

An Industry Orientred project Report On

**Predictive Analysis for Accident Prevention and Response**

Submitted by

**A. NESHITHA**

**N. THANUJA**

to

COAPPS.AI DEVELOPMENT SOLUTIONS Pvt.Ltd

Under the Guidance of

**AMJOY EXSON**



**ABSTRACT :**

Traffic accidents have become severe risks as they are one of the causes of enormous deaths worldwide. Reducing the number of incidents is critical to saving lives and achieving sustainable cities and communities. Machine learning and data analysis techniques interpret the reasons for car accidents and propose solutions to minimize them. However, this needs to take the benefits of big data solutions as the size and velocity of traffic accident data are increasingly large and rapid. This paper explores road car accident data patterns and proposes a predictive model by investigating meaningful data features, such as accident severity, the number of casualties, and the number of vehicles. Therefore, a pre-processing model is designed to convert raw data using missing and meaningless feature removal, data attribute generalization, and outlier removal using interquartile. Four classification methods, including decision trees, random forest, multinomial logistic regression, and naïve Bayes, are used and evaluated to study the performance of road accident prediction. The results address acceptable levels of accuracy for car accident prediction except for naïve Bayes. The findings are discussed through a data-driven approach to understand the factors influencing road car accidents and highlight the key ones to propose accident prevention solutions. Finally, some strategies are provided to achieve healthy and community-friendly cities.

**Keywords**: machine learning; road car accident; prediction model; big data; sustainable community; data-driven approach; community-friendly

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **S.No** | **CONTENTS** | **Pg.No** |
| 1 | CHAPTER-1 :- INTRODUCTION | 4-9 |
| 2 | CHAPTER-2 :- LITERATURE REVIEW | 10-13 |
| 3 | CHAPTER-3 :- PROPOSED SYSTEM | 14-19 |
| 4 | CHAPTER-4 :- SYSTEM DESIGN AND IMPLEMENTATION | 20-33 |
| 5 | CHAPTER-5 :- RESULT AND DISCUSSION | 34-35 |
| 6 | CHAPTER-6 :- CONCLUSION | 36-39 |
| 7 | CHAPTER-7 :- REFERENCES | 40-43 |

**CHAPTER-1**

**INTRODUCTION**

Traffic accidents are unavoidable and can occur anytime and anywhere. The World Health Organization (WHO) reports that approximately 1.35 million lives are lost to traffic accidents each year . Traffic accidents cost about 3% of most counties’ gross domestic product (GDP). To be precise, 1500 to 3500 people were killed each year due to road accidents in the UK in the years between 2000 to 2013 . However, it has been adequately managed recently by using road safety measures that have been implemented due to a better understanding of the determining causes. Road accidents are influenced by external factors, such as weather conditions, road status, and driver skills. According to , more severe injuries were sustained in darker conditions (at night without streetlights) and dry road surface conditions.

The month, season, and weather conditions were also significant factors in the analysis conducted by Wang et al. in China, showing that autumn and winter were likely to cause severe fatalities brought on by difficult weather such as rain and snow. However, Potoglou et al. show that summer and autumn seasons are more likely to result in traffic accidents from data from Italy and Europe, which are more relevant to this research because it uses data from the UK. Traffic accident data patterns can be explored and studied using machine-learning enabled big data analysis techniques. Road accident datasets are variant, unstructured, large, and rapidly changing. Because of this, traditional data processing and storage techniques offer no benefits, especially where the data processing platform is restricted (i.e., a single data server).

Hence, road accident analysis applications (e.g., prediction) need to benefit from robust and scalable data processing frameworks to analyze massive and/or streaming datasets according to distributed computing fashion. Apache Spark is an open-sourced framework that provides a unified data analysis framework to manage big data tasks. It can perform workloads at up to 100 × faster speeds than Hadoop using a directed acyclic graph (DAG) scheduler, supporting operation optimization . Yet, it supports cluster-based computing through which several data servers are linked to run data analysis codes. Spark provides a machine learning library named MLlib, which includes several machine learning algorithms for classification, regression, decision trees, and so forth . This project focuses on predictive data analysis for road car accidents. The prime goal is to propose a predictive machine learning model with the capacity of analyzing and determining accident severity, casualty count, and the number of vehicles involved. The authors performed a validated and online large dataset to train and test the machine learning models. The dataset includes traffic accident data during 2005–2014, except in 2008 in the UK.

Four machine learning techniques, including decision trees, random forest classification, multinomial logistic regression, and naïve Bayes classifications, were set up and evaluated to find the best fit predictive model for road accidents. To do so, each dependent variable was predicted by using several parameters that were chosen because of their high correlation. They are outlined as weather conditions, light conditions, road type, road surface conditions, season, time of day (daytime/nighttime), date of the week (weekday/weekend), area (i.e., rural or urban), and region (including latitude and longitude). The machine learning techniques were fed the training dataset to propose the prediction model. The prediction accuracy of each machine learning technique was measured and compared to highlight the best fit one with maximized accuracy.

The key contributions of this research are outlined below:

• To study road car accident data patterns and figure out the most influential/meaningful predictive and prediction features in accident data analysis.

• To pre-process and prepare a public dataset in order to meet the requirements of a comparative machine-learning-enabled traffic accident data analysis.

• To build, train, and evaluate four machine learning methods to find the best fit solution based on three different targets.

* 1. **PURPOSE**

Predictive analysis plays a vital role in accident prevention and response by leveraging historical data, real-time information, and advanced analytical techniques to anticipate and mitigate potential risks. Here's how predictive analysis serves the purpose in accident prevention and response:

**Identifying High-Risk Areas:**

Predictive analysis can identify geographic locations, road segments, or industrial sites with a higher likelihood of accidents based on historical data. By pinpointing these high-risk areas, preventive measures such as enhanced surveillance, improved infrastructure, or targeted safety campaigns can be implemented to reduce the probability of accidents.

**Predicting Accident Hotspots:**

Analyzing patterns in historical accident data using predictive models can help forecast future accident hotspots. By predicting when and where accidents are likely to occur, authorities can allocate resources more effectively, such as deploying emergency services, adjusting traffic flow, or implementing temporary safety measures during peak risk periods.

**Early Warning Systems:**

Predictive analytics can power early warning systems that detect anomalies or trends indicative of potential accidents. For example, predictive models can analyze real-time data from sensors, weather forecasts, traffic cameras, and social media to identify conditions associated with increased accident risk, such as adverse weather, heavy traffic congestion, or hazardous road conditions. Early detection allows authorities to issue timely warnings to drivers, implement traffic management strategies, or initiate evacuation procedures to prevent accidents or minimize their impact.

**Optimizing Resource Allocation:**

Predictive analysis enables the optimization of resource allocation for accident response and emergency services. By forecasting the likelihood and severity of accidents in different areas, emergency response teams can strategically position personnel, equipment, and supplies to minimize response times and maximize effectiveness. This proactive approach helps save lives, reduce injuries, and minimize property damage in the event of accidents or disasters.

**Enhancing Safety Measures:**

Predictive analytics can inform the development and implementation of targeted safety measures and interventions to mitigate accident risks. By identifying the root causes and contributing factors of accidents, predictive models can recommend specific safety improvements, such as traffic signal optimization, road design modifications, vehicle safety enhancements, or driver behavior interventions. These evidence-based interventions help address underlying safety issues and prevent accidents from occurring in the first place.

**Continuous Improvement:**

Predictive analysis facilitates continuous improvement in accident prevention and response efforts by evaluating the effectiveness of existing safety measures and interventions. By analyzing the outcomes of past interventions and monitoring changes in accident patterns over time, authorities can refine their strategies, adjust resource allocation priorities, and implement new initiatives to stay ahead of emerging risks and evolving threats.

Overall, predictive analysis empowers stakeholders in accident prevention and response to make data-driven decisions, anticipate future events, and take proactive measures to protect public safety, minimize risks, and mitigate the impact of accidents on individuals, communities, and infrastructure.

1.2 **SCOPE**

Predictive analysis has immense potential in accident prevention and response. By analyzing historical data and real-time information, predictive models can identify patterns, trends, and risk factors that contribute to accidents.

By analyzing historical accident data, weather conditions, traffic patterns, and other relevant factors, predictive models can identify high-risk areas and predict the likelihood of accidents occurring. This information can help authorities take proactive measures to prevent accidents.Predictive models can analyze driver behavior data, such as speeding, sudden braking, or distracted driving, to identify patterns that increase the risk of accidents. This information can be used to target interventions, such as driver education programs or enforcement initiatives.

Predictive analysis can assess the condition of roads, bridges, and other infrastructure elements to identify potential hazards or areas in need of repair. This proactive approach can help prevent accidents caused by infrastructure failures.Predictive models can analyze historical accident data, traffic patterns, and other real-time information to optimize emergency response efforts. This includes identifying the most efficient routes for emergency vehicles, predicting the demand for emergency services in specific areas, and allocating resources accordingly.Predictive analysis can be used in intelligent vehicle systems to detect and predict potential accidents. For example, using sensor data, predictive models can identify situations where collision risk is high and trigger warning systems or autonomous emergency braking systems.

By analyzing vehicle sensor data and maintenance records, predictive models can identify potential mechanical failures or malfunctions before they occur. This can help prevent accidents caused by vehicle breakdowns or faulty components.Overall, predictive analysis holds great promise in accident prevention and response by leveraging data to identify risks, predict outcomes, and optimize interventions. It has the potential to save lives, improve road safety, and enhance emergency response efforts. The text or speech output is determined by the user's selections on the virtual keyboard.

* 1. **NEED FOR SYSTEM**

The need for a predictive analysis system in accident prevention and response arises from the desire to enhance road safety, save lives, and improve emergency response efforts. Traditional methods of accident prevention and response often rely on reactive measures, such as investigating accidents after they occur or responding to emergencies as they happen. However, by implementing a predictive analysis system, we can shift towards a more proactive approach.

This system can help identify potential risks and hazards before accidents happen, allowing authorities to take preventive measures and allocate resources more effectively. By analyzing historical data, real-time information, and various factors like weather conditions, traffic patterns, and driver behavior, we can gain valuable insights into accident-prone areas and high-risk situations. This knowledge can be used to implement targeted interventions, such as improving road infrastructure, implementing traffic control measures, or conducting driver education programs.

Furthermore, predictive analysis can optimize emergency response efforts by identifying the most efficient routes for emergency vehicles, predicting demand for emergency services in specific areas, and allocating resources accordingly. This can result in faster response times and more effective emergency assistance.

Overall, the need for a predictive analysis system in accident prevention and response is driven by the goal of improving road safety, minimizing accidents, and enhancing emergency response capabilities. By leveraging data and advanced analytics, we can proactively address potential risks and take preventive measures to create safer roadways and ensure a more efficient emergency response system**.**

**CHAPTER-2**

**LITERATURE REVIEW**

**2.1 AUTHOR:**

Machine learning enabled by big data analytic techniques can propose a predictive road accident model to increase road safety and conserve resources. For this, some solutions have been designed to manage road accidents.

**Nicky Kuttukkaran et al**.: Proposed using **heartbeat to detect accidents**, but Bluetooth range limitations and subjective interpretation of heartbeat make it less reliable.

**Sanjay Kumar Singh**: Found that 50% of Indian cities are prone to road accidents, **estimating** 250,000 accidents by 2025.

**Chaitali Khandbahale et al**.: Used **embedded detection** for accident prevention, but high economic cost and sensor malfunction in rainy weather are limitations.

**Nedjet Dogru et al**.: Used **clustering algorithm** for grouping vehicles based on speed and location, but accuracy during deployment was not optimal.

**Vipul Rana et al.**: Proposed **accident prevention** by asking users to submit a form, but it's limited to cases where the victim can fill out the form.

**Deeksha Gour et al.:** Used optimized **YOLO for accident detection**, but misclassification of target objects can reduce system efficiency.

**Xi Jianfeng et al.**: Adopted **support vector machine (SVM)** for accident detection, but faced challenges in localizing objects and instance segmentation.

**Raad Ahmed Hadi:** Studied **vehicle detection** using background subtraction and feature-based tracking.

**Zhong-Qiu Zhao et al.:** Proposed generic object detection using **masking and recurrent neural networks,** improving detection pipeline and validation.

**Kaiming He et al.**: Extended fast **R-CNN** for instance segmentation, potentially useful for custom object detection like car accidents.

**Sodikov et al.**: Studied accidents in Uzbekistan, providing insights on **accident frequency and severity** distribution over time.

**S. Ramya et al.**: Found **Random Forest Classifier** to outperform other methods in predicting accident severity due to its ability to handle seasonality.

**Jabar H. Yousif et al.**: Presented an **optimal neural network** approach for predicting traffic accident severity, considering weather conditions.

Thus, this project aims to test and evaluate decision trees, random forest, multinomial logistic regression, and naïve Bayes classification models to predict road accidents. It was shown that accident severity, the number of vehicles, and the number of casualties are dependent variables that are predicted using the independent attributes mentioned above.

According to the literature, there is still a research gap to building a comparative machine learning approach to find the best predictive method for road car traffic accident analysis. By this it is meant that road accident data patterns should be studied and processed to figure out the most meaningful predictive and prediction features, find their correlations and dependencies to model (or forecast) road accident behavior.

This research aims to pre-process a big online dataset and highlight the most correlated predictive features according to three predictions—mainly accident severity, the number of casualties, and the number of vehicles. Yet, four well-known machine learning models, including decision trees, random forest classification, multinomial logistic regression, and naïve Bayes classifications, are by the dataset and compared to find the best-fit method to forecast car road accidents.

2.2 **EXISTING SYSTEM**

The existing system related to Predictive analysis for Accident Prevention and Response are:

**Intelligent Transportation Systems (ITS):** ITS encompasses various technologies and systems aimed at improving road safety, traffic management, and transportation efficiency. Within ITS, specific systems related to road accidents include:

* **Traffic Surveillance Systems:** These systems use cameras, sensors, and other monitoring devices to detect traffic incidents, including accidents, congestion, and road hazards.
* **Dynamic Message Signs (DMS):** DMS provide real-time information to drivers about accidents, road closures, and alternative routes, helping to manage traffic flow and reduce congestion.
* **Incident Management Systems (IMS):** IMS facilitate the coordination of emergency response and recovery efforts following road accidents. They include tools for incident detection, response coordination, and traffic diversion to minimize disruptions and improve safety.
* **Collision Avoidance Systems:** These systems use sensors, radar, and vehicle-to-vehicle communication to detect and prevent collisions on the road, thereby reducing the likelihood of accidents.

**Crash Data Analysis Systems:** Various systems analyze crash data to identify trends, patterns, and risk factors associated with road accidents. These systems help transportation agencies and safety organizations prioritize safety improvements, target interventions, and evaluate the effectiveness of countermeasures. Examples include:

* **Crash Data Repositories**: These centralized databases store detailed information about road accidents, including crash reports, vehicle data, and injury statistics, which are used for analysis and research purposes.
* **Crash Prediction Models:** Predictive models use historical crash data and other factors (e.g., road geometry, traffic volume, weather conditions) to forecast the likelihood of future accidents in specific locations or along particular road segments.

**Emergency Response Systems:** These systems facilitate the rapid response of emergency services, such as police, fire, and medical personnel, to road accidents. Examples include:

* **Computer-Aided Dispatch (CAD):** CAD systems enable emergency call centers to receive, prioritize, and dispatch resources to accident scenes efficiently. They include features for call handling, resource tracking, and incident documentation.
* **Mobile Data Terminals (MDTs):** MDTs provide first responders with access to real-time information, maps, and communication tools while en route to accident locations, improving situational awareness and coordination.

**Road Safety Audit Systems:** Road safety audit systems assess the safety performance of road infrastructure and identify potential hazards and deficiencies that contribute to accidents. These systems help transportation agencies prioritize safety improvements and design roadways to better accommodate all users, including pedestrians, cyclists, and motorists.

**Telematics and Vehicle Tracking Systems:** Telematics systems installed in vehicles collect data on driving behavior, vehicle performance, and location, which can be used to identify risky driving practices and prevent accidents. Vehicle tracking systems also enable fleet managers to monitor the location and status of vehicles in real-time, facilitating efficient emergency response and recovery operations.

These existing systems play a crucial role in preventing road accidents, managing incidents when they occur, and improving overall road safety through data analysis, technology deployment, and coordinated response efforts.

**CHAPTER -3**

**PROPOSED SYSTEM**

3.1 **PROPOSED WORK**

Here's a proposed system for road accident analysis that takes date and place as input and generates the number of casualties and road surface condition as output:

**Data Collection Module:**

* Collects data on road accidents, including date, time, location (latitude and longitude), number of casualties, and road surface condition (e.g., dry, wet, icy).
* Sources of data may include police reports, emergency service records, traffic surveillance systems, and citizen reports.

**Preprocessing Module:**

* Cleans and preprocesses the collected data to handle missing values, outliers, and inconsistencies.
* Converts date and time data into a standardized format for analysis.
* Validates and formats the location data (e.g., latitude and longitude) to ensure consistency and accuracy.

**Analysis Module:**

* Analyzes the preprocessed data to identify patterns, trends, and correlations between accidents, date, location, and road surface condition.
* Uses statistical and machine learning techniques to model the relationship between input variables (date, location) and output variables (number of casualties, road surface condition).
* Develops predictive models to forecast the number of casualties and predict road surface conditions based on input parameters.

**Prediction Module:**

* Accepts user input for date and place (latitude and longitude) of interest.
* Utilizes the predictive models developed in the analysis module to generate predictions for the number of casualties and road surface condition at the specified location and date.
* Provides the predicted outcomes as output to the user.

**User Interface:**

* Provides an intuitive interface for users to input the date and location of interest.
* Displays the predicted number of casualties and road surface condition based on the user's input.
* Offers visualizations such as maps, charts, or graphs to present the predicted outcomes in a clear and understandable format.

**Feedback and Improvement Mechanism:**

* Incorporates feedback from users and stakeholders to improve the accuracy and reliability of the predictive models.
* Continuously updates the system with new data and insights to enhance its predictive capabilities and usability over time.

**Integration with External Systems:**

* Integrates with external data sources such as weather forecasts, traffic data, and road condition reports to improve the accuracy of predictions and provide additional context for decision-making.
* Enables interoperability with existing road safety management systems, emergency response platforms, and transportation infrastructure for seamless data exchange and collaboration.
* This proposed system leverages data-driven analysis and predictive modeling techniques to provide insights into road accidents, help stakeholders make informed decisions, and improve road safety measures. By predicting the number of casualties and road surface conditions, authorities can proactively implement preventive measures, allocate resources effectively, and enhance emergency response efforts to mitigate the impact of accidents on public safety.

3.1.1 **PROPOSED ARCHITECTURE**

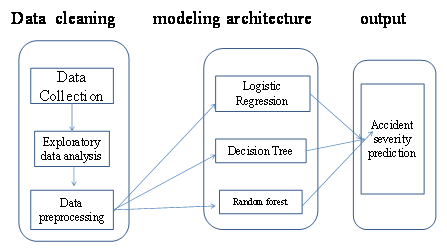


Figure 1 : Proposed System Architecture

3.2 **PROPOSED METHODOLOGY**

Data Preprocessing Data preprocessing is the process in which the data to be used for the machine learning algorithm are manipulated (e.g., transformed, encoded, etc.) to bring them to such a state that the machine can quickly parse it. Additionally, it is a process in which irrelevant data are deleted or modified to be helpful for the predictive analysis process In this research, a validated and online traffic accident dataset is used, which includes UK accidents during 2005–2014, except 2008. This dataset was cleaned up, transformed, and pre-processed to build and train the predictive models. It is because some of the data columns contain redundant data, or are uncorrelated to the prediction target, or consist of missing values. The prepared dataset contains the following columns:

Index

Accident\_Severity

Accident Date

Latitude

Light\_Conditions

District Area

Longitude

Number\_of\_Casualties

Number\_of\_Vehicles

Road\_Surface\_Conditions

Road\_Type

Urban\_or\_Rural\_Area

Weather\_Conditions

Vehicle\_Type

By using the information gathered in the literature review ,it can be concluded that the first three data columns represent the output to be predicted (dependent variables). In contrast, the rest of the data columns are the input required (independent variable) for the models. Then, the interquartile method was applied to trim the outliers—ensuring robustness in our predictive model. After performing the steps above, over a million data records remained over the span of nine years of recorded accident data (2005 to 2007, 2009 to 2011, and 2012 to 2014). The available data may seem to be old, but they fulfill the requirements of this study’s objectives. The data are accurate and consistent, enabling us to conduct holistic modeling for this study. The categorical data columns were then encoded into the numerical format as most machine learning algorithms require all input variables to be in such a form

**Modelling Strategy**

The labels to be predicted are accident-related: accident severity level, number of involved vehicles, and number of casualties. These three factors summarize the overall situation of a car accident under some specific conditions. The resulting system models must predict these three factors given these input conditions. By identifying the accident severity, road managers can focus on areas with greater potential risk. Resources, such as professional personnel to aid accident victims with injuries, are limited in some regions. Still, with the help of predictions from machine learning models, the decision-makers can organize resources in an efficient and need-based manner.

For example, more ambulances should be available in the vicinity of a prediction that indicates many casualties on a specific road. Suppose the result suggests that many vehicles are involved. In that case, more rescue cars, such as tow trucks and fire engines, available on short notice are recommended to reduce the time between requesting and transporting relevant equipment and specialists. We need to measure and obtain all feature values and run the model to perform the prediction. The output labels do not mean the exact number of predicted vehicles and casualties that will occur. Rather they indicate the situation of the most likely scenario predicted by the model. Classification is the machine learning method we chose to use. Regression mainly predicts continuous values, while classification is used for discrete labels. In the dataset, most of the features are discrete. The involved vehicles and the number of casualties are numeric, but the possible values are distributed among several specific numbers due to the characteristics of car accidents. Therefore, they are better treated as discrete labels. Classification has its limitation in predicting unseen results, while regression can predict results that do not appear in the training dataset. Knowing the situation from a macro per-spective can also be meaningful when predicting a car accident. The unseen values, in this case, are large numbers of damaged vehicles and casualties, which indicates a devastating accident. Those tragedies are complicated and may involve many human factors, while the given dataset provides more information about the surrounding environment. Obtaining a predictive result can offer valuable insights to regulation makers and road managers to mitigate accident rates and prevent severe cases. Different algorithms were tested and compared based on accuracy to obtain an optimal prediction model.

There are several classification methods in Spark. However, some methods are not applicable in our case because they only support binary classification. The four algorithms used in this project are the decision tree classifier, random forest classifier, multinomial logistic regression, and naïve Bayes classifier. Each algorithm predicted the three output labels separately, and as a result, they were grouped. To perform the algorithm’s training, we used a global model approach where the data are fed to a pipeline to train a single model, the result of which is a single classifier. This way, higher accuracy is expected from our model because the generated classifier takes advantage of the entire dataset rather than training on local partitions of the dataset. This also means that it is independent of the number of data partitions and produces a single model at the end. Although this may cause longer computational time due to more communication between nodes, higher accuracy is preferred in the case of traffic accidents. Through the analysis of the accuracy of the predictions and the data patterns, the Results and Discussion section interprets the generated results and proposes a new strategy to improve the machine learning model predictions.

3.3 **Steps to be followed for Road Accident Prediction:**

**1. Problem Statement**

**2. Gathering the Data**

**3. Data Preprocessing**

**4. EDA**

**5. Split,Select the Model**

**6. Training Model**

**7. Hyparameter Tuning**

**8. Validate Model**

**9.Deploy**

**CHAPTER-4**

**SYSTEM DESIGN AND IMPLEMENTATION**

4.1 **SOFTWARE AND HARDWARE DETAILS**

4.1.1 **Functional Requirements**

Building a model which would analyse number of casualities and road surface condition in road accidents. Here the model would work accurately.

4.1.2 **Software Requirements**

Operating System - Windows 10

Technology used – Python, Pandas,seaborn,matplotlib

4.1.3 **Hardware Requirements**

Processor- Intel Pentium i5 or above

RAM- 8 GB

4.2 **SOFTWARE DESCRIPTION**

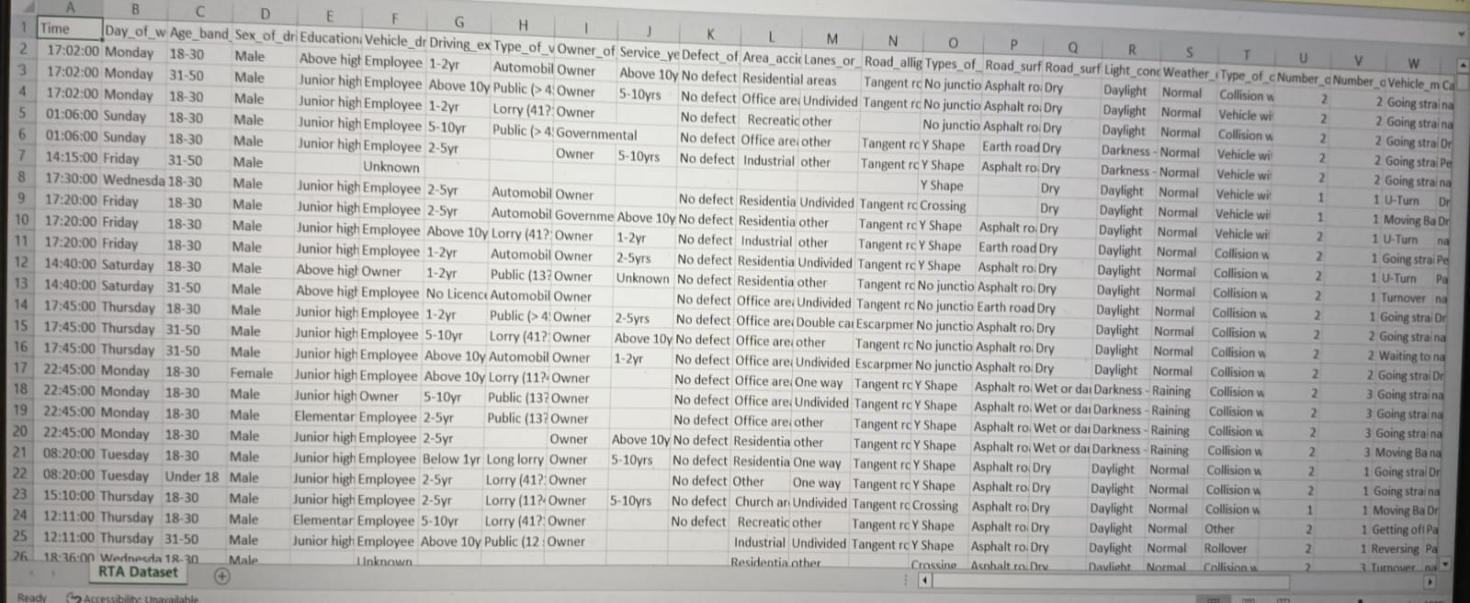
4.2.1 **Problem Statement**

The problem statement for implementing a predictive analysis system in accident prevention and response could be framed as follows:

"The current methods of accident prevention and emergency response are often reactive, relying on investigations after accidents occur or responding to emergencies as they happen. This approach may not effectively address the underlying causes and risks associated with accidents, leading to potential delays in emergency response and a higher number of accidents. Therefore, there is a need for a proactive system that utilizes predictive analysis to identify potential risks, patterns, and factors contributing to accidents. By leveraging historical data, real-time information, and various factors like weather conditions, traffic patterns, and driver behavior, this system aims to predict and prevent accidents before they happen, optimize emergency response efforts, and ultimately improve road safety."

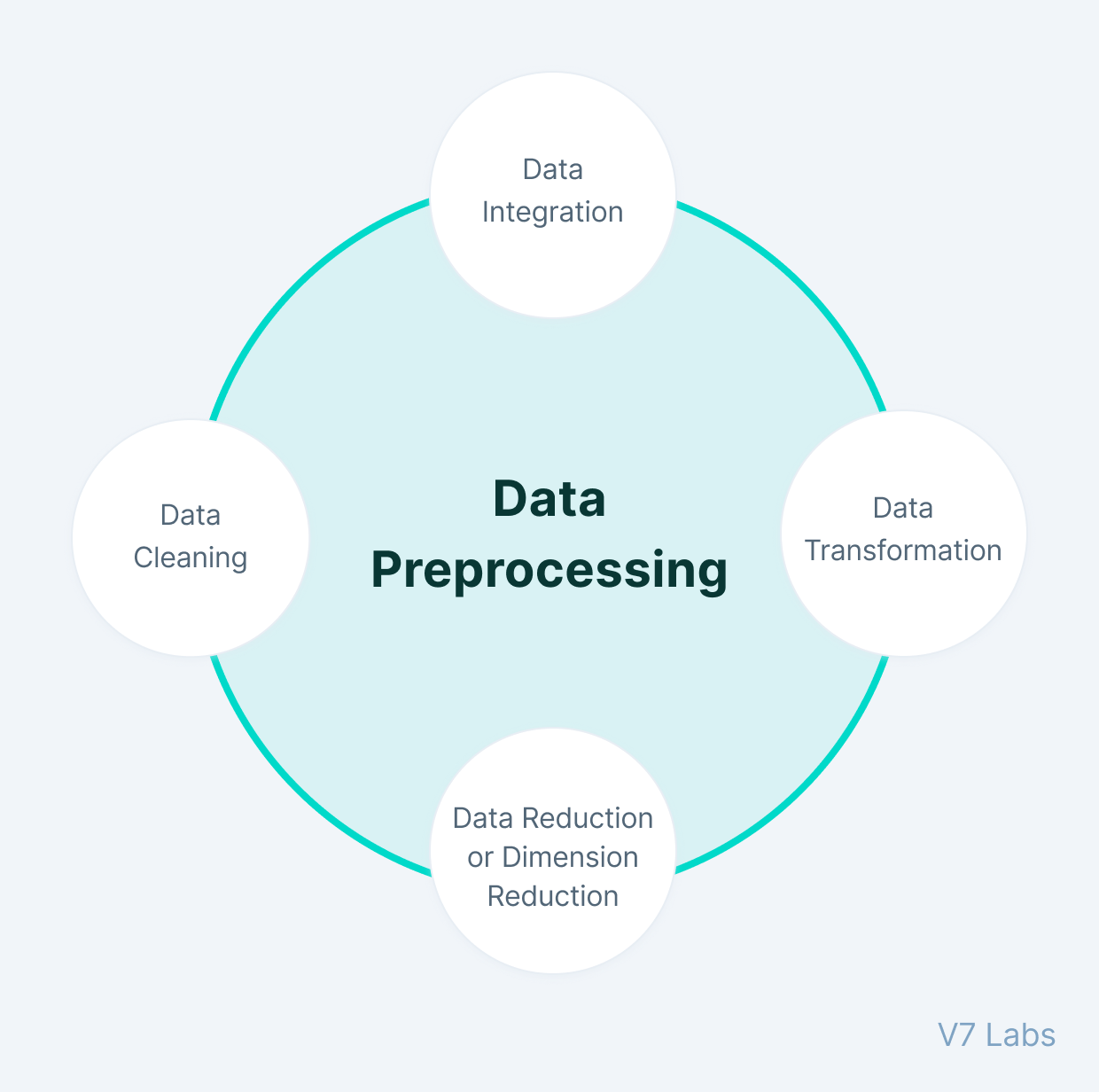
4.2.2 **Gathering the Data**

The purpose of gathering data for accident prevention is to gain insights and understanding about the factors that contribute to accidents. By collecting and analyzing data, we can identify patterns, trends, and potential risks, which can help in developing effective preventive measures. The data allows us to assess the impact of various factors such as road conditions, weather, traffic patterns, and driver behavior on accident occurrence. With this information, authorities can make informed decisions and take targeted actions to reduce accidents and improve road safety. One famous website to collect datasets for various purposes, including accident prevention, is Kaggle. Kaggle is a platform where individuals and organizations share and host datasets for data analysis and machine learning projects. It offers a wide range of datasets on different topics, including transportation and road safety. You can explore Kaggle's website to find datasets specifically related to accident prevention and road safety. Remember to always check the terms and conditions of the dataset and ensure that it aligns with your specific needs and objectives.



4.2.3 **Data Preprocessing:**

Data preprocessing is a crucial step in data analysis and machine learning. It involves preparing and cleaning the raw data to make it suitable for further analysis. The main goal of data preprocessing is to improve the quality and reliability of the data by addressing issues such as missing values, outliers, inconsistencies, and formatting errors.



4.2.4  **EDA:**

**EDA** stands for Exploring Data Analysis.

EDA is a crucial step in analyzing a dataset. It helps us understand the data, identify patterns, and uncover insights. Here's a general process for conducting EDA:

**Data Exploration**: Start exploring the dataset by examining its structure, dimensions, and basic statistics. This includes checking the number of rows and columns, data types, and summary statistics like mean, median, and standard deviation.

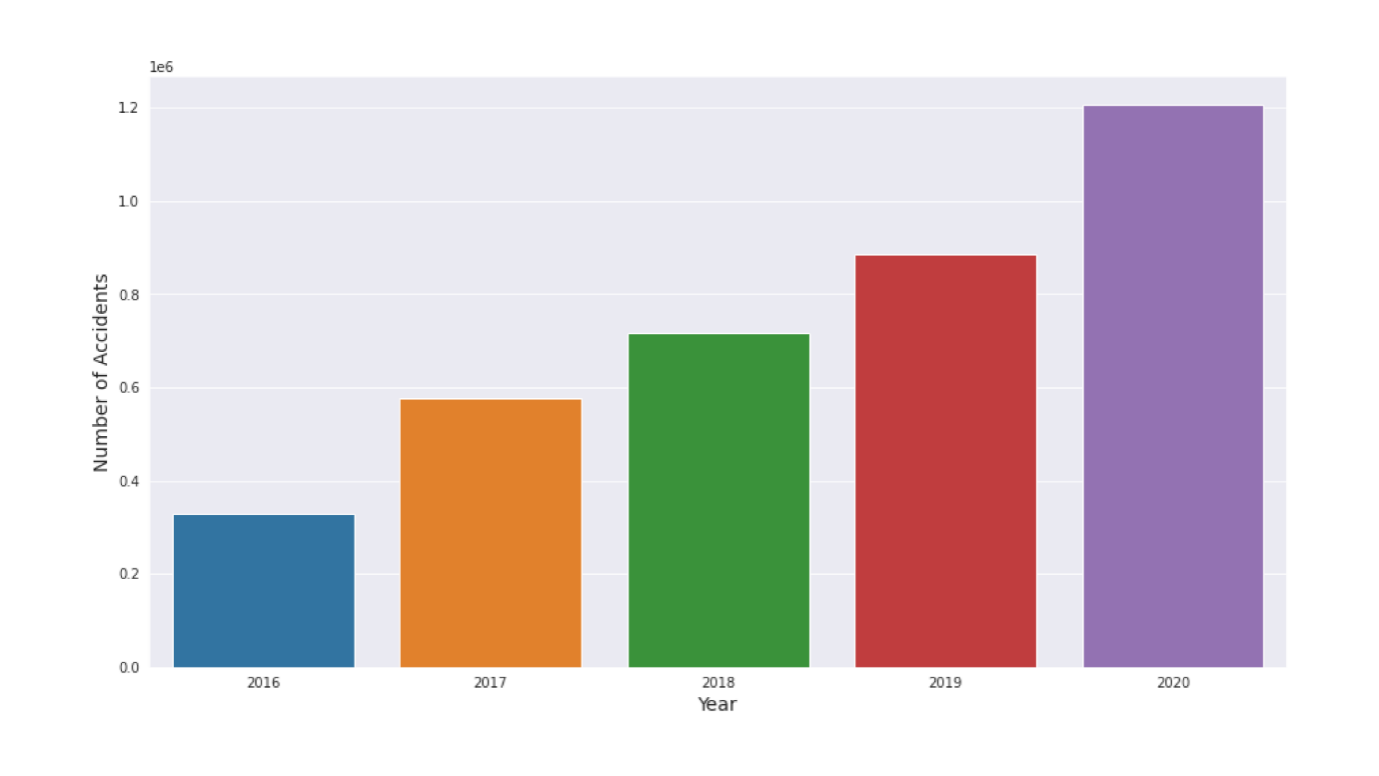
**Visualization**: Create visual representations, such as plots, charts, and graphs, to visually explore the data. This helps in identifying patterns, trends, outliers, and relationships between variables.

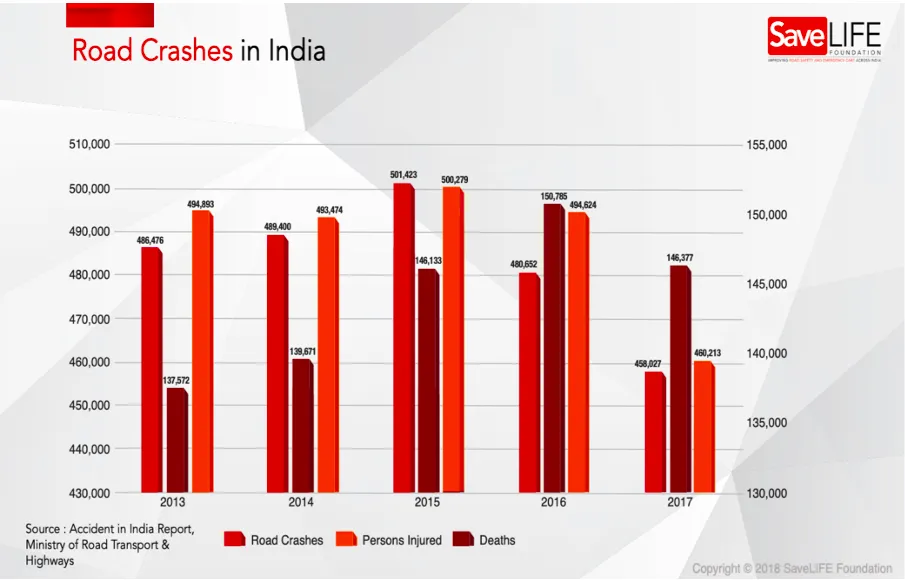
**Statistical Analysis**: Perform statistical tests and calculations to gain deeper insights into the dataset. This may involve calculating correlations, conducting hypothesis tests, or analyzing distributions.

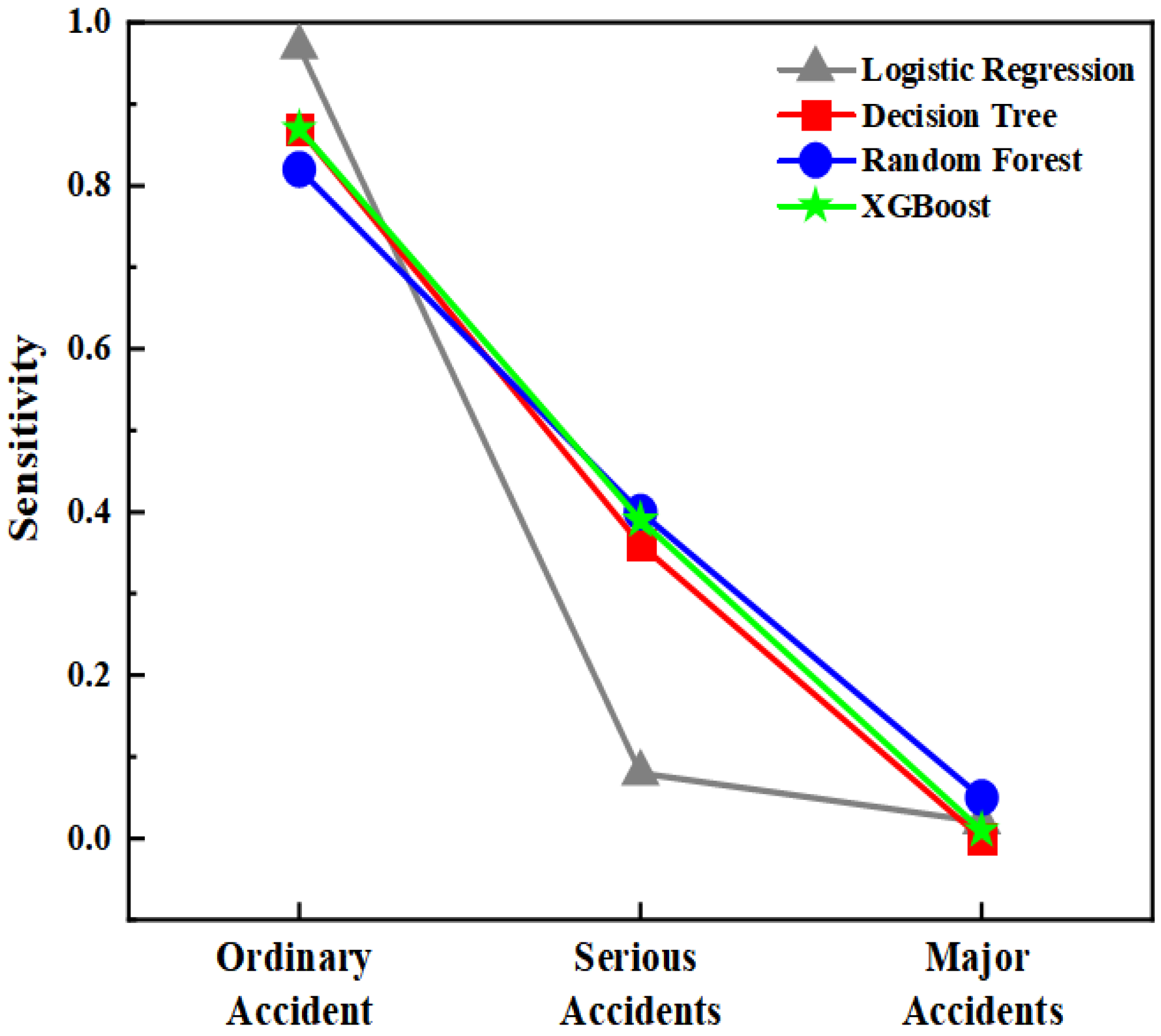
**Feature Engineering:** Transform and engineer new features from existing ones to enhance the dataset's predictive power. This could involve creating new variables, scaling or normalizing data, or encoding categorical variables.

**Iterative Analysis**: Iterate through the previous steps, refining and adjusting the analysis based on the insights gained. This may involve revisiting data cleaning, exploring different visualizations, or conducting additional statistical tests.

Remember, the specific steps and techniques used in EDA can vary depending on the dataset and the goals of the analysis. The key is to explore the data thoroughly, visualize it effectively, and draw meaningful insights to inform further analysis or decision-making.

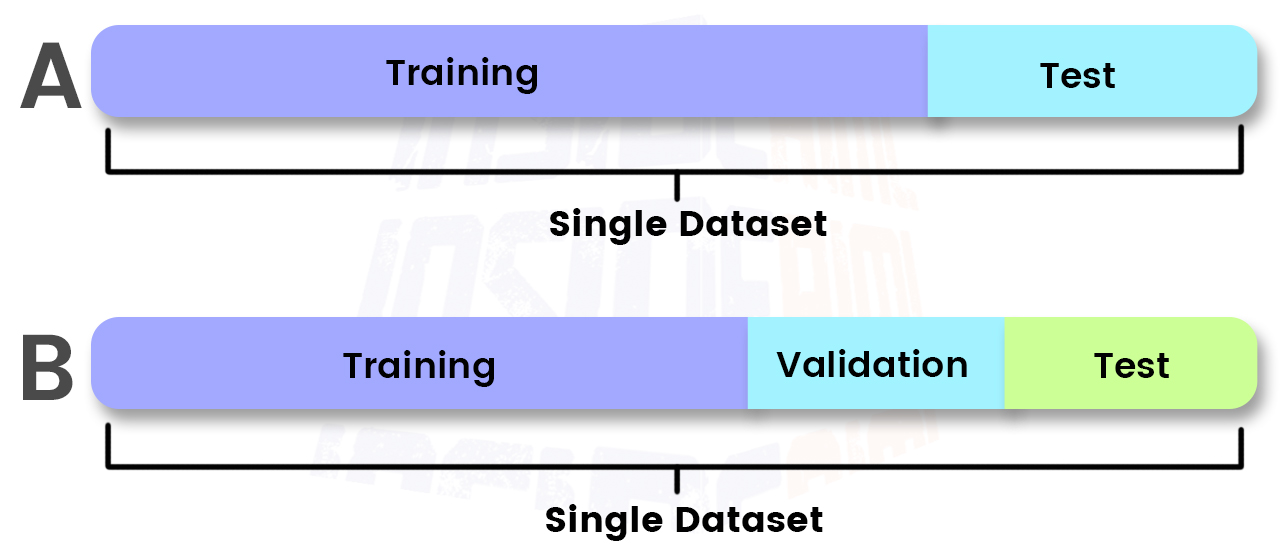






4.2.5  **Split, Select the Model**

Divide the dataset into training and testing sets. The training set is used to build the model, while the testing set is used to evaluate its performance.



4.2.6**. Training Model**

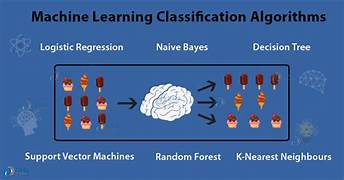
**Select a Model:** Choose the appropriate model architecture or algorithm for your task. This depends on the type of data and the problem you are trying to solve. There are various models to choose from, such as decision trees, neural networks, or support vector machines.

**Train the Model:** Feed the training data into the model and adjust its internal parameters through an optimization process. This process involves minimizing a loss function that measures the difference between the predicted outputs and the actual outputs**.**

**Evaluate the Model**: Use the validation set to assess the model's performance. Calculate metrics such as accuracy, precision, recall, or mean squared error to measure how well the model is performing.

**Fine-tune and Optimize:** Based on the evaluation results, make adjustments to the model's hyperparameters or architecture to improve its performance. This may involve tweaking parameters, trying different optimization algorithms, or adding regularization techniques.

**Test the Model:** Once you are satisfied with the model's performance on the validation set, you can test it on a separate test set or real-world data to assess its generalization ability.

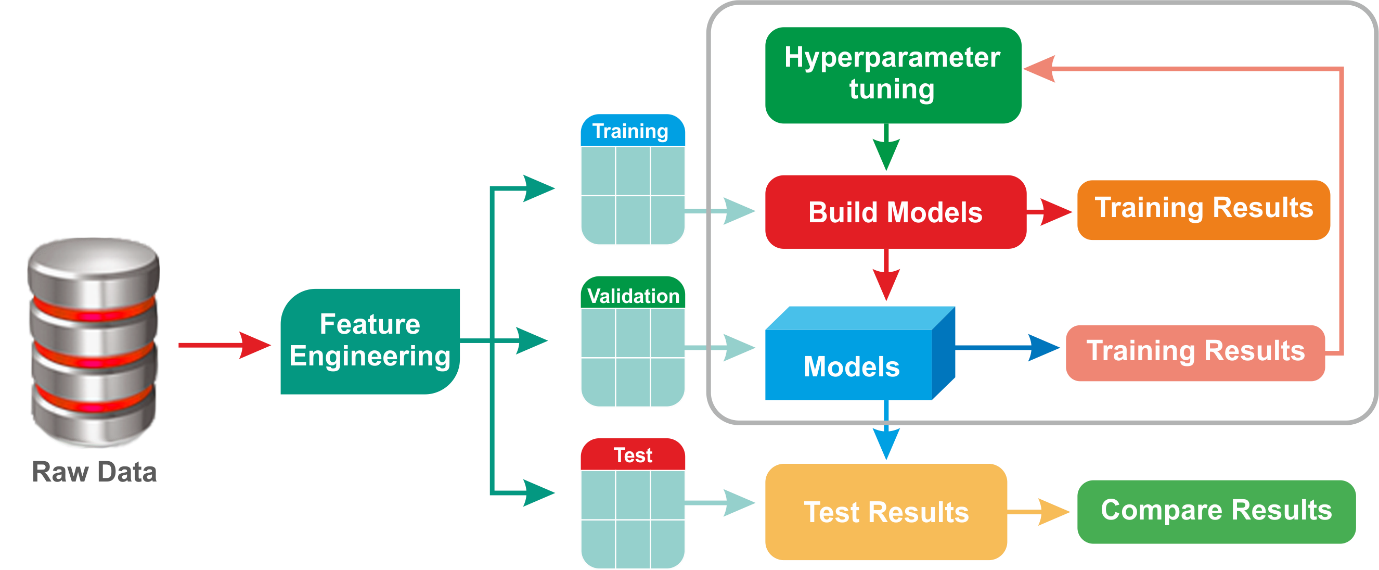


4.2.7**. Hyparameter Tuning**

Hyperparameter tuning is a critical step in machine learning model development that involves selecting the best set of hyperparameters for your model. Hyperparameters are parameters that are set before the learning process begins and control aspects of the learning process itself. Examples include the number of trees in a Random Forest, the learning rate in a Gradient Boosting Machine, or the regularization parameter in a Support Vector Machine.

* Manually try different combinations of hyperparameters based on intuition and domain knowledge. This method is time-consuming and not always efficient, but it can provide insights into how hyperparameters affect model performance.
* Exhaustively search through a specified subset of hyperparameter combinations. You specify a grid of hyperparameter values, and Grid Search evaluates the model's performance for each combination using cross-validation. It's computationally expensive but can be effective for small search spaces.
* Randomly sample hyperparameters from specified distributions. Unlike Grid Search, Random Search does not exhaustively evaluate all combinations but instead randomly samples them. It's more computationally efficient and often yields comparable results to Grid Search.
* Bayesian Optimization builds a probabilistic model of the objective function (model performance) and uses it to decide which hyperparameters to try next. It's more efficient than Grid Search and Random Search and can handle a larger search space.
* Genetic Algorithms mimic the process of natural selection to find the best hyperparameters. They maintain a population of candidate solutions (sets of hyperparameters), apply genetic operators (mutation, crossover), and select the best-performing solutions for the next generation.

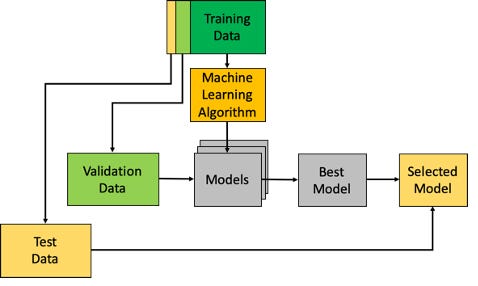
When tuning hyperparameters, it's important to use a separate validation set (or cross-validation) to avoid overfitting to the test set. You can also use techniques like early stopping to prevent overfitting during training.

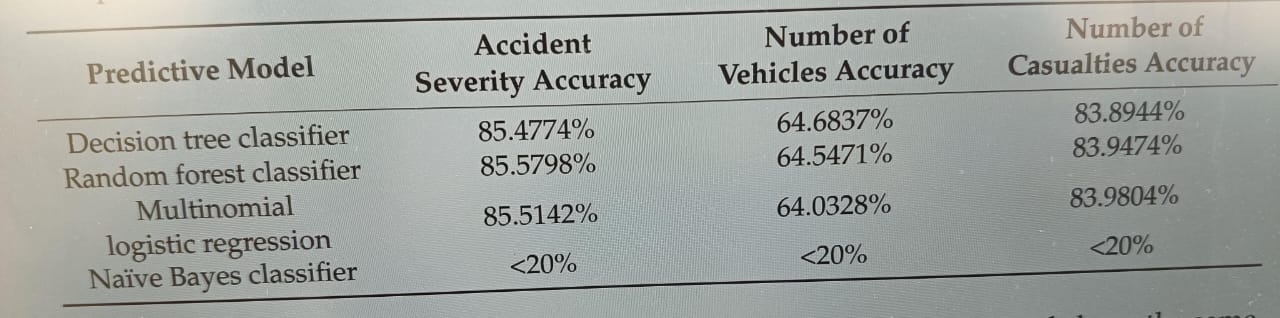


4.2.8**. Validate Model**

Validating a machine learning model involves assessing its performance on data that was not used during training. This helps ensure that the model can generalize well to unseen data and provides an estimate of its real-world performance. The process typically involves the following steps:

* Divide your dataset into at least two parts: a training set and a testing set. Optionally, you can also include a validation set for hyperparameter tuning.
* Use the training set to train your machine learning model. This involves feeding the training data into the model and adjusting its parameters (weights and biases) based on the input-output pairs.
* After training, evaluate the model's performance on the validation set. This step is especially important if you have hyperparameters to tune. For each set of hyperparameters, train a model on the training set and evaluate its performance on the validation set. This helps you select the best-performing model and hyperparameters.

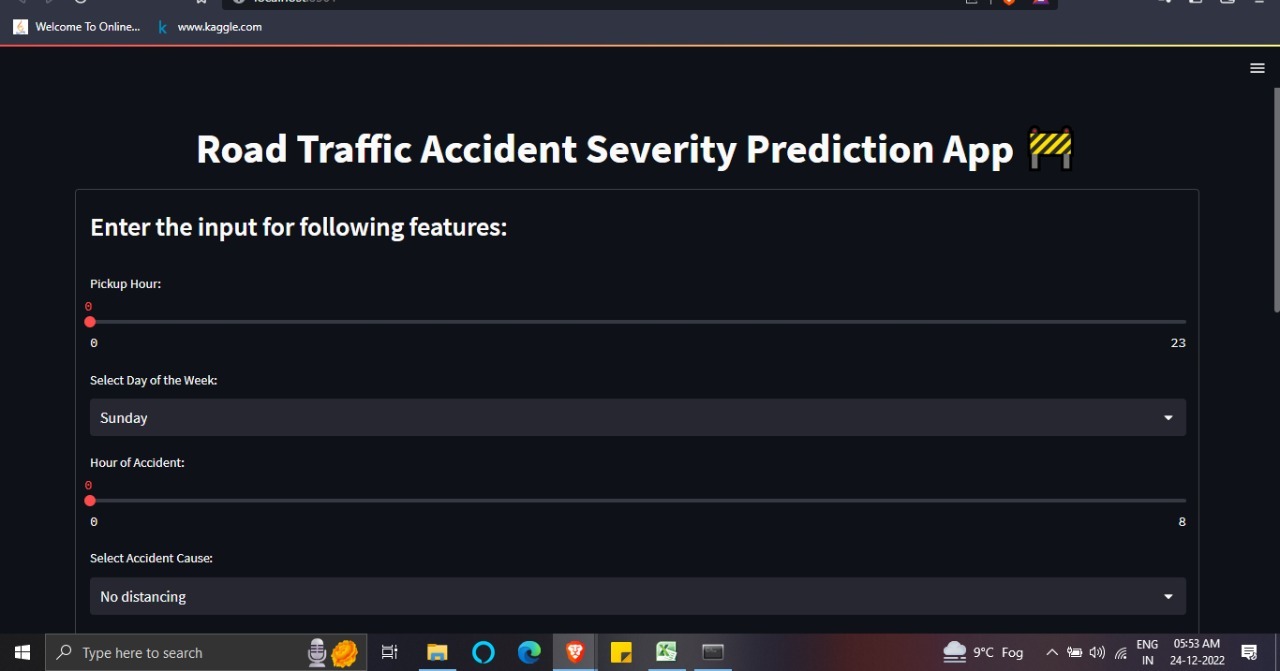




4.2.9**.Deploy**

Deploying a machine learning model involves making the model available for use in a production environment, where it can make predictions on new data. The process typically involves several steps:

* **Prepare the Model**: Ensure that your model is trained and validated on a representative dataset. It should be able to make accurate predictions on new, unseen data.
* **Choose a Deployment Environment**: Decide where you want to deploy your model. Common options include cloud platforms (such as AWS, Google Cloud, or Azure), on-premises servers, edge devices, or containers.
* **Model Serialization**: Serialize your trained model into a format that can be saved to disk and loaded into memory during inference. Common serialization formats include pickle (for scikit-learn models), HDF5 (for Keras and TensorFlow models), and ONNX (for interoperability across different frameworks).
* **Create an Inference Pipeline**: If necessary, create an inference pipeline that preprocesses input data before passing it to the model for prediction. This may involve scaling, normalization, encoding categorical variables, etc. Ensure that the preprocessing steps are consistent with those used during model training.
* **Set up an API**: Expose your model through an API (Application Programming Interface) that allows clients to send input data and receive predictions. You can use web frameworks like Flask or Django to create the API endpoints.
* **Deploy the API**: Deploy the API to your chosen deployment environment. This may involve setting up servers, containers, or serverless functions depending on your infrastructure requirements.
* **Monitor and Scale:** Once deployed, monitor the performance of your deployed model and scale resources as needed to handle varying loads. This may involve autoscaling, load balancing, and performance monitoring tools.
* **Security:** Ensure that your deployed model is secure and protected from unauthorized access. Implement authentication, encryption, and other security measures as necessary.
* **Version Control**: Keep track of different versions of your deployed model to facilitate rollback in case of issues or to support A/B testing.
* **Documentation and Maintenance**: Provide documentation for your API endpoints, including input/output formats, expected data types, and error handling. Regularly update and maintain your deployed model to incorporate new data and improve performance over time.



4.3 **IMPLEMENTATION**

It summarizes how the model was coded in **Python**.

The key components like are described **Random Forest** **Classifier**.

**Random Forest** **Classifier** is used in this for classification purpose.

4.4 **CODE IMPLEMENTATION**:-

from datetime import datetime

import pandas as pd

df = pd.read\_csv("C:/Users/jaya2/OneDrive/Documents/final data.csv")

def manual\_testing(date, district):

print(date)

r=df.loc[(df['Accident Date']==date) & (df['District Area']==district),['Number\_of\_Casualties','Road\_Surface\_Conditions','Weather\_Conditions']]

print(r)

date=input("Enter a date(dd-mm-yyyy format):")

district=input()

manual\_testing(date,district)

4.5 **web app:-**

import pandas as pd

import pickle

import streamlit as st

#df = pd.read\_csv("path\_to\_your\_dataset.csv")

df = pd.read\_csv("C:/Users/jaya2/OneDrive/Documents/final data.csv")

def manual\_testing(date, district):

    # Filter the DataFrame based on date and district

    filtered\_data = df[(df['Accident Date'] == date) & (df['District Area'] == district)]

    # For now, let's return the filtered data as a placeholder

    return filtered\_data

def main():

    st.title("Accident Prediction Web App")

    # User input for date and district

    date = st.text\_input("Enter a date (dd-mm-yyyy format):")

    district = st.text\_input("Enter district:")

    if st.button("Predict Road Accident"):

        # Perform manual testing and get the result

        accident\_data = manual\_testing(date, district)

        # Display the prediction result

        st.success("Prediction Result:")

        st.write(accident\_data)

if \_\_name\_\_ == '\_\_main\_\_':

    main()

try:

    df = pd.read\_csv("C:/Users/jaya2/OneDrive/Documents/final data.csv")

    # Proceed with further processing of the DataFrame

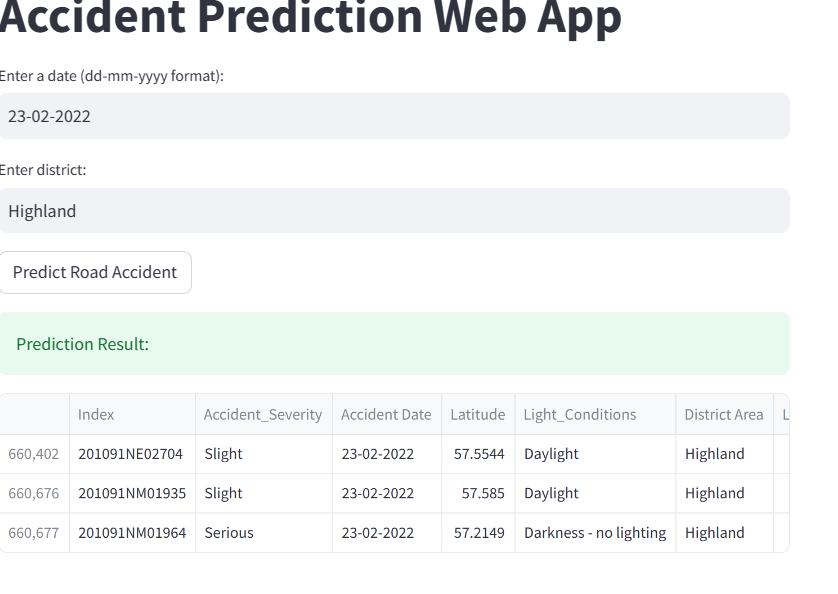
except Exception as e:

    print("Error reading CSV file:", e)

**CHAPTER-5**

**RESULT AND DISCUSSION**

5.1 **RESULT SCREENSHOTS**

****

5.2 **DISCUSSION**

According to the statistical analysis result of the dataset, most records have the same accident severity and the number of casualties. Over 90% of car accident records have a severity level of 3 (the lowest severity), and over 95% have many casualties of 1. In other words, minor accidents occupy most of the records in the dataset. During the modeling stage, many records with different features are classified into the same label, thereby reducing the model’s sensitivity. When dividing the dataset for training and testing, many minor accidents are more likely to be involved in the testing data; hence, it is not surprising that the accuracy is high because the model can predict minor accidents very well. Hence, the classification model can achieve a lower accuracy than the other two labels. An improved training strategy is proposed to improve the accuracy in predicting accidents with the severity of 1 and 2, the number of casualties above 1, and the number of vehicles. Since the problem is mainly caused by the unbalanced portion of each of the labels, the population of records with severity of 3 or the number of casualties that is 1 will be reduced in the new strategy.. The resulting model may become more accurate in classifying data into different labels, then the testing results accuracy could be higher. Since the three models’ accuracy is very close, we discuss the three algorithms and car accident problems based on other aspects. After data processing, the attributes of a car accident are either nominal or ordinal. Multinomial logistic regression is based on linear algebra, it can be used to classify data points, but the performance might reduce when the attributes become more complex. DT and RF are more suitable in this case. Random forest is a method derived from the decision tree. Random forest generates many decision trees randomly and produces the final classification result by launching voting among the temporary effects of those decision trees. The decision tree’s performance is ensured by correctly determining the important factors. In the car accident dataset, our knowledge makes it hard to determine their importance in an accident. Therefore, the random forest’s methodology is more sensible in such a complicated case.

**CHAPTER-6**

# CONCLUSION

## This project has covered the need to reduce traffic accidents since the number of accidents produced is still relatively high and causes significant economic expenditure. To find insights into where and how these accidents could be reduced, the project aimed to generate predictive models to understand the causes of the accidents. We determined the independent determinants from the extensive literature review , including weather, light, road surface conditions, and others. We first had to drop unnecessary columns that were not needed for the prediction model from the dataset we used. Outlier data were dropped to make the data more robust to uncertainties and errors.

## Furthermore, we performed some pre-processing to convert some of these variables into more useful formats, such as latitude and longitude, into geographical regions for more general solutions. The output variables to predict are the number of vehicles, the number of casualties, and accident severity. They were treated as discrete labels due to the limited values and concentration around that small range. Since the dataset was large in volume, big data analysis is important to handle large datasets of various types and generate them quickly. The dataset used in this research is quite large, and therefore, big data analysis helps generate faster results. Additionally, big data analysis allows us to extract useful information from the datasets through classification to generate predictive models. By using the analyzed results, decision-makers can easily understand accident patterns, driver behavior, time of day, road and weather conditions causing traffic congestion and other key factors contributing to accidents, such as fatalities and serious injuries, thus improving traffic safety control strategies. They can also use predictive models to adopt new policies in road safety and accident prevention.

## Future work may include determining the determinant factors from this dataset before developing predictive models using methods such as correlation matrix to find the significant dependent variables for the dataset. Apache Spark’s framework handles distributed data processing and streaming datasets to overcome traditional storage restrictions, and is useful for big data analysis. Using MLLib in Spark, we used the machine learning models implemented in classification: decision tree, random forest, multinomial logistic regression, and naïve Bayes classifier. Then, the input (independent) variables and output (dependent) variables from the dataset were split into training sets to fit the data and testing sets to obtain the accuracy of the models we used. Matplotlib was used to draw the figures for the data exploration and discussion of the effect of the independent variables. Null values were dropped from the dataset, and outlier data were dropped using the interquartile method to make the data more robust. More advanced machine learning algorithms, such as neural networks, could be used to determine a higher accuracy. Other techniques to fill in missing data, such as using mean values, could improve accuracy rather than dropping the null values. Three algorithms produced relatively good results (60 to 80% accuracy). Accident severity and the number of casualties were above 80%, and the number of vehicles was slightly lower at around 64%. This might have been caused because the possible range for the number of vehicles was higher.

## Additionally, because most of the dataset’s records, over 90%, had minor accident severities, the modeling stage may have reduced the model’s sensitivity by joining records with different features into the same label. Hence, an improved strategy can be proposed in which the unbalanced proportion of the labels can be reduced before feeding the dataset to the model for training, such as reducing the number of records with the severity of three casualties equal to one but keeping it as the dominant value. Hence, the model may become more accurate in classification . Some valuable findings are generated in data exploration. Traffic accidents are more likely to occur on foggy days than on rainy days and more likely to occur on snowy roads than on flooded. Neutrosophic statistics, an extension of interval statistics, are more efficient than classical statistics when the data are imprecise and uncertain . More applications of neutrosophic statistics can be seen in the intelligent traffic system .

## The limitation of the current study is that some of the factors may have effects on accidents that are not considered. In the future, comprehensive datasets that include more important variables, such as air humidity and wind speed, could be used to possibly improve the accuracy of the prediction.The implementation in this study can take accurate decisions and experiences to manage the situation and assist traffic authorities in reducing the number of accidents, as they are validated and more accurate in predicting the severity of traffic accidents. To make this more feasible, the authors propose building a system or mobile app that can predict and warn road users of traffic accidents, providing users with accurate predictions.

6.2  **PERFORMANCE EVALUATION**

When it comes to evaluating the performance of predictive analysis for accident prevention and response, there are several metrics you can consider. Here are a few commonly used ones:

**1. Accuracy:** This measures the overall correctness of the model's predictions. It is the ratio of the number of correct predictions to the total number of predictions made.

**2. Precision:** This metric focuses on the proportion of true positive predictions out of all positive predictions made by the model. It helps assess the model's ability to avoid false positives.

**3. Recall:** Also known as sensitivity or true positive rate, recall measures the proportion of actual positive instances that the model correctly identifies. It helps evaluate the model's ability to avoid false negatives.

**4. F1 Score:** The F1 score is the harmonic mean of precision and recall. It provides a single metric that balances both precision and recall, giving you an overall measure of the model's performance.

**5. Area Under the ROC Curve (AUC-ROC):** This metric evaluates the model's ability to discriminate between positive and negative instances. It plots the true positive rate against the false positive rate, providing anoverall measure of the model's performance across different classification thresholds.

**CHAPTER-7**

**REFERENCES**

1. World Health Organization. Road Traffic Injuries. Available online: https://www.who.int/news-room/fact-sheets/detail/roadtraffic-injuries (accessed on 20 June 2021).

2. International Transport Forum. Road Safety Annual Report 2015. Road Saf. Annu. Rep. 2015, 486. [CrossRef]

3. Wang, Y.; Zhang, W. Analysis of Roadway and Environmental Factors Affecting Traffic Crash Severities. Transp. Res. Procedia 2017, 25, 2119–2125. [CrossRef]

4. Wang, J.; Lu, H.; Sun, Z.; Wang, T.; Wang, K. Investigating the Impact of Various Risk Factors on Victims of Traffic Accidents. Sustainability 2020, 12, 3934. [CrossRef]

5. Potoglou, D.; Carlucci, F.; Cirà, A.; Restaino, M. Factors associated with urban non-fatal road-accident severity. Int. J. Inj. Control. Saf. Promot. 2017, 25, 303–310. [CrossRef] [PubMed] Sustainability 2023, 15, 5939 14 of 15

6. Alomari, E.; Mehmood, R.; Katib, I. Road Traffic Event Detection Using Twitter Data, Machine Learning, and Apache Spark. In Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Leicester, UK, 19–23 August 2019; pp. 1888–1895. [CrossRef]

7. Asor, J.R.; Catedrilla, G.M.B.; Estrada, J.E. A Study on the Road Accidents Using Data Investigation and Visualization in Los Baños, Laguna, Philippines. In Proceedings of the 2018 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 6–7 March 2018; pp. 96–101. [CrossRef]

8. Aziz, K.; Zaidouni, D.; Bellafkih, M. Real-Time Data Analysis Using Spark and Hadoop. In Proceedings of the 2018 4th International Conference on Optimization and Applications (ICOA), Mohammedia, Morocco, 26–27 April 2018; pp. 1–6. [CrossRef]

9. Fisher-Hickey, D. 1.6 million UK Traffic Accidents. 2021. Available online: https://www.kaggle.com/daveianhickey/2000-16- traffic-flow-england-scotland-wales (accessed on 20 June 2021).

10. Li, Y.; Xing, L.; Wang, W.; Wang, H.; Dong, C.; Liu, S. Evaluating impacts of different longitudinal driver assistance systems on reducing multi-vehicle rear-end crashes during small-scale inclement weather. Accid. Anal. Prev. 2017, 107, 63–76. [CrossRef]

11. Nofal, F.; Saeed, A. Seasonal variation and weather effects on road traffic accidents in Riyadh City. Public Health 1997, 111, 51–55. [CrossRef]

12. Hammad, H.M.; Ashraf, M.; Abbas, F.; Bakhat, H.F.; Qaisrani, S.A.; Mubeen, M.; Fahad, S.; Awais, M. Environmental factors affecting the frequency of road traffic accidents: A case study of sub-urban area of Pakistan. Environ. Sci. Pollut. Res. 2019, 26, 11674–11685. [CrossRef]

13. Elassad, Z.E.A.; Mousannif, H.; Al Moatassime, H. Class-imbalanced crash prediction based on real-time traffic and weather data: A driving simulator study. Traffic Inj. Prev. 2020, 21, 201–208. [CrossRef] [PubMed]

14. Banerjee, K.; Bali, V.; Sharma, A.; Aggarwal, D.; Yadav, A.; Shukla, A.; Srivastav, P. Traffic Accident Risk Prediction Using Machine Learning. In Proceedings of the 2022 International Mobile and Embedded Technology Conference (MECON), Noida, India, 10–11 March 2022; pp. 76–82. [CrossRef]

15. Kumeda, B.; Zhang, F.; Zhou, F.; Hussain, S.; Almasri, A.; Assefa, M. Classification of Road Traffic Accident Data Using Machine Learning Algorithms. In Proceedings of the 2019 IEEE 11th International Conference on Communication Software and Networks (ICCSN), Chongqing, China, 12–15 June 2019; pp. 682–687. [CrossRef]

16. Moghadam Gilani, V.N.; Hosseinian, S.M.; Ghasedi, M.; Nikookar, M. Data-Driven Urban Traffic Accident Analysis and Prediction Using Logit and Machine Learning-Based Pattern Recognition Models. Math. Probl. Eng. 2021, 2021, 1–11. [CrossRef]

17. Mohanta, B.K.; Jena, D.; Mohapatra, N.; Ramasubbareddy, S.; Rawal, B.S. Machine learning based accident prediction in secure IoT enable transportation system. J. Intell. Fuzzy Syst. 2022, 42, 713–725. [CrossRef]

18. Abdulhafedh, A. Incorporating the Multinomial Logistic Regression in Vehicle Crash Severity Modeling: A Detailed Overview. J. Transp. Technol. 2017, 7, 279–303. [CrossRef]

19. Bozorg, S.; Tetri, E.; Kosonen, I.; Luttinen, T. The Effect of Dimmed Road Lighting and Car Headlights on Visibility in Varying Road Surface Conditions. Leukos 2018, 14, 259–273. [CrossRef]

20. Sameen, M.I.; Pradhan, B. Severity Prediction of Traffic Accidents with Recurrent Neural Networks. Appl. Sci. 2017, 7, 476. [CrossRef]

21. Aslam, M. Monitoring the road traffic crashes using NEWMA chart and repetitive sampling. Int. J. Inj. Control. Saf. Promot. 2020, 28, 39–45. [CrossRef] [PubMed]

22. Lee, J.; Chae, J.; Yoon, T.; Yang, H. Traffic accident severity analysis with rain-related factors using structural equation modeling—A case study of Seoul City. Accid. Anal. Prev. 2018, 112, 1–10. [CrossRef]

23. Yau, K.K.W. Risk factors affecting the severity of single vehicle traffic accidents in Hong Kong. Accid. Anal. Prev. 2004, 36, 333–340. [CrossRef]

24. Petrova, E.G.; Shiryaeva, A.V. Road accidents in Moscow: Weather impact. Adv. Environ. Sci. 2019, 11, 19–30. Available online: http://www.aes.bioflux.com.ro/docs/2019.19-30.pdf (accessed on 20 June 2021).

25. Novikov, A.; Novikov, I.; Shevtsova, A. Study of the impact of type and condition of the road surface on parameters of signalized intersection. Transp. Res. Procedia 2018, 36, 548–555. [CrossRef]

26. Ismael, K.; Razzaq, N.A. Traffic Accidents Analysis on Dry and Wet Road Bends Surfaces in Greater Manchester-UK. Kurd. J. Appl. Res. 2017, 2, 284–291. [CrossRef]

27. Shafabakhsh, G.A.; Famili, A.; Bahadori, M.S. GIS-based spatial analysis of urban traffic accidents: Case study in Mashhad, Iran. J. Traffic Transp. Eng. Engl. Ed. 2017, 4, 290–299. [CrossRef]

28. Labib, F.; Rifat, A.S.; Hossain, M.; Das, A.K.; Nawrine, F. Road Accident Analysis and Prediction of Accident Severity by Using Machine Learning in Bangladesh. In Proceedings of the 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, 28–30 June 2019; pp. 1–5. [CrossRef]

29. Adanu, E.K.; Hainen, A.; Jones, S. Latent class analysis of factors that influence weekday and weekend single-vehicle crash severities. Accid. Anal. Prev. 2018, 113, 187–192. [CrossRef] [PubMed]

30. Kumar, S.; Mahima, S.K.; Srivastava, D.K.; Kharya, P.; Sachan, N.; Kiran, K. Analysis of risk factors contributing to road traffic accidents in a tertiary care hospital. A hospital based cross-sectional study. Chin. J. Traumatol. 2020, 23, 159–162. [CrossRef] Sustainability 2023, 15, 5939 15 of 15

31. Acker, B.; Yuan, M. Network-based likelihood modeling of event occurrences in space and time: A case study of traffic accidents in Dallas, Texas, USA. Cartogr. Geogr. Inf. Sci. 2018, 46, 21–38. [CrossRef]

32. Ait-Mlouk, A.; Gharnati, F.; Agouti, T. Application of Big Data Analysis with Decision Treefor Road Accident. Indian J. Sci. Technol. 2017, 10, 1–10. [CrossRef]

33. Cheng, R.; Zhang, M.-M.; Yu, X.-M. Prediction Model for Road Traffic Accident Based on Random Forest. DEStech Trans. Soc. Sci. Educ. Hum. Sci. 2019. [CrossRef]

34. Fountas, G.; Fonzone, A.; Gharavi, N.; Rye, T. The joint effect of weather and lighting conditions on injury severities of single-vehicle accidents. Anal. Methods Accid. Res. 2020, 27, 100124. [CrossRef]

35. Dogru, N.; Subasi, A. Traffic accident detection using random forest classifier. In Proceedings of the 2018 15th Learning and Technology Conference (L&T), Jeddah, Saudi Arabia, 25–26 February 2018; pp. 40–45. [CrossRef]

36. Joni, H.H.; Al-Dahawi, A.M.; Al-Tamimi, O.J. Analysis of traffic accident severity in Baghdad city using Binary Logistic Regression Model. IOP Conf. Series Mater. Sci. Eng. 2020, 737, 012140. [CrossRef]

37. AlMamlook, R.E.; Kwayu, K.M.; Alkasisbeh, M.R.; Frefer, A.A. Comparison of Machine Learning Algorithms for Predicting Traffic Accident Severity. In Proceedings of the 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, 9–11 April 2019. [CrossRef]

38. Bakhouyi, A.; Bakali, A.; Talea, M.; Dhital, A.; Deivanayagampillai, N. An Intelligent Traffic Control System Using Neutrosophic Sets, Rough sets, Graph Theory, Fuzzy sets and its Extended Approach: A Literature Review. Neutrosophic Sets Syst. 2022, 50, 11–46. Available online: https://digitalrepository.unm.edu/cgi/viewcontent.cgi?article=2113&context=nss\_journal#page=23 (accessed on 16 March 2023).