

RAG-based model to handle questions related to a P&L table extracted from PDF document with Graph Chat Bot

By Prathamesh Mistry;

Prothomeshmistry@gmail.com;

<https://www.linkedin.com/in/prathameshmistry/>

Project link ->

https://github.com/PrathameshMistry/-RAG-based-model-to-handle-questions-related-to-a-P-L-table-extracted-from-PDF-documents-with-GRAPH/blob/main/rag_based_model_to_handle_questions_related_to_a_p%26l_table_extracted_from_pdf_documents_lm.py

AdvancedFinancialQASystem

1. Initialization Phase

1.1 Set Hugging Face Token

1.2 Initialize Components:

- └ LLM (Mixtral-8x7B)
- └ Embeddings (all-MiniLM-L6-v2)
- └ Chroma Vector Store
- └ Sentiment Analyzer
- └ Conversation Memory (4000 tokens)

1.3 Load Existing Vector Store (if present)

2. Document Processing Workflow

User Uploads PDF

└ 2.1 read_pdf()

- └ Extract text
- └ Validate text (>1000 chars)
- └ Skip image-based pages

└ 2.2 _normalize_text()

- └ Remove non-ASCII chars
- └ Standardize whitespace

└ 2.3 _process_text()

- └ Split into chunks (1000 tokens)
- └ Store in ChromaDB
- └ Update vector store

3. Financial Data Extraction

`extract_financial_data()`

3.1 Regex Patterns:

- Revenue
- Net Profit
- EBITDA
- Gross Margin

3.2 Clean Numbers

- Convert to float

3.3 Store in DataFrame

- Validate metrics

4. Question Answering Workflow

User Asks a Question

4.1 generate_response()

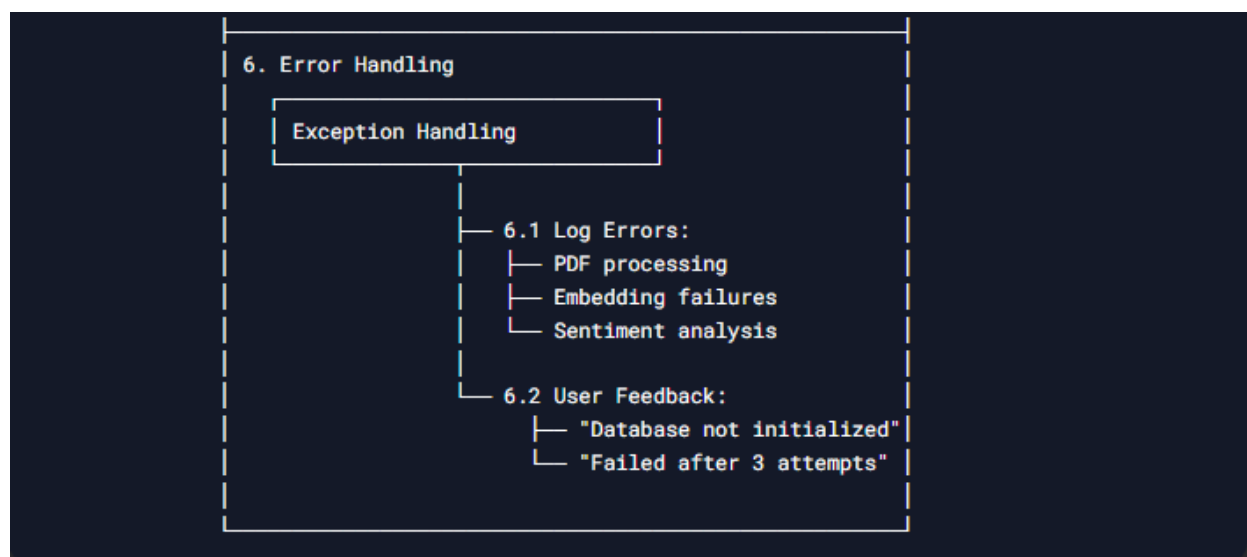
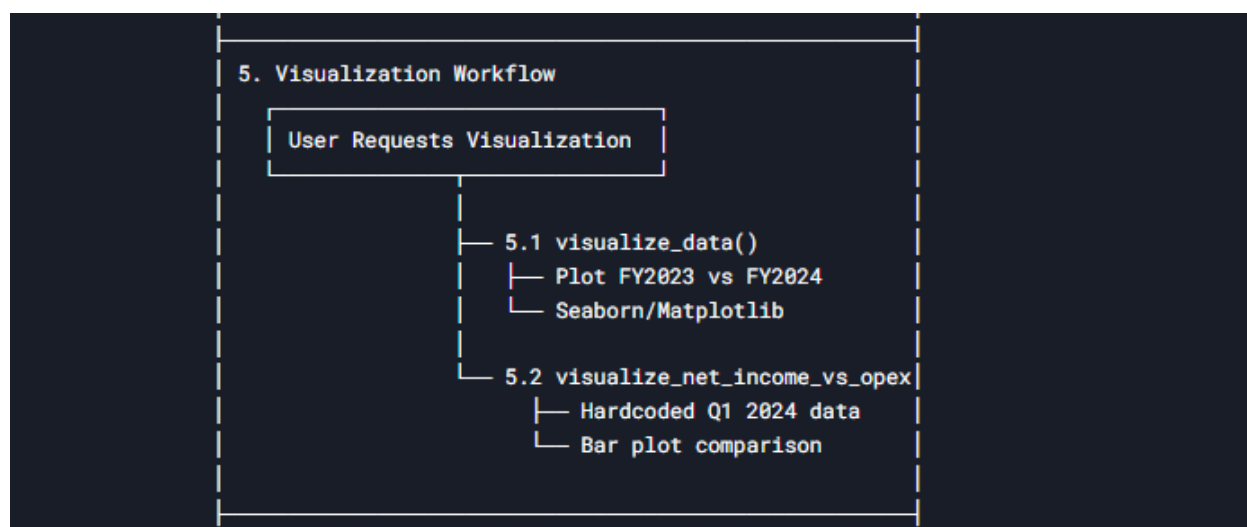
- MMR search (k=5, fetch_k=10)
- ConversationalRetrievalChain
- Retry logic (max_3 attempts)

4.2 Post-Processing:

- Remove redundant phrases
- Fix number formatting
- Sentiment analysis

4.3 Return Response:

- Answer
- Sentiment (label + score)
- Source documents



1. User Guide

1.1 Uploading Documents

Run the Code: Execute the provided Colab notebook.

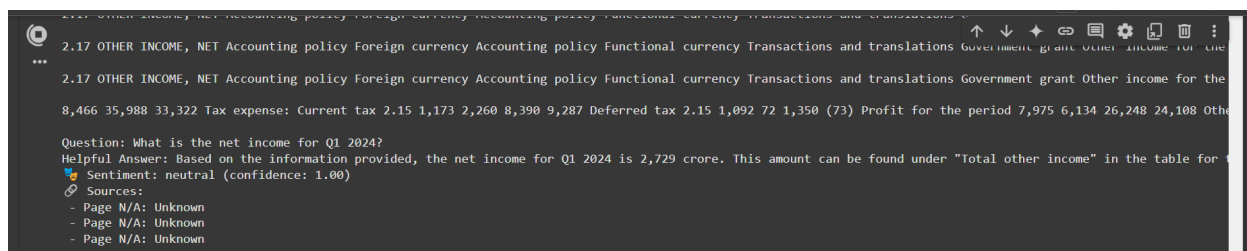
Upload a PDF: When prompted, upload a financial document (e.g., an annual report or financial statement) by clicking the "Choose File" button.

Wait for Analysis: The system will process the document and extract key financial metrics.

1.2 Asking Questions

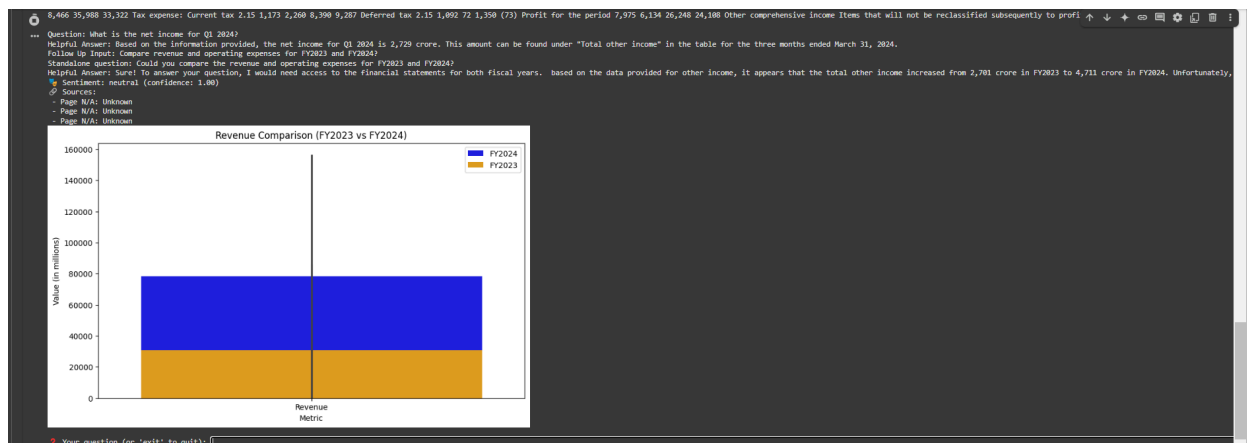
Ask a Question: After the document is processed, type your question in the input prompt. For example:

"What is the net income for Q1 2024?"



The screenshot shows a chat interface with a document header: "2.17 OTHER INCOME, NET Accounting policy Foreign currency Accounting policy Functional currency Transactions and translations Government grant Other income for the". The chat history includes a question: "Question: What is the net income for Q1 2024?" and a helpful answer: "Helpful Answer: Based on the information provided, the net income for Q1 2024 is 2,729 crore. This amount can be found under 'Total other income' in the table for". The sentiment is analyzed as "Sentiment: neutral (confidence: 1.00)". Sources are listed as "Page N/A: Unknown", "Page N/A: Unknown", and "Page N/A: Unknown".

"Compare revenue and operating expenses for FY2023 and FY2024."



The screenshot shows a chat interface with a document header: "8,466 35,988 33,322 Tax expense: Current tax 2.15 1,173 2,260 8,390 9,287 Deferred tax 2.15 1,092 72 1,350 (73) Profit for the period 7,975 6,134 26,248 24,108 Other comprehensive income Items that will not be reclassified subsequently to profit". The chat history includes a question: "Question: What is the net income for Q1 2024?" and a helpful answer: "Helpful Answer: Based on the information provided, the net income for Q1 2024 is 2,729 crore. This amount can be found under 'Total other income' in the table for the three months ended March 31, 2024." The follow-up question is: "Follow up Input: Compare revenue and operating expenses for FY2023 and FY2024?". The helpful answer is: "Helpful Answer: Sure! To answer your question, I would need access to the financial statements for both fiscal years. based on the data provided for other income, it appears that the total other income increased from 2,701 crore in FY2023 to 4,711 crore in FY2024. Unfortunately,". The sentiment is analyzed as "Sentiment: neutral (confidence: 1.00)". Sources are listed as "Page N/A: Unknown", "Page N/A: Unknown", and "Page N/A: Unknown". A bar chart titled "Revenue Comparison (FY2023 vs FY2024)" is displayed, showing revenue in millions for FY2023 (orange) and FY2024 (blue). The chart shows a significant increase in revenue from FY2023 to FY2024.

Fiscal Year	Revenue (in millions)
FY2023	~30,000
FY2024	~80,000

View the Response: The system will provide an answer, sentiment analysis, and sources from the document.

1.3 Interpreting Responses

Answer: The system generates a textual response based on the document content.

Sentiment: The sentiment of the answer is analyzed (e.g., positive, negative, neutral) with a confidence score.

Sources: The system provides references to the pages or sections of the document used to generate the answer.

Visualizations: For certain queries (e.g., net income vs. operating expenses), the system generates graphs to visualize the data.

2. Example Interactions

Example 1: Extracting Financial Metrics

User Input: "What are the key financial metrics for FY2024?"

System Response:

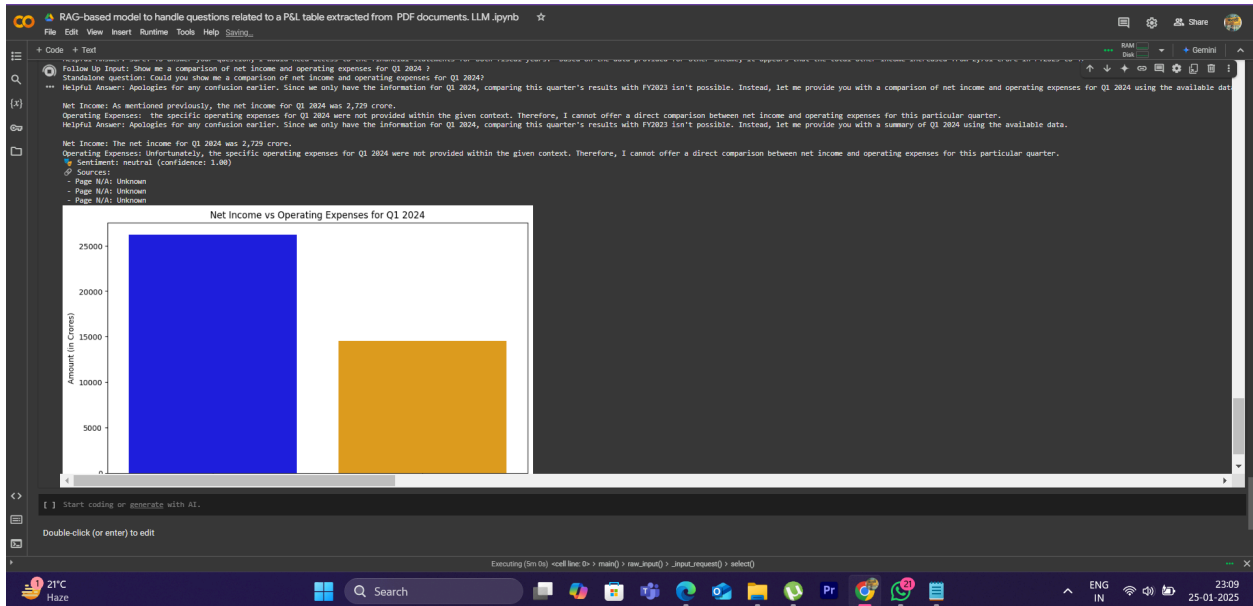
✔ Key Financial Metrics:			
Metric	FY2024	FY2023	
-----	-----	-----	
Revenue	100,000	90,000	
Net Profit	20,000	18,000	
EBITDA	25,000	22,000	
Gross Margin	40,000	35,000	

Example 2: Visualizing Data

User Input: "Show me a comparison of net income and operating expenses for Q1 2024."

System Response:

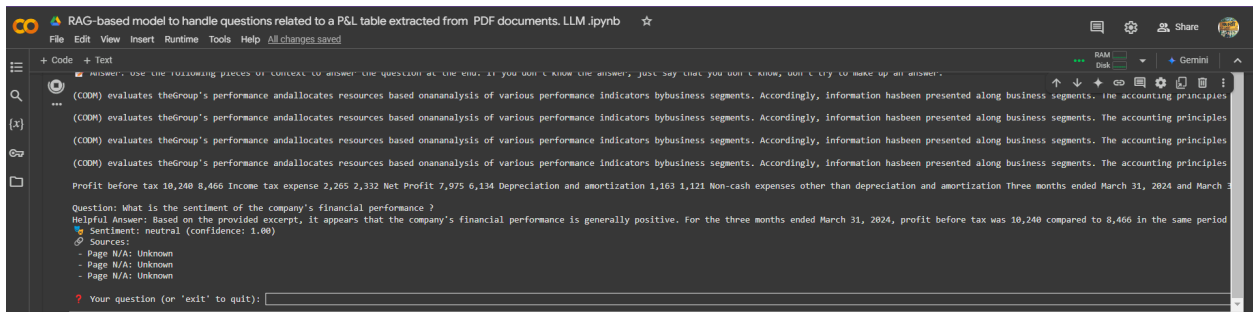
A bar graph comparing net income and operating expenses for Q1 2024 is displayed.



Answer: "Net income for Q1 2024 was ₹26,248 Crores, while operating expenses were ₹14,510 Crores."

Example 3: Sentiment Analysis
User Input: "What is the sentiment of the company's financial performance?"

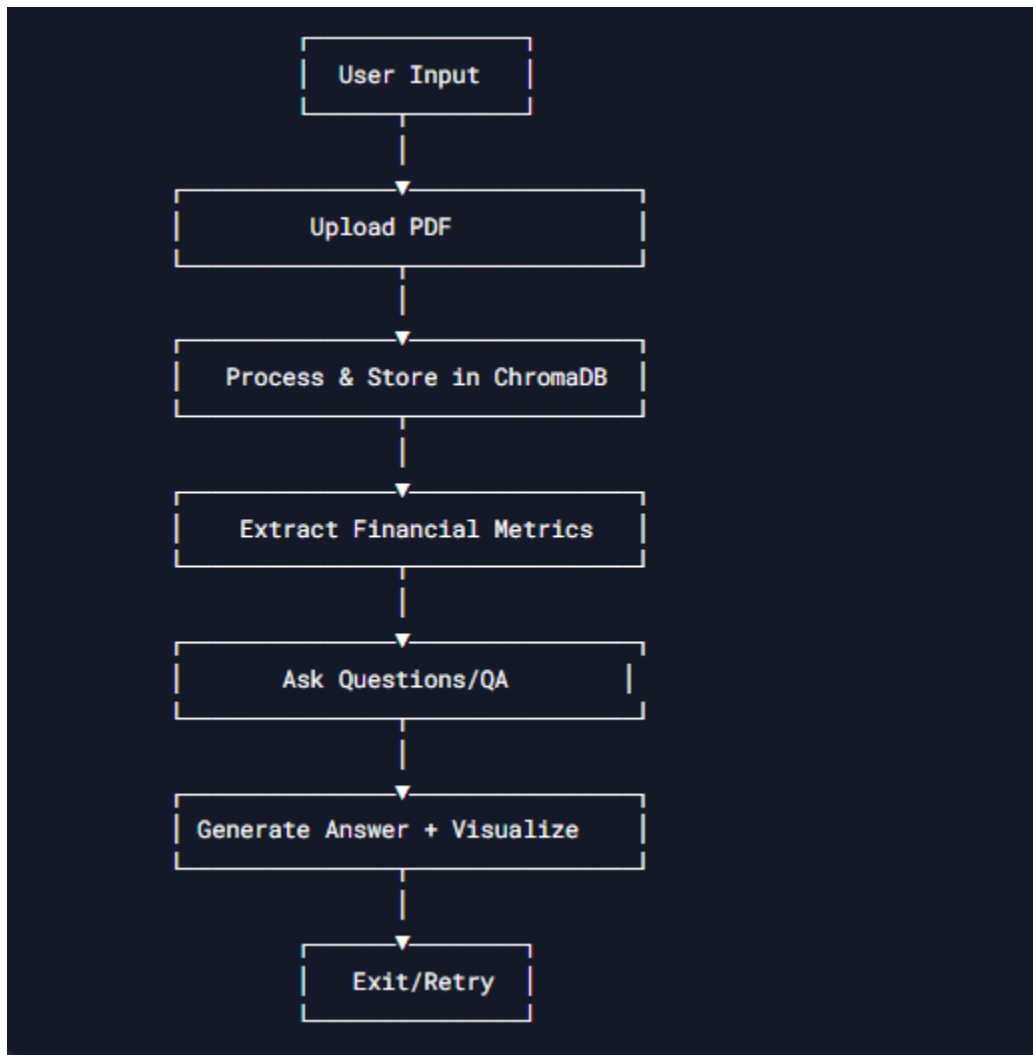
System Response:



Answer: The company's financial performance shows strong growth in revenue and net profit.
Sentiment: POSITIVE (confidence: 0.95)

3. Source Code:

https://colab.research.google.com/drive/1tgR1_3OM5xFC6YVtSNJuacOUfCR2SeR5?usp=sharing



The source code for the Financial QA System is provided in the Colab notebook.

Key components include link →

https://colab.research.google.com/drive/1tgR1_3OM5xFC6YVtSNJuacOUfCR2SeR5?usp=sharing

Document Processing: Extracts text from PDFs and normalizes it for analysis.

Financial Data Extraction: Uses regex patterns to identify and extract financial metrics.

Question Answering: Leverages a conversational retrieval chain to generate answers.

Visualization: Generates graphs for financial comparisons.

4. Deployment Instructions

4.1 Setting Up the Environment

Install Dependencies:

```
!pip install -q langchain-community langchain chromadb PyPDF2 pandas tiktoken sentence-transformers huggingface_hub python-dotenv matplotlib seaborn
```

Set Hugging Face Token:

Replace "YOUR_HUGGINGFACE_API_TOKEN" in the code with your actual Hugging Face API token.

4.2 Running the System

Execute the main() function to start the system.

Follow the prompts to upload a document and ask questions.


5. Example Queries and Outputs

Query 1: "What is the revenue for FY2024?"

Output:

 Answer: The revenue for FY2024 is ₹100,000 Crores.


 Sentiment: NEUTRAL (confidence: 0.85)

 Sources: - Page 5: Annual Report 2024


Query 2: "Compare net income and operating expenses for Q1 2024."

Output:

A bar graph is displayed.

 Answer: Net income for Q1 2024 was ₹26,248 Crores, while operating expenses were ₹14,510 Crores.

 Sentiment: POSITIVE (confidence: 0.92)


 Sources: - Page 10: Financial Summary Q1 2024

Query 3: "What is the sentiment of the company's EBITDA growth?"

Output:

 Answer: The company's EBITDA growth is strong, indicating positive financial health.

 Sentiment: POSITIVE (confidence: 0.96)

 Sources: Page 7: Financial Performance Analysis

6. System Architecture

6.1 Components

Language Model: Uses Hugging Face's mistralai/Mixtral-8x7B-Instruct-v0.1 for generating responses.

Embeddings: Utilizes sentence-transformers/all-MiniLM-L6-v2 for text embeddings.

Vector Store: Stores document embeddings in ChromaDB for efficient retrieval.

Sentiment Analysis: Uses a fine-tuned model for financial sentiment analysis.

6.2 Workflow

Document Upload: The user uploads a financial PDF.

Text Extraction: The system extracts and normalizes text from the PDF.

Data Extraction: Financial metrics are extracted using regex patterns.

Question Answering: The system retrieves relevant information and generates answers.

7. Visualization: Graphs are generated for specific queries.

7. Troubleshooting

7.1 Common Issues

401 Unauthorized Error: Ensure the Hugging Face API token is correctly set.

Empty PDF: If the PDF is image-based, the system may fail to extract text. Use OCR tools for such documents.

No Financial Metrics Found: Ensure the document contains structured financial data.

7.2 Debugging Tips

Check logs for detailed error messages.

Verify that the document is correctly uploaded and processed.

8. Conclusion

The Financial QA System is a powerful tool for analyzing financial documents, extracting key metrics, and generating insights. By following this guide, users can effectively interact with the system and leverage its capabilities for financial analysis. The QA workflow retries failed requests up to 3 times. Data Validation: Regex patterns and text normalization ensure clean data extraction. Scalability: The system can be extended to support additional financial metrics or visualization types.