**Final Report: Smart City Public Transportation Project**

Optimizing Smart City Public Transportation Through Data and Technology

DS-670-HYB2-23WNTR

Capstone: Big Data & Bus Analysis

Reda Mastouri

30/01/2024

Kethana devi kadiyala

Venkata Sairam Rohith Mechineni

Deva Sai Vikas Vakkalagadda

**1. Exploratory Data Analysis and Clustering:**

Data Preprocessing:

* Explored the dataset to understand its structure and characteristics.
* Removed columns with a significant number of missing values to ensure data quality.
* Imputed missing values in numeric columns with the mean and in categorical columns with the mode.

KMeans Clustering:

* Investigated the optimal number of clusters using the Elbow method.
* Utilized KMeans clustering with the optimal number of clusters for grouping the data.
* Evaluated the quality of clustering using the silhouette score, indicating the cohesion and separation of clusters.

Hierarchical Clustering:

* Visualized hierarchical clustering using a dendrogram to understand the hierarchical structure.
* Determined an optimal number of clusters by inspecting the dendrogram.
* Applied hierarchical clustering to the data and assessed the resulting clusters.

DBSCAN:

* Implemented DBSCAN for identifying spatial clusters and detecting outliers.
* Evaluated the silhouette score to measure the effectiveness of clustering.

Gaussian Mixture Model (GMM):

* Applied GMM to model the data distribution and assign probabilities to each point belonging to a cluster.
* Evaluated the silhouette score to assess the quality of GMM clustering.

Agglomerative Clustering:

* Employed Agglomerative Clustering to group data points into clusters based on their similarity.
* Visualized the spatial distribution of clusters on geographical maps.

**2. Flask Application (app.py):**

Data Integration:

* Loaded the preprocessed dataset into the Flask application for real-time predictions.
* Applied standardization to incoming user data to align with the preprocessing steps.

GMM Integration:

* Integrated the trained GMM model into the Flask application for predicting clusters.
* Used the GMM model to dynamically cluster user input and visualize the results.

Real-time Plotting:

* Utilized **seaborn** and **matplotlib** for real-time plotting of clusters on geographical maps.
* Saved the generated cluster plot as a static image for display in the application.

**3. User Interface (index.html):**

Form Design:

* Designed a clean and intuitive HTML form with input fields for relevant attributes.
* Ensured proper alignment of input fields with appropriate labels for user-friendly interaction.

Cluster Prediction:

* Leveraged Flask to process user inputs and predict the cluster using the GMM model.
* Displayed the predicted cluster to users and visually represented the cluster on the geographical map.

**4. Testing and Iteration:**

Rigorous Testing:

* Conducted extensive testing to validate the accuracy and reliability of cluster predictions.
* Iteratively refined the application based on user feedback and testing outcomes.

Performance Optimization:

* Explored opportunities to optimize the application's performance, ensuring responsiveness even with larger datasets.
* Addressed potential bottlenecks to enhance user experience.

**5. Future Steps:**

Model Refinement:

* Plan to refine clustering models based on additional data and insights gathered during application usage.
* Explore advanced clustering techniques and assess their impact on prediction accuracy.

Feature Enhancement:

* Consider implementing advanced features such as dynamic route planning and real-time updates for commuters.
* Enhance the application's capabilities to provide more valuable information to users.

User Feedback:

* Encourage users to provide feedback for continuous improvement.
* Prioritize user experience and satisfaction in further iterations, incorporating user suggestions and preferences.

**Prerequisites:**

1. **Python Installed:**
   * Make sure you have Python installed on your system. You can download it from Python's official website.
2. **Required Libraries:**
   * Install the required Python libraries by running the following command in your terminal or command prompt:

**pip install flask pandas matplotlib seaborn scikit-learn**

**Steps to Run the Flask App:**

1. **Download Code:**
   * Download the Flask app code (app.py, index.html) and the traffic dataset.
2. **Navigate to the Project Directory:**
   * Open a terminal or command prompt and navigate to the directory where your Flask app code is located.
3. **Run the Flask App:**
   * Execute the following command to run the Flask app:

**python app.py**

* + This command will start the Flask development server.

1. **Access the App:**
   * Once the Flask app is running, open your web browser and go to http://127.0.0.1:5000/.
2. **Use the Application:**
   * You will see the input form on the webpage. Enter values for the specified attributes.
   * Click the "Predict Cluster" button to submit the form.
3. **View Results:**
   * The app will predict the cluster for the input data using the trained GMM model.
   * The predicted cluster will be displayed, and a geographical map with the clusters will be generated.
4. **Explore and Test:**
   * Explore the application, test different inputs, and observe how the clustering results change.
5. **Close the App:**
   * To stop the Flask app, go back to the terminal or command prompt and press Ctrl+C.