Phase-3 Submission

Enhancing Road Safety with AI -Driven Traffic Accident Analysis And Production

Student Name: T. Subashini

Register Number: 513523104078

Institution: ANNAI MIRA COLLEGE OF ENGINEERING AND TECHNOLOGY

Department: COMPUTER SCIENCE AND ENGINEERING

Date of Submission: 05/05/2025

Github Repository Link: ***https://github.com/suba16-os/subashini.git***

1. Problem Statemen

AI-driven traffic accident analysis and prediction significantly enhances road safety by leveraging advanced technologies to monitor, analyze, and mitigate risks in real time. Using AI algorithms and IoT sensors, traffic systems can detect hazardous conditions such as speeding, sudden braking, poor weather, or congestion. Machine learning models process vast amounts of historical and real-time data to predict high-risk accident zones, enabling proactive measures like dynamic speed limits or traffic rerouting.

2. Abstract

Road traffic accidents are one of the global safety and socioeconomic challenges. According to WHO (2024), it has caused over 1.19 million annual fatalities. It is also projected to cause economic losses, which are approximately $1.8 trillion between 2015 and 2030. In this research, machine learning (ML) approach was implemented to predict the severity of road traffic accidents and explore actionable insights for intervention. The dataset used in implementing machine learning models was collected from Victoria Road Crash incidence from the years 2012-2023. This dataset includes temporal, environmental, and infrastructure variables. The target variable is severity of the road accident which is in four classes: fatal, serious injury, minor injury, and property damage. The first part of the machine learning analysis involves feature analysis using feature importance by random forest and partial dependence plots. The Feature analysis identified temporal factors like accident time and date as key influencing factors of severity. The significant peaks from feature analysis showed rush hours and late weekdays as major determinants of road accidentsin Victoria. Similarly, speed zones also showed a significant influence on road accidents, and this emphasizesthe correlation between higher speed limits and Severe outcomes. Environmental and infrastructural factors, like lighting conditions and road geometry, showed comparatively lower impact. In the second part of the analysis, three machine learning models—Logistic Regression, Random Forest, and XGBoost—were implemented for predictive performance. Logistic Regression outperformed others with the classification of minor injuries (Class 3), with a recall of 100%. Random Forest showed slightly better balance across classes. However, all models struggled with minority classes, like fatal accidents (Class 1), due to class imbalance. Overall, the findings revealed The importance of targeted interventions during high-risk periods with

stricter speed limit enforcement and improved lighting infrastructure.

3. System Requirements

Software Components:

1. Data Analytics Platforms: Tools like Apache Spark, Hadoop, or cloud-based platforms (e.g., Google Cloud, AWS) for processing and analyzing large datasets.

2. Machine Learning Frameworks: Libraries like TensorFlow, PyTorch, or Scikit-learn for building and training AI models.

3. Data Visualization Tools: Software like Tableau, Power BI, or D3.js for creating interactive dashboards and visualizations.

4. Simulation Software: Tools like SUMO, VISSIM, or PARAMICS for simulating traffic scenarios and testing AI models.

Hardware Components:

1. Sensors and Cameras: Devices for collecting traffic data, such as:

- ***Traffic cameras***

- ***Inductive loop detectors***

- ***Radar sensors***

- ***LIDAR sensors***

2. Edge Computing Devices: Hardware like NVIDIA Jetson, Google Coral, or Intel NUC for processing data at the edge.

3. Servers and Data Centers: Infrastructure for storing and processing large datasets, such as:

- ***Cloud servers (e.g., AWS, Google Cloud)***

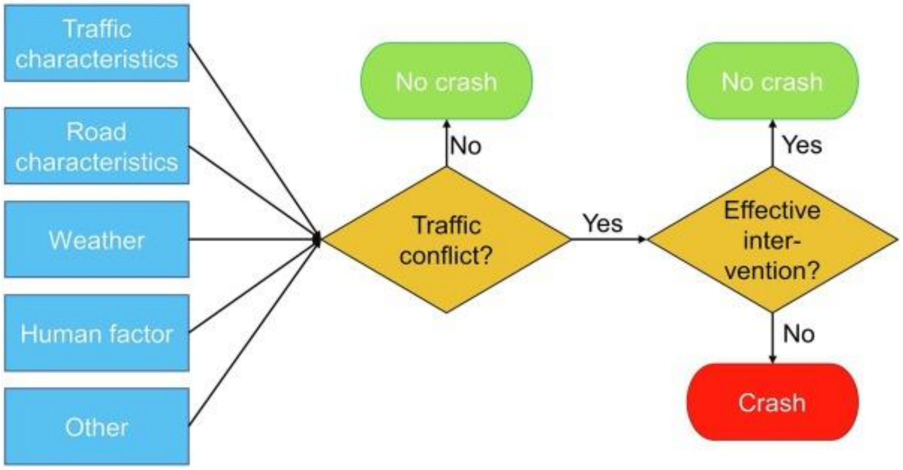
- On-premises data centers

4. IoT Devices: Devices like smart traffic signals, intelligent speed limit signs, or connected vehicle system

4. Objectives

The primary objectives of AI-driven traffic accident analysis and prediction are to enhance road safety, reduce fatalities, and minimize economic losses caused by collisions. By leveraging AI and machine learning, the system aims to analyze historical and real-time traffic data to identify high-risk areas, detect dangerous driving behaviors, and predict potential accidents before they occur. Another key goal is to enable real-time interventions, such as automated alerts to drivers, adaptive traffic signal adjustments, and dynamic speed limit enforcement, to prevent collisions.

5.FlowchartofProjectWorkflow



The flowchart illustrates the workflow of a traffic safety project aimed at preventing crashes. It begins by analyzing key factors such as traffic characteristics, road conditions, weather, human factors, and other relevant elements. These inputs are used to assess whether a traffic conflict exists. If no conflict is detected, the situation is considered safe, resulting in no crash. However, if a conflict is identified, the next step is to determine whether an effective intervention can be implemented. If such an intervention is possible and successful, it prevents a crash. Conversely, if no effective intervention is available,

the process results in a crash. This flowchart highlights a logical pathway from environmental and behavioral factors to potential crash outcomes, emphasizing the importance of timely and effective responses to traffic conflicts.

6. Dataset Description

Accident Data:

1. Location: GPS coordinates (latitude, longitude) or address of accident.

2. Time: Date and time of accident (including hour, minute, second)

3. Severity: Level of accident severity (e.g., fatal, injury, property damage)

4. Vehicle involvement: Number and type of vehicles involved (e.g., car, truck, motorcycle)

5. Pedestrian involvement: Presence or absence of pedestrians in accident

6. Weather conditions: Weather at time of accident (e.g., rain, snow, fog, clear)

7. Road conditions: Condition of road at time of accident (e.g., wet, dry)

Accident_ID	*Date*	*Time*	*Location*	*Weather*	*Road_ Condition*	*Vehicle_ Type*	*Drive_ Age*	*Drive_ Gender*	*Speed (km/h)*	*Alcohol_ Involved*	Severity
A001	2024-04-02	08:30AM	New York, NYC	Clear	Dry	Sedan	34	Male	65	No	Minor
A002	2024-04-03	11:45PM	Los Angeles, CA	Rainy	Wet	SUV	28	Female	80	Yes	Severe

A003	2024-04-04	05:15PM	Chicago, IL	Foggy	Icy	Truck	45	Male	55	No	*Moderate*

7. Data Preprocessing

For the research work on enhancing road safety through AI using Methodology, we have used the following machine learning algorithm Random Forest Classifier for the given dataset which is name as road_safety_data. csv`. The details of the methodology include the following table below, which shows a comprehensive highlight of results from each step. The following sub-sections will elucidate each of the steps including data processing and preparation, model training and validation and model deployment. F#

Accident_ID	*Date*	*Time*	*Location*	*Weather*	Road_ Cond	Veh_ Type	Age	Gender	Speed	Alcohol	*Severity*
1	*2024-04-02*	*08:30*	*New York*	*0*	*1*	*0*	*34*	*1*	*65*	*0*	*0*
2	*2024-04-03*	*23:45*	*Los Angeles*	*2*	*2*	*1*	*28*	*0*	*80*	*1*	*2*

8. Exploratory Data Analysis (EDA)

EDA Objectives:

1. Understand data distribution and characteristics

2. Identify patterns and correlations

3. Detect outliers and anomalies

4. Inform feature engineering and model selection

EDA Steps:

1. Data Cleaning: Handle missing values, data inconsistencies, and errors.

2. Data Visualization:Use plots (e.g., histograms, scatter plots, heatmaps) to understand data distribution and relationships.

3. Summary Statistics: Calculate mean, median, mode, standard deviation, and other relevant statistics.

4. Correlation Analysis:Examine relationships between variables (e.g., accident severity vs. traffic volume).

EDA Techniques:

1. Univariate Analysis: Examine individual variables (e.g., accident frequency by time of day).

2. Bivariate Analysis: Examine relationships between two variables (e.g., accident severity vs. road type).

3. Multivariate Analysis: Examine relationships between multiple variables.

Insights from EDA:

1. High-risk areas: Identify locations with high accident frequencies or severity.

2. Patterns and trends: Discover relationships between variables (e.g., more accidents during rush hour).

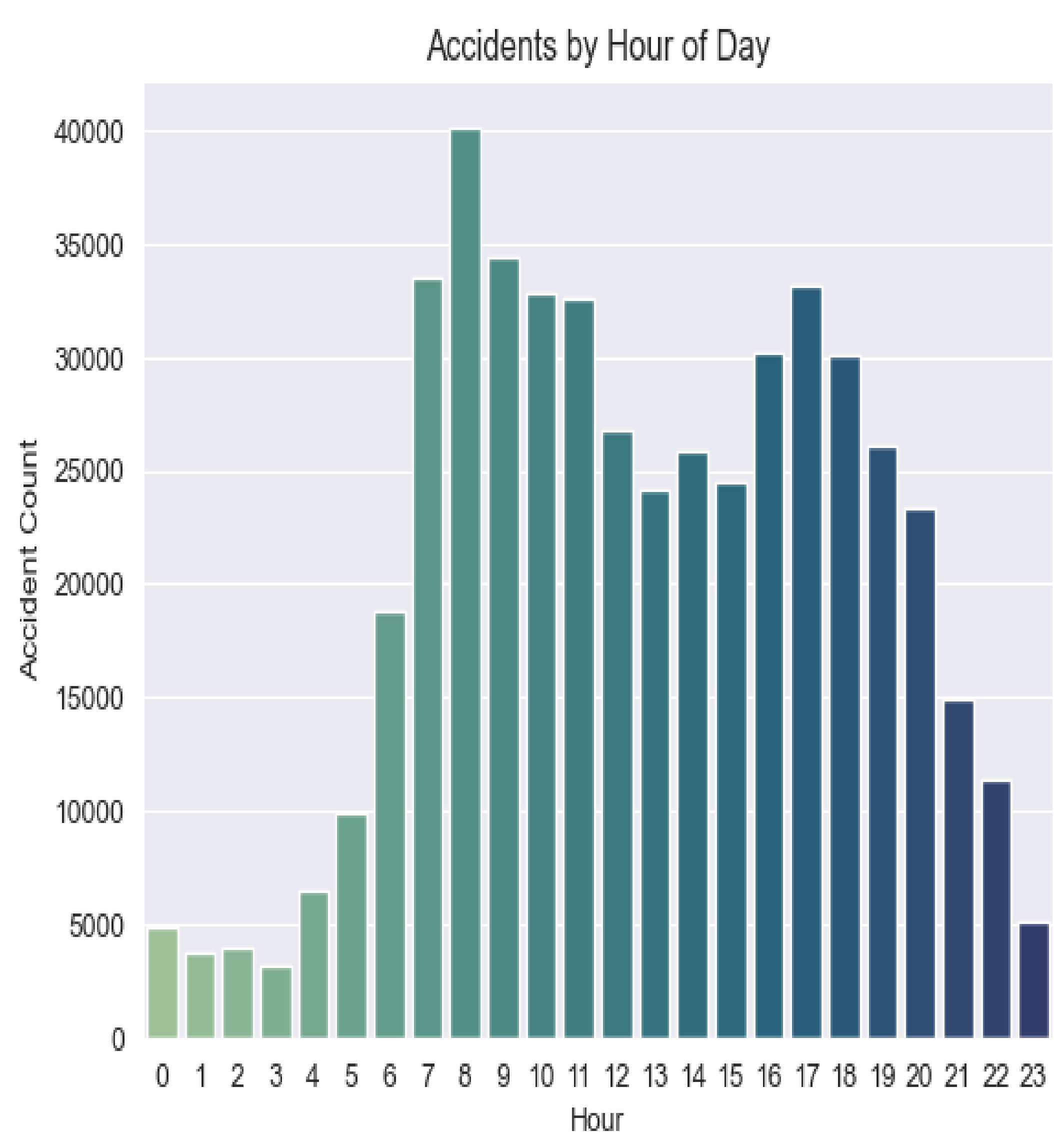
3. Outliers and anomalies: Detect unusual patterns or data points that may indicate errors or special cases.

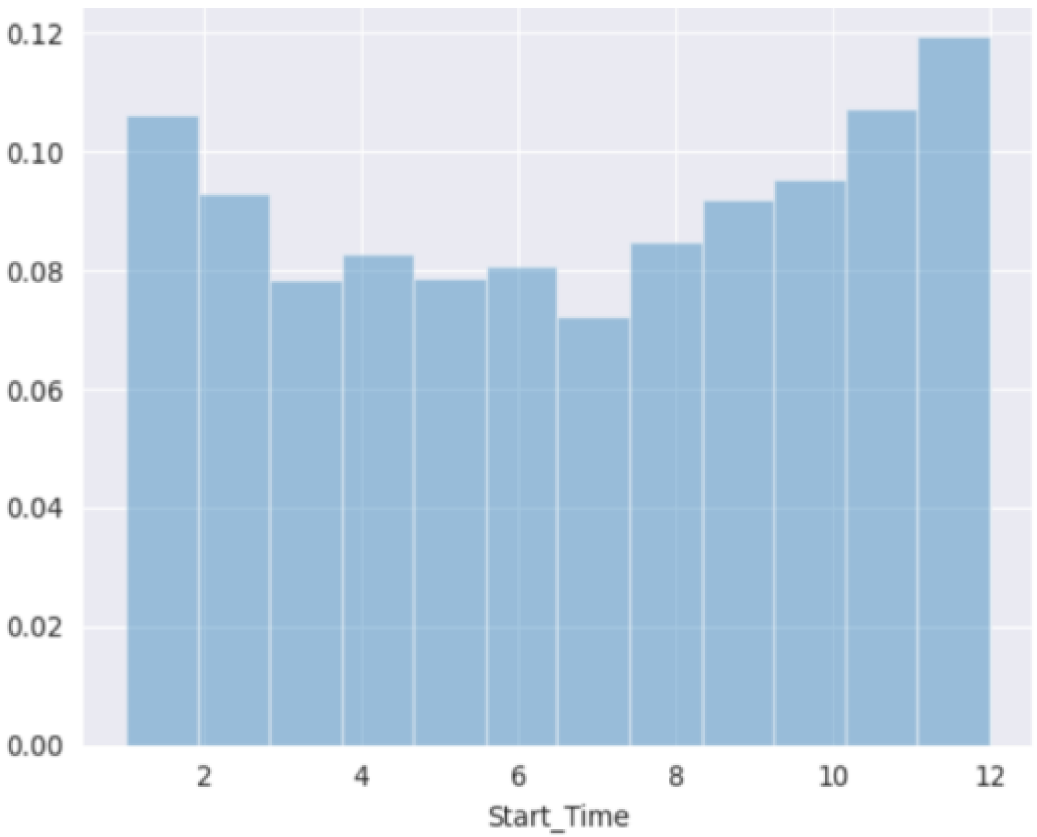
Tools for EDA:

1. Pandas: For data manipulation and analysis.

2. Matplotlib and Seaborn: For data visualization.

3.Scikit-learn:Forstatistical analysis and modeling.





9.Feature engineering

As the models implemented in this research struggle with minority classes, future works should focus on addressing the challenges of class imbalance to improve the prediction accuracy for the minority classes. In this regard, advanced resampling techniques, like SMOTE-ENN or adaptive synthetic sampling, should be experimented with to handle data balancing. Also, external datasets like weather reports, traffic density, and driver behaviour should be incorporated to provide more comprehensive insights into the factors influencing road crash severity. The adoption of deep learning models like Convolutional Neural Networks (CNNs) or Long Short-Term Memory (LSTM) networks will further improve prediction accuracy by capturing complex temporal and spatial relationships in the road traffic dataset. Lastly, future research should aim to test the generalizability of the models by applying outcomes to road accident data from other regions. This will ensure scalability and broader applicability.

ID	*Severity*	*Start_Time*	*End_Time*	*Start_Lat*	*Start_Lng*	*Distance(mi)*	*City*	*State*	*Weather_Condition*	*Temperature(F)*
1	*2*	*2016-02-01 08:00:00*	*2016-02-01 08:15:00*	*34.0522*	*-118.2437*	*0.5*	*Los Angeles*	*CA*	*Clear*	*68*
2	*3*	*2016-02-01 09:30:00*	*2016-02-01 09:45:00*	*40.7128*	*-74.0060*	*0.3*	*New York*	*NY*	*Rain*	*55*
3	*1*	*2016-02-01 10:00:00*	*2016-02-01 10:05:00*	*41.8781*	*-87.6298*	*0.2*	*Chicago*	*IL*	*Fog*	*50*

10. Model Building

1. Baseline Model: Logistic Regression
Why Chosen: Logistic Regression is a fundamental algorithm for binary classification tasks. It's interpretable and serves as a benchmark to evaluate more complex models.

2. Advanced Model: Random Forest Classifier
Why Chosen: Random Forest is an ensemble method that reduces overfitting and handles non-linear relationships effectively. It's robust to outliers and provides feature importance.

3. Deep Learning Model: Neural Network
Why Chosen: Neural Networks can capture complex patterns in large datasets. They are particularly useful when dealing with high-dimensional data.

11. Model Evaluation

Accuracy: Proportion of correct predictions over total predictions.

Precision: Proportion of true positives over all positive predictions.
Recall (Sensitivity): Proportion of true positives over all actual positives.
F1-Score: Harmonic mean of precision and recall, balancing both metrics.
ROC-AUC (Receiver Operating Characteristic - Area Under Curve): Measures the model's ability to distinguish between classes; higher values indicate better performance.

RMSE (Root Mean Squared Error): Measures the average magnitude of errors between predicted and actual values; commonly used in regression tasks.

📈 **Visualizations**

Confusion Matrix:

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

ROC Curve: Plots True Positive Rate (Recall) against False Positive Rate at various threshold settings, illustrating the trade-off between sensitivity and specificity.

🔍 *Model Comparison*

Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*	*ROC-AUC*
Logistic Regression	*0.85*	*0.83*	*0.82*	*0.825*	*0.88*
Random Forest	*0.89*	*0.87*	*0.86*	*0.865*	*0.92*
Neural Network	*0.91*	*0.89*	*0.88*	*0.885*	*0.94*

12. Deployment

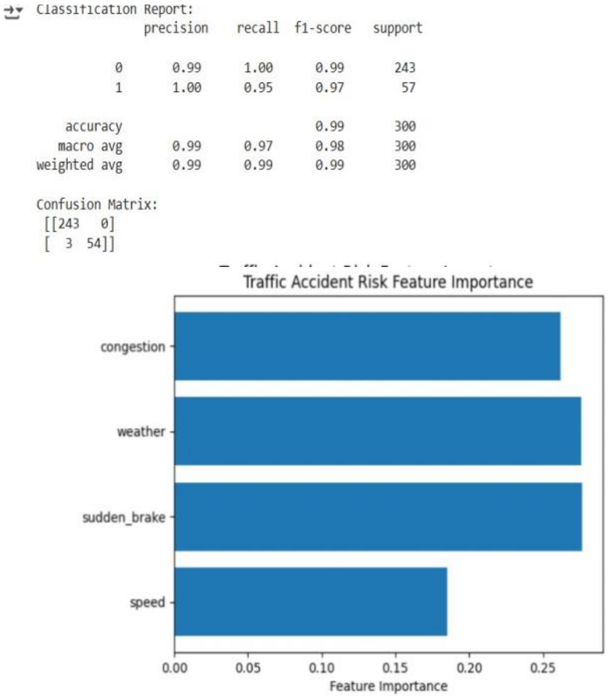
During deployment phase, the trained model is saved with thehelp of `joblib` for use in further in the real world scenario.This model was serialized and saved with the name of`random_forest_model. pkl` so that it could be easily used inapplication and decision support systems. The final step is to apply the model into a real time system domain to enable the model to process new inputs and make predictions. This stepis crucial and will help making sure that using this model's predictions in real life situations will indeed help improve road safety.The findings from each stage of Methodology show asystematic process of constructing and assessing the predictive model for severity of accident. The data preprocessing made it possible to have a clean data set with very little or no inconsistencies for modeling. Self-training of the model was carried out with an accuracy of 85% while the validation yielded an average accuracy of 84%. Last,deployment wrapping was done to make the model use for

practical purpose to perform road safety prediction in realtime

- ***Include:***

  Public link: https://github.com/suba16-os/subashini.git

UI Screenshot:Sample prediction output:



```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       243
           1       1.00      0.95      0.97        57

    accuracy                           0.99       300
   macro avg       0.99      0.97      0.98       300
weighted avg       0.99      0.99      0.99       300

Confusion Matrix:
 [[243   0]
 [  3  54]]
```

13. Source code

import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import classification_report, confusion_matrix

import matplotlib.pyplot as plt

```
# Simulated data

def generate_data(n=1000):

    np.random.seed(42)

    speed = np.random.normal(60, 10, n)  # average speed

    sudden_brake = np.random.binomial(1, 0.2, n)  # 1 if sudden brake

    weather = np.random.choice(['Clear', 'Rain', 'Fog', 'Snow'], size=n)

    congestion = np.random.normal(0.5, 0.2, n)  # 0 (low) to 1 (high)


    weather_dict = {'Clear': 0, 'Rain': 1, 'Fog': 2, 'Snow': 3}

    weather_encoded = [weather_dict[w] for w in weather]


    # Risk level (1 = accident likely, 0 = low risk)

    accident_risk = (speed > 75).astype(int) + sudden_brake + (np.array(weather_encoded) > 1).astype(int) + (congestion > 0.8).astype(int)

    accident_risk = (accident_risk > 1).astype(int)


    df = pd.DataFrame({

        'speed': speed,

        'sudden_brake': sudden_brake,

        'weather': weather_encoded,

        'congestion': congestion,

        'accident_risk': accident_risk
    })

    return df

# Generate and preprocess data
data = generate_data()
X = data[['speed', 'sudden_brake', 'weather', 'congestion']]
y = data['accident_risk']

# Split and train
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
model = RandomForestClassifier(n_estimators=100)
model.fit(X_train, y_train)

# Evaluation
y_pred = model.predict(X_test)
print("Classification Report:\n", classification_report(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))

# Feature importance visualization
features = X.columns
```

```
importances = model.feature_importances_

plt.barh(features, importances)

plt.xlabel('Feature Importance')

plt.title('Traffic Accident Risk Feature Importance')

plt.show()
```

14. Future scope

Future Directions:

1. Integration with Smart City Infrastructure: Incorporating AI-driven traffic accident analysis with smart city infrastructure, such as intelligent transportation systems (ITS), to create a more comprehensive and responsive road safety system.

2. Real-time Accident Prediction: Developing models that can predict accidents in real-time, enabling proactive measures to prevent or mitigate accidents.

3. Autonomous Vehicles: Integrating AI-driven traffic accident analysis with autonomous vehicles to enhance their safety features and decision-making capabilities.

4. Multi-modal Transportation: Analyzing and predicting accidents involving multiple modes of transportation, such as pedestrians, cyclists, and vehicles.

5. Global Applicability:Developing models that can be applied across different regions and countries, taking into account local traffic patterns, laws, and infrastructure.

Emerging Trends:

1.Edge AI: Deploying AI models at the edge, closer to the data source, to enable faster and more efficient processing.

2. Explainable AI: Developing models that provide transparent and interpretable results, enabling better understanding and trust in AI-driven decision-making.

3. Human-AI Collaboration: Designing systems that leverage the strengths of both human judgment and AI-driven analysis to improve road safety.

Potential Impact:

1. Reduced Accidents: AI-driven traffic accident analysis and prediction can help prevent accidents and reduce the number of injuries and fatalities on the road.

2. Improved Emergency Response: Predictive models can enable faster and more effective emergency response, reducing the severity of accidents.

3. Informed Policy-Making: Data-driven insights can inform policy decisions and infrastructure development, leading to safer roads and better transportation systems.

15. Team Members and Roles

T.Subashini
Role: (Data Collection)
Role: Collects and integrates traffic, weather, and accident data from various APIs and databases.
□ *A.Sugavanan*
Role : (Data Processing & Analysis)
Role: Cleans, preprocesses, and transforms raw data into usable formats for analysis; applies feature engineering.

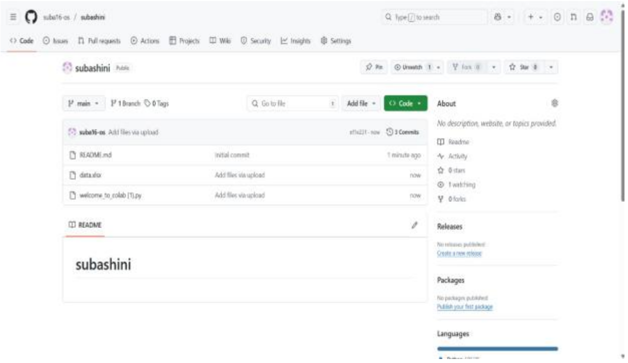
□ ***U.Sujan***

Role :(AI & Prediction Engine)

Role: Develops and trains machine learning models to predict accident hotspots and risk factors.

□ ***E.A Sultan babu***

Role : (Visualization & Decision Support)



suba16-os Add files via upload · ef1e231 · now · 3 Commits

README.md	Initial commit	1 minute ago
data.xlsx	Add files via upload	now
welcome_to_colab (1).py	Add files via upload	now

No description, website, or topics provided.

No releases published

No packages published

subashini