Phase-2

Enhancing road safety with ai driven traffic accident analysis and prediction.

NAME: V Malini

REGISTER NUMBER: 422623104048

INSTITUTION: University college of engineering panruti

DEPARTMENT: computer science and engineering

DATE OF SUBMISSION:10.05.2025

Github repository link:[https://github.com/Malini-28/ENHANCING-ROAD-SAFETY-WITH-AI-DRIVEN.git]

PROBLEM STATEMENT

Traffic accidents remain a leading cause of death and injury worldwide, with significant economic and social consequences. Despite advancements in vehicle safety technologies and road infrastructure, the occurrence of traffic accidents continues to rise in many regions. Traditional methods for addressing road safety primarily focus on reactive measures, such as responding to accidents after they happen, and implementing blanket safety policies that may not be optimally targeted.There is a need for proactive and predictive systems that can anticipate traffic accidents before they occur. By leveraging artificial intelligence (AI) and data analytics, we can create more effective models to analyze historical traffic accident data, predict accident risks.

➢ OBJECTIVE OF THE PROJECT

- The Objective project's goal is to be below the following below,
- This initiative aims to improve road safety by forecasting traffic accidents through AI-driven analysis and prediction.
- Using both historical and current data, forecast the frequency and severity of accidents.
- To enable focused actions, identify high-risk locations and accident times.
- Provide useful insights to traffic management to enhance road safety protocols.
- Give decision-makers the resources they need to allocate funds wisely and create safer road systems.

➢ ENHANCING ROAD SAFETY WITH AI-DRIVEN TRAFFIC ACCIDENT ANALYSIS AND PREDICTION"

➢ DATA DESCRIPTION

❖ **NAME AND ORIGIN OF THE DATASET**:

❖ Traffic Accident Data for Road Safety Improvement is the name of the dataset.

❖ **ORIGIN:** The dataset may have come from a number of open sources, such as:

❖ **KAGGLE**: datasets such as "UK Road Safety Data" or "US Traffic Accidents."

❖ Databases like the "Road Safety Data" and other accident-related datasets are available in the UCI Machine Learning Repository.

❖ Real-time information from governmental or local traffic monitoring systems (such as Open Data Portals and City Traffic Monitoring Systems) is available through Open traffic ai.

❖ **LOCAL TRAFFIC AGENCIES**: Transportation departments or local or national traffic agencies that offer accident statistics.

➢ FLOW CHART OF THE PROJECT

→ Data Collection (e.g., traffic cameras, sensors, accident reports)

→ Data Preprocessing (cleaning, normalization, feature extraction)

→ Exploratory Data Analysis (visualizations, trend identification)

→ Model Selection (e.g., ML models like Random Forest, LSTM

→ Model Training & Testing (split data, train, validate, test)

→ Accident Prediction (based on real-time or historical input)

→ Result Visualization & Alerts (dashboard, alert systems)

→ Feedback & Model Improvement (based on new data or performance)

➢ Flow chart

➢ DATA PREPROCESSING

- ❖ Preprocessing ensures the dataset is in a usable state, enhancing the accuracy and reliability of predictive models.

- ❖ Data preprocessing is the process of cleaning and preparing raw data for analysis or machine learning. In the context of traffic accident prediction, this step includes:

- **HANDLING MISSING DATA**: Filling in or removing missing values to ensure the dataset is complete.

- **DATA CLEANING**: Correcting errors and inconsistencies in the data, such as fixing formatting issues or removing duplicates.

- **FEATURE ENCODING**: Converting categorical data (like weather conditions or road types) into numerical values using methods like one-hot encoding.

- **SCALING DATA**: Standardizing numerical features (such as traffic volume or speed) to a similar range for better model performance.

- **DATA SPLITTING**: Dividing the dataset into training and test sets to train and evaluate machine learning models effectively.

EXPLORATORY DATA ANALYSIS (EDA)

- Exploratory Data Analysis (EDA) is an essential step in understanding the structure and characteristics of the dataset. It involves both statistical and visual exploration to identify patterns, trends, and relationships in the data. Below are key steps in the EDA process:

- **UNIVARIATE ANALYSIS:**

- **Distribution of Features**:

- **Histograms** and **Boxplots** are used to analyze the distribution of numerical features (e.g., traffic volume, accident severity). This helps identify skewness, outliers, and the spread of data.

- **BIVARIATE/MULTIVARIATE ANALYSIS:**

- **Correlation Matrix**: A heatmap of the correlation matrix is used to identify relationships between numerical features (e.g., traffic volume and accident severity). Strong correlations can inform feature selection for the model.

- **Pairplots** and **Scatterplots**: Visualize relationships between multiple features and how they relate to the target variable. For example, plotting **accident severity** against **weather conditions** or **traffic volume** helps identify patterns.

- **INSIGHT SUMMARY**

- **Patterns and Trends**: EDA helps uncover patterns, such as whether accidents are more likely to occur during specific weather conditions or at certain times of the day.
- **Feature Influence**: Features like **traffic volume**, **weather conditions**, and **time of day** may have a significant impact on accident severity. For example, heavy traffic combined with rain may increase the likelihood of severe accidents.

➢ FEATURE ENGINEERING

- **Extract time-based features:** Break down timestamp into hour, day, month, weekday, and rush hour indicator to capture temporal patterns.
- **Location encoding:** Convert coordinates into area types (urban/rural), proximity to intersections, or use clustering for zone classification.
- **Weather conditions:** Transform weather data into binary flags (e.g., rainy, foggy, icy) to identify hazardous conditions.
- **Road characteristics:** Include road type, number of lanes, presence of traffic lights/signs, and speed limits.
- **Accident severity mapping:** Convert textual severity labels to numeric scales or categorical bins (e.g., low, medium, high).
- **Vehicle and driver behavior features:** Count vehicle types involved, speeding, alcohol influence indicators, or driver age groups.
- **Interaction Features**: Create new features by combining others, e.g., "Weather × Time" or "Speed × Road Type".
- **Binning:** Group continuous variables like driver age or visibility distance into bins (e.g., age 18–25, 26–40, etc.).
- **Dimensionality Reduction (optional)**: Use PCA to simplify complex data (especially useful for sensor/image data).

➢ MODEL BUILDING

- **Model selection:** Choose models like Random Forest (handles non-linearity and feature importance) and Logistic Regression (interpretable, baseline model for classification).
-
- **Problem type consideration:** Use classification models if predicting accident severity or likelihood; regression models if predicting time to clearance or injury cost.
- **Data splitting:** Divide data into training and test sets (e.g., 80/20) with stratification to maintain class balance in classification problems.
- **Model training:** Train each selected model on the training set using cross-validation for robust performance estimates.
- **Evaluation metrics:**
- For classification (e.g., accident severity): Use accuracy, precision, recall, F1-score.
- For regression (e.g., accident duration): Use MAE, RMSE, and R^2 score.

➢ TOOLS AND TECHNOLOGIES USED

Plot the confusion matrix to show correct vs incorrect predictions.

- Use the ROC curve to evaluate classification model performance.
- Show feature importance to identify key factors causing accidents.
- Use bar charts to compare different model accuracies and F1-scores.
- Display residual plots to assess prediction errors in regression models.
- Highlight and explain top contributing features to the model's decisions.
-
- Python – Main language for data analysis and modeling.
- Pandas & NumPy – Data cleaning and preprocessing.
- Git & GitHub – Version control and collaboration.

TEAM MEMBERS AND ROLES:

1. **K Mathesh kumar :** Project lead; overseas the entire project, co-ordinates tasks and ensure deadline are mets.

2. **G Harini :**Handles data collection, cleaning and exploratory data analysis.

3. **V Malini** : Focuses on building and training the AI models for accident prediction.

4. **T Arun Kumar** : Manages data storage, preprocessing, and feature engineering.

5. **F Flora** : Works on deploying the AI model into a web application and ensures it runs smoothly.