**Chatbot Project Report: RAG-based System for GDP Analysis**

**By Sanyogita Kharat**

1. **Project Overview**

**Project Statement**

2011-12 to 2022-23 financial year taken from [data.gov](https://www.data.gov.in/high-value-datasets?sortby=resource_node_rating)) and press note from ([mospi.gov](https://www.mospi.gov.in/sites/default/files/press_release/PressNoteQ4_FY2022-23_31may23.pdf)). Due to the **critical nature of financial data**, our RAG system emphasizes **high accuracy and contextual relevance** in responses. We have implemented multiple solutions using **LlamaIndex** to optimize the retrieval and generation process.

1. **Technologies Used**

* **LlamaIndex**: For efficient document indexing and retrieval.
* **LLMs (Large Language Models)**: Used in conjunction with LlamaIndex for generating detailed responses.
* **Vector Databases**: To enhance semantic search efficiency.
* **Jupyter Notebook**: For testing and fine-tuning the model.
* **Python Libraries**: llama\_index,, langchain, pandas, numpy.

1. **System Architecture & Flowchart**

The RAG-based chatbot system follows the following architecture:

**System Workflow:**

1. **Data Collection & Preprocessing**: GDP datasets are sourced from **data.gov** and **mospi.gov**, then cleaned and structured.
2. **Indexing Using LlamaIndex**: The data is split into chunks and indexed using vector embeddings.
3. **User Query Processing**: The user submits a question about GDP.
4. **Retrieval Step**: Relevant data chunks are retrieved using **semantic search**.
5. **LLM Response Generation**: The retrieved context is passed to the **LLM**, which generates a detailed and context-aware answer.
6. **Response Delivery**: The chatbot returns a well-structured answer to the user.

**Flowchart of the RAG System with llamaIndex:**

A diagram of a dog

Description automatically generated

1. **Key Features & Enhancements**

**✅ High Accuracy Retrieval**

* Implemented **hybrid search** (BM25 + Vector Search) to improve answer quality.
* Optimized **document chunking** to maintain coherence in responses.

**✅ Efficient Query Processing**

* Used **SubQuestionQueryEngine** to break down complex financial queries.
* Fine-tuned **retrieval thresholds** for maximizing relevant context.

**✅ Scalability & Performance**

* Used **FAISS/ChromaDB** for **efficient vector storage and retrieval**.
* Implemented **caching mechanisms** to reduce redundant computations.

1. **Challenges & Solutions**

| **Challenge** | **Solution Implemented** |
| --- | --- |
| **Data Complexity** | Applied structured chunking & hierarchical retrieval |
| **Accuracy Issues** | Used hybrid search for better results |
| **Performance Bottlenecks** | Optimized indexing & vector search storage |

1. **Future Improvements**

* **Fine-tuning LLM responses** using domain-specific training data.
* **Integrating feedback loops** for continuous improvement.
* **Expanding datasets** to include more economic indicators.

1. **Conclusion**

This project successfully demonstrates how **LlamaIndex** and **RAG-based retrieval** can be leveraged to build an accurate and scalable chatbot for **GDP analysis**. The system effectively **retrieves, processes, and generates** responses that provide insightful analysis of India's financial trends over the years.

By continuously refining the retrieval mechanisms and improving accuracy, this chatbot can serve as a valuable tool for **economists, policymakers, and researchers** seeking data-driven financial insights.