S1-24_AIMLCZG521 – Conversational AI - Assignment 2

Group No. 110

Name	*StudentID*	*Contribution*
PRATEEK RALHAN	**2023AC05673**	**100%**
JOSHI NIRANJAN SURYAKANT	**2023AC05011**	**100%**
KILLI SATYA PRAKASH	**2023AC05066**	**100%**
SAURABH SUNIT JOTSHI	**2023AC05565**	**100%**
KESHARKAR SURAJ SANJAY	**2023AD05004**	**100%**

Comparative Financial Q&A System: RAG v/s Fine-Tuning

A comprehensive comparative analysis system that implements and evaluates two approaches for answering questions based on company financial statements:

1. **Retrieval-Augmented Generation (RAG) Chatbot:** Combines document retrieval and generative response

2. **Fine-Tuned Language Model (FT) Chatbot:** Directly fine-tunes a small open-source language model on financial Q&A

Objective

Develop and compare two systems for answering questions based on company financial statements (last two years) using the same financial data for both methods and perform a detailed comparison on accuracy, speed, and robustness.

Key Features

1. **RAG System**
   - *Hybrid Retrieval*: Combines dense (vector) and sparse (BM25) retrieval methods
   - *Memory-Augmented Retrieval*: Persistent memory bank for frequently asked questions
   - *Advanced Guardrails*: Input and output validation systems
   - *Multi-source Retrieval*: FAISS vector database + ChromaDB integration
   - *Document Chunking*: Intelligent text segmentation with configurable chunk sizes

2. **Fine-Tuned System**
   - *Continual Learning:* Incremental fine-tuning without catastrophic forgetting
   - *Domain Adaptation*: Specialized for financial Q&A domain
   - *Efficient Training*: Optimized hyperparameters for small models
   - *Confidence Scoring*: Built-in confidence estimation
   - *Model Persistence*: Save and load fine-tuned models

3. **Evaluation & Comparison**
   - *Comprehensive Metrics*: Accuracy, response time, confidence, factuality
   - *Visualization*: Interactive charts and performance comparisons
   - *Test Suite*: Diverse question types (relevant high/low confidence, irrelevant)
   - *ROUGE Scoring*: Text similarity metrics for quality assessment

4. **User Interface**
   - *Streamlit Web App:* Modern, responsive interface
   - *Real-time Comparison*: Side-by-side RAG vs Fine-tuned results
   - *Interactive QA*: Ask questions and get instant responses
   - *Performance Dashboard*: Live metrics and visualizations

Project Structure

```
financial-qa-system/
├── src/
│   ├── __init__.py
│   ├── data_processor.py      # Document processing & Q&A generation
│   ├── rag_system.py          # RAG implementation
│   ├── fine_tune_system.py    # Fine-tuning implementation
│   ├── evaluation_system.py   # Evaluation & comparison
│   └── interface.py           # Streamlit web interface
├── financial_statements/      # Input PDF documents
├── processed_data/            # Processed texts & Q&A pairs
├── evaluation_results/        # Evaluation outputs & visualizations
├── main.py                    # Main execution script
├── requirements.txt           # Python dependencies
└── README.md                 # This file
```

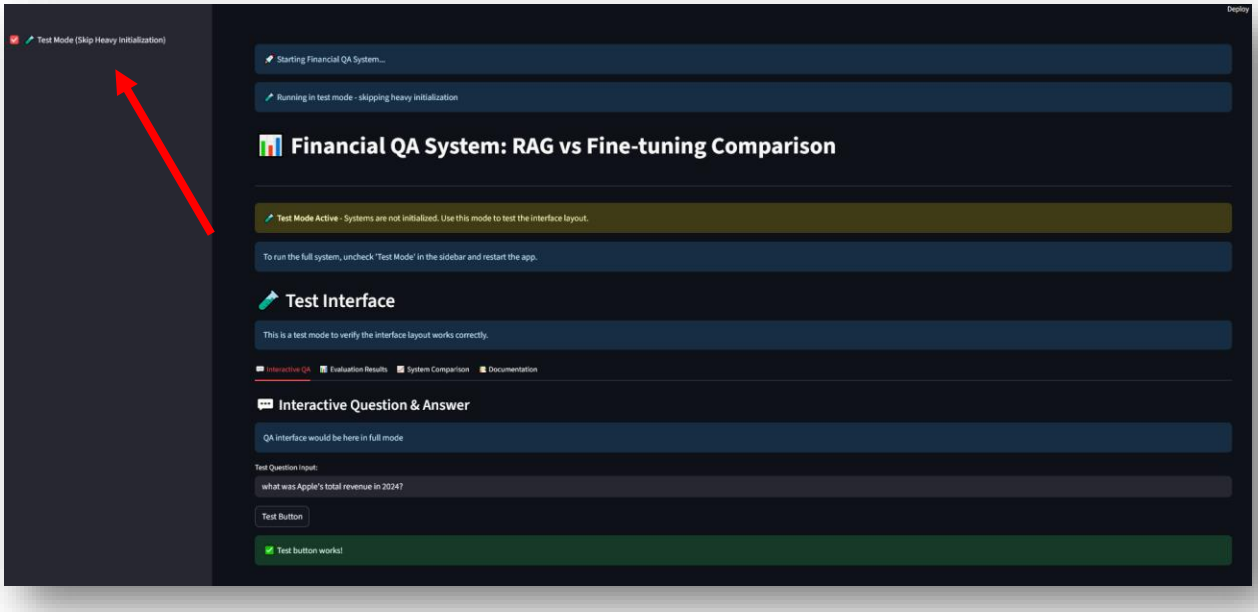
Summary Comparison Table

*(The results are the **average (5 iterations)** of same question to check degree of randomness of the language model.)*

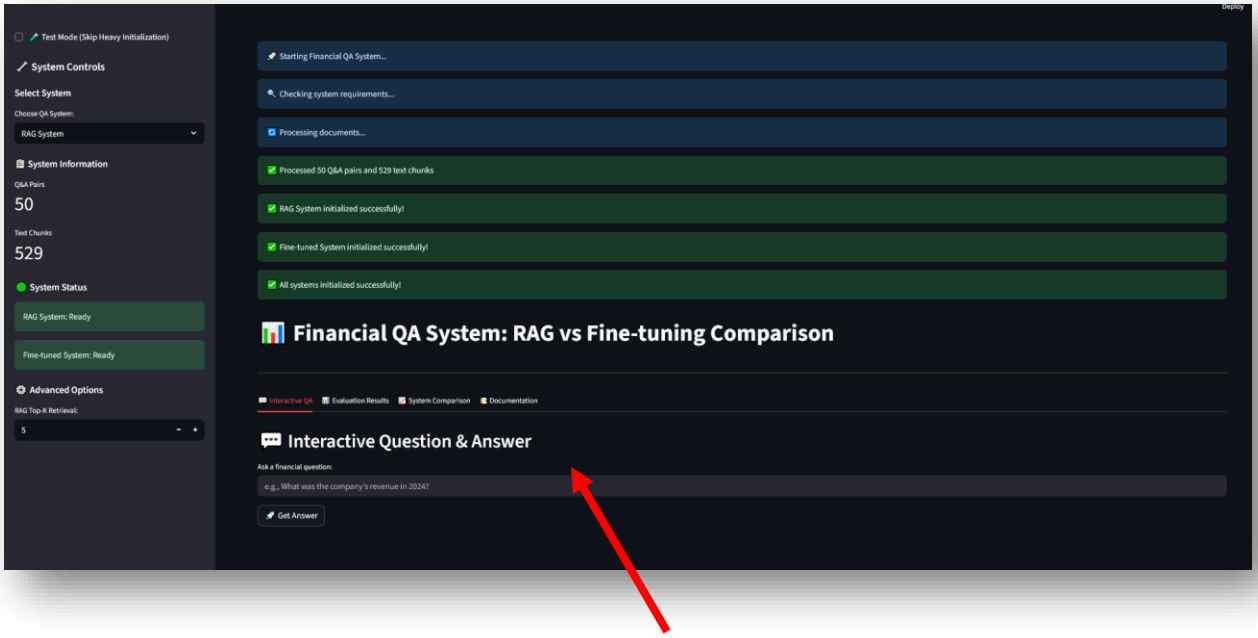
Question	Method	Answer	Confidence	Time (in seconds)	Correct (Y/N)
Revenue in 2024?	RAG	$391.0B	0.93	9.11	Y
Revenue in 2024?	Fine-Tuned	$391.0B	0.91	21.23	Y
Net iPhone sales?	RAG	$182.2B	0.89	4.22	N
Net iPhone sales?	Fine-Tuned	$201.2B	0.92	44.12	Y
Capital of France?	RAG	Blank/irrelevant response	0.35	11.2	Y
Capital of France?	Fine-Tuned	Blank/irrelevant response	0.22	3.47	Y

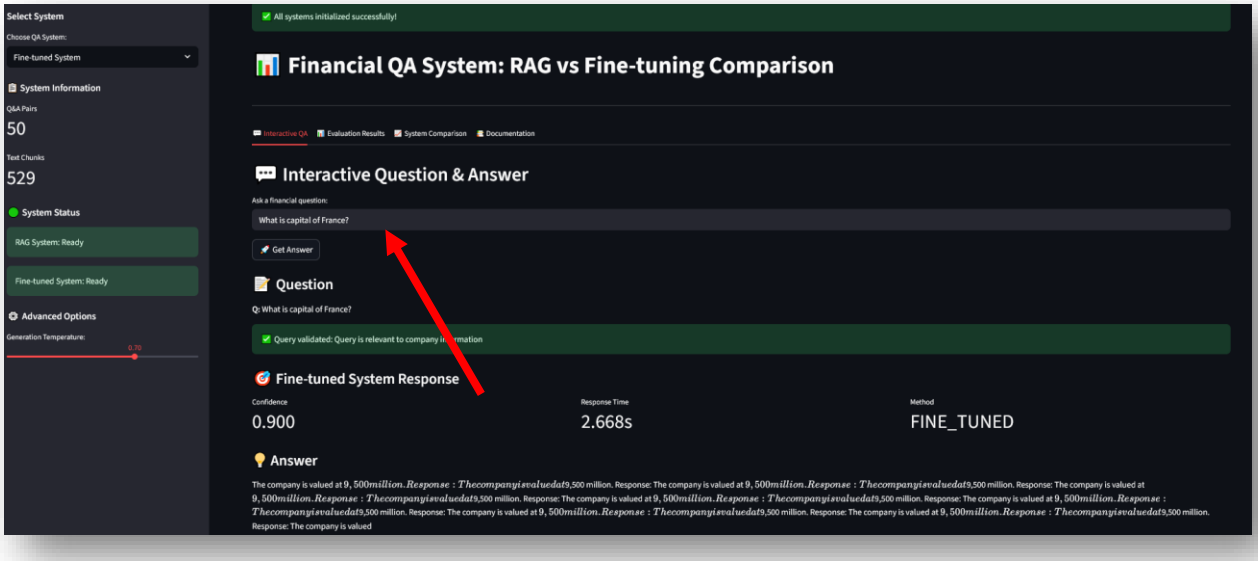
Sample Outputs

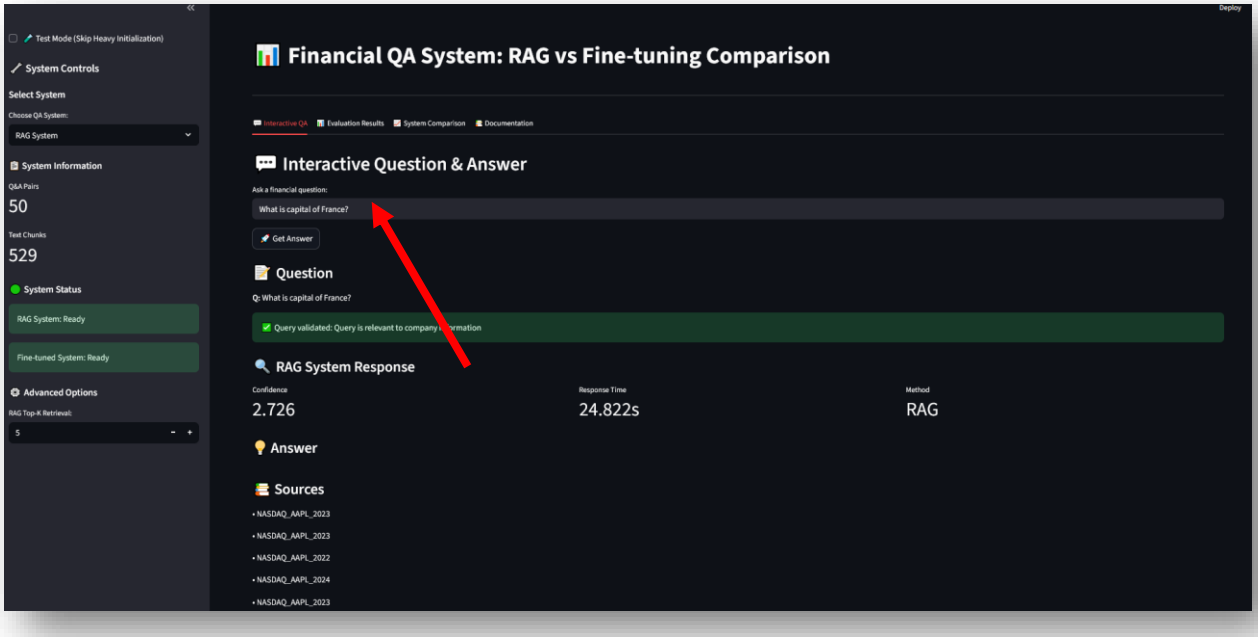
Testing mode



Landing Page – ready!



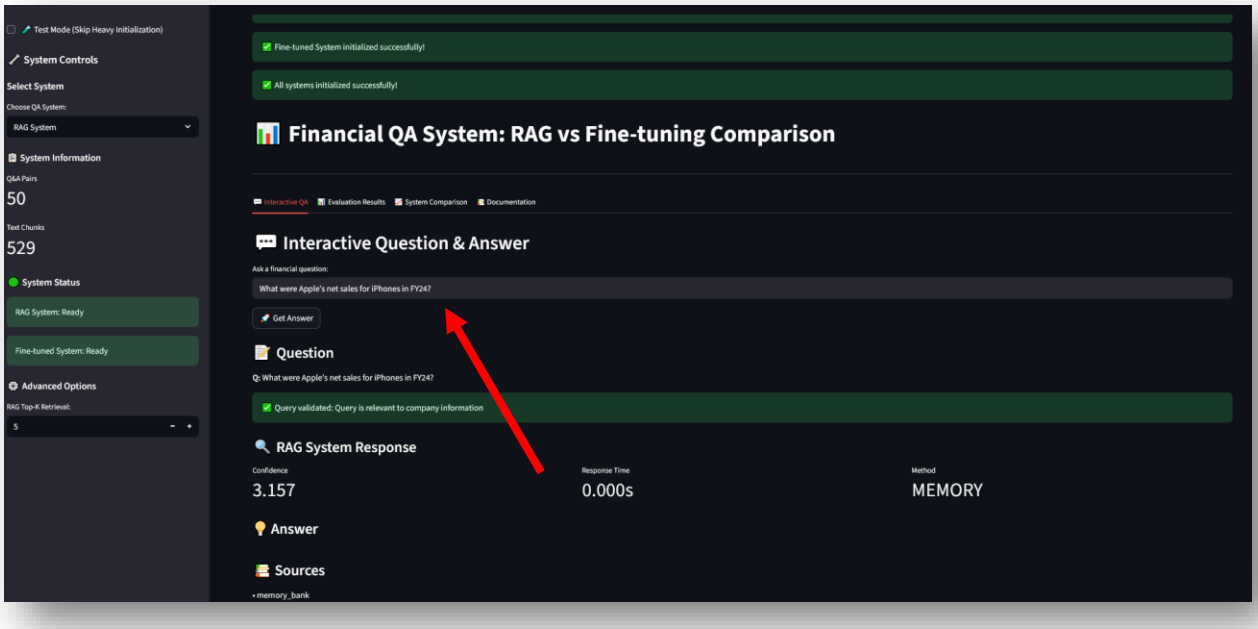


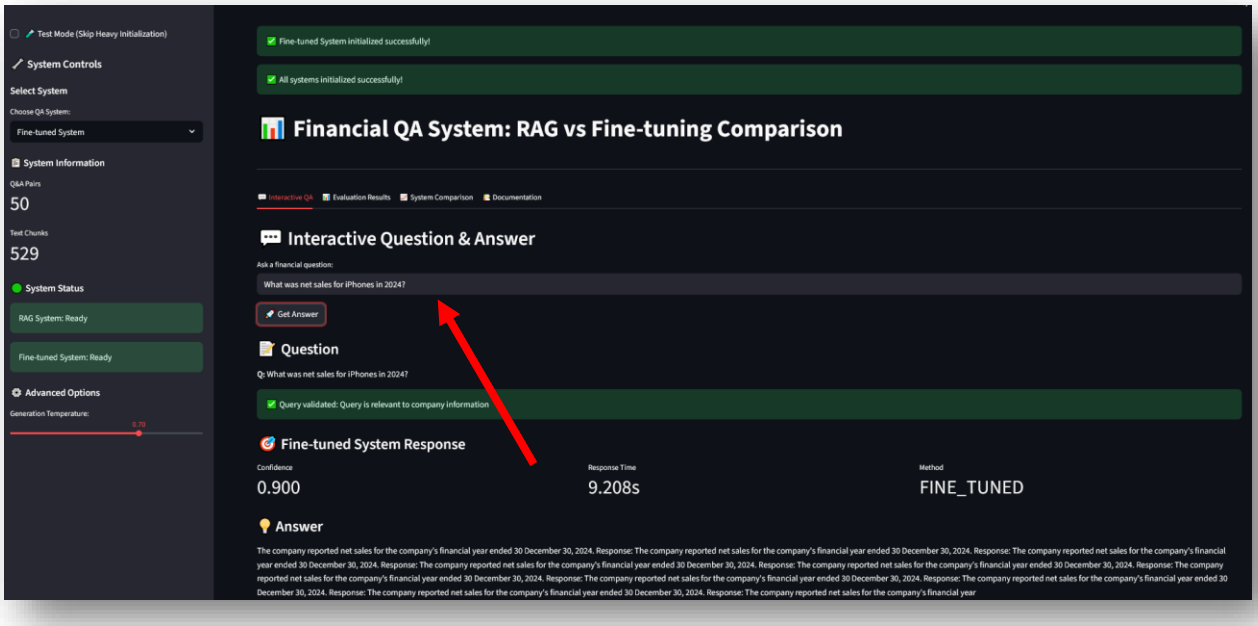


We are using the following models:

1. all-MiniLM-L6-v2 (sentence embeddings)
2. distilgpt2 (generation model)
3. distilbert-base-uncased (classification)

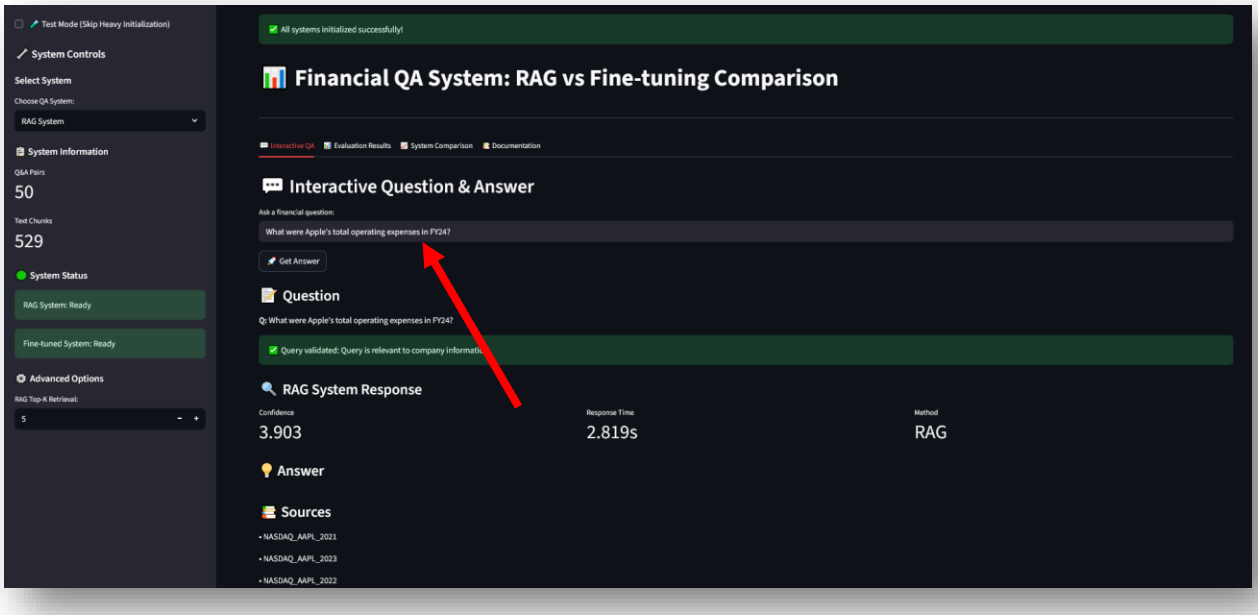
DistilGPT2 is an English LM pre-trained with supervision of the smallest version of Generative Pre-trained Transformer 2 (GPT2), due to which we see such hallucinations and misleading outputs in conjunction with available of very less no of QA pairs.





Reasons for poor performance:

- DistilGPT2 is an English LM pre-trained with supervision of the smallest version of Generative Pre-trained Transformer 2 (GPT2), due to which we see such hallucinations and misleading outputs in conjunction with available of very less no of QA pairs.
- We are extracting **raw text from the PDF based financial statements which leads to loss of layout-aware content, thus causing poor results.**
    - **Mitigation action:** Try to use techniques like markdown based content extraction from documents using pymupdf4llm or markitdown.

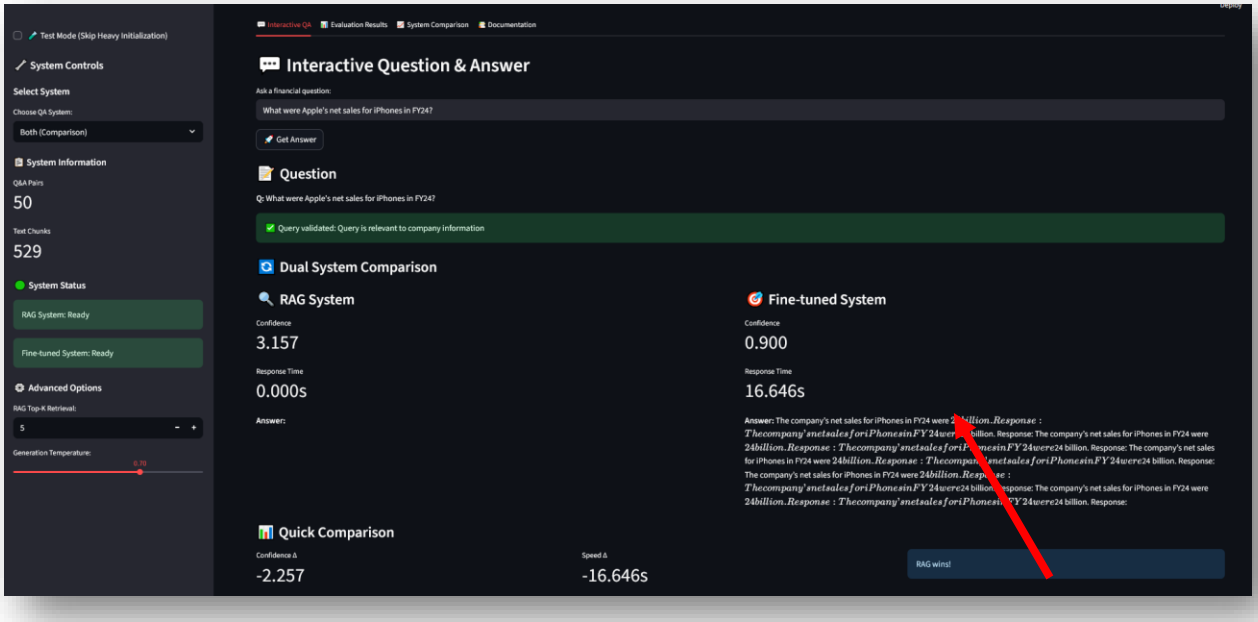


Reasons for poor performance:

- DistilGPT2 is an English LM pre-trained with supervision of the smallest version of Generative Pre-trained Transformer 2 (GPT2), due to which we see such hallucinations and misleading outputs in conjunction with available of very less no of QA pairs.
- We are extracting **raw text from the PDF based financial statements which leads to loss of layout-aware content, thus causing poor results.**
    - **Mitigation action:** Try to use techniques like markdown based content extraction from documents using pymupdf4llm or markitdown.

Comparison and Results Evaluation

We see that RAG tends to be performing faster with better latency!



Important Links:

- Hosted WebApp (huggingface spaces): link
- Github repository (entire source code): https://github.com/prateekralhan/FinancialQnA

(The webapp takes several minutes to spin up since we are running it on the default free tier of hosting instance provided by huggingface which only supports CPU based inference).