

Scheme Research Application

Overview:

At Haqdarshak, our dedicated research team meticulously sifts through various information portals for government schemes, extracting pertinent details based on a standardized criteria. Armed with a keen eye for detail and a thorough understanding of our mission, they comb through the vast array of available information to ensure accuracy and relevance. Once they have gathered all the necessary data, they organize and update it on our research portal. This ensures that the information provided to our users is comprehensive and presented in a user-friendly manner, empowering individuals with the knowledge they need to access government schemes effectively.

Assignment Goal:

The goal of this assignment is to develop an automated Scheme Research Tool. This tool will take the URL of a scheme article as input, create an accurate and relevant summary and enable users to ask questions based on the content of the article. The tool will provide a summary covering four key criteria: Scheme Benefits, Scheme Application Process, Eligibility, and Documents required; along with the features mentioned below:-

- Load URLs or upload text files containing URLs to fetch article content.
- Process article content through LangChain's UnstructuredURL Loader.
- Construct an embedding vector using OpenAI's embeddings and leverage FAISS, a powerful similarity search library, to enable swift and effective retrieval of relevant information.
- Interact with the LLM's (ChatGPT etc.) by inputting queries and receiving answers along with source URLs and its summary.

Code Submission Criteria:

The final solution should be a web application using Streamlit, a framework for building interactive web applications with Python. The structure of the solution should include:

- main.py: The main Streamlit application Python script.
- requirements.txt: A list of required packages for the project.

- faiss_store_openai.pkl: A pickle file to store the FAISS index.
- .config: Configuration file for storing your OpenAI API key.

Solution Acceptance Criteria:

- The web app will open in your browser.
- On the sidebar, users can input URLs directly.
- Data loading and processing are initiated by clicking "Process URLs."
- The system performs text splitting, generates embedding vectors, and efficiently indexes them using FAISS.
- The embeddings are stored and indexed using FAISS, enhancing retrieval speed.
- The FAISS index is saved in a local file path in pickle format for future use.
- Users can ask a question and receive an answer based on the news articles, along with the URL and its summary.
- Please provide a recorded video of the web application in action that you created.

Note: Submit a well-documented report containing all parts of the assignment. Also, ensure the code contains proper comments wherever required.

[Sample url Link](#)