What is RAG?

Empower Your Data with Conversational AI

Workshop

Hey 👋

I’m Simran Anand

@CSE Insights by Simran Anand

Senior Software Engineer at Bosch Global Software Technologies

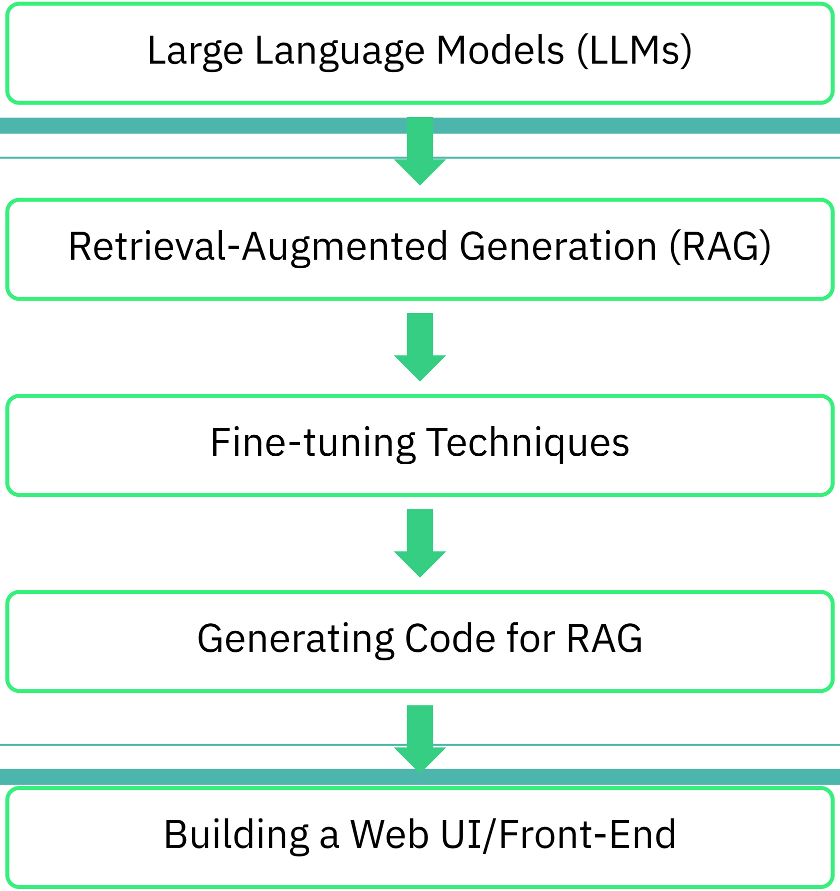
AI & Data Science Expert

Mentor & Educator | Trained 500+ people

I also make educational YouTube videos

Your AI-Powered Journey for Chat-Bot Development Begins Here!

I hope you guys are excited to learn from me 🚀
Now, let’s straight away dive into today’s session without any further adieu.



Large Language Models (LLMs)

Retrieval-Augmented Generation (RAG)

Fine-tuning Techniques

Generating Code for RAG

Building a Web UI/Front-End

Goal – Why Build an AI Chatbot?

Objective: Develop an AI chatbot that:

- Engages in seamless, natural conversations.
- Fetches relevant information from documents to answer user queries.

Outcome: Intelligent query resolution, enhanced user experience, and data accessibility.

Large Language Models (LLMs)

LLMs are AI models trained on massive datasets to understand and generate human-like text.

Use Cases:

- Summarizing content.
- Answering complex questions.
- Conversing in multiple languages.

Key Features:

- Context awareness.
- Scalability across domains.

Retrieval-Augmented Generation (RAG)

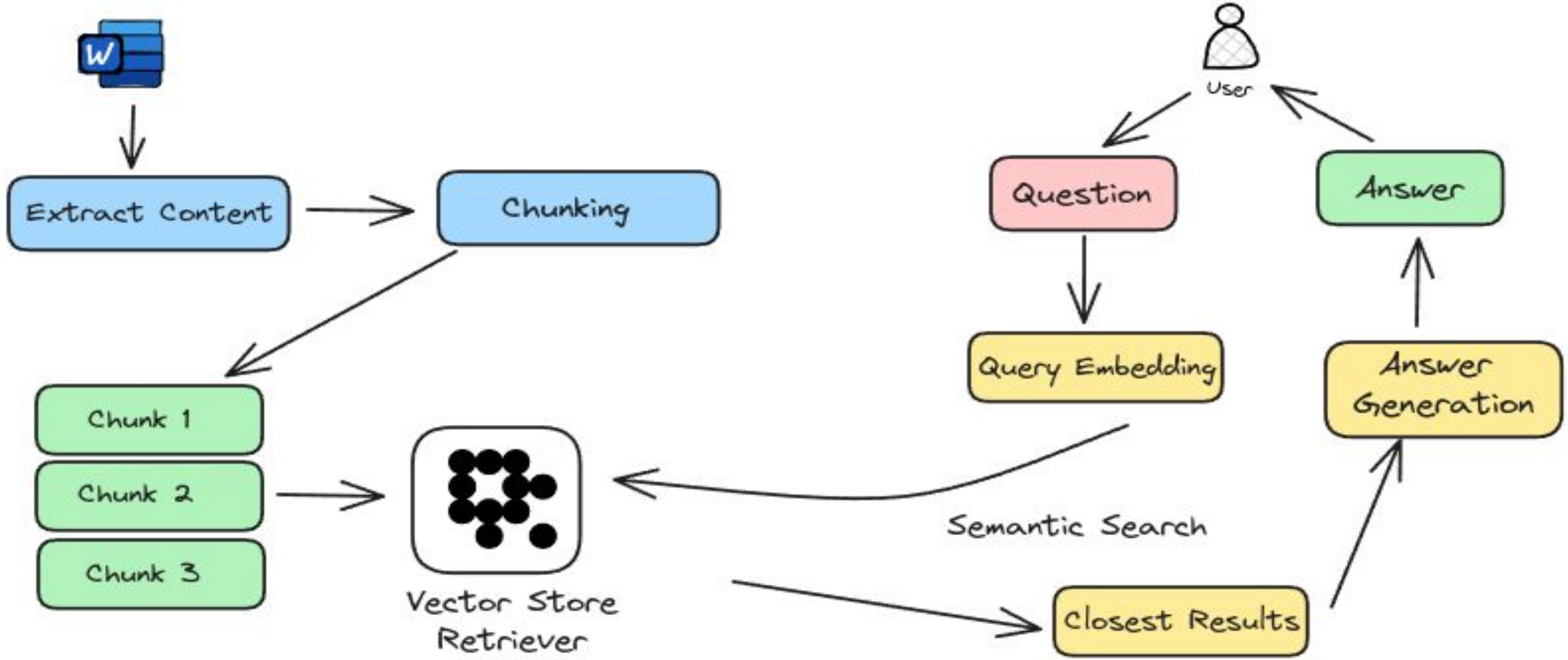
Making AI Smarter and Relevant

What is RAG?: Combines retrieval of relevant documents with generative AI.

- **Components**:
  - **Retriever**: Locates relevant chunks of data.
  - **Generator**: Uses retrieved data to craft accurate responses.
- **Advantages**:
  - Enhanced accuracy.
  - Dynamic, context-driven responses.

Example: “Instead of vague responses, RAG provides direct answers pulled from your document repository.”

Methodology of RAG



Why RAG is a Game-Changer

Combines **retrieval** of accurate, up-to-date information with the **generation** capabilities of LLMs.
Provides context-aware responses by grounding answers in real-world, domain-specific data.
Ensures precision, reduces hallucinations, and enhances trust in AI outputs

RAG is a versatile tool for turning static data into actionable, conversational intelligence tailored to your needs.

Fine Tuning

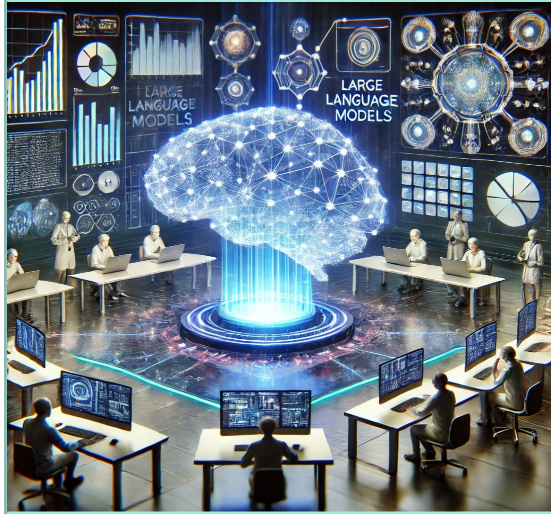
Purpose: Tailor the LLM to your specific domain or dataset

Techniques:

- **Domain Adaptation**: Training on domain-specific data.
- **Parameter Freezing**: Fine-tuning only relevant layers.

Benefits:

- Improved accuracy for niche queries.
- Alignment with organizational tone and style.



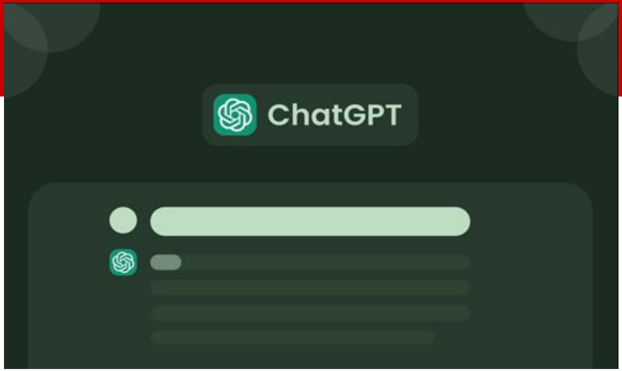
Generating Code for RAG

From Concept to Code

- **Tools**: Use ChatGPT or similar tools for generating RAG code snippets.

Key Steps:

1. Define the retriever (e.g., FAISS, Chroma DB, ElasticSearch).
2. Use pre-trained models for text generation.
3. Combine both into a functional RAG pipeline.



From Concept to Code

- **Tools**: Use ChatGPT or similar tools for generating RAG code snippets.

Key Steps:

1. Define the retriever (e.g., FAISS, Chroma DB, ElasticSearch).
2. Use pre-trained models for text generation.
3. Combine both into a functional RAG pipeline.

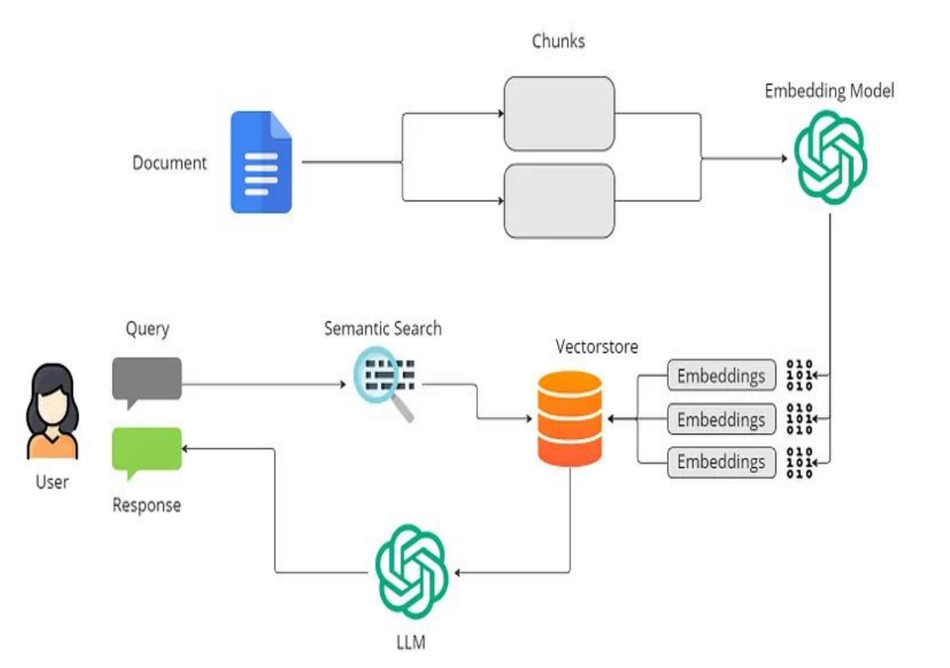


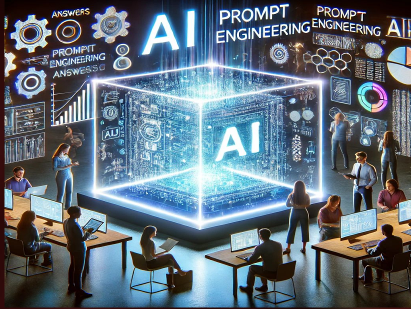
Image Source: Towards AI

Prompt Engineering

- Design prompts to guide AI for desired outputs.
- Example: “Retrieve content related to [topic] and summarize.”

Types

- Zero Shot Prompting
- Few Shot Prompting
- One Shot Prompting
- Chain Of Thought (CoT) Prompting



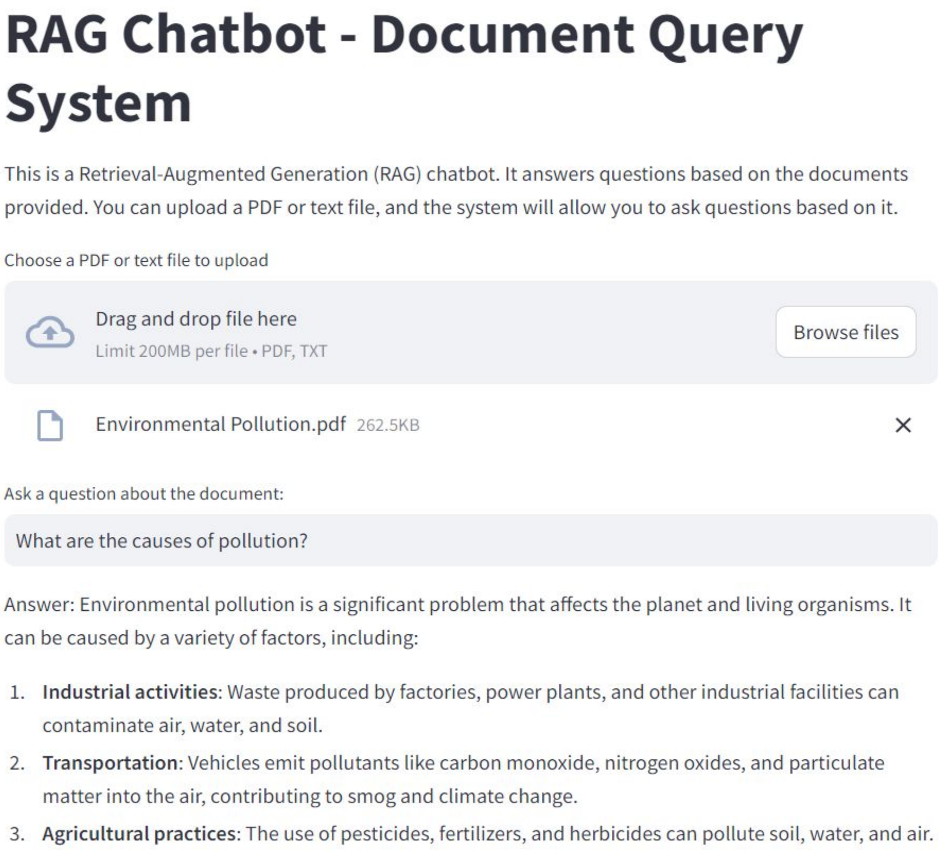
Building a Web UI for Front End

Bringing Your AI Chatbot to Life

Frameworks:

- **Streamlit**: Simple, Python-based UI development.
- **Gradio**: Quick prototyping for interactive AI demos.

Tools for Code Generation: ChatGPT, Blackbox.ai or Claude can help build UI elements.



RAG Chatbot - Document Query System

This is a Retrieval-Augmented Generation (RAG) chatbot. It answers questions based on the documents provided. You can upload a PDF or text file, and the system will allow you to ask questions based on it.

Choose a PDF or text file to upload

Drag and drop file here

Limit 200MB per file • PDF, TXT

Browse files

Environmental Pollution.pdf 262.5KB ✕

Ask a question about the document:

What are the causes of pollution?

Answer: Environmental pollution is a significant problem that affects the planet and living organisms. It can be caused by a variety of factors, including:

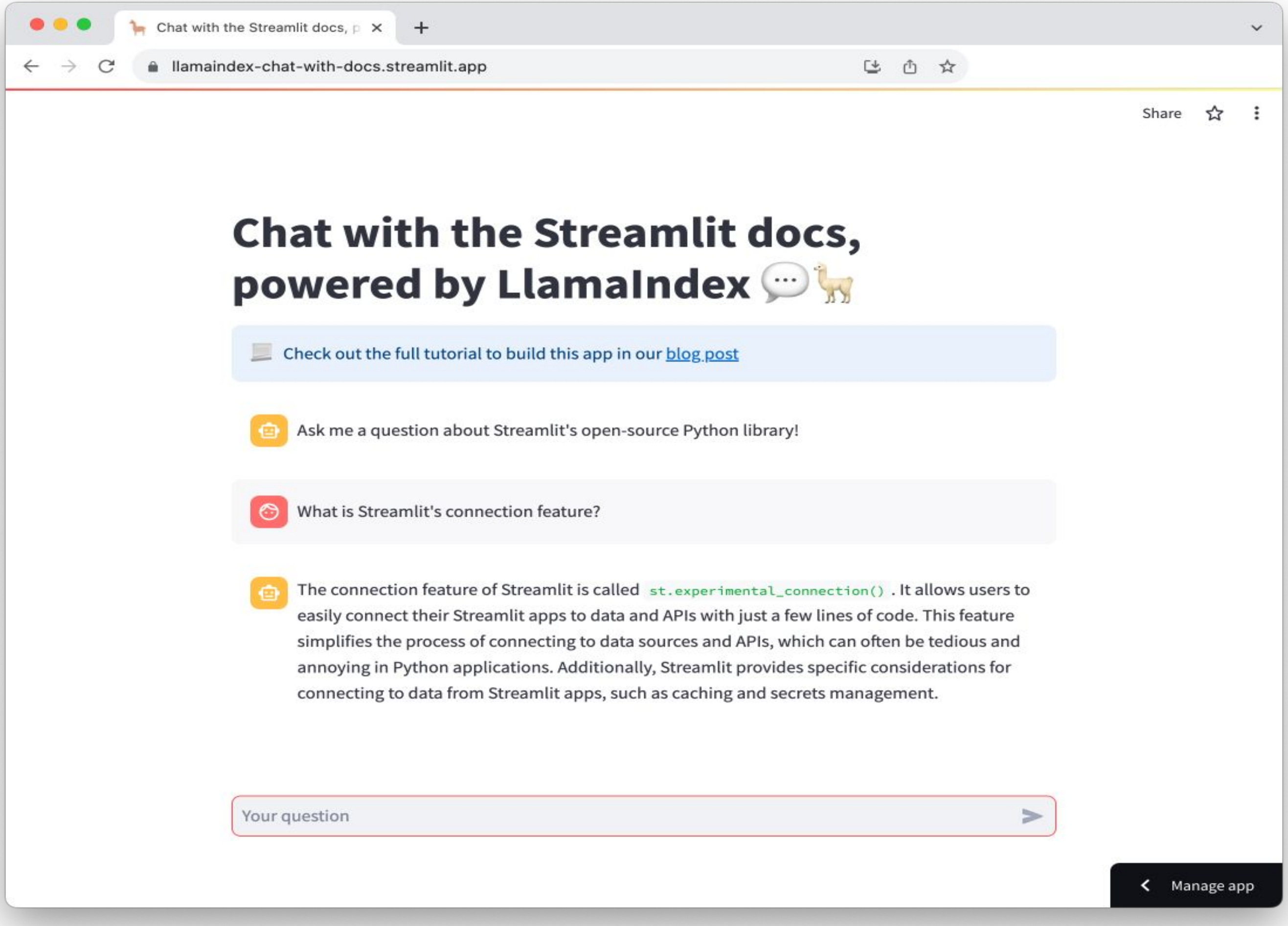
1. **Industrial activities**: Waste produced by factories, power plants, and other industrial facilities can contaminate air, water, and soil.
2. **Transportation**: Vehicles emit pollutants like carbon monoxide, nitrogen oxides, and particulate matter into the air, contributing to smog and climate change.
3. **Agricultural practices**: The use of pesticides, fertilizers, and herbicides can pollute soil, water, and air.

Workflow

Develop backend with RAG pipeline

Further enhance responses with Prompt Engineering or Fine Tuning

Integrate with a user-friendly web interface



Chat with the Streamlit docs, p
llamaindex-chat-with-docs.streamlit.app
Share

Chat with the Streamlit docs, powered by LlamaIndex 💬🦙

Check out the full tutorial to build this app in our blog post

Ask me a question about Streamlit's open-source Python library!

What is Streamlit's connection feature?

The connection feature of Streamlit is called `st.experimental_connection()` . It allows users to easily connect their Streamlit apps to data and APIs with just a few lines of code. This feature simplifies the process of connecting to data sources and APIs, which can often be tedious and annoying in Python applications. Additionally, Streamlit provides specific considerations for connecting to data from Streamlit apps, such as caching and secrets management.

How You Can Use RAG in Your Work

1. **Customer Support**: Create chatbots that fetch and answer FAQs from support databases.
2. **Knowledge Management**: Enable employees to query internal documents for instant insights.
3. **Education & Training**: Build tools that summarize or explain large texts interactively.
4. **Research Assistance**: Quickly extract and synthesize insights from multiple resources.

Transform Conversations with AI

Key Takeaways:

- LLMs + RAG + Fine Tuning = Intelligent Chatbot
- Code generation and no-code tools simplify implementation
- Interactive UIs make the chatbot accessible to all

Next steps: Deploy your AI chatbot and transform how users interact with information!

“Your journey to smarter conversations begins today!”

Any final questions or doubts ?

You’ve taken the first step—now go try this out in your projects!

Thank you