

## Final Report: Traffic Volume Prediction System

### TEAM MEMBERS

**Team ID :** LTVIP2025TMID56158

**Team Size :** 2

**Team Leader :** Gadamsetty Venkata Priyanka

**Team member :** Karnati Nagendra Prasad

---

## 1. INTRODUCTION

### 1.1 Project Overview

This project aims to predict the volume of traffic using a machine learning model based on factors such as date, time, weather conditions, and holiday information. The solution integrates a Random Forest Regressor with a Flask-based web application that takes user input and returns predicted traffic volume.

### 1.2 Purpose

The primary purpose is to enable traffic authorities or users to predict vehicle congestion levels at a given time, helping with traffic management and planning.

---

## 2. IDEATION PHASE

### 2.1 Problem Statement

Traffic congestion is a common issue in urban areas. Predicting traffic volume can help city planners and drivers make informed decisions.

### 2.2 Empathy Map Canvas

- **Who are we empathizing with?** Commuters, traffic managers, city planners
- **What do they need to do?** Avoid congested routes, plan trips better
- **What do they see?** Long travel times, inconsistent traffic reports
- **What do they hear?** Complaints, alerts, advisories
- **What do they say and do?** Use maps, plan around peak times
- **What do they think and feel?** Frustrated, stressed, time-pressured

### 2.3 Brainstorming

- Use machine learning to predict traffic volume
- Provide a user-friendly interface

- Include weather, date, and time features for better prediction
- 

### 3. REQUIREMENT ANALYSIS

#### 3.1 Customer Journey Map

1. User visits the prediction page
2. Inputs date, time, weather, and holiday details
3. Receives traffic volume prediction
4. Plans accordingly

#### 3.2 Solution Requirements

- Preprocessed dataset with traffic and weather data
- Trained machine learning model
- Flask server with prediction API
- HTML frontend for inputs

#### 3.3 Data Flow Diagram

1. User inputs data
2. Flask app receives input
3. Model makes prediction
4. Flask app returns result to user

#### 3.4 Technology Stack

- Python
  - Pandas, NumPy
  - Scikit-learn (Random Forest)
  - Flask
  - HTML/CSS (Bootstrap optional)
- 

### 4. PROJECT DESIGN

#### 4.1 Problem Solution Fit

We use historical data and weather metrics to build a model capable of predicting traffic volume with high accuracy.

#### 4.2 Proposed Solution

A Random Forest Regressor model trained on traffic data and deployed using Flask to interact with users through a web form.

#### 4.3 Solution Architecture

1. Data Cleaning and Feature Engineering in Jupyter Notebook
2. Model Training and Pickle File Creation

3. Flask Backend to Load Model and Handle Requests
  4. HTML Form to Collect User Input
  5. Return Prediction Output on the Same Page
- 

## 5. PROJECT PLANNING & SCHEDULING

### 5.1 *Project Planning*

- Week 1: Dataset exploration, cleaning
  - Week 2: Feature engineering, model training
  - Week 3: Web application setup with Flask
  - Week 4: Integration, testing, and report creation
- 

## 6. FUNCTIONAL AND PERFORMANCE TESTING

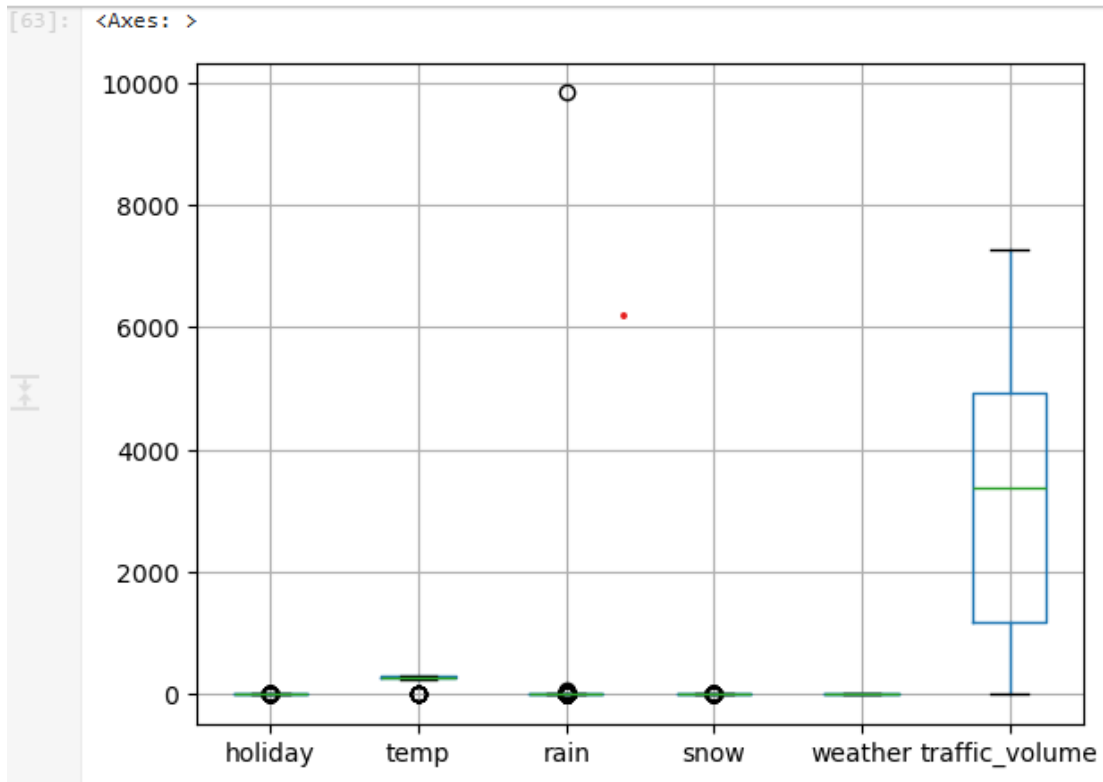
### 6.1 *Performance Testing*

- Model tested using R2 Score and Mean Squared Error
  - Model performs with acceptable accuracy on test data
- 

## 7. RESULTS

### 7.1 *Output Screenshots*

- Screenshot of Jupyter Notebook with model results



- 
- Screenshot of HTML form interface

127.0.0.1:5000

Please enter the following details

oliday:

emp:

ain:

ow:

weather:

ear:

onh:

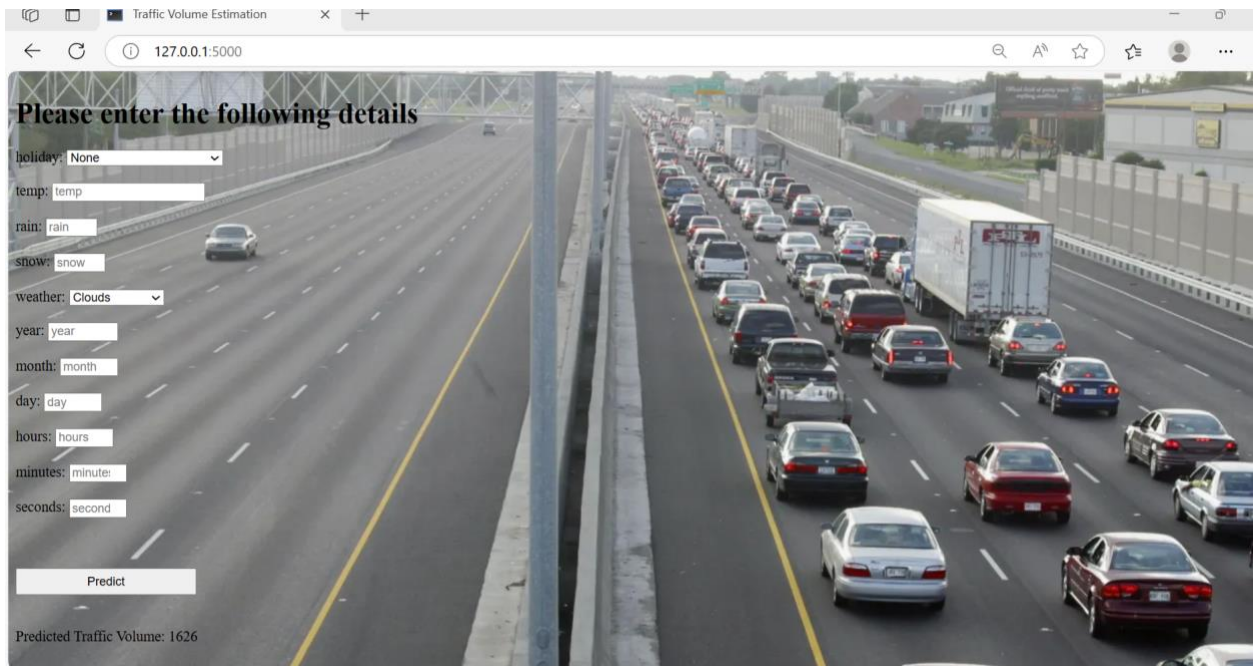
ay:

ours:

inutes:

econds:

- 
- Screenshot of predicted output in browser



---

## 8. ADVANTAGES & DISADVANTAGES

**Advantages:** - Real-time prediction - User-friendly - Scalable with additional features like location, vehicle type

**Disadvantages:** - Dependent on quality of input data - Limited to historical patterns

---

## 9. CONCLUSION

This project successfully integrates machine learning and web development to provide a usable traffic volume prediction system. It enables smarter urban planning and helps users avoid high traffic periods.

---

## 10. FUTURE SCOPE

- Integrate with Google Maps API
  - Use live weather and traffic data
  - Build mobile app version
- 

## 11. APPENDIX

- **Source Code:** Available upon request or in GitHub repo

- **Dataset Link:**  
[https://drive.google.com/file/d/1iV5PfYAml6YP0\\_0S4KYy1ZahHOqMgDbM/view](https://drive.google.com/file/d/1iV5PfYAml6YP0_0S4KYy1ZahHOqMgDbM/view)
- **GitHub & Project Demo:**  
[https://github.com/Priyanka171203/Traffic\\_Telligence\\_Project](https://github.com/Priyanka171203/Traffic_Telligence_Project)