Readme

Team YES

December 7, 2023

Introduction

This document provides an overview and instructions for running the Flask application that utilizes the LangChain library for creating conversational AI models. The application uses OpenAI GPT-3, LangChain, and DeepLake for natural language processing and conversation retrieval.

Requirements

To run the application, you need the following dependencies:

- Flask: A micro web framework for Python.
- langchain: A Python library for creating conversational AI models.
- langchain-vectorstores: For DeepLake vector database integration.
- python-dotenv: A library for managing environment variables.
- OpenAI GPT-3 API key: Ensure you have API credentials from OpenAI.

Installation

1. pip install -r requirements.txt
2. Create a `.env` file in your project directory and add your OpenAI API credentials: `OPENAI_API_KEY=your_openai_api_key_here`
3. In the .env file in your project directory add your ActiveLoop API: `ACTIVELOOP_TOKEN = your_activeloop_api_key_here`

Configuration

1. Set your OpenAI API key in the `.env` file.
2. Adjust any other configuration parameters in the code as needed.

Running the Application

1. Run the Flask application using `python your_app_file.py`.
2. Access the application in your browser at `http://localhost:5002`.

Endpoints

- `/` - Home route, renders the index.html template.
- `/generate_answer` - POST request endpoint to generate answers based on user prompts.

Usage

1. Visit the home route (`http://localhost:5002`) in your browser.
2. Use the provided user interface or make a POST request to `/generate_answer` to interact with the LangChain model.

Contact

For questions or issues, please contact [Yogitha] at [palukuri@uwm.edu].

For questions or issues, please contact [Saaiheswar] at [sgamidi@uwm.edu].

For questions or issues, please contact [Siddartha] at [pullakh2@uwm.edu].