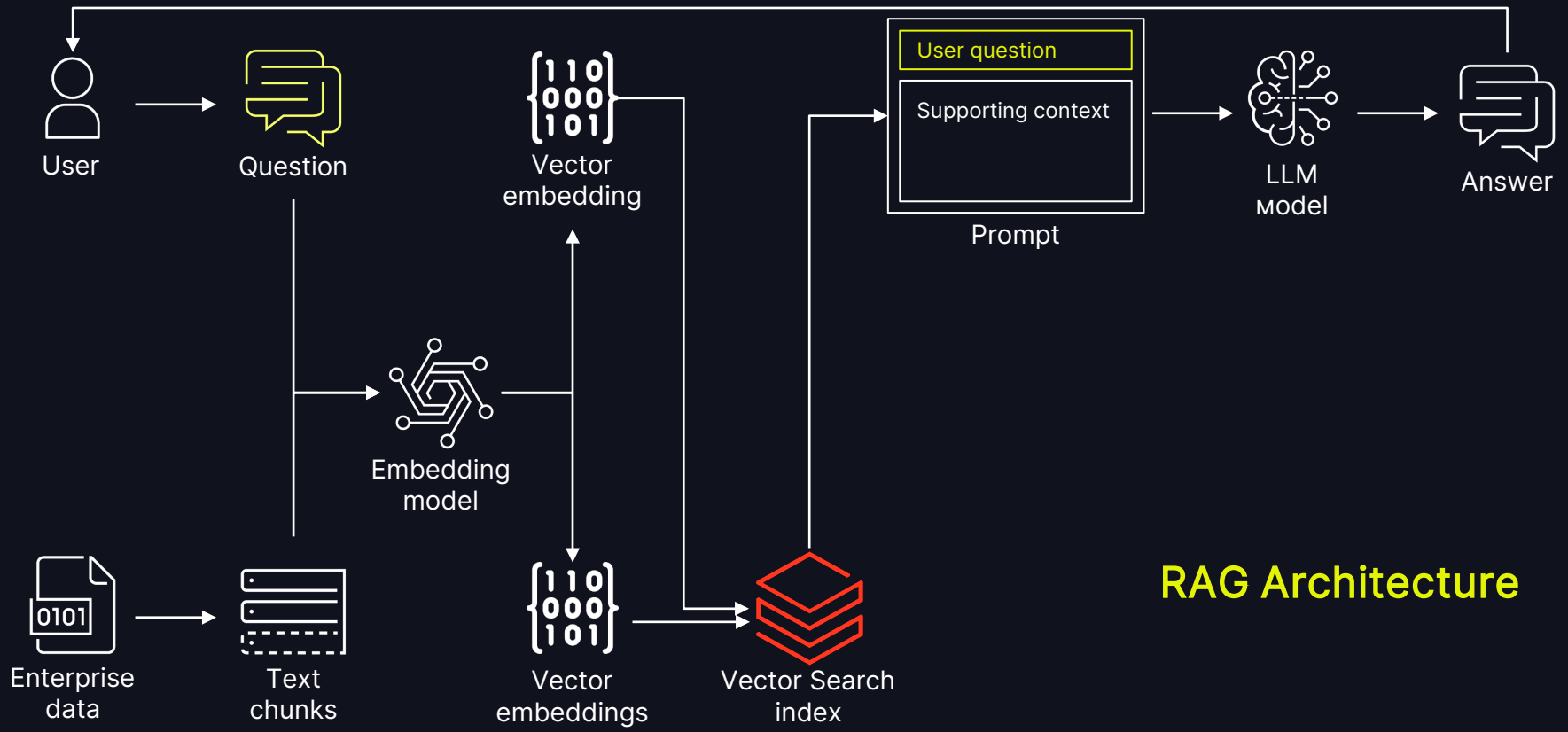
- Quality results with high Recall
- Low Latency
- Effectively use the Metadata as filters
- Additional capabilities like Hybrid search

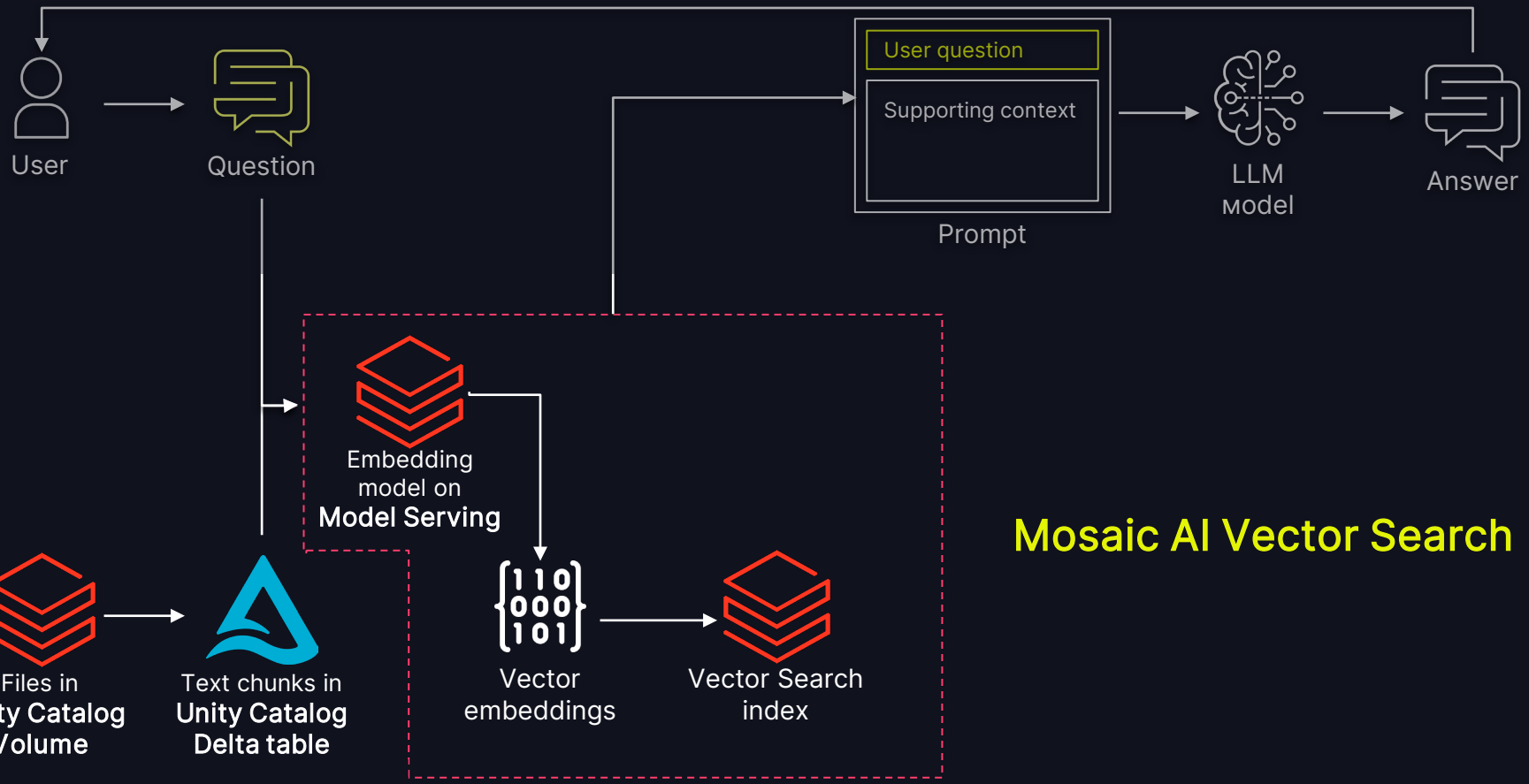
Mosaic AI Vector Search

Simplifying Semantic Search

Serverless and Highly Performant Search Engine designed for Production applications.

- Simple and Fully Managed
- Native Lakehouse Integration
- Fast and Scalable
- Built in Governance via Unity Catalog

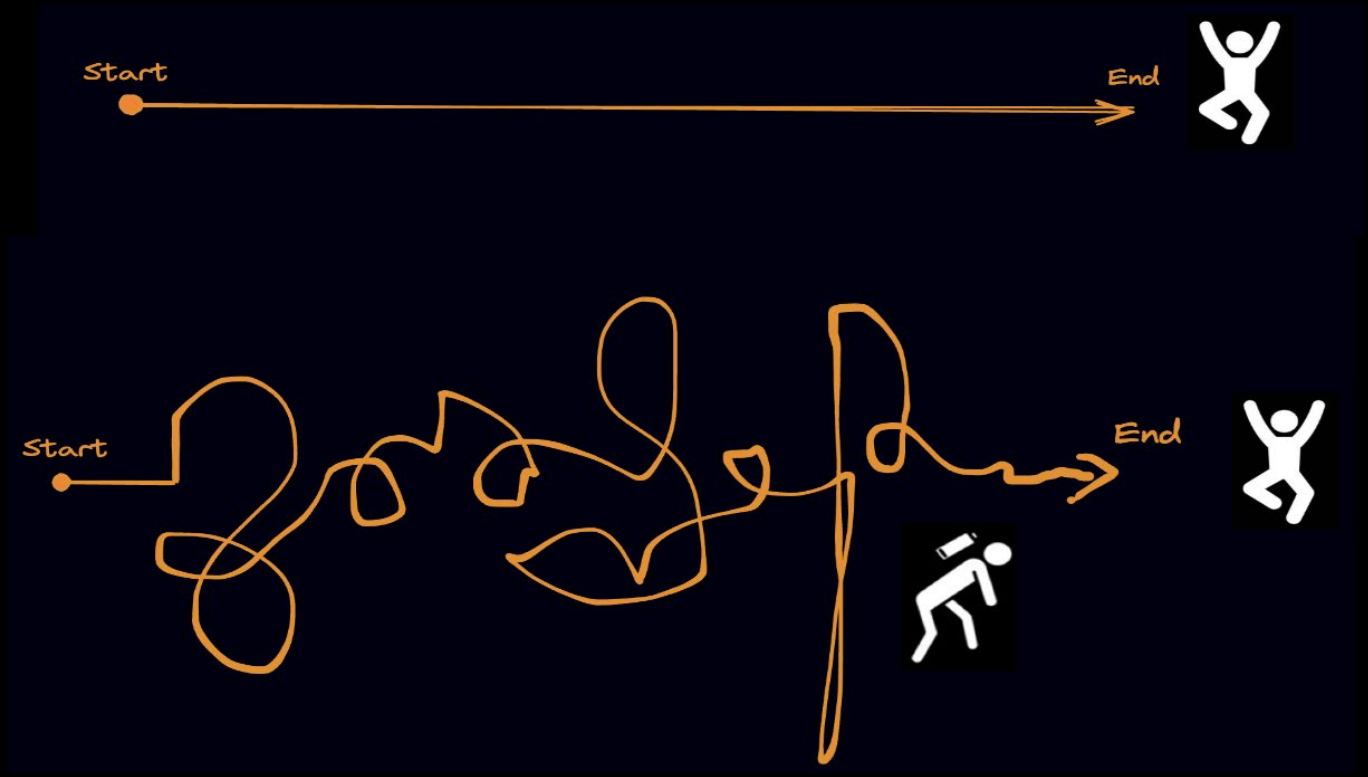




Mosaic AI Vector Search

But Path to Production is Hard…

Expectation vs Reality

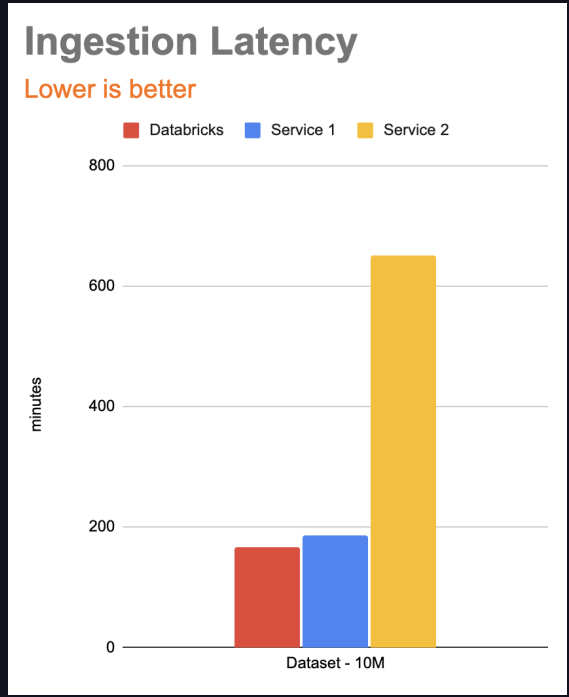


Managing & Scaling Ingestion

Databricks does all of the heavy lifting

- Use Delta Sync Indices
  - Auto-syncs w/ source Delta Table
  - Auto-scales for different dataset sizes
  - Columns in Delta Table are filters
  - Integrated with Model Serving for embedding generation

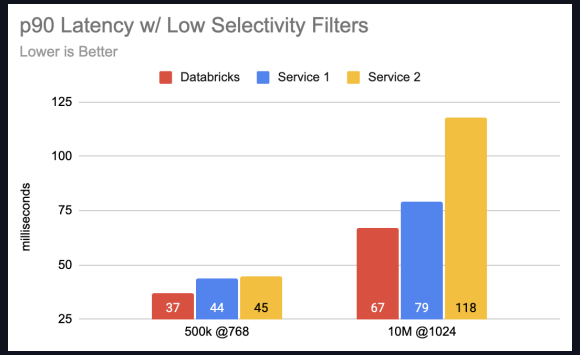
- Triggered vs Continuous
- Be cautious about overwriting source tables
- Assess the performance of Model Serving endpoints

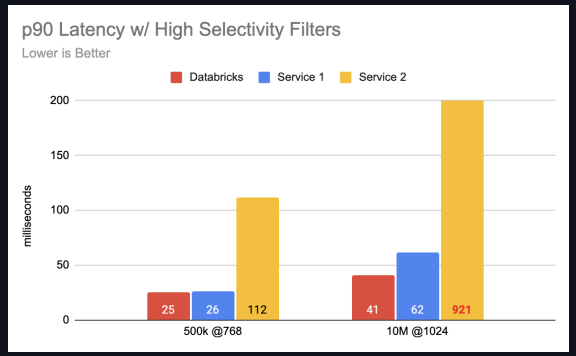


* As of June 2024

Improving Retrieval Efficiency

Improving query latency





- Identify the source of latency
- Be cognizant on fetching large number of results and additional metadata columns
- Ensure you are using the latest SDK version
- Use Service Principals for network optimized routes in Production.
- Assess the performance of model serving endpoints

* As of June 2024

Mastering Chunking

Elevate Your Search

- Chunk size- Too Big or Too Small?
- Fixed vs Dynamic size
- Element based- by section, table or sentence
- Semantic Chunking
- Include metadata with chunks (especially unstructured data)
- Chunking Overlays

Credit Line Increase?

Incorrect Chunking Example

User: Can I get a credit line increase?

System: You have a pre-approved offer to increase your credit limit.

Actual Response

Depending on the evaluation of your profile, you may be eligible for a credit line increase. You will have a better chance of an increase if you meet the following requirements:

- If you have a good credit history
- You do not have any late payments in the last 3 years.
- You have a high income relative to your existing debt obligations

- *You have a pre-approved offer to increase your credit limit.*

Embedding Model Selection

Selecting the Right Model for High Quality Embeddings

- Different data types needs different embedding models.
- (Re)Verify the Embedding Model:
  - Similar Training Data
  - Truncated chunks- check context window
  - Dimension Size
- Task specific Models.
- Check for Normalization.

Retrieval English leaderboard

- **Metric:** Normalized Discounted Cumulative Gain @ k (ndcg_at_10)
- **Languages:** English

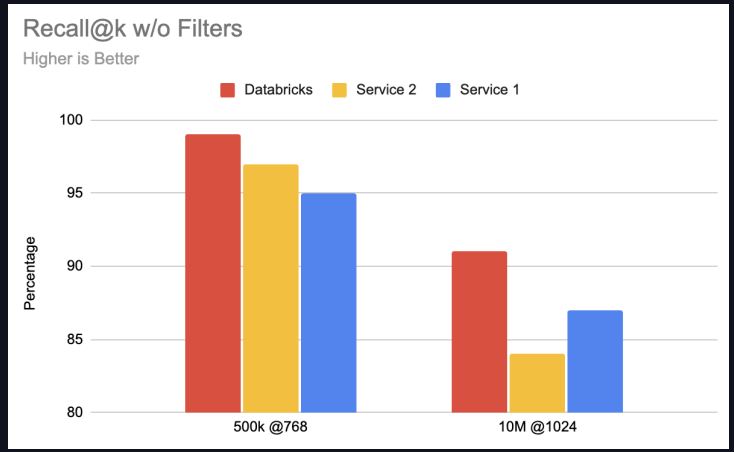
Rank	Model	Model Size (Million Parameters)	Memory Usage (GB, fp32)	Average
1	NV-Embed-v1			59.36
2	SFR-Embedding-Mistral	7111	26.49	59
3	voyage-large-2-instruct			58.28
4	gte-large-en-v1.5	434	1.62	57.91
5	GritLM-7B	7242	26.98	57.41
6	e5-mistral-7b-instruct	7111	26.49	56.89
7	LLM2Vec-Meta-Llama-3-supervis	7505	27.96	56.63
8	voyage-lite-02-instruct	1220	4.54	56.6
9	SE_v1			56.55
10	gte-Qwen1.5-7B-instruct	7099	26.45	56.24

HF MTEB leaderboard

Enhancing Retrieval Accuracy

Efficient Recall

- Hybrid Search.
- Use filtering on accompanying columns or metadata.
- Re-ranking.
- Verify the embedding model for questions- automatic for Managed delta sync.



Some More Best Practices

Path to Production

- Use Unity Catalog to leverage unified governance.
- Evaluate effectively.
- Leverage Lakehouse Monitoring and system tables.
- Enable automatic Inference to log and audit the requests and responses.

- Make sure the model endpoint is properly configured and has enough resources.

Conclusion

Empowering Innovation, Unlocking Potential, and Shaping the Future of Intelligent Applications

- Vector Search is still a data and ML problem.
- Iteration is going to be the key.
- Evaluate and Re-evaluate methodology.

- Pick a great technology partner, who can solve majority of the scalability and performance issues for you.

Call to Action

Gen AI Cookbook

https://ai-cookbook.io/

RAG QuickStart

Thank you!