TrafficTelligence Advanced Traffic Volume Estimation with Machine Learning

1.Introduction

Project Title: TrafficTelligence Advanced Traffic Volume Estimation with Machine Learning

Team Members:

1. ***R. Akhila***
2. ***R.Mounika***
3. ***R.Pratima***
4. ***S. Musharaf***

2.Project Overview

2.1 **Purpose:** This project aims to develop a machine learning-based predictive model to estimate traffic volume using historical traffic sensor data and real-time features like weather, time, and road type. It applies advanced algorithms and hyperparameter tuning to achieve accurate volume predictions. The

best-performing model is deployed via Flask, making it suitable for real-time integration into smart city traffic systems.

2.23 **Features:** TrafficTelligence is a robust machine learning solution that predicts traffic flow based on various contextual and sensor data. The project includes data preprocessing (handling missing values, encoding categorical variables, and normalizing data), training on regression algorithms like Linear Regression, Random Forest, Gradient Boosting, and XGBoost. Evaluation metrics such as MAE, MSE, RMSE, and R2 score guide model selection. The final model is deployed using a Flask web app for live input and prediction, contributing to better urban traffic planning and congestion management.

3.Architecture

3.1 **Frontend:** Built with HTML and CSS, the interface allows users to input features like time of day, weather conditions, and road details. Upon submission, this data is sent to the backend, and the predicted traffic volume is displayed for user interpretation.

3.2 Backend: Implemented with Flask, the backend processes the frontend data, applies preprocessing, and feeds it to the trained ML model. It returns predictions for display and can store the results in a database if required.

3.3**Database:** MongoDB stores the user inputs and their respective predicted traffic volume. Each record includes input features, predicted output, and a timestamp. Flask integrates with MongoDB using PyMongo.

4.Setup Instructions

4.1 **Prerequisites:**

- Python 3.x
- Pandas
- NumPy
- Scikit-learn
- XGBoost
- Flask
- Flask-PyMongo
- Joblib or Pickle
- MongoDB

4.2 Installation:

pip install flask pandas numpy scikit-learn xgboost flask-pymongo git clone <your-repo-url> cd https://github.com/SHAIKMUSHARAF123/TrafficTelligence-Advanced-Traffic-Volume-Estimation-with-Machine-Learning.git

Visit: http://127.0.0.1:5000

Demo link:
https://drive.google.com/file/d/1nidBBM3QvHSUon_IPPL07A9bnn1tFzoV/view?usp=drive_link

5.Folder Structure

5.1 **Client (Frontend):** Built with HTML/CSS. A form allows user entry for features like day, hour, temperature, humidity, and weather. On submit, data is sent to the Flask API and results are shown on the same or redirected page.

5.2 Server (Backend): The Flask server processes the input, applies preprocessing, loads the model, and returns the result. It also logs predictions to MongoDB if enabled.

6.API Documentation

1. POST /predict

Request: application/json

```
{
  "hour": 17,
  "day_of_week": "Monday",
  "weather": "Clear",
  "temp": 24.5,
  ...
}
```

Response:

```
{
  "prediction": 352,
  "result": "Estimated traffic volume: 352 vehicles/hour"
}
```

2. GET /

Response:

```
{
  "message": "Welcome to the Traffic Volume Estimation API"
}
```

3. POST /store-data

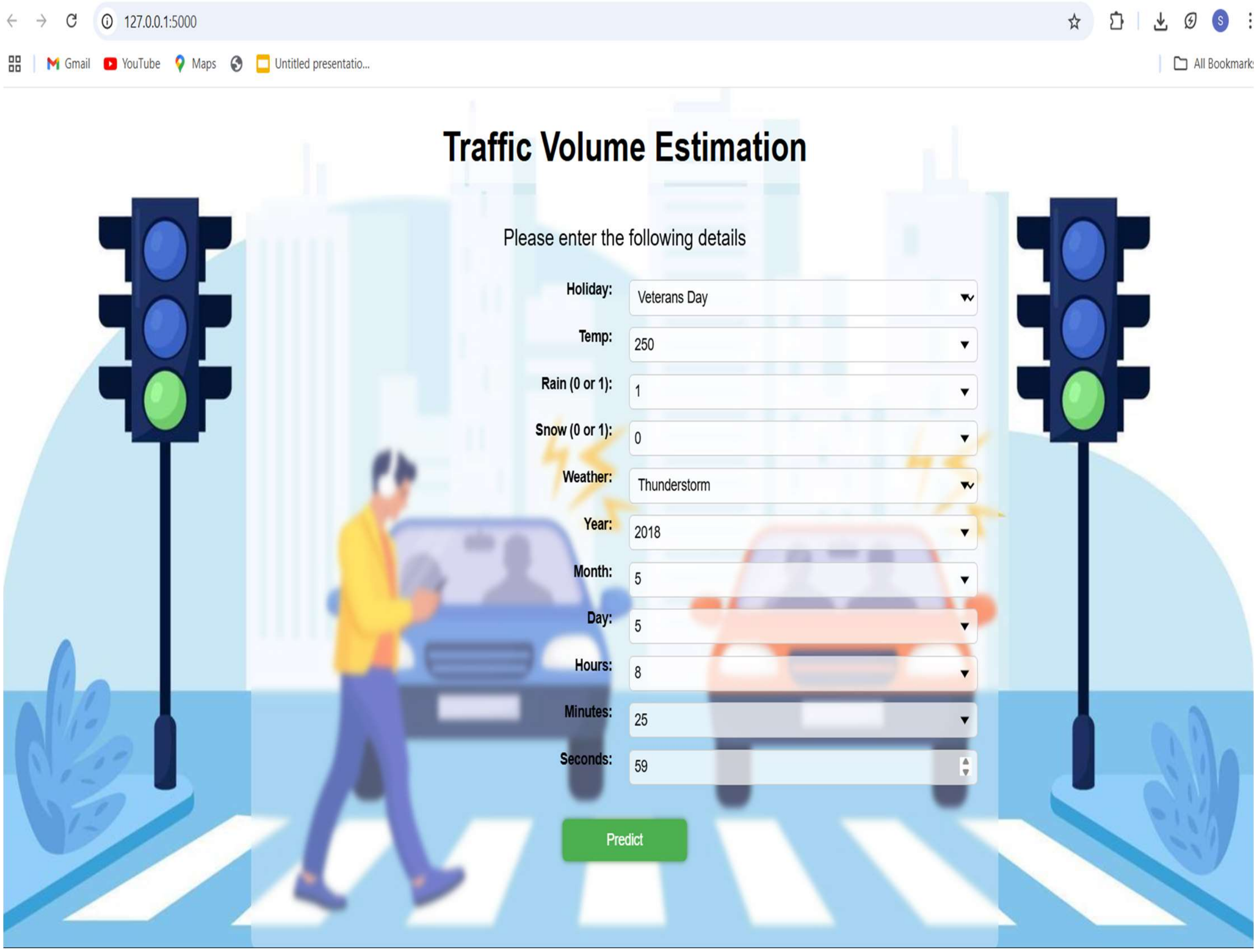
Optional for storing input and prediction in DB.

8.Authentication

No authentication added; the tool is open for general use.

9.User Interface

Clean, responsive interface for data input and instant traffic volume predictions. Designed for planners, researchers, and analysts.



Traffic Volume Estimation

Please enter the following details

Holiday: Veterans Day

Temp: 250

Rain (0 or 1): 1

Snow (0 or 1): 0

Weather: Thunderstorm

Year: 2018

Month: 5

Day: 5

Hours: 8

Minutes: 25

Seconds: 59

10.Testing

Data Validation: Cleaned missing values, normalized features.

Model Evaluation: MAE, RMSE, R2 score for regression models.

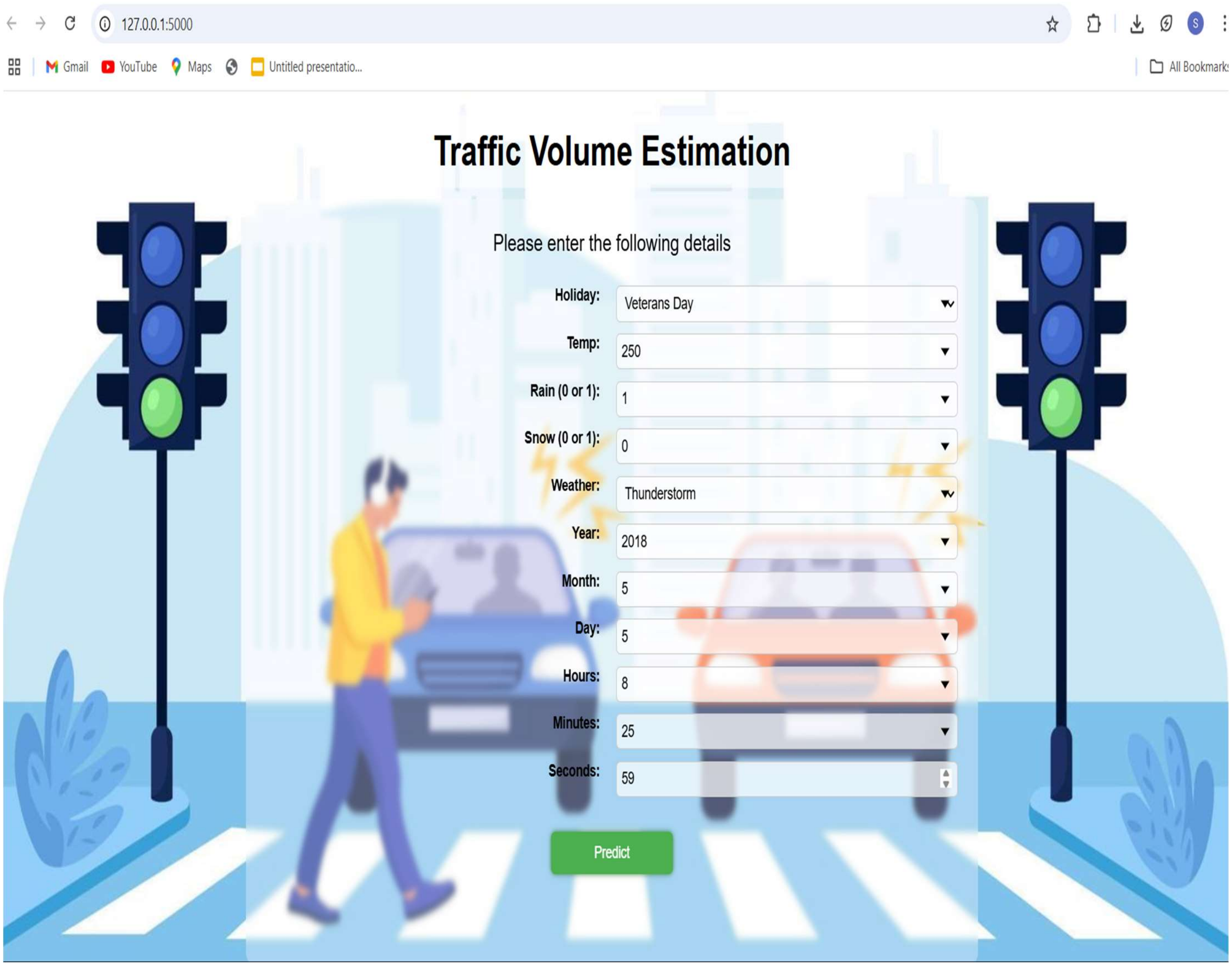
API Testing: Ensured endpoints function with valid/invalid input.

Integration: Verified flow from form to prediction.

Edge Cases: Tested for unusual conditions like weather or time.

Tools: scikit-learn, XGBoost, Pytest, Postman, Jupyter, DevTools.

11.Screenshots



Traffic Volume Estimation

Please enter the following details

Holiday: Veterans Day

Temp: 250

Rain (0 or 1): 1

Snow (0 or 1): 0

Weather: Thunderstorm

Year: 2018

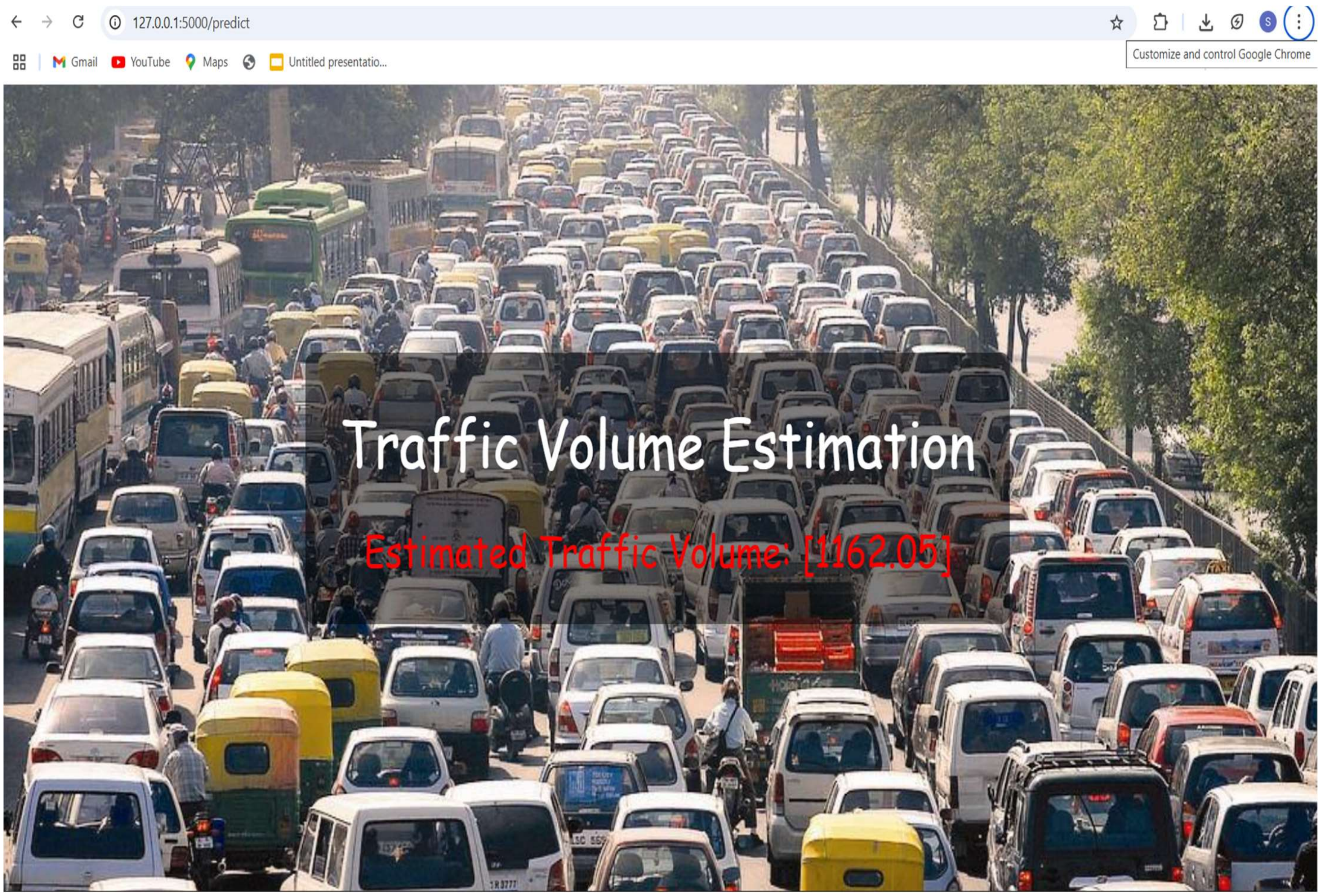
Month: 5

Day: 5

Hours: 8

Minutes: 25

Seconds: 59



Traffic Volume Estimation

Estimated Traffic Volume: [1162.05]

12.Known Issues

1. Data Imbalance Low volume segments (e.g., night) affect predictions.
2. Sensor Errors Raw data may have noise/missing values.
3. Deployment Limitations Flask isn't ideal for large-scale use.
4. Input Validation Minimal frontend checks.
5. No Role-Based Access Public tool.

13.Future Enhancements

SHAP/LIME for Explainability

Mobile UI Optimization

Role-based Access (Doctor/Admin/Researcher)

Real-time Dashboard for Predictions

Cloud Deployment with Docker + AWS/Heroku