

## Phase-2

**Student Name** : Bakkiyalakshmi.P

**Register Number** : 422723104015

**Institution**: V.R.S College of Engineering  
and Technology

**Department**: Computer science  
Engineering

**GitHub Repository Link**:

[https://github.com/bakkiyalakshmi868/  
Bakkiya-lakshmi.git](https://github.com/bakkiyalakshmi868/Bakkiya-lakshmi.git)

# Enhancing Road Safety through AI driven Traffic Analysis and Prediction

---

## 1. Problem Statement

Road traffic accidents cause significant loss of life and property every year. Despite advancements in

transportation infrastructure, the prediction and prevention of accidents remain a challenge due to complex

and dynamic traffic conditions. This project aims to use AI to analyze traffic data and predict accident-prone

scenarios, enabling authorities to implement preventive measures.

- Problem Type: Classification (Accident vs. No Accident) and/or Regression (Severity Prediction)

- Impact: Enhances road safety by enabling proactive interventions, resource allocation, and informed urban planning.

## 2. Project Objectives

- Build AI models that predict traffic accidents based on real-time and historical data.

- Achieve high accuracy and recall to minimize false negatives.

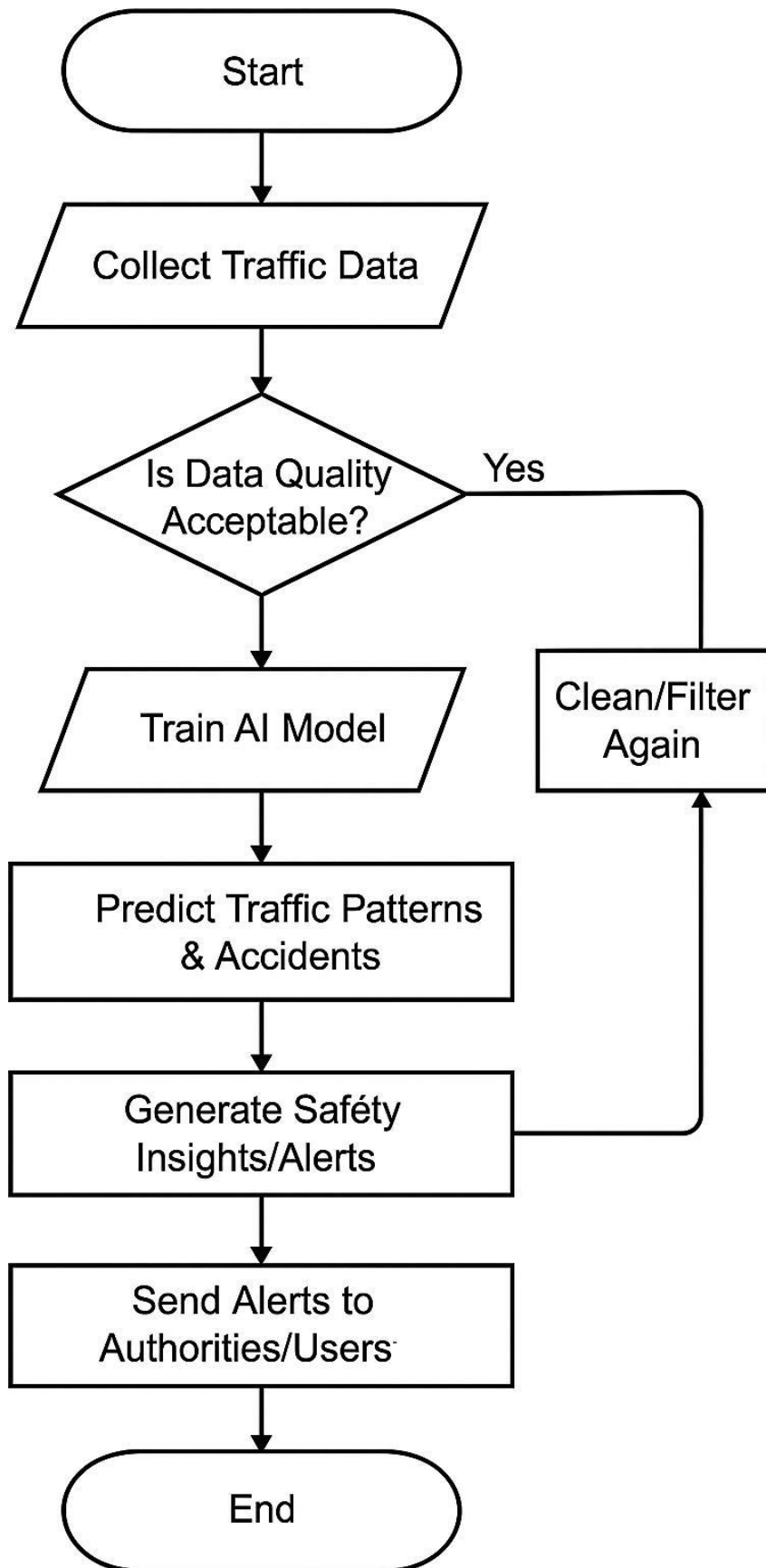
- Provide interpretable insights for authorities to act upon.

- Evolve feature design and modeling based on data exploration outcomes.

## 3. Flowchart of the Project Workflow

The diagram below represents the workflow of the AI-driven movie recommendation system:

Start → Data Collection → Data Preprocessing → User Profile Building → Model Selection →  
Model Training → Recommendations Generation → User Feedback Collection → Model  
Evaluation and Tuning → End (Loop)



## 4. Data Description

- Dataset Name: Road Traffic Accidents Dataset
- Source: Kaggle / Open Government API / UCI
- Type: Structured, Time-Series
- Size: ~200,000 records, 25+ features
- Static/Dynamic: Dynamic
- Target Variable: Accident Occurrence / Severity

## 5. Data Preprocessing

### 1.Data Collection

Sources: CCTV footage, traffic sensors, GPS data, social media feeds, weather reports, and traffic apps.

Formats: Images, video, tabular logs, JSON, etc.

### 2. Data Cleaning

Removing Noise: Eliminate irrelevant or redundant data (e.g., blurry images, corrupted GPS points).

Handling Missing Values: Imputation (mean/mode), removal, or interpolation.

Outlier Detection: Identifying anomalies in speed, volume, or travel time data.

### 3. Data Integration

Combining Multiple Sources: Merging GPS, sensor, and weather data for a unified view.

Synchronization: Aligning data based on time stamps and location.

### 4. Data Transformation

Normalization/Scaling: Bringing data into a standard range for machine learning models.

Encoding Categorical Variables: Converting data like road types or weather conditions into numerical format.

### 5.Data Annotation (for AI models)

Manual or Semi-Automated Labeling: For tasks like object detection in videos (e.g., vehicles, pedestrians).

Labeling Events: Accidents, congestion, or rule violations.

## 6. Exploratory Data Analysis (EDA)

### 1. Data Overview

Dimensions: Number of records and features (e.g., time, vehicle count, speed, location)

Data Types: Numerical (speed), categorical (road type), temporal (timestamp)

Missing Values: Identify and visualize missing or null values

#### ->Univariate analysis is

Histograms & Boxplots: Vehicle speed, traffic volume, accident frequency

Distribution Checks: Normality and skewness of features

#### ->Bivariate Analysis

Scatter Plots: Speed vs. accident count

Correlation Matrix: Identify relationships between variables (e.g., congestion vs. weather)

#### ->Time Series Analysis

Traffic Trends: Daily, weekly, and seasonal traffic patterns

Peak Hours: Time-of-day analysis for congestion

#### ->Geospatial Analysis

Traffic Hotspots: Map visualizations of high-accident or high-congestion zones

GPS Data Clustering: Identify patterns in vehicle movement

#### ->Outlier Detection

Visual & Statistical Methods: Z-score, IQR, boxplots for spotting anomalies

## 7. Feature Engineering

Creating New Features: Time of day, day of the week, average vehicle speed, congestion index.

Dimensionality Reduction: PCA or t-SNE for handling high-dimensional datasets.

## 8. Model Building

### 1. Define the Problem

Types:

1. Classification: Accident vs. non-accident
2. Regression: Predict vehicle count, speed, or delay
3. Time Series Forecasting: Predict future traffic flow

Clustering: Group similar traffic patterns (e.g., using GPS data)

### 2. Data Preparation

Train-Test Split: Usually 70/30 or 80/20

Cross-Validation: Ensures model generalization

Feature Selection: Choose relevant features using correlation, importance scores

### 2. Model Selection

Traditional ML Algorithms:

Linear Regression

Decision Trees / Random Forest

Support Vector Machines (SVM)

K-Means / DBSCAN (for clustering)

Deep Learning Models:

CNNs (for video/image data)

RNNs / LSTM (for time series traffic data)

Autoencoders (for anomaly detection)

### 4. Model Training

Input: Cleaned and structured dataset

Tools: Scikit-learn, TensorFlow, PyTorch, Keras

Hyperparameter Tuning: Grid search, Random search

### 5. Model Evaluation

Metrics:

Accuracy, Precision, Recall (classification)

MAE, RMSE (regression)

F1 Score, AUC-ROC

Visualization: Confusion matrix, prediction plots

## 6. Model Optimization

Techniques:

Feature scaling

Regularization

Ensemble methods (e.g., XGBoost, Bagging)

## 9. Visualization of Results and Model Insights

Model Performance:

Confusion matrix, accuracy, MAE/RMSE, ROC curve.

Feature Importance:

Bar charts, SHAP/LIME for explainability.

Trend Analysis:

Line plots for traffic flow, anomaly detection.

Geospatial Maps:

Heatmaps for accident/congestion zones.

Cluster & Dimensionality Visuals:

K-Means clusters, PCA/t-SNE plots.

Dashboards:

Real-time traffic, predictions, filters using Tableau/Power BI.

## 10. Tools and Technologies Used

1. Machine Learning & Deep Learning



ML: Random Forest, SVM, XGBoost

DL: CNNs (for images), LSTM/RNN (for time series)

## 2. Data Processing & Analysis

Libraries: Pandas, NumPy, Scikit-learn

EDA & Visualization: Matplotlib, Seaborn, Plotly

## 3. Geospatial Analysis

Tools: QGIS, Folium, GeoPandas

Techniques: Heatmaps, GPS clustering

## 4. Time-Series Forecasting

Techniques: ARIMA, Prophet, LSTM

Use: Predict future traffic volume, speed

## 5. Big Data & Real-time Processing

Tools: Apache Spark, Kafka, Hadoop

## 6. Deployment & Dashboarding

Tools: Flask, Dash, Tableau, Power BI

## **Team members:**

**Cathrine rejina mary.J-***data preprocessing,model training*

**Aswini.K-** deployment

**Bakkiyalakshmi.P-**EDA

**Gowri.J-**report preparation

