Airport Analyzer

Team: Data Wizards

Members:

Bhairavi Brahme (bkb6542@mavs.uta.edu)

Mehul Hivlekar (mxh6380@mavs.uta.edu)

Sakshi Deshmukh (sxd5431@mavs.uta.edu)

Sumit De (sxd0218@mavs.uta.edu)

Tejesh Rambothu (txr2671@mavs.uta.edu)

Abstract

The project aims to revolutionize the analysis of airport reviews by leveraging generative AI and advanced NLP techniques. It addresses the challenge of extracting relevant, nuanced insights from vast digital content, which traditional methods struggle with. By automating the collection, analysis, and comparison of reviews from various sources, the project provides stakeholders with actionable intelligence on consumer sentiments and preferences. Utilizing a suite of tools including Streamlit for web application creation, Selenium & BeautifulSoup for web scraping, and OpenAI APIs for data processing, the solution offers a comprehensive, cost-effective approach to understanding and improving airport services based on customer feedback.

Problem Statement:

In the era of vast digital content, extracting relevant and comprehensive reviews for various products and services, such as airlines, airports, and HPE products, poses a significant challenge. Traditional methods of gathering and analysing reviews are time-consuming and often fail to capture the nuanced sentiments and specific areas of concern expressed by consumers. This deficiency impedes the ability of stakeholders, including product manufacturers and service providers, to gain actionable insights and make informed decisions.

To address this challenge, our project aims to leverage generative AI powered by advanced natural language processing (NLP) techniques. By harnessing the capabilities of AI, we seek to automatically collect, analyse, and compare reviews across diverse sources, enabling stakeholders to gain a holistic understanding of consumer sentiments and preferences. Specifically, our initiative focuses on extracting insights from airport evaluations, which encompass a wide range of experiences and perspectives.

Our goal is to develop an AI algorithm capable of identifying key themes, sentiments, and areas of improvement expressed in reviews, thereby providing stakeholders and airport managers with actionable intelligence. By aggregating and synthesizing large volumes of review data, our solution aims to unveil patterns, highlight strengths and weaknesses of airports, and facilitate data-driven decision-making.

Solutions Overview

The core objective of this project lies in effectively utilizing the OpenAI API, managing costs and optimizing the processing of data for any application. Following are the limitations associated with using the OpenAI API while building an application:

1. **Cost Management:** At $0.0015 per input token and $0.0020 per output token, it is crucial to optimize the use of words.
2. **Data Accessibility:** ChatGPT does not have access to organizations internal data. This means that even though ChatGPT is a powerful tool, it operates independently of any proprietary information within an organization. This limits the scope of the problems that can be addressed.
3. **Token Limitation:** ChatGPT has an input token limit of 3000 words. As a result, it imperative for businesses to carefully break the data into chunks and provide only those chunks which are most relevant to the prompt.
4. **Knowledge Limitation:** The knowledge of ChatGPT is limited to September 2021. Hence, there is a dire need of for creating an aggregated knowledge database which contains relevant information from the chunks.

Example:

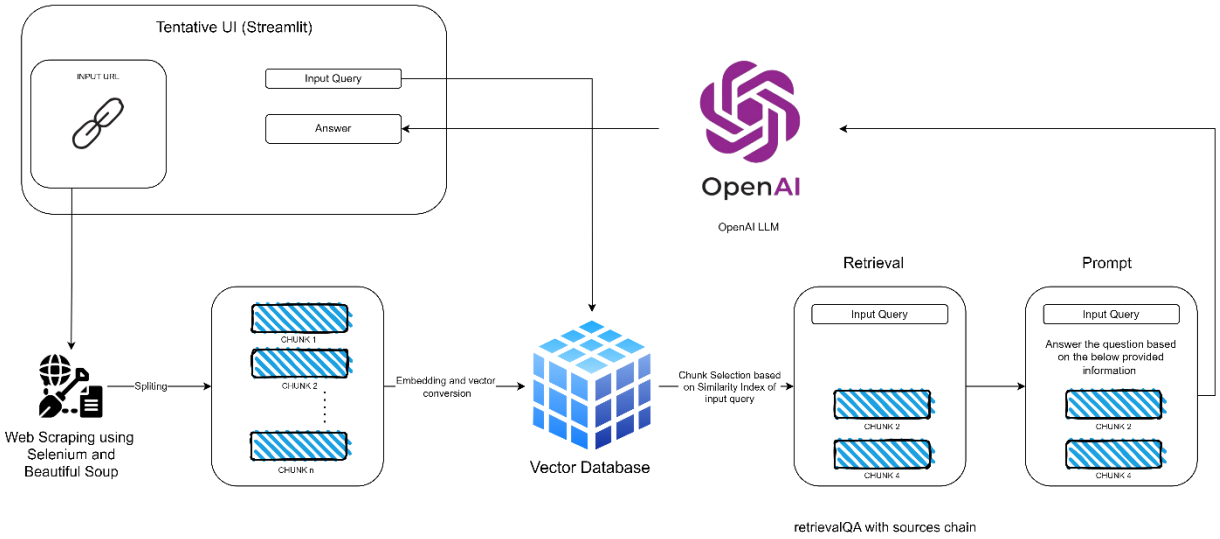
Consider a business analyst using a product analytics tool that integrates reviews across different vendors from not only their product but also competitor's products. By aggregating sentiments and comments, the business analyst can gain insights about market trends and what customers like, in order to see where their product might lag.

Business analyst can also correlate their internal sales data with customer reviews for deeper understanding of the impact of specific product features on customer satisfaction and identify areas of improvement.

From the above two examples it can be observed that to build a successful product analytics app the usage of tokens has to be optimized & access to internal organization data is required. The incorporation of Langchain addresses these issues.

Solution Details

System Architecture



Tool /Model	Function
Streamlit	Web Application Creator
Selenium & BeautifulSoup	Web scraping data
PyCharm	Python IDE
OpenAI Embeddings	Text to Embedding
FAISS (Facebook AI Similarity Search)	Used as a Vector DB and Similarity Search Engine
Retrieval QA with Source Chain	Retrieve Relevant Information Chunk
Open AI LLM	Large Language Model

System Flow

Step 1 – User Selection

The application prompts the user to select an airport for review analysis.

Step 2 – Web Scraping

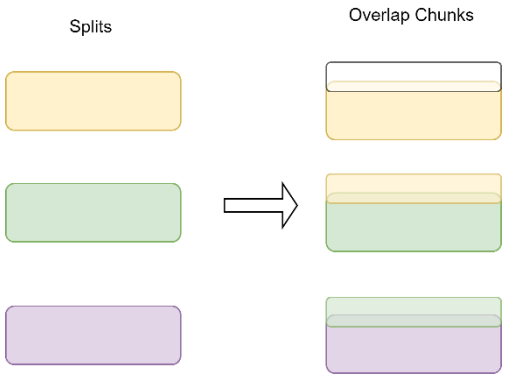
The application scrapes all reviews for the selected airport from the Skytrax website using Selenium.

Step 3 – Splitting the data

The scraped data is divided into multiple chunks using RecursiveCharacterTextSplitter. This segmentation helps filter only the relevant chunks for the LLM model in the subsequent step, thereby reducing the number of tokens sent to the OpenAI LLM and saving costs.

Step 4 – Overlapping chunks

This step involves overlapping the chunks in controlled amounts to maintain the context continuity of each chunk, ensuring the integrity of the splits' meanings. This prevents meaningless splits that could lead to unreasonable answers.



Step 5 – Embedding Creation

The overlapped chunks are then converted into embedding vectors using OpenAI embeddings. This facilitates semantic searches within the database.

Step 6 – Storing Vectors and Semantic Search
The embeddings are then stored as indices in FAISS, where a semantic search is performed based on the similarity index between the chunks and the user-provided query.

Step 7 – Text analysis and response generation

The chosen data segments and user query are forwarded to the OpenAI LLM, accompanied by a standard prompt, for data examination and response formulation.

Step 8 – LLM Response display

The response generated by the LLM is displayed on the web application.

Libraries Used:

The libraries used are:

1. `streamlit` (imported as `st`)

2. `pickle`

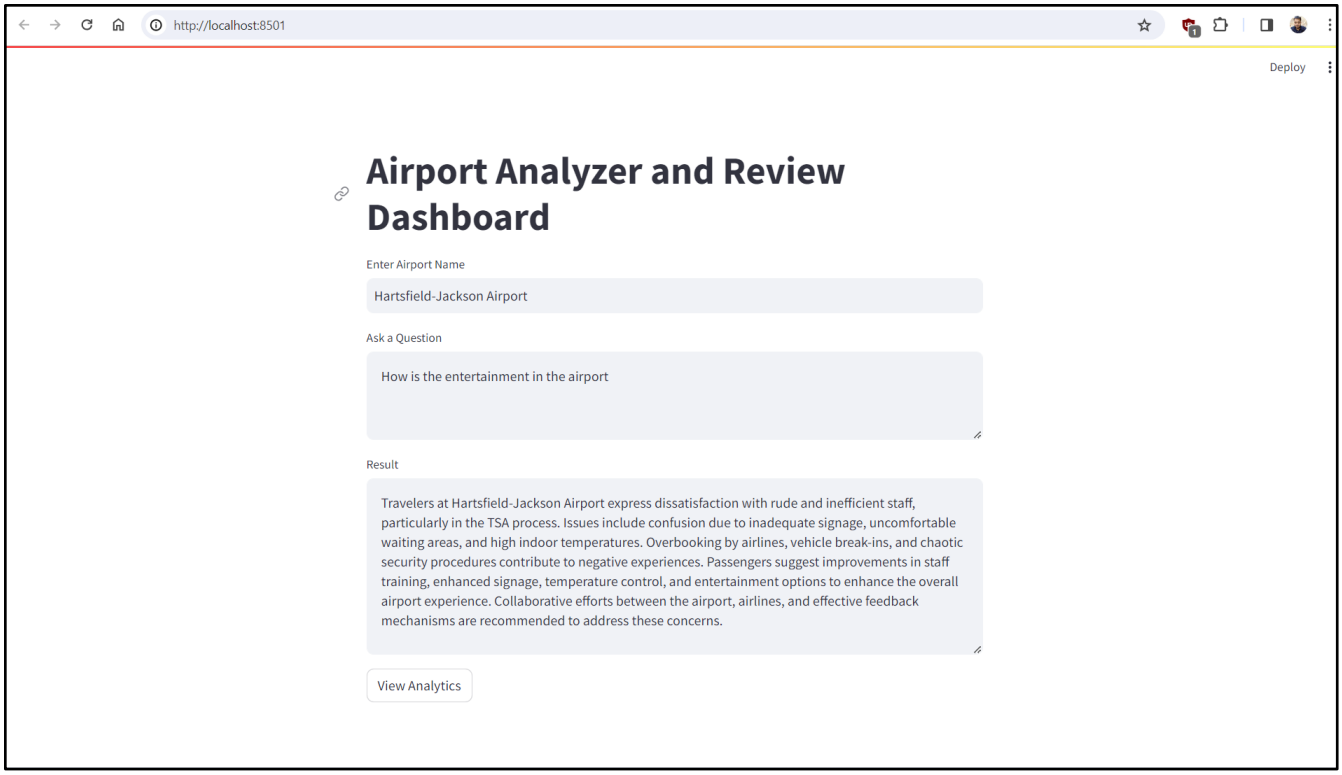
3. `time`

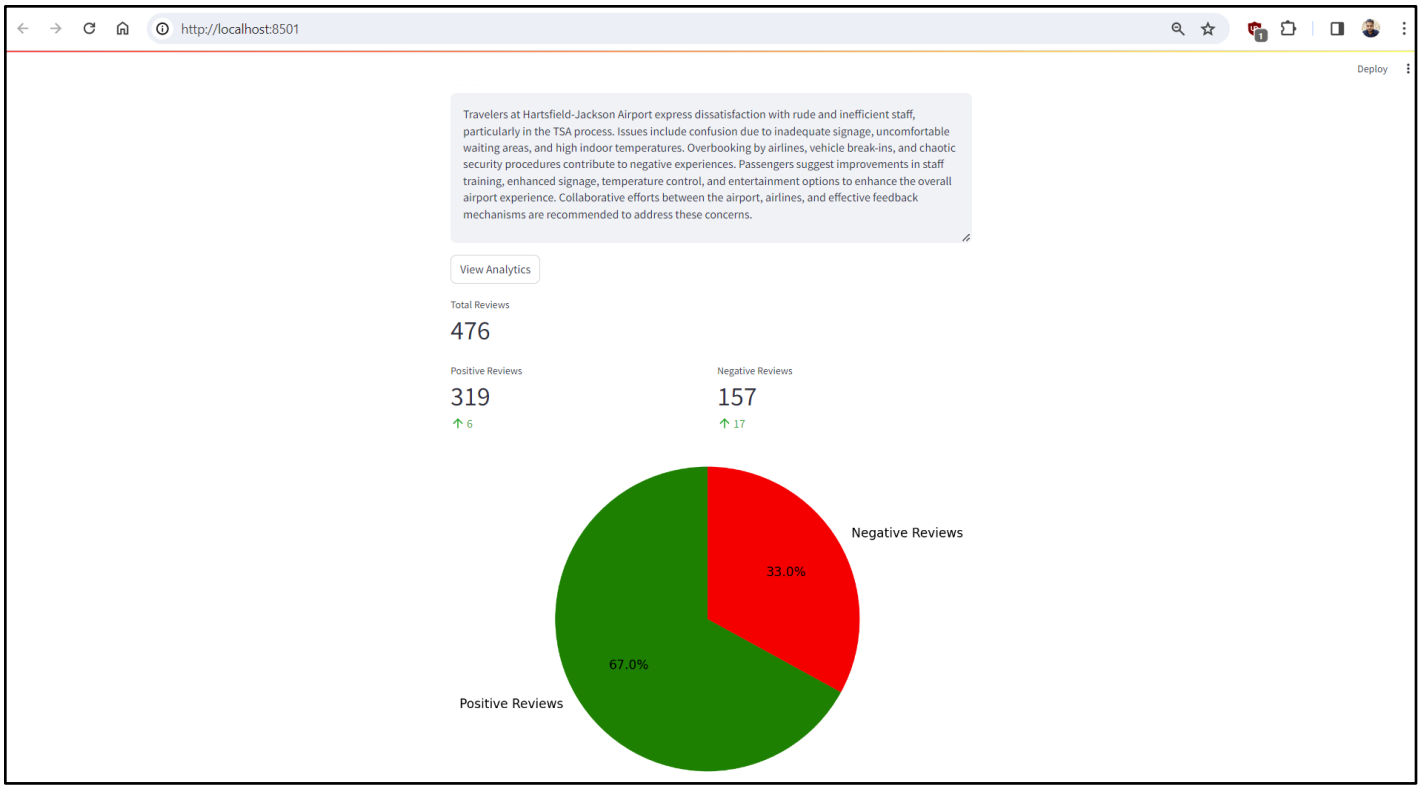
4. `langchain` (specifically, `OpenAI` and various components from `langchain.chains`, `langchain.text_splitter`, `langchain.document_loaders`, `langchain.embeddings`, and `langchain.vectorstores`)

Expected Outcome

The aim of the application is to provide a user-friendly interface for analyzing airport reviews. After a user selects an airport, the application comprehensively reads all associated reviews. It then stands ready to address any questions regarding these reviews. Users can, for instance, prompt the application to identify the most frequently mentioned feature of a chosen airport. Inquiries can range from discerning the best and worst facilities to gauging the overall public sentiment surrounding the airport. The depth and focus of the application's insights are directed by the specificity of the user's prompts, as seen in the provided user interface.

Mock UI Image





Relevance to Business Analytics

The Airport Analyzer application holds immense promise in the realm of business analytics, offering invaluable insights into customer experience, operational efficiency, competitive analysis, and strategic decision-making. Its global reach further amplifies its significance, providing a comprehensive understanding of market dynamics and enabling the customization of services to meet diverse customer needs.

- **Data-driven Decision Making:** The project uses NLP approaches to extract significant insights from unstructured data (such as airport reviews). These insights enable airport administrators to make data-driven decisions about infrastructure upgrades, service improvements, and resource allocation.
- **Operational Efficiency**: Insights from customers can provide valuable feedback on various operational aspects, such as check-in processes, security procedures and overall airport facilities. Businesses analysts can use this insight to improve the overall passenger experience.
- **Predictive Analytics:** By analysing historical airport evaluations, the AI system may discover trends and patterns, allowing for predictive analytics. This enables preventive steps to resolve potential issues before they escalate, resulting in higher overall customer satisfaction and loyalty.
- **Customer Experience Optimization**: The application's ability to discern sentiments and preferences from user reviews allows airports to successfully modify their services and facilities to match consumer expectations. This promotes improved customer experiences, resulting in increased consumer satisfaction and loyalty.
- **Strategic Decision Making:** Insights from AI can help airport authorities, airlines, and stakeholders make strategic decisions in the aviation business. Business analysts can help airports and airlines develop data-driven plans, optimize resource allocation, and increase their worldwide competitiveness.

References

1. Hasib, Khan & Naseem, Usman & Keya, Ashfia & Maitra, Shithi & Mithu, Kamruzzaman & Alam, Md Golam Rabiul. (2023). A Systematic Review on Airlines Industries based on Sentiment Analysis and Topic Modeling. 10.21203/rs.3.rs-3475984/v1.
2. Jeong, Cheonsu. (2023). Generative AI service implementation using LLM application architecture: based on RAG model and LangChain framework. Journal of Intelligence and Information Systems. 19. 129-164. 10.13088/jiis.2023.29.4.129.
3. Kondurkar, Isha & Raj, Akanksha & Lakshmi, D.. (2023). Modern Applications with a Focus on Training ChatGPT and GPT Models: Exploring Generative AI and NLP. 10.4018/979-8-3693-0502-7.ch010.