Instructions

Applicant Name: Shelender Kumar

FastAPI

Step 1: Install Python

Ensure you have Python installed on your machine. It's recommended to use Python 3.7 or later. It's also recommended to use a virtual environment for Python projects to manage dependencies efficiently. You can create one using Python's built-in venv module.

Step 2: Install Required Libraries

Run the following command:
pip install fastapi uvicorn pydantic pandas "transformers[torch]" aiofiles

Installing PyTorch: You also need to install PyTorch by visiting their official website which will be provide you with the command based on your device and requirements.

Step 3: Prepare the SQLite Database

Ensure you have a SQLite database file created at the path specified by **DATABASE_URL** in your script. If not, you'll need to create one or adjust the **DATABASE_URL** to point to your database file's location.

The table sentiment_analysis that stores records has following structure:

CREATE TABLE sentiment_analysis (

   comment_id TEXT PRIMARY KEY,

   campaign_id TEXT,

   description TEXT NOT NULL,

   sentiment TEXT

);

Step 4: Prepare the Sentiment Analysis Model

Place your sentiment analysis model and tokenizer files at the specified **model_path**. Ensure the path is correct and accessible by your script.

Running the Application

Step 5: Save the Script

Save the provided Python code into a file within your project directory. For example, name it main.py

Step 6: Run the FastAPI Server

Start the FastAPI application by running the following command in terminal:
uvicorn main:app --reload
The **--reload** option enables automatic reloading of the server upon code changes, which is useful during development.

Interacting with the API

Step 8: Accessing the API Documentation

Once the server is running, open a web browser and navigate to **http://127.0.0.1:8000/docs** This page provides an interactive API documentation where you can test and explore the available endpoints.

Step 9: Using the Endpoints

Through the interactive documentation, you can execute API calls directly. Each endpoint requires specific inputs:
- **/predict/:** Submit a POST request with a JSON body containing the comment_id, campaign_id, and description to receive a sentiment prediction.
- **/insert/:** Submit a POST request with similar data to insert a record into the database.
- **/delete/{comment_id}/:** Submit a DELETE request with the comment_id in the URL path to delete a specific record.
- **/update/:** Submit a PUT request with a JSON body containing all the fields required to update an existing record.
- **/bulk_insert/:** Submit a POST request with a file to insert multiple records at once.