Module 6: Working with Financial Documents

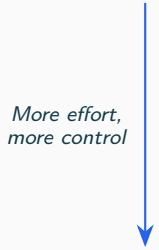
MGMT 675: Generative AI for Finance

Kerry Back

Beyond Prompting

Three Ways to Give AI Knowledge

Prompting and skills customize *how* an LLM responds. But what if you need it to know things it wasn't trained on?



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

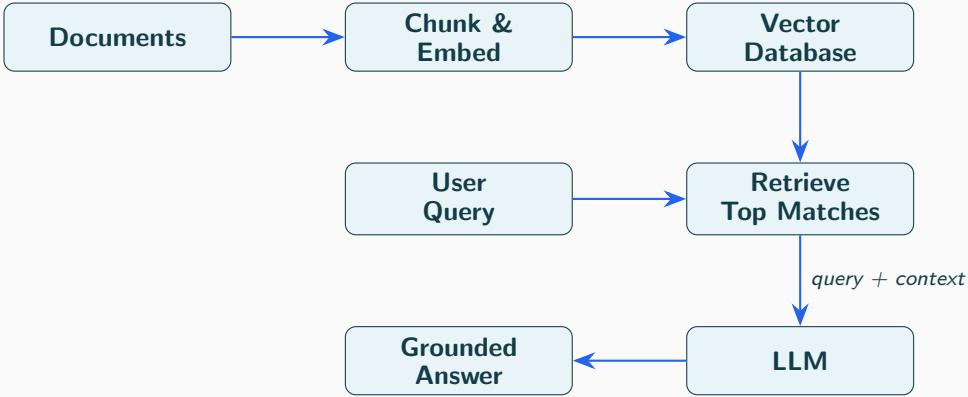
How RAG Works

What is RAG?

RAG = Retrieval-Augmented Generation. 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

The RAG Pipeline



RAG: Key Concepts

Embeddings

- Text 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

Finance Applications of RAG

Document Types

- 10-K and 10-Q filings
- Earnings call transcripts
- Analyst reports
- Deal documents and contracts
- Internal policies and memos

Use Cases

- **Compliance Q&A**: query regulatory filings, internal policies
- **Earnings analysis**: search and summarize transcripts across quarters
- **Due diligence**: search deal documents with citations
- **Research synthesis**: combine multiple sources

RAG: Strengths and Limitations

Strengths

- No training required
- Data can be updated in real time
- Answers are traceable to source pages
- Works with any document format

Limitations

- Quality depends on retrieval quality
- Doesn't change how the model reasons
- Context window limits how much can be passed
- Chunking can split important context

NotebookLM: RAG Without Code

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: PDFs, Google Docs, Slides, web pages, YouTube videos
- Ask questions and get answers with **inline citations**
- No API keys, no vector databases, no code required

NotebookLM Features

Query & Summarize

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

Audio Overview

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

Visual Outputs: Generate slide decks and infographics from your sources — useful for turning research into presentation-ready visuals.

NotebookLM for Finance

- **Earnings analysis**: Upload 10-K/10-Q filings and earnings transcripts, ask comparative questions
- **Deal prep**: Load pitch books, CIMs, and contracts for quick reference
- **Research synthesis**: Combine analyst reports, news articles, and filings into one queryable source
- **Year-over-year comparison**: Upload two years of 10-Ks, ask AI to identify changes in risk factors, revenue composition, and guidance

NotebookLM is a practical example of RAG that you can use **today**.

Building a RAG Pipeline

Under the Hood: The RAG Notebook

For those who want to understand the internals, the **Agentic RAG for Dummies** Colab notebook walks through building a RAG system step by step.

1. **Install libraries**: LangChain, Qdrant (vector database), HuggingFace embeddings
2. **Upload a PDF**: annual report converted from PDF to structured text
3. **Chunk and embed**: text split into overlapping pieces, converted to vectors
4. **Store**: chunks indexed in Qdrant for fast similarity search
5. **Query**: type a question $\rightarrow$ retrieve similar chunks $\rightarrow$ pass to LLM $\rightarrow$ grounded answer
6. **Chat interface**: Gradio web UI for interactive querying

Example Questions to Try

Upload a company's 10-K and ask:

- What was total revenue in the most recent fiscal year?
- How did services revenue grow compared to the prior year?
- What are the main risk factors related to supply chain?
- What changed in the company's accounting policies?
- Summarize management's outlook for the coming year

Notice how answers are grounded in the actual document — the key benefit of RAG over plain prompting.

Exercises

Exercise 1: NotebookLM Analysis

1. Upload 3+ financial documents for the same company into NotebookLM (10-K, earnings transcript, analyst report)
2. Ask 5+ questions across the documents
3. Note how citations trace back to specific sources
4. Submit: Q&A pairs + quality assessment (were answers grounded? any hallucinations?)

Exercise 2: RAG Pipeline

1. Open the Agentic RAG for Dummies Colab notebook
2. Upload a corporate annual report (e.g., Apple 10-K)
3. Ask 5 finance-specific questions
4. Evaluate: are answers grounded in the document, or does the model hallucinate?
5. Submit: notebook + evaluation

Colab notebook: https://colab.research.google.com/gist/GiovanniPasq/ddfc4a09d16b5b97c5c532b5c49f7789

Exercise 3: Document Comparison

1. Upload two years of 10-Ks for the same company into NotebookLM
2. Ask AI to identify the most significant changes in:
   - Risk factors
   - Revenue composition
   - Management guidance
   - Accounting policies
3. Submit: summary of key changes with source citations

Summary

RAG

- Retrieve, then generate
- Grounds answers in sources
- No training required

NotebookLM

- Free RAG without code
- Inline citations
- Audio overviews

Finance Uses

- 10-K analysis
- Earnings call Q&A
- Due diligence

RAG gives AI knowledge it doesn't have — grounded in your documents, with citations.