Vipul251/AI-Powered-Solar-FAQ-Assistant(Github Link)

Solar FAQ Assistant

This is an AI-powered FAQ chatbot that helps users get answers related to solar energy. The chatbot uses **FAISS** for efficient search over a set of FAQ data.

1.Solar Industry AI Assistant - Project Documentation

1.1 Overview

This project is designed to create an AI-powered assistant specializing in solar industry consulting. It provides accurate and helpful information about solar energy to both technical and non-technical users. The assistant leverages Natural Language Processing (NLP) techniques and a custom FAQ dataset stored in a FAISS vector index, allowing it to retrieve relevant answers to user queries about solar panel technology, installation processes, maintenance, cost analysis, and more.

1.2 Project Goal

The goal of this project is to create an intelligent assistant that:

- Responds to user queries about solar energy.
- Provides detailed, accurate, and contextually relevant answers from a predefined knowledge base.

2.Technologies Used

- **Streamlit**: For building the web interface.
- **LangChain**: For managing the interaction with the knowledge base and handling responses.
- **FAISS**: For efficient similarity search over a large set of embeddings.
- **SentenceTransformer**: For converting text queries and documents into vector embeddings for similarity comparison.
- **Pickle**: For saving and loading the preprocessed data and FAISS index.
- **Custom Data**: The FAQ dataset is stored in FAISS for efficient retrieval.

---

3. Components (Setup Instructions)

3.1 File Structure:

```
/solar-faq-assistant
│
├── /knowledge_base
│   ├── faq_index.faiss          # FAISS index file for FAQ embeddings
│   ├── faq_vectors.pkl          # Preprocessed FAQ embeddings
│
├── app.py                    # Streamlit app for the user interface
├── langchain_faiss.py             # Backend logic for querying FAISS index
├── faq_data.py                  # Contains FAQ data (questions and answers)
├── requirements.txt               # Python dependencies
└── README.md                    # Project documentation (this file)
```

3.2 Setup Instructions

3.2.1 Clone the Repository:

```
git clone https://github.com/your-repo/solar-faq-assistant.git
cd solar-faq-assistant
```

3.2.2 Create a Virtual Environment (Optional):

```
python -m venv venv
source venv/bin/activate  # On Windows: venv\Scripts\activate
```

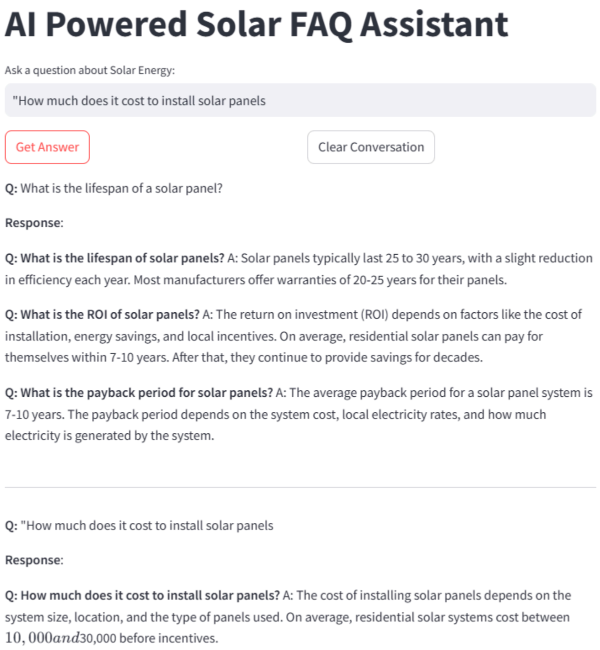
3.2.3 Install Dependencies:

```
pip install -r requirements.txt
```

3.2.4 Run the App:

```
streamlit run app.py
```

3.3 Output:



AI Powered Solar FAQ Assistant

Ask a question about Solar Energy:

"How much does it cost to install solar panels

Q: What is the lifespan of a solar panel?

Response:

Q: What is the lifespan of solar panels? A: Solar panels typically last 25 to 30 years, with a slight reduction in efficiency each year. Most manufacturers offer warranties of 20-25 years for their panels.

Q: What is the ROI of solar panels? A: The return on investment (ROI) depends on factors like the cost of installation, energy savings, and local incentives. On average, residential solar panels can pay for themselves within 7-10 years. After that, they continue to provide savings for decades.

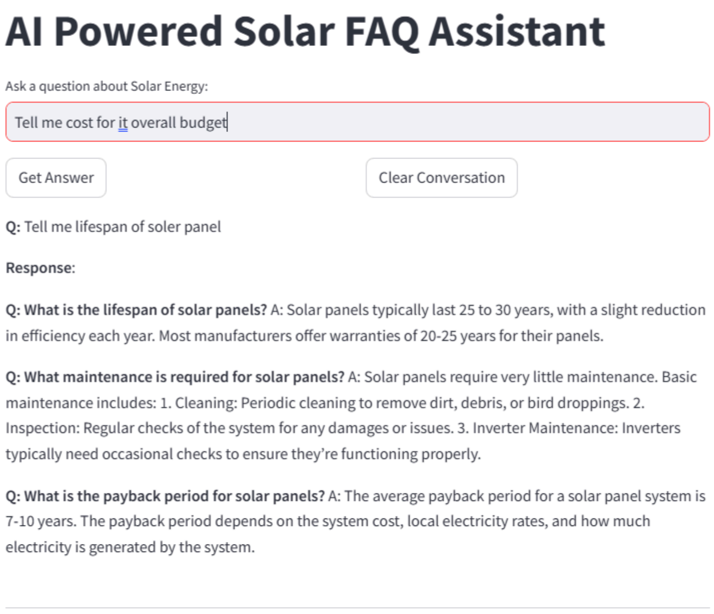
Q: What is the payback period for solar panels? A: The average payback period for a solar panel system is 7-10 years. The payback period depends on the system cost, local electricity rates, and how much electricity is generated by the system.

Q: "How much does it cost to install solar panels

Response:

Q: How much does it cost to install solar panels? A: The cost of installing solar panels depends on the system size, location, and the type of panels used. On average, residential solar systems cost between $10,000and$30,000 before incentives.

This interface enhances usability by allowing users to engage in an interactive Q&A session, retrieve past responses, and reset conversations as needed. The **Clear Conversation** feature ensures that users can start fresh when switching topics, improving clarity and experience.



AI Powered Solar FAQ Assistant

Ask a question about Solar Energy:

Tell me cost for it overall budget

Q: Tell me lifespan of soler panel

Response:

Q: What is the lifespan of solar panels? A: Solar panels typically last 25 to 30 years, with a slight reduction in efficiency each year. Most manufacturers offer warranties of 20-25 years for their panels.

Q: What maintenance is required for solar panels? A: Solar panels require very little maintenance. Basic maintenance includes: 1. Cleaning: Periodic cleaning to remove dirt, debris, or bird droppings. 2. Inspection: Regular checks of the system for any damages or issues. 3. Inverter Maintenance: Inverters typically need occasional checks to ensure they're functioning properly.

Q: What is the payback period for solar panels? A: The average payback period for a solar panel system is 7-10 years. The payback period depends on the system cost, local electricity rates, and how much electricity is generated by the system.

3.3.1 Streamlit Frontend (`app.py`)

This component is responsible for creating the user interface. It allows users to input their queries about solar energy and get responses from the knowledge base. The chat history is maintained during the session.

Key Features:

- Text Input: Users can type in their queries about solar energy.
- Response Display: The assistant displays relevant answers.
- Conversation History: All interactions are logged and displayed during the session.
- Clear Chat: Users can clear the chat history at any time.

3.3.2 Backend (`langchain_faiss.py`)

This component handles the logic for querying the knowledge base. It uses FAISS to perform similarity search over the preprocessed FAQ data stored as embeddings. The SentenceTransformer model is used to embed user queries and compare them against stored embeddings.

Key Features:

- FAISS Index: The index stores vector representations of FAQs for efficient similarity search.
- Custom Data Storage: The FAQ data is stored in FAISS for fast and efficient retrieval.
- Sentence Embedding: The SentenceTransformer model converts user queries and stored FAQs into vector embeddings.
- Query Handling: The `get_answer_from_faq` function retrieves the most relevant FAQ answers based on the user's query.

---

4. Implementation Documentation

The chatbot is built using a combination of **Streamlit** for the frontend and **LangChain** for managing interactions. The **FAISS** library is used for storing and retrieving vector embeddings, ensuring efficient similarity search.

4.1 How It Works

1. **FAISS Indexing**: The FAQ data is preprocessed and converted into vector embeddings using the SentenceTransformer model. These embeddings are indexed using FAISS to allow efficient nearest-neighbor search.
2. **User Query Handling**: When a user submits a query, the query is embedded using the same model, and a similarity search is performed against the FAISS index.
3. **Answer Retrieval**: The top **k** most similar FAQs are retrieved, and their corresponding answers are displayed to the user.
4. **Streamlit UI**: The user interacts with the app through a clean interface, submitting queries and receiving responses in real-time.

---

5. Example Use Cases

1. Technical Consultation

User: "What is the lifespan of a solar panel?"

Assistant: "The typical lifespan of a solar panel is 25-30 years, depending on the quality of the panels and the environmental conditions."

2. Cost Estimation

User: "How much does it cost to install solar panels?"

Assistant: "The cost of installing solar panels varies depending on the location, the size of the installation, and the type of panels. On average, residential solar installations cost between $10,000 to $30,000."

3. Maintenance Guidance

User: "What maintenance is required for solar panels?"

Assistant: "Basic maintenance includes periodic cleaning, regular system inspections, and inverter maintenance to ensure efficiency."

4. Financial Viability Analysis

User: "What is the payback period for solar panels?"

Assistant: "The average payback period is around 7-10 years, depending on system cost, electricity rates, and energy production."

"

---

6. Future Improvement Suggestions

- **Expansion of Knowledge Base**: Include more detailed topics such as solar energy incentives, international market trends, and advanced technical information.
- **Multi-Language Support**: Implement support for multiple languages to reach a wider audience.
- **Real-time Data Integration**: Integrate with APIs for real-time data on solar energy production, cost, and regulations.
- **Advanced Query Understanding**: Improve context awareness and follow-up responses.

- **Authentication Management System**: Implement an authentication system that integrates with different departments, ensuring private data is accessible only to specific members with the required authentication level.

---

7. Code Quality

The code is organized into two main components: a frontend (Streamlit) and a backend (LangChain and FAISS). The code is modular, with functions for embedding queries, searching the knowledge base, and displaying results. Error handling is implemented to ensure that invalid queries or missing data do not break the application.

Contact:

For queries or contributions, feel free to reach out:

Name : Vipul Bhatt

Contact No: 6399577058

Linked in :LinkedIn

Email: stl.1547vipul@gmail.com