CS432/532: Final Project Report

Project Title: Airline Review Analysis

Team Member(s): Atharva Patil, Shreyas Kadam

/* Your project will be graded based on the significance of the project, the success of the demo, and the clarity of your report. Your final report excluding the references and source code should be 2-3 pages. */

/* **Please organize your report for clarity, and always check spelling and grammar. An excellent style manual for science writers is [2].** */

I. N+1 NOSQL QUERIES

/* Clearly describe the N+1 NoSQL queries that you have implemented. */

Query 1 - Food & Beverages Feedback Analysis:
Analyzes customer sentiments from food and beverage reviews, groups them by sentiment, calculates the average 'value for money' rating per group, counts reviews, and sorts by average rating. It outputs sorted sentiment categories with average value ratings, providing insight into customer satisfaction related to cost perception.

Query 2 - Analysis of Sentiment Versus Value for Money:
Investigate the relationship between sentiment expressed in review and the 'valueForMoneyRating'.
Determine if higher satisfaction correlates with perceptions of value for money, or if expectations vary significantly across different price points.

Query 3 - Sentiment by Route Popularity:
Categorize routes by popularity based on the number of reviews.
Analyze sentiments to determine if more popular routes have better or worse reviews, possibly indicating operational focus areas.

II. NOSQL DATABASE AND DATASET

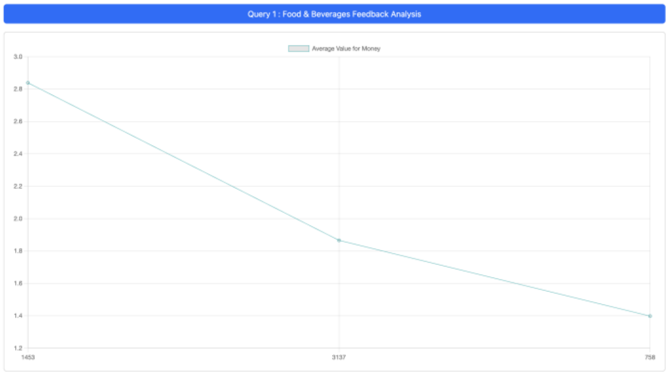
For this project, we will be using the Airline Reviews dataset from Kaggle website. The URL for this dataset is:-

https://www.kaggle.com/datasets/juhibhojani/airline-reviews

This dataset contains a csv file containing data for airline reviews. The CSV file has 20 attributes and over 5000 samples.

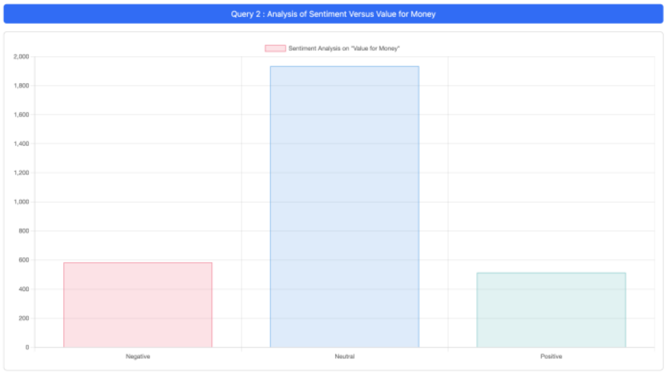
III. PROJECT OUTCOME

Query 1



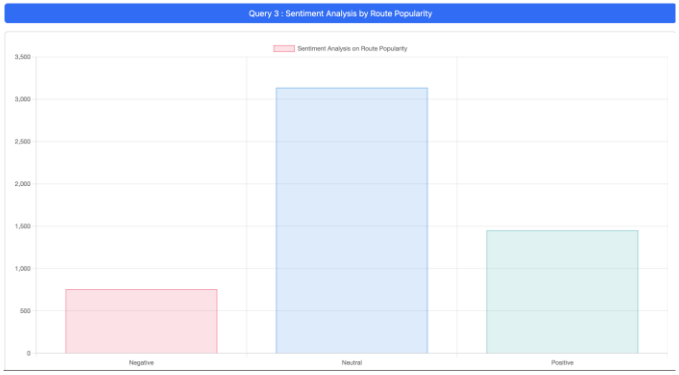
This Flask route analyzes airline food and beverage feedback by executing a MongoDB pipeline. It assesses reviews with both food and beverages and value for money ratings, then groups them by sentiment—likely derived from a text analysis algorithm (`sentiment_query`). It calculates the average value for money rating for each sentiment, counts the reviews, and sorts the results by the average rating in descending order. The outcome is a chart plotting these averages, with a downward trend indicating that the sentiment may correlate inversely with the perceived value for money. The X-axis labels are unclear but seem to be sentiment-related identifiers.

Query 2



This code defines a Flask route that processes airline review data to perform sentiment analysis through a MongoDB aggregation pipeline. It first injects a sentiment label into each review, then groups the results by the 'value for money' rating. The pipeline counts reviews and computes the sentiment distribution for each rating group. The bar chart visualization shows the count of reviews for each sentiment type, indicating that neutral feedback predominates, with fewer instances of positive and even fewer negative sentiments. This analysis helps understand customer perception of the value for their money in the context of airline services.

Query 3



This Flask route executes a MongoDB aggregation pipeline analyzing sentiments across different flight routes. It groups reviews by route, counts them, pushes sentiments into arrays, and sorts by the number of reviews to determine route popularity. Then it calculates sentiment distribution—positive, neutral, negative—for each route. The operation adds sentiment counts, and the final projection removes the MongoDB `_id` field. The outcome, visualized in a bar chart, reveals most reviews are neutral, suggesting a general customer satisfaction without strong feelings. Positive reviews are noticeable but less than neutral, while negative feedback is the least, indicating relatively favorable overall perceptions.

References

[1] R. Elmasri and S. B. Navathe, Algorithms, Fundamentals of Database Systems, Seventh Edition, Pearson, 2017.

[2] Michael Ernst, How to Write a Technical Paper, https://homes.cs.washington.edu/~mernst/advice/write-technical-paper.html .