Smart Internz

SMARTBRIDGE
Let's Bridge the Gap

Traffic Telligence

Advanced Traffic Volume Estimation with Machine Learning

Team Members

- Mulla Ansar
- S M Ibrahim
- Potladurthi harshitha
- Baska Thulasi Kumar

Purpose:

Urban transportation is a critical infrastructure that significantly affects daily life, logistics, and environmental health. Traditional traffic monitoring systems rely on manual counters or static sensors which often fail to provide real-time or predictive insights.

TrafficTelligence leverages machine learning to estimate and forecast traffic volume accurately using a wide range of features like time, weather, historical traffic patterns, and events. The main goal is to help city planners, traffic departments, and commuters optimize travel time and make informed decisions.

Features:

- Accurate traffic volume predictions using trained ML models
- Web interface to visualize predictions, trends, and graphs
- RESTful API integration for model access
- User authentication for secured access
- Admin dashboard to manage prediction history and logs
- Scalable architecture using the MERN stack

- Architecture:-
- The system is structured around a machine learning core supported by data preprocessing and a lightweight server for prediction requests.
- Components: -
- Data Collection & Cleaning (CSV/Excel) - Machine Learning Model (Regression or Ensemble) - Flask API for model interaction - Visualization interface (optional via Streamlit/Flask UI)

- Setup Instructions

- Prerequisites: -
- Python 3.x - Libraries: Flask, Pandas, Scikit-learn, NumPy, Matplotlib

- Installation:
- 1. Clone repository
- 2. Set up virtual environment and install dependencies
- 3. Place training dataset in the designated folder
- 4. Train model using the provided script
- 5. Launch Flask server to access prediction endpoint

- Name
- >ipynb_checkpoints
- Flask
- templates
- app.py
- encoder.pkl
- model.pki
- IBM
- >Flask
- >traffic volume_lbm_scoring end point.ipynb
- Requirements.txt
- Traffic volume estimation.docx
- traffic volume.csv
- traffic volume.ipynb

- Running the Application:

- Run the following command from the project root to start the backend server
- $ python api/app.py For UI (if any), navigate to localhost and follow the interface to input parameters for prediction.

- API Documentation POST /predict:

- Description:
- Predicts traffic volume based on input features -
- Input: JSON { "hour": 9, "weather": "Clear", "holiday": false }
- Output: JSON { "prediction": 422 } GET /history - Returns all previous prediction logs (if implemented)

- Authentication :

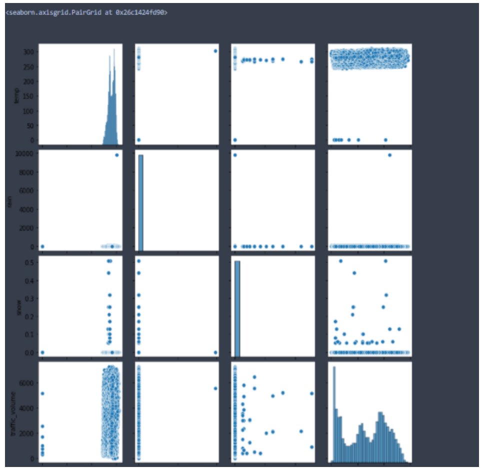
- Currently, no authentication is implemented. Future versions may include:
- -Admin-only access to logs -
- Token-based user-level acces

- User Interface :

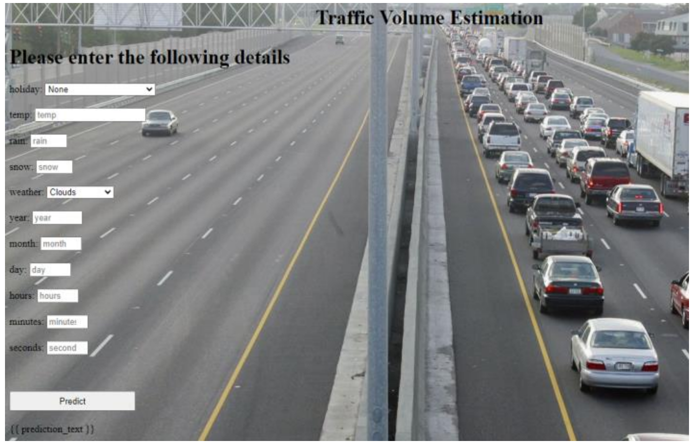
- A basic UI may be included to allow non-technical users to input time, date, and other parameters for traffic prediction. Results are shown in real-time along with visual charts.

- Testing Model testing included:

- Train/test split evaluation (e.g., 80/20)
- Accuracy metrics like R2 Score, MAE
- Manual testing through the API using Postman or cURL
- Demo :



```
<seaborn.axisgrid.PairGrid at 0x26c1424fd90>
```



Traffic Volume Estimation

Please enter the following details

holiday: None

temp: temp

rain: rain

snow: snow

weather: Clouds

year: year

month: month

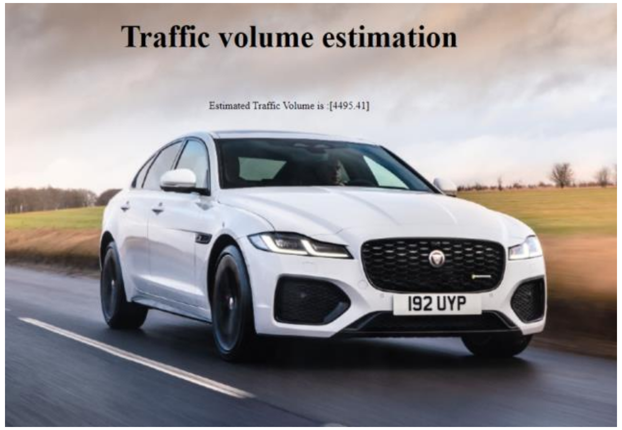
day: day

hours: hours

minutes: minutes

seconds: second

{{ prediction_text }}



Traffic volume estimation

Estimated Traffic Volume is :[4495.41]

- Known Issues :
- Model accuracy drops during rare weather conditions due to lack of data
- No real-time sensor data yet
- No built-in data validation for malformed requests

- Future Enhancements :
- Integrate live data from traffic sensors or APIs
- Use deep learning models (e.g., LSTM for time-series forecasting)
- Deploy the project to cloud for real-time accessibility
- Develop a companion mobile app for public use