AI-Powered FAQ Chatbot

Introduction

In the age of rapid digitalization, users expect instant and accurate responses to their queries. Manually answering frequently asked questions (FAQs) can be inefficient and unscalable. This project aims to develop a lightweight, AI-powered chatbot that can respond to FAQs using Natural Language Processing (NLP), providing an interactive and intelligent user experience on websites or portals.

Abstract

The project is a web-based FAQ chatbot that leverages a locally hosted NLP model to match user questions to predefined FAQs. The chatbot interface allows users to type questions and receive AI-generated responses, simulating a natural conversation. It supports feedback options for continual improvement and logs unhandled queries for future enhancements. The focus is on simplicity, speed, and offline operability without relying on heavy cloud APIs, making it ideal for constrained environments.

Tools Used

- **Frontend**: React.js, HTML, CSS
- **Backend**: Node.js, Express.js
- **NLP Library**: natural (TF-IDF similarity model)
- **Other Packages**: body-parser, cors
- **Data**: faqs.json (stores sample FAQs and answers)
- **Optional**: fs module for logging unanswered questions

Steps Involved in Building the Project

1. **Frontend Setup**
    - Built using React with a clean chat UI.
    - Messages from users are aligned to the right; bot responses appear on the left.
    - Each bot response includes feedback buttons (👍 / 👎) with confirmation on click.
    - Added auto-scroll and “Enter” key support for message input.
2. **Backend Setup**
    - Developed using Node.js and Express.

- A local JSON file (faqs.json) contains FAQ pairs.
- On server start, questions are processed using the natural library's TF-IDF algorithm.

3. **NLP and Logic**
    - User input is compared against all FAQ questions using cosine similarity.
    - If the best match score exceeds a threshold, its answer is returned.
    - Otherwise, the question is logged to unanswered.json and a default response is shown.
4. **Feedback System**
    - Users can rate responses.
    - While feedback is currently displayed in the UI, it can be extended to store feedback in the backend for analytics.
5. **Deployment and Testing**
    - The app is tested locally on localhost:3000 (frontend) and localhost:5000 (backend).
    - Easily deployable on any static or Node-based host.

Conclusion

This project successfully demonstrates a lightweight, intelligent FAQ chatbot using basic NLP. By avoiding reliance on external APIs, it ensures cost-efficiency and control. With a user-friendly interface and a customizable knowledge base, this chatbot can be adapted for businesses, educational institutions, or any service platform looking to automate customer interactions. Future enhancements may include integrating intent classification, database-backed question logging, and multilingual support.