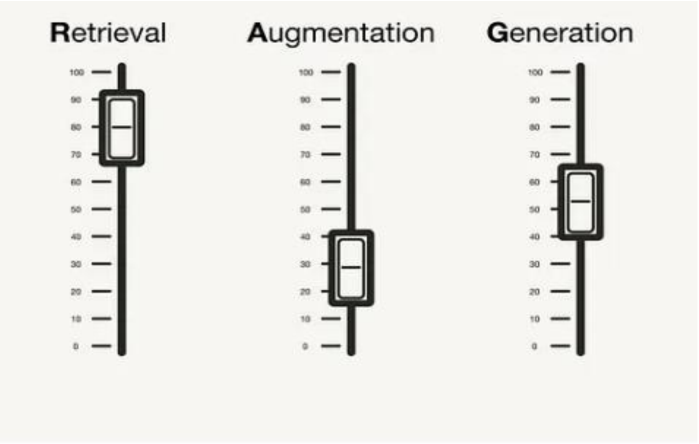


Tuning Strategies for Retrieval-Augmented Generation Applications

Tuning Strategies for Production-Ready RAG Applications

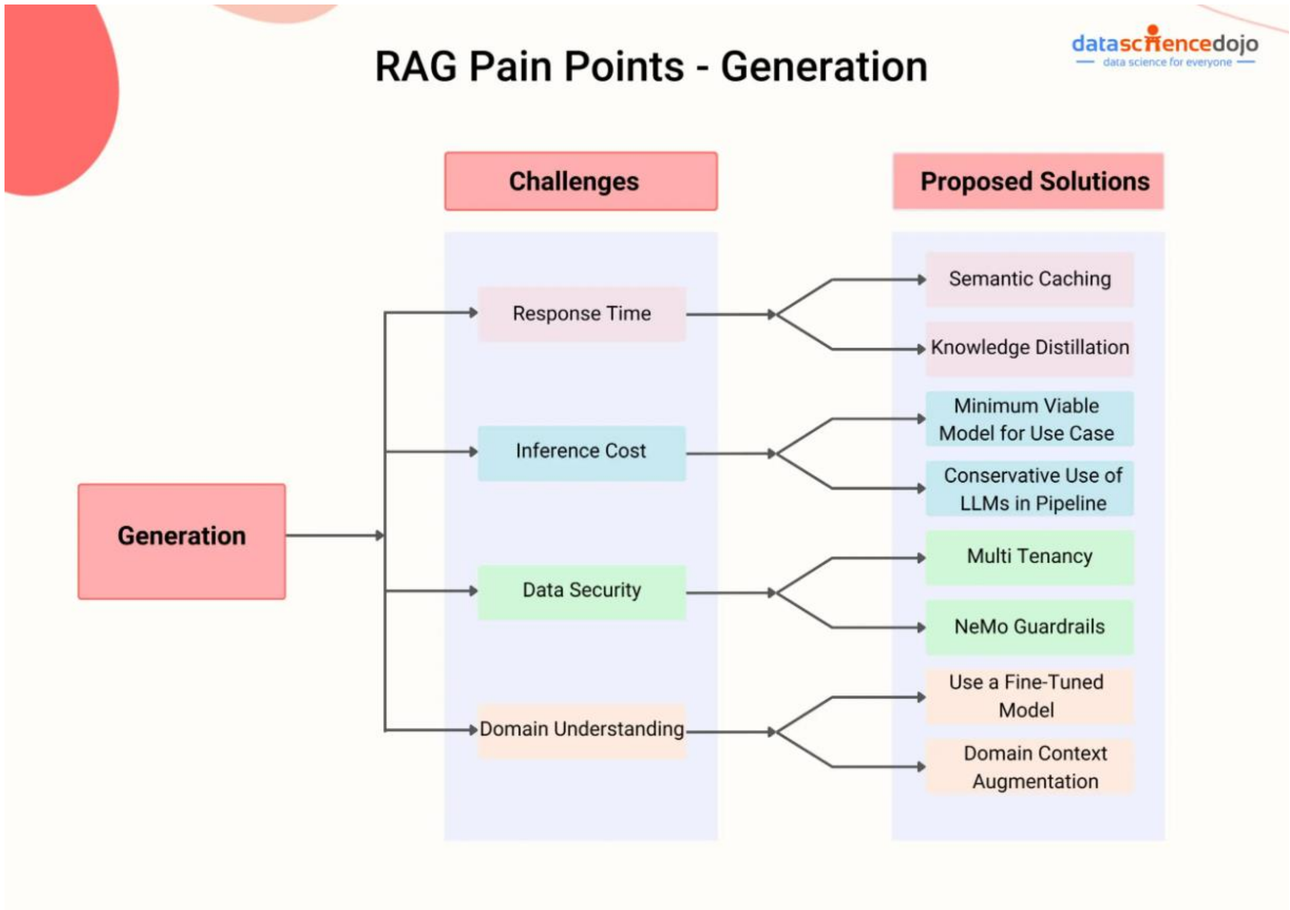
Karn Singh

Tuning Strategies for RAG Applications

Explore hyperparameters and strategies to enhance your RAG pipeline across different stages, focusing on text-based applications.

Stages:

- Ingestion stage
- Inferencing stage (retrieval and generation)



Ingestion Stage Enhancements

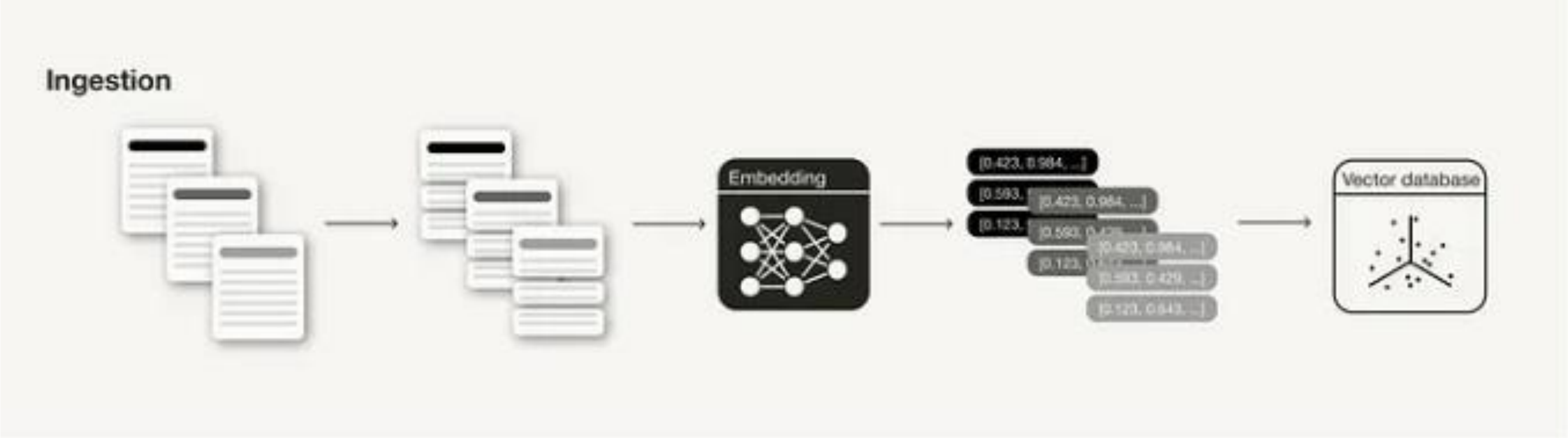
Data Cleaning: Ensuring data quality by removing inconsistencies and encoding errors.

Chunking: Adjusting chunk size and overlap based on the application (e.g., summarization vs. question answering).

Embedding Models: Selection and tuning of embedding models affect the precision of retrievals, possibly requiring domain-specific adjustments.

Metadata and Multi-indexing: Enhances retrieval by organizing data more efficiently, allowing for quicker and more relevant results.

Indexing Algorithms: Use of ANN for fast retrieval, potential fine-tuning of parameters like vector compression for optimal performance.



Ingestion stage of a RAG pipeline

Retrieval Stage Optimization

Query Transformations: Modify queries for better search results using LLMs. You can experiment with various query transformation techniques.

1. **Rephrasing:** Use an LLM to rephrase the query and try again.
2. **Hypothetical Document Embeddings (HyDE):** Use an LLM to generate a hypothetical response to the search query and use both for retrieval.
3. **Sub-queries:** Break down longer queries into multiple shorter queries.

•**Retrieval Parameters**: Adjust search parameters, considering hybrid search techniques.

•**Advanced Retrieval Strategies**: Implement strategies like sentence-window or auto-merging retrieval for context enhancement.

Data Cleaning (Ingestion Stage)

Objective: Ensure high-quality, reliable data for effective retrieval.

How to achieve it:

- Employ basic NLP cleaning techniques.
- Correctly encode special characters.
- Maintain factual accuracy to prevent data conflicts.
- Regularly update and validate data sources.
- Remove duplicate or irrelevant information.

Chunking (Ingestion Stage)

Objective: Optimize document breakdown for better context retrieval.

How to achieve it:

- Segment long documents into smaller, coherent units.
- Adjust chunk sizes based on specific application needs (e.g., QA or summarization).
- Implement overlapping windows for continuity in data.
- Choose chunking techniques appropriate for the data type (code, prose, etc.).
- Ensure chunks provide sufficient context without excess irrelevance.

Embedding Models (Ingestion Stage)

Objective: Enhance the precision of retrieval through better vector embeddings.

How to achieve it:

- Select embedding models that suit the domain and data specificity.
- Consider high-dimensionality models for increased precision.
- Explore fine-tuning opportunities to tailor models to specific needs.
- Evaluate models using benchmarks like the MTEB Leaderboard.
- Stay updated on limitations regarding the fine-tuning capabilities of certain models.

Metadata (Ingestion Stage)

Objective: Utilize metadata to enhance searchability and retrieval accuracy.

How to achieve it:

- Incorporate useful metadata such as dates, authors, or tags.
- Use metadata for filtering and refining search results.
- Store metadata alongside vector embeddings for easy access.
- Develop a schema for consistent metadata application.
- Regularly review and update metadata to ensure relevance.

Multi-indexing (Ingestion Stage)

Objective: Manage diverse document types with tailored indexing strategies.

How to achieve it:

- Implement separate indexes for different types of content.
- Design index routing logic for efficient retrieval.
- Use multi-indexing to improve search performance and specificity.
- Continuously evaluate the effectiveness of indexing structures.
- Explore native multi-tenancy for enhanced data organization.

Indexing Algorithms (Ingestion Stage)

Objective: Optimize search algorithms for faster and more accurate retrieval.

How to achieve it:

- Choose between ANN and kNN based on precision needs.
- Tune parameters like efConstruction and maxConnections in HNSW.
- Consider vector compression to balance precision and storage.
- Evaluate different ANN algorithms like Faiss, Annoy, and ScaNN.
- Keep abreast of industry benchmarks to guide algorithm selection.

Query Transformations (Inferencing Stage)

Objective: Enhance the effectiveness of search queries.

How to achieve it:

- Rephrase queries for better accuracy using LLMs.
- Implement Hypothetical Document Embeddings (HyDE) for deeper context.
- Break down complex queries into simpler sub-queries.
- Experiment with different phrasing styles.
- Analyze the impact of transformations on retrieval outcomes.

Retrieval Parameters (Inferencing Stage)

Objective: Fine-tune retrieval settings to optimize search results.

How to achieve it:

- Adjust the alpha parameter to balance semantic and keyword searches.
- Set optimal numbers for search result retrieval.
- Decide on the similarity measures best suited for the embeddings.
- Experiment with hybrid search configurations.
- Monitor the impact of parameter changes on search quality.

Advanced Retrieval Strategies (Inferencing Stage)

Objective: Implement sophisticated retrieval techniques for enhanced accuracy.

How to achieve it:

- Use sentence-window retrieval for broader contextual capture.
- Apply auto-merging retrieval for consolidated context from related chunks.
- Explore the feasibility of more complex retrieval frameworks.
- Continuously test and refine advanced strategies.
- Study the latest research for potential implementation.

Re-ranking Models (Inferencing Stage)

Objective: Improve the relevance of search results post-retrieval.

How to achieve it:

- Deploy models to reassess the semantic relevance of results.
- Fine-tune re-rankers to specific use cases.
- Decide on the number of contexts for input into re-ranking models.
- Evaluate the impact of re-ranking on final output quality.
- Consider proprietary vs. open-source re-ranking solutions.

LLMs and Prompt Engineering (Inferencing Stage)

Objective: Optimize response generation through tailored LLMs and prompts.

How to achieve it:

- Select the appropriate LLM for specific needs and constraints.
- Design prompts that effectively guide LLM responses.
- Use few-shot examples to improve completion quality.
- Fine-tune LLMs to capture desired tone and style.
- Monitor and adjust the number of contexts used in prompts.