Retrieval Augmented Generation

MGMT 675: Generative AI for Finance

Kerry Back

Beyond Prompting

- Prompting and skills customize *how* an LLM responds
- But what if you need it to know things it wasn't trained on?
- Three approaches, in order of increasing effort and cost:
  1. **RAG** — Retrieval-Augmented Generation
  2. **Fine-tuning** — adjust a pre-trained model's weights
  3. **Training a small language model** — build from scratch on your data

When to Use Each Approach

RAG	Need up-to-date or proprietary facts; data c quently
Fine-Tuning	Need a specific tone, format, or domain exp into the model
Small Language Model	Need full control, privacy, or a highly specia

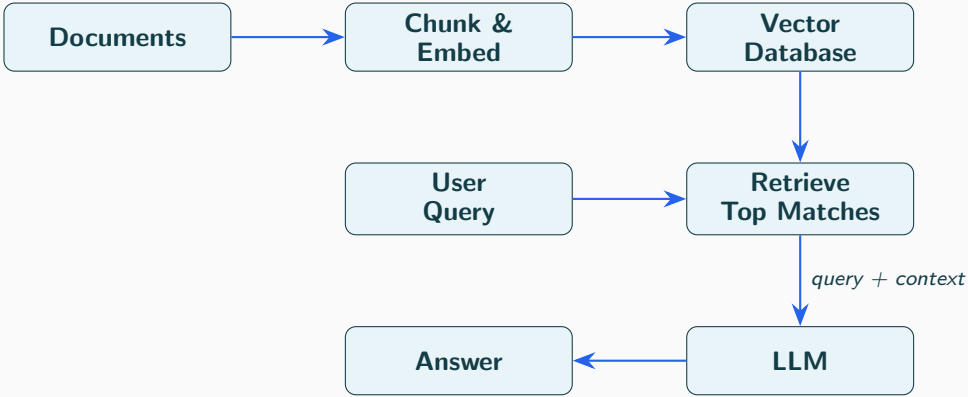
How RAG Works

What is RAG?

RAG = retrieve relevant documents first, then pass them to the LLM along with the user's question. The LLM generates an answer grounded in the retrieved text.

- The LLM's training data may be stale or lack your proprietary information
- RAG injects current, domain-specific context at query time
- No model weights are changed — the base LLM is used as-is

How RAG Works



RAG: Key Concepts

Embeddings

- Text is converted into numerical vectors
- Similar meaning $\rightarrow$ nearby vectors
- Enables semantic search (not just keyword matching)

Vector Database

- Stores document chunks as vectors
- Fast similarity search
- Examples: Pinecone, Chroma, FAISS

Chunking

- Documents are split into small, overlapping pieces (chunks)
- Chunk size matters: too large = noisy context, too small = lost meaning
- Typical sizes: 200–1000 tokens per chunk

RAG in Finance

- **Compliance Q&A** — query regulatory filings, internal policies
- **Earnings call analysis** — search and summarize transcripts
- **Client reporting** — answer questions about portfolio holdings and performance
- **Due diligence** — search across deal documents, contracts, memos
- **Market research** — query analyst reports and news archives

RAG: Strengths and Limitations

Strengths

- No training required
- Data can be updated in real time
- Answers are traceable to sources
- Works with any LLM

Limitations

- Quality depends on retrieval quality
- Doesn't change how the model reasons
- Context window limits how much can be passed
- Chunking and embedding require design choices

NotebookLM: RAG Made Easy

What is NotebookLM?

Google NotebookLM is a free, consumer-friendly RAG tool. Upload your documents, and it builds a personal knowledge base you can query with natural language.

- Available at https://notebooklm.google
- Upload up to 50 sources per notebook: PDFs, Google Docs, Slides, web pages, YouTube videos, and audio files
- Ask questions and get answers grounded in your sources, with citations
- No coding, no setup — RAG in a browser

NotebookLM Features

Query & Summarize

- Chat with your documents
- Answers include inline citations
- Generate summaries, FAQs, study guides, timelines, and briefing docs

Audio Overview

- Generates a podcast-style audio discussion of your sources
- Two AI hosts discuss the key points in a conversational format
- Great for reviewing material on the go

Visual Outputs

- Generate slide decks and infographics from your sources
- Useful for quickly turning research into presentation-ready visuals

NotebookLM for Finance

- **Earnings analysis** — upload 10-K/10-Q filings and earnings transcripts, then ask comparative questions
- **Deal prep** — load pitch books, CIMs, and contracts into a notebook for quick reference
- **Research synthesis** — combine analyst reports, news articles, and company filings into one queryable source
- **Study and review** — generate audio overviews of dense financial documents

NotebookLM is a practical example of RAG that you can use today — no API keys, no vector databases, no code required.

Hands-On: RAG with an Annual Report

Hands-On: RAG with an Annual Report

Try RAG yourself using the **Agentic RAG for Dummies** Google Colab notebook. Upload Walmart's 2024 annual report and ask questions about it.

- **Colab notebook:** https://colab.research.google.com/gist/GiovanniPasq/ddfc4a09d16b5b97c5c532b5c49f7789/agentic_rag_for_dummies.ipynb
- **Walmart 2024 Annual Report** (PDF): https://corporate.walmart.com/content/dam/corporate/documents/newsroom/2024/04/25/walmart-releases-2024-annual-report-and-proxy-statement/walmart-inc-2024-annual-report.pdf
- **GitHub repo:** https://github.com/GiovanniPasq/agentic-rag-for-dummies

What You'll See in the Notebook

1. **Install libraries** — LangGraph, LangChain, Qdrant (vector database), and HuggingFace embeddings
2. **Upload a PDF** — the Walmart annual report is converted from PDF to structured text
3. **Chunk and embed** — the text is split into small, overlapping pieces and converted into numerical vectors (embeddings)
4. **Store in a vector database** — chunks are indexed in Qdrant for fast similarity search
5. **Ask questions** — type a question and the system retrieves the most relevant chunks, then passes them to an LLM to generate a grounded answer
6. **Gradio chat interface** — a web UI lets you interact with the RAG system like a chatbot

Example Questions to Try

Once the notebook is running, try asking:

- What was Walmart's total revenue in fiscal year 2024?
- How did e-commerce sales perform compared to the prior year?
- What are Walmart's main risk factors?
- What is Walmart's strategy for international markets?
- Summarize Walmart's capital expenditure plans.

Notice how the LLM's answers are grounded in the actual document — this is the key benefit of RAG over plain prompting.