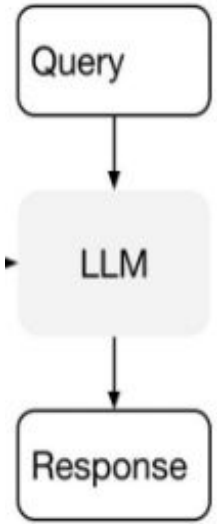
Diving Deep into RAG: How Retrieval Augmentation Transforms Language Models

Introduction

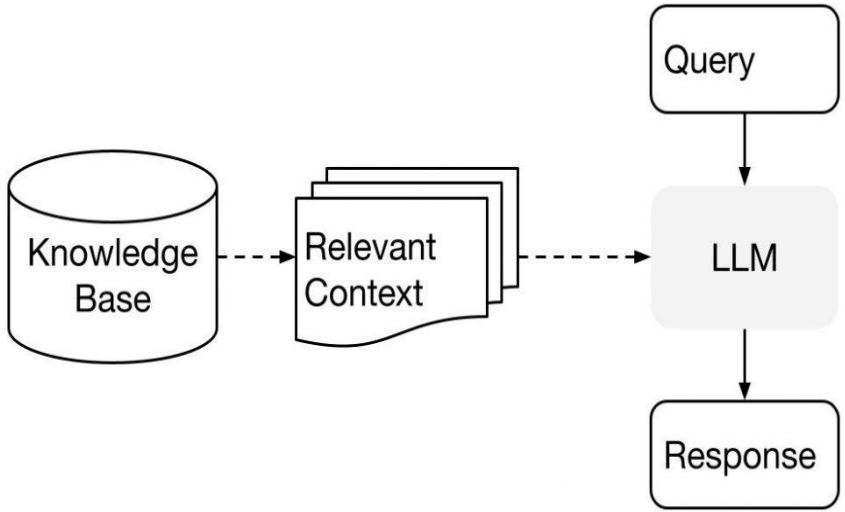
- RAG offering a solution to enhance the capabilities of LLMs and enabling them to provide responses based on information they have not been explicitly trained on.
- Prompt Engineering
- RAG
- Fine Tuning



crossml

What is RAG?

- RAG offering a solution to enhance the capabilities of LLMs and enabling them to provide responses based on information they have not been explicitly trained on.
- Prompt Engineering
- RAG
- Fine Tuning



crossml

Components of RAG

- Key components of RAG, including retrieval mechanisms, document encoders, and generation models.
- Two main components of RAG
- Retrieval Based Model
- Lack ability to generate or creative or novel content
- Generative Based Model
- Train on huge data, learn pattern in NLP
- But struggle with factual accuracy or relevance to the specific context

Retrieval Mechanisms

- Different types retrieval mechanisms used in RAG, such as sparse retrieval, dense retrieval, and hybrid approaches.
- Neural network embeddings
- BM25 (Best Match 25)
- Term frequency-inverse document frequency
- Hybrid Search - combination of these techniques

crossml

Document Encoders

- Document encoders used in RAG to encode retrieved documents into a latent space for better representation.
- BoW
- TF-IDF
- Word Embeddings
- Word-to-vec

Applications of RAG

- Question answering,
- Text summarization,
- Dialogue systems.

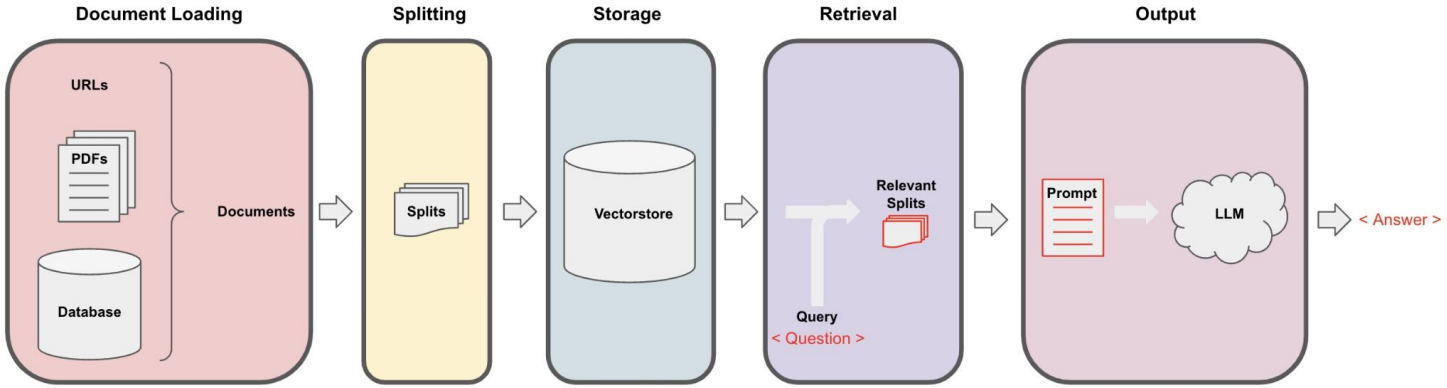
Benefits of RAG

- Including improved response quality
- Better context understanding,
- Enhanced user interaction.

crossml

Understanding RAG

A deep dive into the workings of RAG, its components, and how it contributes to the overall system.



crossml

Implementing RAG

- Steps how to implement RAG in your own projects, with tips and tricks for best practices.
- Load the documents from various sources such as pdfs, sets of videos, URLs etc.
- Splitting the large documents into smaller chunks of data
- Store embeddings in vectorstore
- Create a retriever for querying the vector store database
- Haystack and Langchain are the open source solutions

crossml

Challenges and Limitations

- Identification of challenges and limitations associated with RAG
- Scalability issues,
- Model biases,
- Hallucinations
- Data quality concerns.

crossml

Future Directions

- Exploration of potential future directions for RAG research and development,
- Including advancements in retrieval techniques
- Model architectures.

crossml

Thank you. Please feel free to ask any questions.