"Development of an Intelligent Financial News Research Tool using Langchain, OpenAI, and Streamlit"

The LLM Project: Revolutionizing Financial News Research with Intelligent Chat-bot Capabilities

In the pursuit of efficient financial news research, the LLM project leverages data from a MySQL database to create a sophisticated tool capable of answering questions posed in natural human language. By converting questions into SQL queries and executing them on the database, users experience an interactive conversation with the database in plain English.

Key Components:

Integration of Langchain, OpenAI, and Streamlit:
Employing cutting-edge technologies such as Langchain, OpenAI, and Streamlit, our project aims to build a chat-bot resembling Chat GPT specifically designed for investment-related inquiries.

Automated Retrieval and Summarization of Financial News Articles:
The tool is designed to process a collection of financial news article URLs. When users pose questions related to financial news, the tool retrieves relevant responses from the articles and provides accompanying links. Additionally, the tool can generate concise summaries of entire articles.

Rationale for Development:

Copying and pasting information on Chat GPT can be tedious.
A comprehensive knowledge base is essential for accurate responses.
Chat GPT has a word limit of 3000 words, while GPT-4 accommodates up to 25,000 words at a time**.**

Development Process:

Document Loader:
The initial step involves creating a document loader that aggregates all financial news articles and loads them into a designated object.

Data Processing and Storage:
The loaded object is then divided into multiple chunks, and each chunk is converted and stored in a vector database. This vector database facilitates faster search capabilities when responding to user queries.

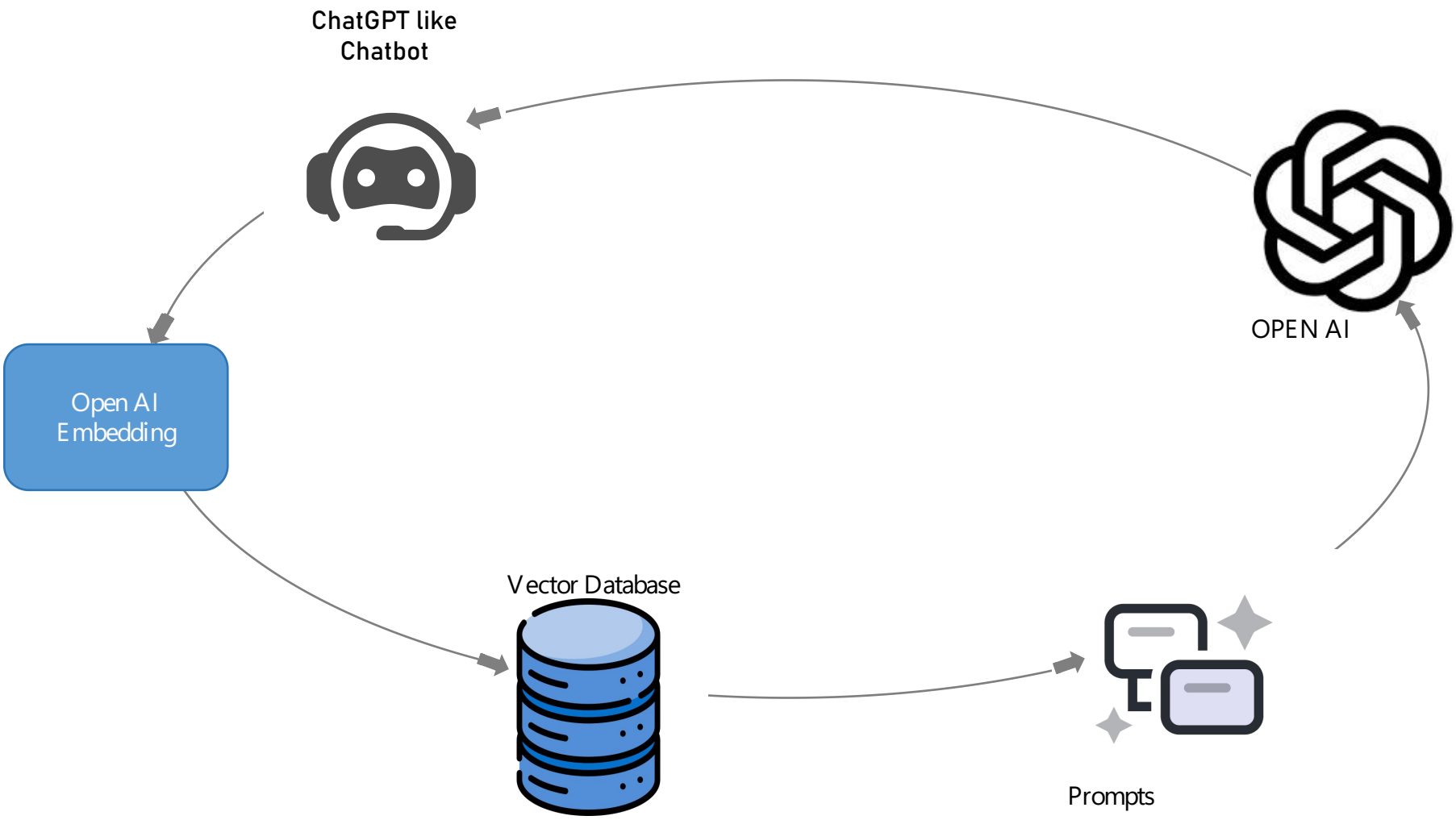
Front-end Development with Streamlit:
The user interface is constructed using Streamlit to provide an intuitive and user-friendly experience.

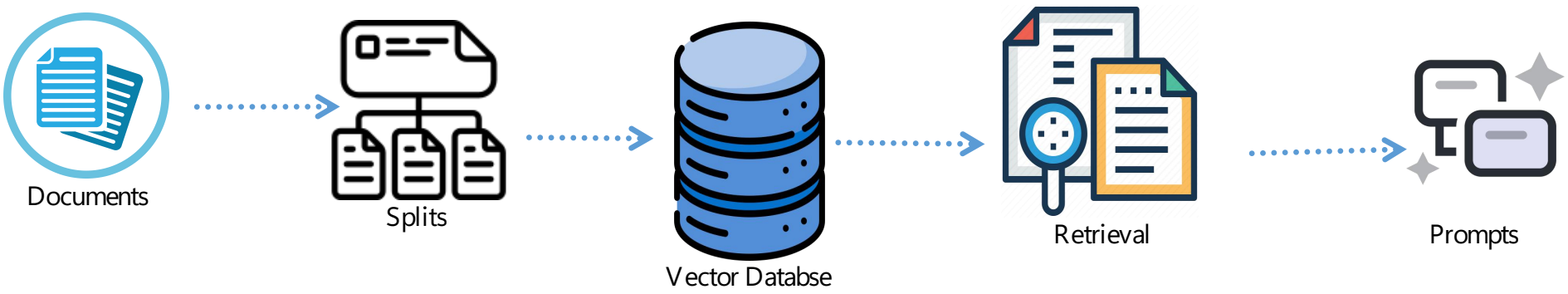
Database Ingestion System:

To populate the vector database, a web scraper is implemented in Python. This scraper runs on a scheduled interval using a Cron Job Scheduler. The extracted text is converted into embedding vectors using OpenAI and stored in a vector database, leveraging Pinecone for efficient storage.

Chat-bot Functionality:

Using a UI framework, a chat-bot interface is developed, closely resembling Chat GPT. Users input questions, which are then converted into embeddings using OpenAI. Relevant chunks from the vector database matching the questions are retrieved. Subsequently, the questions and prompts are fed into the LLM, generating responses that are sent back to the UI interface**.**


ChatGPT like
Chatbot
OPEN AI
Open AI
Embedding
Vector Database
Prompts


Documents
Splits
Vector Databse
Retrieval
Prompts

LLM ops

How to build an end-to-end work flow for your base applications
How to prepare data for tuning a large language model. Using PEFT(parameter efficient fine tunning) .

How to automate and orchestrate the LLM tuning work flow for multiple used cases including when you have large dataset that might be to large for memory.

How to delopy model as a REST API and deploy the model.

How to build an overall LLM workflow

Concept of LLM ops
Data-management Concepts and deployments.

What is ML ops : it is a practice that aims to unify ML development and ML operations.

Automation and monitoring at all steps of ML construction system.
Integration
Testing
Releasing
Deployment
Infrastructure management

ML ops workflow.

Data ingestion	Data Validation	Data transformation	Model	Model analysis	Serving	Logging
Taking data for application databases	How the data is validated	Taking data and transforming or converting it into suitable formats . check for missing data		Evalution model to see how it perform on your data	Model serving and what fits for your use case	

Job Orchestration , is telling the system which job need to be executed first

MLops for LLMs; this focused on the LLM development and how managing the model production

LLM system Design is Looking at the entire application : this include the front end , back end and data engineers