RAG for Large Number of PDFs

# Data Extraction

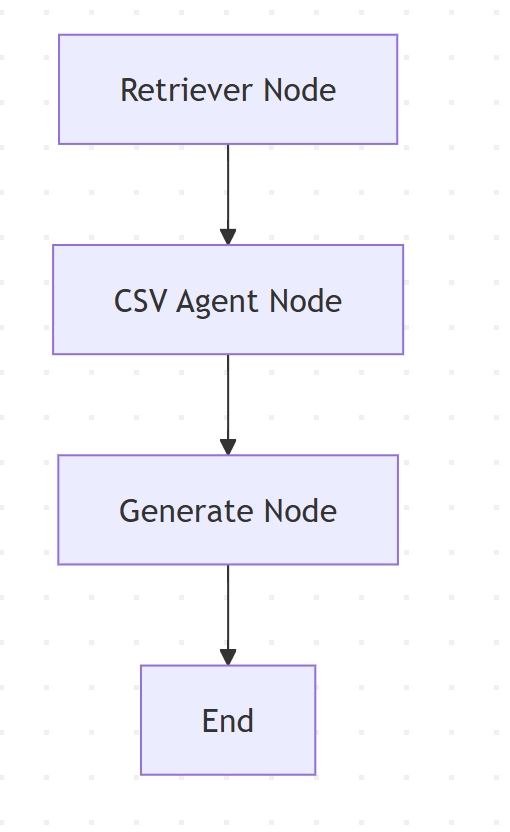
Text Extraction  
- Uses PyMuPDF.  
- Each page is a chunk (context preservation + embedding efficiency).  
- Text-based tables are retained for LLM interpretation.  
  
Image-Based Table Extraction  
- Pages with image-based tables are converted into images (full-page).  
- Grouped into 25-page PDFs due to Adobe API limits.  
- OCR is applied using Adobe PDF Services.  
- Extracted tables are saved as CSV files.

# Why Two Vectorstores?

Conflicts exist between text and table versions in some documents. Having separate stores ensures accurate, source-specific retrieval.

# 🧠 Algorithm Overview

The system uses two separate FAISS vectorstores:  
  
Source: Text Vectorstore  
- Use Case: Paragraphs, in-line tables  
- Details: Pages are chunked individually to preserve context and reduce token load.  
  
Source: CSV Vectorstore  
- Use Case: Tables from image-based PDFs  
- Details: Tables extracted via OCR and stored as CSVs for structured querying.  
  
Retrieval Logic:  
- Text selected → Raw text chunks are passed directly to the LLM.  
- Table selected → CSV tables are passed to an LLM (Mistral) for interpretation.  
- Output is converted into natural language.  
- Final answer is generated using Claude.



# 🛠️ Tools Used

Task: Text Extraction  
- Tool / Library: PyMuPDF  
  
Task: OCR (Image Tables)  
- Tool / Library: Adobe PDF Services  
  
Task: Vector Store  
- Tool / Library: FAISS  
  
Task: LLM for Answer Generation  
- Tool / Library: chatbedrock – Claude 3.5 Sonnet(Temperature -0.3)  
  
Task: LLM for CSV Interpretation  
- Tool / Library: chatbedrock – Mistral 7B Instruct(Temperature-0.3, Recommended temperature-0.0)  
  
Task: Embedding Model  
- Tool / Library: amazon.titan-embed-text-v2:0  
  
Task: Storage  
- Tool / Library: SQLite3

# 🚀 Potential Improvements

- Use smarter OCR tools like LlamaParse, PDF.co, or Tesseract.  
- Add multi-agent orchestration to:  
 - Grade retrieved chunks  
 - Validate final responses  
- Implement metadata filtering for more relevant chunk retrieval.  
- Use fine-tuned models to:  
 - Summarize long text documents (reduce token load)  
 - Better interpret complex CSV tables  
-Add multiple workflows for summarization and other purposes

-Storing the csv files with metadata like source and table name