National College of Ireland

Comparison of the Ensemble and Stacking Approaches in Predicting Mortality in Vehicle Collisions in New York

MSc Research Project
Programme Name: MSc in Data Analytics

Subramanyam Dhandapani
Student ID: x22245421

School of Computing
National College of Ireland

Supervisor: Anderson Simiscuka

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Subramanyam Dhandapani
Student ID:	x22245421
Programme:	Programme Name: MSc in Data Analytics
Year:	2023-2024
Module:	MSc Research Project
Supervisor:	Anderson Simiscuka
Submission Due Date:	12/08/2024
Project Title:	Comparison of the Ensemble and Stacking Approaches in Predicting Mortality in Vehicle Collisions in New York
Word Count:	7382
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Subramanyam Dhandapani
Date:	12th August 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Comparison of the Ensemble and Stacking Approaches in Predicting Mortality in Vehicle Collisions in New York

Subramanyam Dhandapani
x22245421

Abstract

Traffic collisions are becoming a major cause of deaths and injuries worldwide which draws attention for the need of technology in addressing and identifying the important aspects which makes these accidents to happen. This Research aims to analyse the New York Motor Collision data by applying various machine learning techniques and statistical methods to identify the factors and trends which has a huge impact on the survivability of the person during the motor collision. This research uses KDD methodology to identify patterns and trends in the collision dataset. Also, for this research two datasets are used where one contain the data about the crashes and the other contains the person's information. This study compares the stacking model to an ensemble approach with the evaluation of metrics F1 score, accuracy, precision and recall. This Research uses three classification algorithms namely decision tree, random forest and k-nearest neighbor comparing to each other by applying stacking and ensemble approaches among them.

Keywords: KDD Methodology, Stacking Approach, Motor Collision, Ensemble Approach, Injury Severity.

1 Introduction

1.1 Background and Motivation

The growth in the population mostly in the urban areas has given way in increasing motor collisions. The Fatalities or the Injuries happened by motor collision in the cities has become into a global concern where most of the people choose to live in the urbanized cities instead of living in the outskirts of cities. According to a recent article released by (Organization, 2023) on road safety and motor collisions shows that the traffic collision fatalities has reduced to 1.5 million people per year, but still, it remains a need for application of technologies to identify and mitigate the risk factors that causes the collision. Also, the motor collisions mostly happen in certain specific areas and have high possibility to happen at specific time periods as said by (Bíl et al., 2019). Since, there is a significant increase in the number of road vehicles for the past decades motor vehicle collision has become a major concern globally. Motor Vehicle collisions remain a big threat to the children and the youths who are in the age between 5 to 29 years.

The number of collisions happening also differs on various factors such as driver age and gender. In UK 54% drivers involved in the motor collisions were asked to provide breath analysis test during the years of 2003 to 2015 (Lloyd et al., 2015).

The report released by the WHO shows that around 53% of the fatalities happening to the road users includes pedestrians, cyclists and motorists. The deaths of the pedestrian have risen around 4% in the year 2010 to 2021 and it is around 23% of global deaths. A Report released by WHO on the road safety shows that the road collisions will jump from ninth position of leading cause of death to fifth by the year 2030. Researches are still going on analyzing the crash factors which increases the mortality ratio (Organization, 2023). Motor Collisions deaths happen due to various reasons like body injury of the person, safety equipment used during the time of collision, position in the vehicle, the place of the injury for the person in the body, crash location, contributing factors for the collision. A study done by (Wahab et al., 2019) used random forest, Decision Tree and Instance based learning with parameter comparing to each other in the prediction of crash severity of the person in the collision. These algorithms were evaluated using a 10-fold cross validation technique.

The Identification of the determinants of the motor vehicle collisions is a complex and critical task. There are various factors that which ranges from human behaviour, weather conditions and all other external factors should be factored in which can significance the severity of the outcomes in the event of collisions. Data Mining involves a series of techniques from different fields by including database concepts, statistical and machine learning techniques to uncover the patterns and insights that can help in identifying the crash severity of the collision. Most of the General Data Mining Principles involves identification of sequential patterns, clustering, classification and predictions which are applied to various areas. Applying classification techniques gives interesting results (Santos et al., 2022). The application of machine learning techniques in risk assessment has been increasing over the years where Artificial Intelligence is the most common method used which is followed by Support vector machine, decision trees and K-Nearest Neighbors and many other techniques have been developed over a period of time for the prediction of crash severity in traffic collisions.

A comparative study between the ML models was done by (Ahmed et al., 2023) in predicting the Accident severity in New Zealand by comparing the single mode models and ensemble models. The single mode models used were Logistic Regression, K-Nearest Neighbor and Naive Bayes classifier. The Ensemble models used were in the order Random Forest, XGBoost and Adaboost. The ensemble models were able to predict the severity of collision more accurately than the single mode models. The Dataset used for this research consists of two files crash which contains the data of the crash like crash date, location, number of persons injured or killed in the collision and contributing factors of the vehicle. The other file contains data about the Persons like age, gender, emotional status of the person and the bodily injury of the person. The Data contains 20+ attributes and 4276413 collisions combining both the datasets. Both the datasets are combined using a common key called collision ID in both the datasets. For this Research both the datasets are merged together and data preprocessing steps are applied to the dataset.

1.2 Research Question and Objectives

Traffic collisions remains a significant challenge and addressing this with the help of machine learning is significant. This study proposes an approach to compare ensemble and stacking.

"How well the stacking and ensemble approaches with models such as Decision Tree, Random Forest and K-Nearest Neighbor compare to each other in terms of accuracy, F1 Score, Precision and Recall in Predicting Mortality in Vehicle Collisions in New York City?"

The objective of this Research is to evaluate the performance of both ensemble and stacking approaches to improvise the prediction models for mortality in motor collisions in New York. This Research aims in comparing the Accuracy, F1 Score, Recall and Precision of ensemble and stacking approach of these three machine learning models: Decision Tree Classifier, Random Forest and K-Nearest Neighbor. By implementing and analyzing these algorithms in combination of ensemble and stacking this Research aims to determine which approach yields robust predictions. The Research uses datasets from New York government website containing two datasets crashes and person which contains the data about the crashes and the person dataset contains the data about the person involved in the collisions. The Dataset undergoes various data processing stages such as Data cleaning, Exploratory Data Analysis, Pre-processing of data.

2 Related Work

The Development of Collision Prediction model is a complex aspect of road safety design due to various factors such as location, environmental characteristics, human behaviour and vehicle characteristics. Building and Implementation of Collision Prediction Model can help forecast traffic collisions in specific region and collect the information for the users who are nearby in that location which makes them to be precarious measures. Most of the existing approaches used in the prediction of motor collision as classification problem where the crash severity will be the target variables. This Section reviews about the previous literature by highlighting their contributions in the data mining research.

2.1 Analytical Techniques Used in Predicting Crash Severity

One of the studies examined around 900 accidents on the N5 National Highway in Bangladesh. They used different decision tree induction algorithms to find out the traffic accidents trends and patterns and following that derived norm from these trees and reduced the road accidents on the highway as mentioned by (Satu et al., 2017). Many studies have been investigated the use of Decision Tree and random forest in predicting the injury severity since there are numerous numbers of contributing factors in road accidents. The author (Sharma et al., 2016) has modelled the traffic collision injury as a classification problem using Decision Tree Classifier and gaussian kernel.

(Delen et al., 2017) compared and analyzed the prediction performance of Artificial Neural Network, Support Vector Machine and Decision Trees in predicting the crash severity with the data of the national automotive sampling system and the results showed that the Decision Tree predicted the best with good accuracy. Machine learning techniques have been

primarily been used for forecasting traffic collisions owing to their foreseeing capabilities and their capability to handle complex and multi-dimensional data. The researchers in the study used predetermined set of conditions or the features to predict traffic collision situations which enables the analysis and development models for predicting traffic collision fatalities by (Lord and Mannering, 2010). Various Data visualization methods have been applied to uncover the significant patterns and trends within the traffic collision data.

Most of the traditional methods and to classify the seriousness of the injuries used statistical techniques such as Poisson, Binomial and other statistical techniques. These techniques have some limitations because each of the models includes its predefined correlations between the independent and the target variables (Qiang Zeng, 2016). Machine Learning techniques like support vector machines, K-Nearest Neighbor and Decision Trees have been used to analyze the variety of road collision problems and is considered as most common analytical methods due to the capacity of those to handle large volumes of data. In Addition to that the flexibility of the machine learning techniques and the generalization of these models led to accept as a accurate and generic model in the domain of road safety (Islam et al., 2022).

2.2 Overview of Machine Learning Techniques and Algorithms

According to the study (Jian Zhang, 2018) which is done on multi class classification problem of predicting a crash severity comparing statistical and machine learning techniques. The Statistical technique used two common model which are order probit model and multinomial logit model. Comparing with four machine learning algorithms K-Nearest Neighbor, Random Forest, Decision Tree and Support vector Machine. The results of the machine learning algorithms were better in terms of predicting the crash severity comparing to the statistical techniques. The Random Forest and K-Nearest Neighbor had the best prediction in the machine learning techniques.

This Study which was done by (Evwiekpaefe and Umar, 2022) in predicting crash severity in Nigeria. This Study used five different algorithms like K-Nearest Neighbor, Naive Bayes, Decision Tree, JRIP and Multilayer perceptron. The target column is a multi class variables contained three classes serious, fatal and minor injury severity. The K-Nearest Neighbor came out as the best model with an accuracy score of 94.8% wihtout applying feature selection and 96.1% after applying feature selection techniques.

Stacking is a technique which is used to create a strong model that is aimed at performing well and generalizes the model by minimising the bias in the data. This approach involves the combination of different sets of classifiers. To achieve a good accuracy, a learning algorithm is applied. The base learner is trained using the original dataset. The predictions which are made by these base learners acts as a new input data. Another learning algorithm is then applied which is the meta learner which is then trained using the new input data and the outputs is generated corresponding to it to make the final prediction. This was introduced by (David et al., 1992).

The study (Tang et al., 2019) which was done in prediction of crash severity on 5538 crashes which used stacking technique to achieve better prediction results. The study used Gradient Boosting, Ada boosting and Random Forest as the base learner. The second learner used Logistic Regression in identifying the crash severity. The prediction result

showed stacking technique performed better while evaluating with evaluation metrics like recall and accuracy. The base learners are trained using the training dataset and then the inputs for the second model is the outputs of the individual models.

This study proposed by (Hussain and Ashraf, 2023) applied ensemble learning like random forest, voting classifier and compared with algorithms like Decision Tree and KNN. They applied on predicting crash severity with multi class classification variables. A Study done by (Chandra et al., 2021) shows use of feature selection and ensemble learning has good accuracy compared to individual models. Bagging classifier is applied to achieve high accuracy. These models were applied to an imbalanced dataset.

From the above inputs the bagging classifier can perform better on multi-class classification problem comparing to support vector machine due to its handling of high volume of data, robustness to noise and imbalanced data. The method an ensemble technique which combines multiple weak learners and it captures complex patterns, linear and non-linear relationship between the data. It works well where there feature selection and interpretation of the data is important since it can highlight the feature importance of the dataset. The stacking is another ensemble technique where the predictions of one algorithm is taken as input to the meta-learners and the final prediction is obtained.

2.3 Key Findings from the Literature

Most of the techniques which are used in prediction of the traffic collision involved statistical techniques and some of the researchers failed to apply feature engineering techniques by deriving a new feature from the existing feature. Also, this research uses stacking technique in prediction of accuracy. This study uses stacking technique for which the models which are chosen are compared with each other where the above researches failed to implement in their study. The base models and the meta models are changed by applying all the three meta models and the base models from the models which are chosen. The above researches applied stacking technique and didn't compare to each other by changing the base and the meta models which will be done in this study.

This Study uses various pre-processing techniques, feature selection and feature engineering to derive new features from the existing features and also to identify anonymous patterns in the dataset. This Research also aims in identifying the important factors which has high influence in the prediction of mortality rate with the help of the classification of Injury Severity.

3 Methodology

This section is about the detailed overview of the research methodology. This section outlines each step which is necessary for implementing this research successfully and gives a technical view for the methodology. The objective of the research is to identify the collision patterns and to identify the important features which has huge impact of the person's mortality. This Research uses KDD (Knowledge Discovery Database). The Methodology section begins with pre-processing of data which includes data cleaning, normalization of data, identifying missing values to make sure the data quality. It involves applying feature selection and feature engineering are applied to identify and derive new features from the existing features to enhance the model's performance.

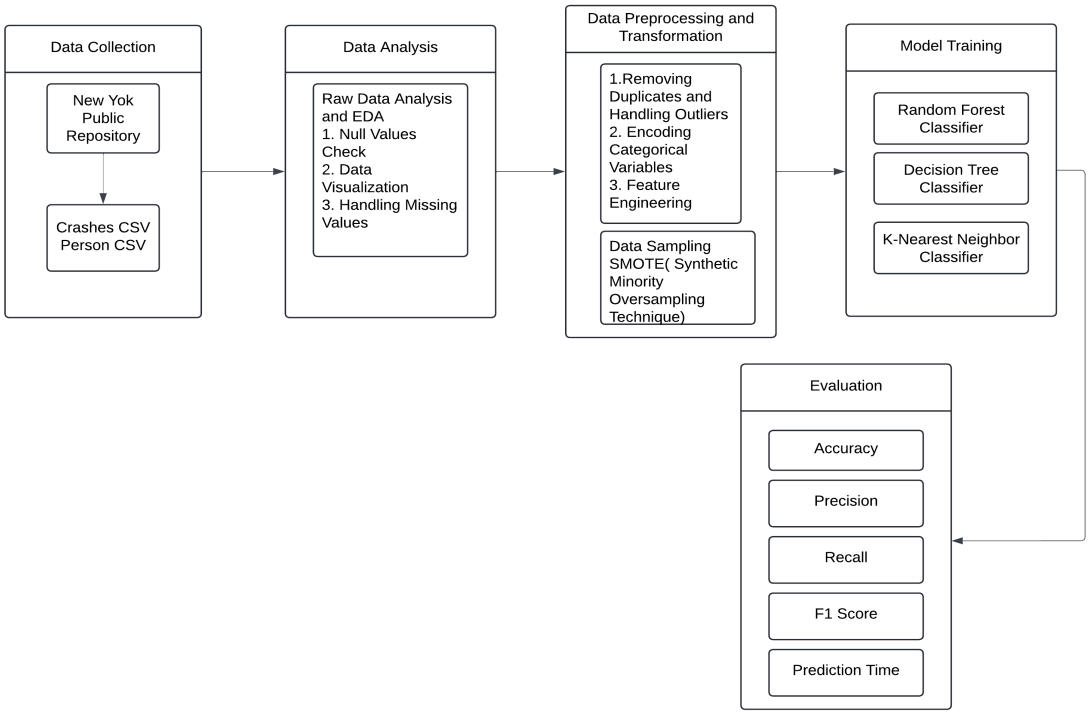

Figure 1: Research Methodology

3.1 Selection of Data

The Exploration of data and the thorough understanding of the domain are the important steps in finding any solution to the real world problems. The domain understanding helps in comprehending the problem. The mortality increase in the traffic collision is a significant challenge the humanity faces. The dataset used for this research contains data about two the data about the crashes and the data about the persons. The data is downloaded New York government website a public repository (City, 2024). Identification of factors which causes proportional impact on the mortality or crash severity of the person. These features helps this research in predicting the crash severity of the person.

3.2 Data Exploration

The Exploration of data involves finding relevant data sources for the study. The next step involves identifying tools which can be used to continue this study such as programming language, visualisation tools or libraries. This research uses python as the programming language and libraries like pandas, numpy, matplotlib and other libraries to perform data pre-processing steps.

- Crash Data: This Dataset contains data on the crash type, contributing factors of the vehicle, location of the collision, number of persons injured,killed in the collision and crash date and time from 2012 to 2024.
- Person Data: The person Dataset contains the data about the persons involved in

[0]https://catalog.data.gov/dataset/motor-vehicle-collisions-crashes
[0]https://catalog.data.gov/dataset/motor-vehicle-collisions-person

the crash. The features include person type, person injury, person age, ejection status, bodily injury, safety equipment, emotional status and contributing factors.

3.3 Data Pre-Processing

The Crashes data contains 20 columns and the person's data contains 21 columns both of the datasets are combined using a common column name collision ID which is there is both the datasets. The initial stage of data pre-processing where the data is selected and cleaned before performing Exploratory Data Analysis and Model Building as shown in Figure 1. This initial step is done to filter out the columns which are not necessary for problem statement and removing the redundant data which affects the model's performance. Since, the dataset is taken from a public repository website it contains the dataset contains many missing values and null values which lets it allow to process raw data. The dataset is loaded using python by using pandas library to read the csv file into dataframe. Most of the columns with duplicate values and redundant data has been removed prior in performing Exploratory Data Analysis and Model Building. The below figure shows the columns that are used for performing EDA and Model Building.

3.3.1 Data Cleaning

```
Data columns (total 15 columns):
 #   Column                          Dtype
---  ------                          -----
 0   BOROUGH                         object
 1   NUMBER OF PERSONS INJURED       float64
 2   NUMBER OF PERSONS KILLED        float64
 3   CONTRIBUTING FACTOR VEHICLE 1   object
 4   CONTRIBUTING FACTOR VEHICLE 2   object
 5   CRASH_DATE                      object
 6   CRASH_TIME                      object
 7   PERSON_INJURY                   object
 8   PERSON_AGE                      float64
 9   EJECTION                        object
 10  EMOTIONAL_STATUS                object
 11  BODILY_INJURY                   object
 12  POSITION_IN_VEHICLE             object
 13  SAFETY_EQUIPMENT                object
 14  PERSON SEX                      object
```

Figure 2: Filtered Columns

Data Cleaning is a significant step in data pre-processing where it involves identification of outliers in the data, correction of errors in the data, handling inconsistencies in the data for data quality and reliable data. This process also includes removing duplicate values in the dataset, filling or removing missing values, outlier detection where leaving it may skew the analysis process. Effective cleaning of data helps in improving the accuracy of the machine learning models and ensures that the insights gained from the data are valid and actionable. Also, Data cleaning is the most time consuming task in the entire process of data pre-processing process.

From Figure 2 the target column is the column named PERSON INJURY column which has the data of the crash severity of the person having three categories Alive, Injured and Killed. The Borough column is a categorical data that represents the location of the

crash took place, Number of persons killed and injured columns represents the number of person involved in the crash and whether they are injured or killed in the crash, Contributing factor 1 and Contributing Factor 2 both of the columns are categorical values which represents the contributing factors of the 1st and the 2nd vehicle which made to happen a collision, the Crash Date column represents the date at which the crash is recorded, crash time column shows the time at which collision happened, the Ejection column contains categorical data where it represents whether the person is thrown from the vehicle during the crash or not, Person age column describes the age of the person involved in the crash, Emotional status columns represents the emotional status of the person at the crash, Bodily Injury column shows the injury place in the person's body, position in the vehicle column contains categorical variable and represents the position of the person in the vehicle such as driver, passenger at back seat, middle seat and front seat, Safety equipment column too contains categorical values where it tells the type of safety equipment used by the person during the collision and Person sex column shows the gender of the person.

BOROUGH	NUMBER OF PERSONS INJURED	NUMBER OF PERSONS KILLED	CONTRIBUTING FACTOR VEHICLE 1	CONTRIBUTING FACTOR VEHICLE 2	PERSON_INJURY	PERSON_AGE	EJECTION	EMOTIONAL_STATUS	BODILY_INJURY
BRONX	2.0	0.0	Unspecified	Unspecified	Injured	41.0	Not Ejected	Shock	Chest
BRONX	2.0	0.0	Unspecified	Unspecified	Injured	20.0	Not Ejected	Shock	Head
MANHATTAN	0.0	0.0	Passing Too Closely	Unspecified	Alive	37.0	Not Ejected	Does Not Apply	Does Not Apply
MANHATTAN	0.0	0.0	Passing Too Closely	Unspecified	Alive	22.0	Not Ejected	Does Not Apply	Does Not Apply
QUEENS	0.0	0.0	Turning Improperly	Unspecified	Alive	25.0	Not Ejected	Does Not Apply	Does Not Apply

Figure 3: Head of DataFrame

POSITION_IN_VEHICLE	SAFETY_EQUIPMENT	PERSON_SEX	Year	Day
Driver	Lap Belt & Harness	F	2021	Tuesday
Driver	Lap Belt & Harness	M	2021	Tuesday
Driver	Lap Belt	M	2021	Tuesday
Driver	Lap Belt	M	2021	Tuesday
Driver	Lap Belt & Harness	M	2021	Tuesday

Figure 4: Head of DataFrame

The dataset is further analysed where the analysis includes applying statistical techniques to the numerical columns for outlier detection. The Null and the missing values from the dataset is removed. Most of the outlier detection uses box plot or histogram to check

the distribution of the data. The below histplot shows the distribution of the person age column. The person age indicates the age of person who is involved in the crash.

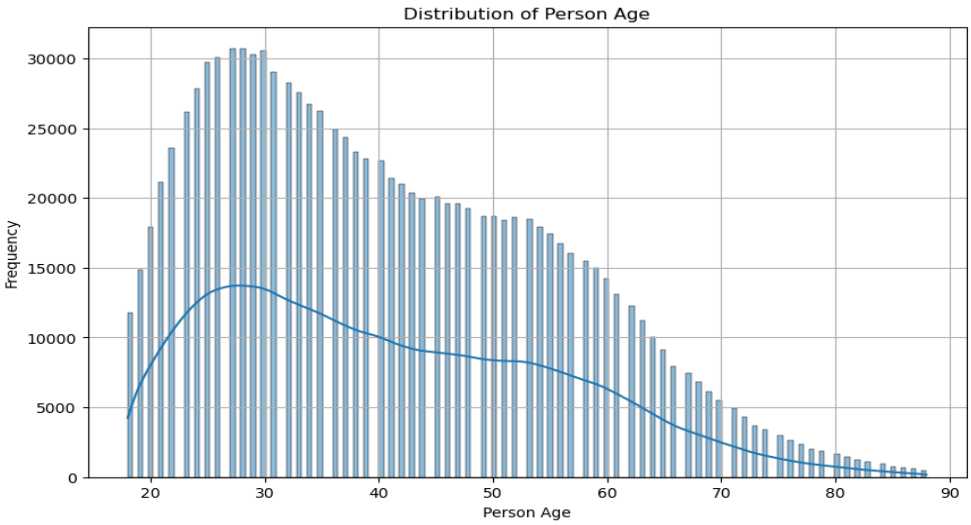

Figure 5: Distribution of the Person Age

The above hist plot shows that there is age distribution above after identifying and removing outliers in that column. The outlier is removed by applying the Interquartile Range. The Interquartile range, is calculated for identifying the middle 50% of the data.

$$\text{IQR} = Q_3 - Q_1$$
$$\text{Lower Bound} = Q_1 - 1.5 \times \text{IQR}$$
$$\text{Upper Bound} = Q_3 + 1.5 \times \text{IQR}$$

The Q1 is the age below the 25% of the data falls and the Q3 is the age, below where the 75% of the data falls. The difference calculated, between them is the Interquartile Range. The standard process of defining an outlier is multiplying the Q1 and Q3, with 1.5 value to get the lower and the upper bounds of the data points.

Handling Missing Values

Handling of missing values is a crucial step in data pre-processing where it has a major impact on the quality and the reliability of the analysis and affects the model's performance. Some of methods used in handling missing values are deletion of the missing or the null values and filling the missing values with the most common values repeated in the column. The deletion of the values can have significant loss in data if the data is crucial in terms of analysis and model building. Imputing is another method where the missing or the null values is replaced by statistical formulas such as mean, median or mode. The Contributing factor 1 and 2 columns has a value named unspecified where that value is filled using mode operation. The most frequent value is then filled in the place of the unspecified value. The missing values in the Ejection column is in negligible so the values were removed.

3.3.2 Feature Engineering

Feature Engineering is an important step in data analysis where a new feature is derived from an existing feature or the existing feature is modified which helps in the analysis and also improves in prediction of machine learning model. By selecting and creating features feature engineering aims in enhancing the model's ability to capture the patterns in the data which allows the model to predict with high accuracy. The Crash dataset has a column named crash date with the date as string format.The crash date column is converted into date time format and the month and the year features are derived from the crash date. The new features are derived to perform Exploratory Data Analysis to analyse the number of accidents occurred during the month or year.

3.3.3 Exploratory Data Analysis

Exploratory Data Analysis involves analysing of the motor collision dataset to identify patterns, relationships and insights that helps the modelling process better. The First Phase of the process involves identifying key features of the dataset such as age, bodily injury, safety equipment, gender, contributing factor which influenced the collision to take place. It involves data visualization techniques as well to plot box and hist plots to understand the distribution of the data and identify the outliers or anomalies in the data. The categorical data such as safety equipment, location, body injury, contributing factor can be analyzed using bar charts. Usage of scatterplots and correlation matrices helps in identifying trends between the numerical columns.

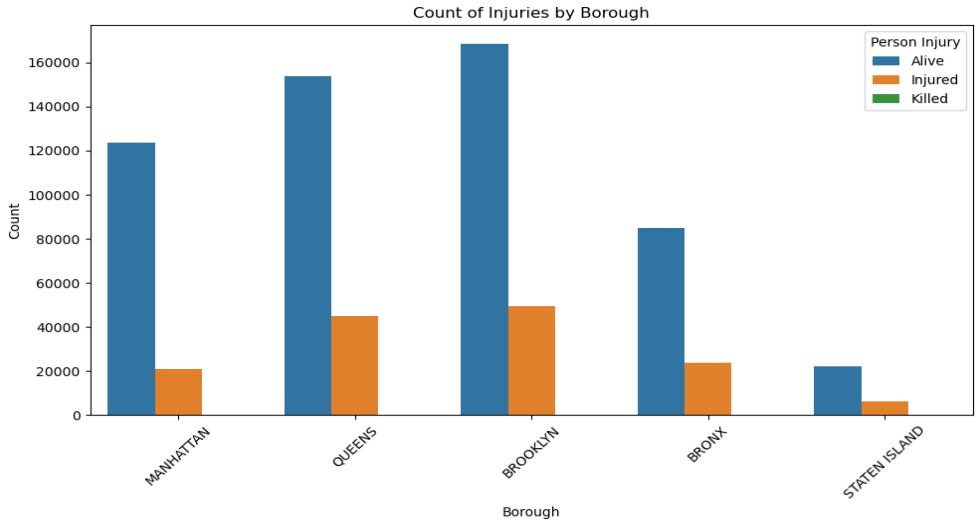

Figure 6: Collision by Borough

The above Figure 6 is a bar chart shows the counts of collisions happens per borough and it is evident from the above image that Brooklyn has the highest count of collisions followed by Queens and Manhattan. The Staten Island stands as the lowest in terms of the counts of collision. The above graph is plotted per borough using the crash severity of the people involved in the collision.

The above figure 7 illustrates the contributing factor which influence the collision to happen. The Most contributing factor which influences the collision is Driver Inattention/ Distraction having the most number of counts which caused collision. The second most

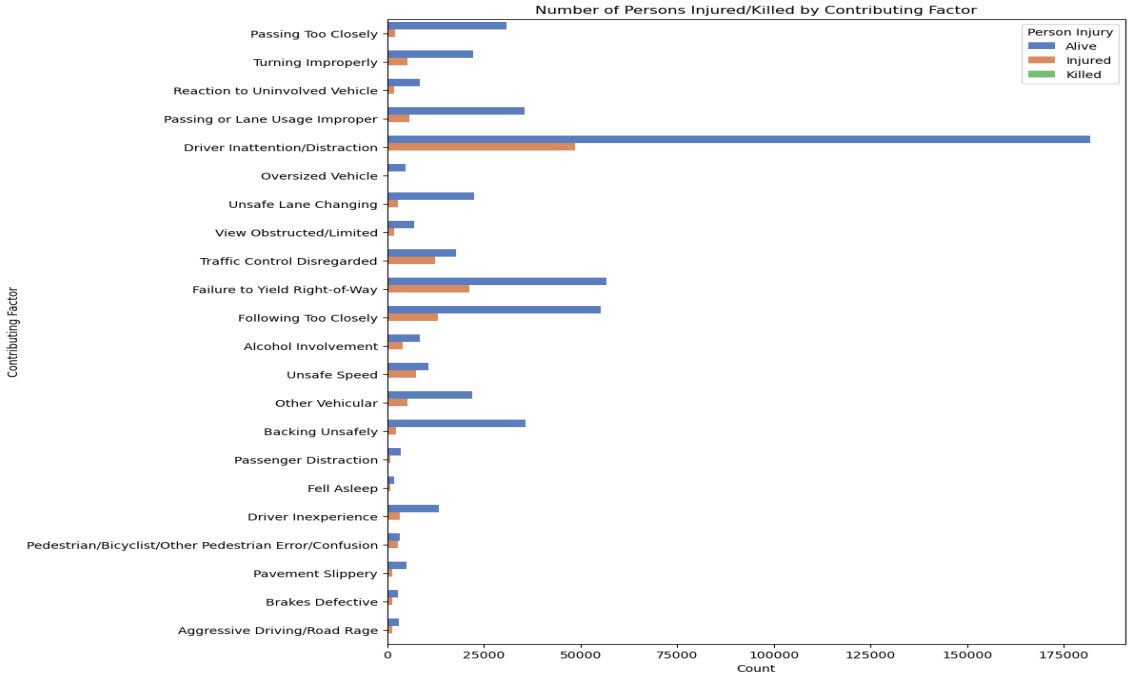

Figure 7: Contributing Factor Influencing Collision

common factors are Failure to yield right way and Following of vehicles too closely. Passing too closely, Improper lane usage and Turning vehicle improperly are also some prominent factors which influence the collision.

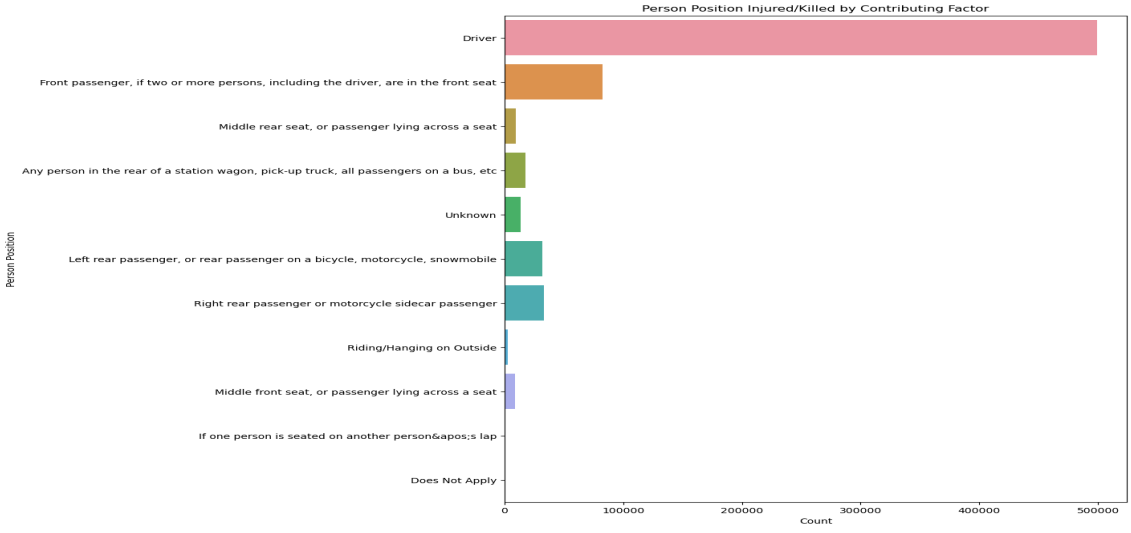

Figure 8: Position of Person Injured/Killed

The person's position in the vehicle involved in the collision is in the Figure 8 shows that the position of the driver tend to have more injuries than the people who are in different position in the vehicle. Front seated Passengers tend to get affected the second most after the position of the driver. From the above it is evident that the people who are positioned in the front of the vehicle tend to have high impact. The people in the rear seat have some risk but it is less compared to the people who are positioned in the front of the vehicle.

3.3.4 Using SMOTE(Synthetic Minority Oversampling Technique) for Data Balancing

The Dataset used for this study has imbalanced data in the target column Person Injured. The category killed is very low in count in the dataset. SMOTE is used in oversampling the reduced data in the target column before model building. SMOTE is a oversampling technique used in machine learning to address the issue of class imbalance in the data. SMOTE involves synthetic generation of minority class instead of replacing the existing samples which done will increase the chances of over fitting of the model. To overcome the over fitting of the model SMOTE is proposed (Chawla, 1970). The data set is split into training and testing and then the SMOTE is applied.

4 Design Specification

The Prediction of Mortality in Motor Collision involves series of steps. The data sources consist of two files namely Crashes.csv and Person.csv. Both the files are merged using the common column named COLLISION ID column. Initial analysis is done to comprehend the dataset which includes identification and handling of missing and null values. Feature Engineering is performed to extract the new features from the existing features. This section contains Algorithms and methods used in this research.

Exploratory Data Analysis is performed once the data cleaning is done using the dataset. The Categorical variables are encoded and the target and the independent variables are split using X and y in the ratio of 70:30 for Training and Testing. Once the training and testing is split SMOTE is used in oversampling the imbalanced data. The model used in this research includes Random Forest Classifier, K-Nearest Neighbor and Decision Tree Classifier.

4.1 Machine Learning Approaches

The Research compares the Ensemble and the Stacking Approach where the models are trained by Ensemble and Stacking. The Models are trained Individually and then they are trained in Ensemble approach by using Voting Classifier. In Simple Terms the Majority Voting Classifier is where each classifier is given equal weights and casts only one vote per classifier. The final prediction of the classifier is determined by majority of the votes which in terms the class which receives the majority votes becomes the final prediction. Let the n represent the number of classifiers each of the classifiers is represented as C_t and the ensemble E=C_1,C_2..C_n. The decision will give d_t, j 0,1 where j= 1,2..k and k represents the number of classes. The decision will give $d_t, j = 1$ if t-th classifier selects the class C_j and d_t, j is 1 else d_t, j returns 0 (Dogan and Birant, 2019). The Flow Chart for ensemble approach is shown in Figure 9. The Models used in this research are Random Forest Classifier, Decision Tree Classifier and K-Nearest Neighbor.

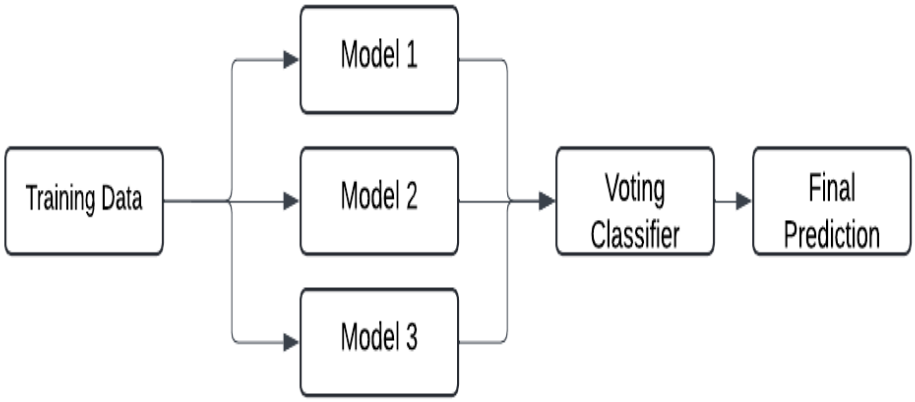

Figure 9: Flow of Ensemble Approach

The Following combinations are used in Ensemble Approaches:

- Ensemble Approach using Voting Classifier by using all the models
- Ensemble Approach using Voting Classifier with Random Forest and Decision Tree
- Ensemble Approach using Voting Classifier with K-Nearest Neighbor and Decision Tree
- Ensemble Approach using Voting Classifier with Random Forest and K-Nearest Neighbor

Stacking Generalization which is also called as stacking is a type of ensemble method where that involves one or multiple base learners to train a base model, meta learner which learns how to combine the predictions of these learners. In a stacking framework, base learners are first trained independently on the dataset, and their predictions are then used as input for a meta-learner, which makes the final prediction.

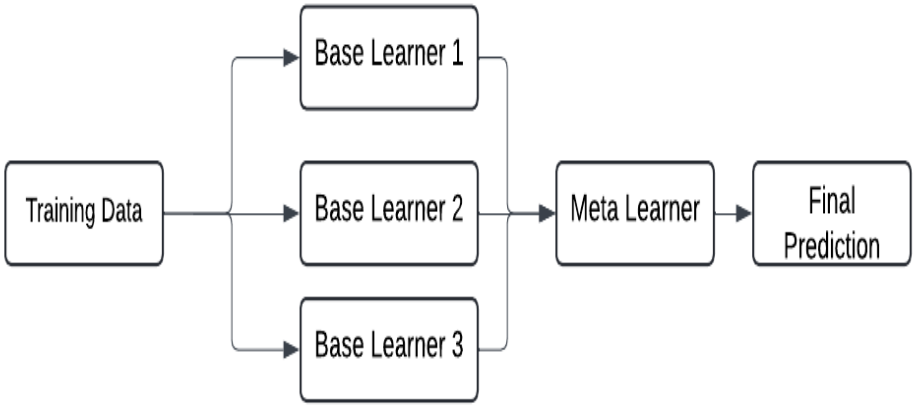

Figure 10: Flow of Stacking Approach

When shuffling base and meta learners, different models are tried in both the base and meta levels to find the best combination. In this case, Random Forest Classifier, Decision Tree Classifier, and K-Nearest Neighbors (KNN) are used as models. For example, you might use a Random Forest and KNN as base learners and a Decision Tree as the meta-learner. Alternatively, the roles could be reversed, using a Decision Tree and KNN as base learners with a Random Forest as the meta-learner. The following are the combinations of stacking approaches:

- Stacking approach using random forest as meta learner classifier by using all the other as base models.
- Stacking using decision tree as meta learner and random forest classifier as base learner.
- Stacking using K-nearest neighbor as meta learner and decision tree classifier as base learner.
- Stacking using random forest classifier as meta learner and K-nearest neighbor as base learner.
- Stacking using random forest classifier as meta learner and decision tree classifier as base learner.
- Stacking with K-nearest neighbor as meta learner and random forest classifier as the base learner.
- Stacking decision tree classifier as the meta learner and K-nearest neighbor as the base learner.

The privilege of shuffling these models, is that it allows the strength of the each model to complement the others. Random Forests are known for their robustness and ability to handle over-fitting of the training data which captures complex relationships in the data. Decision Tree provide better interpretability and simple decision-making while KNN works well in cases where, the data structure is simple and local relationships are significant. With different configurations of these models in the base and meta layers, stacking can potentially deliver better predictive performance instead of individual model alone.

5 Implementation

This section is about the tools and methods used in this research. Python programming language is used in this research for building machine learning models. Various libraries were used in building these machine learning models. Libraries like sklearn, K-neighbors classifier, random forest classifier, decision tree classifier, stacking classifier, voting classifier were used in building these models. Jupyter notebook is used as the execution model to run the python codeartefacts.

The dataset contains two files crashes and person which are taken from new york government public repository. Both the datasets are merged using a common column in both the datasets named collision id. The data is then loaded into a dataframe using pandas library. Data cleaning is performed using pandas and numpy libraries to check null and missing values in the dataframe. Some of the columns where data is more crucial the

null values are replaced with the mode value of the column. Outliers in the person age column is removed by using the Interquartile range. After removing the outliers hist plot is used to check the distribution of the age column. Different count plots were generated using matplotlib package in analysing different features of the dataset.

Feature engineering is applied to the crash date column of the dataset to extract the features such as day, year and month. Count plots is generated based on the borough to understand the counts of collisions happening per borough. The ID columns such as vehicle id, crash id, unique id, person id and other factors which don't have value to the dataset are dropped using the drop column function. EDA is performed on the cleaned dataset to visualise the data in count plots and other bar charts to understand the data in more detail. The dataset contains imbalanced data, oversampling techniques like SMOTE(synthetic minority oversampling technique) is used to balance the data for better model training and prediction.

The category columns in the dataset are encoded using label encoder before splitting the data into training, testing and model building. The data is split into 70:30 where 70 is the training split and 30 is the testing split. Random forest, K-nearest neighbor and decision tree are the models used for this research. This research focuses on comparing stacking and ensemble approaches by shuffling the base learners and the meta learners in the stacking approach and applying a combination of models in the ensemble approach. Evaluation metrics such as accuracy, f1score, precision and recall are used to evaluating and comparing them with each other to get the best accuracy of the models. Confusion matrix is plotted to evaluate the accuracy of the classification models, the matrix plots between true positive and negatives, false positives and negatives.

6 Evaluation

Evaluation metrics is crucial in assessing the performance of the model which provides insights in how well the model is able to predict. This section gives overview of the approaches of machine learning used and their evaluation metrics used in assessing them. Several evaluation metrics such as accuracy, precision, recall and f1 score were used to evaluate the model's performance. Confusion matrix is plotted to evaluate the actual and the predicted values of the model. This has two subsections discussing both ensemble and stacking approaches of machine learning.

- Accuracy: It is an evaluation metric used for assessing both regression and classification model. It calculates the ratio of correctly predicted values to the total number of values in the set.
- Precision and Recall: Precision is a ratio which is used for calculating correctly identified positive values to the total number of positive values. It calculates the accuracy of the positive predictions by the model. Recall or sensitivity is the ratio of calculating the true positive values to the actual number of positive values.
- F1 Score: This score is used to calculate a balance between both the precision and recall. It can also be called as the harmonic mean of precision and recall where it make sure both the false positive and the false negatives are accounted in the evaluation of model's performance.

- Confusion Matrix: Confusion matrix is used in assessing the classification model's performance. It plots a matrix between the actual and the predicted values. The values includes true positive and negatives, false positives and negatives.

6.1 Case Study 1: Ensemble Approach

The first case study is about the ensemble approach where the voting classifier is used to get the majority voting prediction. The voting classifier has all the three models the number of neighbors is 5 for k-nearest neighbor, number of estimators is given as 100 and max depth is 10 for random forest and random state is given as 42 for both random forest and decision tree for reproducibility of the results. The voting parameter is set as hard in the voting classifier function. The model which used all the three models as estimators gives an accuracy of 98.9%. There are series of combinations used in the ensemble approach.

Models	Accuracy (%)	Time (s)	Category	Precision	Recall	F1 Score
Ensemble with all the models	98.90%	35.98	Alive	0.99	0.99	0.99
			Injured	0.97	0.98	0.97
			Killed	0.54	0.72	0.62
Ensemble with KNN and random forest	96.78%	34.74	Alive	0.96	1	0.98
			Injured	0.98	0.86	0.92
			Killed	0.39	0.2	0.26
Ensemble with KNN and decision tree	96.75%	2.26	Alive	0.96	1	0.98
			Injured	0.99	0.86	0.92
			Killed	0.27	0.16	0.2
Ensemble with random forest and decision tree	98.93%	33.25	Alive	0.99	0.99	0.99
			Injured	0.97	0.98	0.97
			Killed	0.76	0.68	0.72

Figure 11: Evaluation results of all the ensemble approaches

6.1.1 Voting classifier with Knn and random forest as base models

The base models used in this includes K-nearest neighbor and random forest where the voting parameter is hard of the voting classifier. The accuracy of this model's combination is 96.78%. The model took 34s for training. This model has the highest recall score compared to other model's. The voting parameter hard specifies that the final prediction is determined by the majority class predicted by the base models. This model performs well in predicting class alive and injured with a higher precision value but has less recall value compared to class alive.

6.1.2 Voting classifier with Knn and decision tree as base models

This section consists of the combination of knn and decision tree classifier as the base models with the same voting parameter of the voting classifier. The accuracy of this combination came out to be 96.7% which is less compared to the previous combination but the training time of the model is 2.26s which makes it computationally effective compared to the previous one. The number of neighbors used for knn in this combination is 20. The max depth of the decision tree classifier is none so that the nodes of the decision tree

are expanded till all the leaves are pure. The decision tree parameter criterion is 'gini' by default.

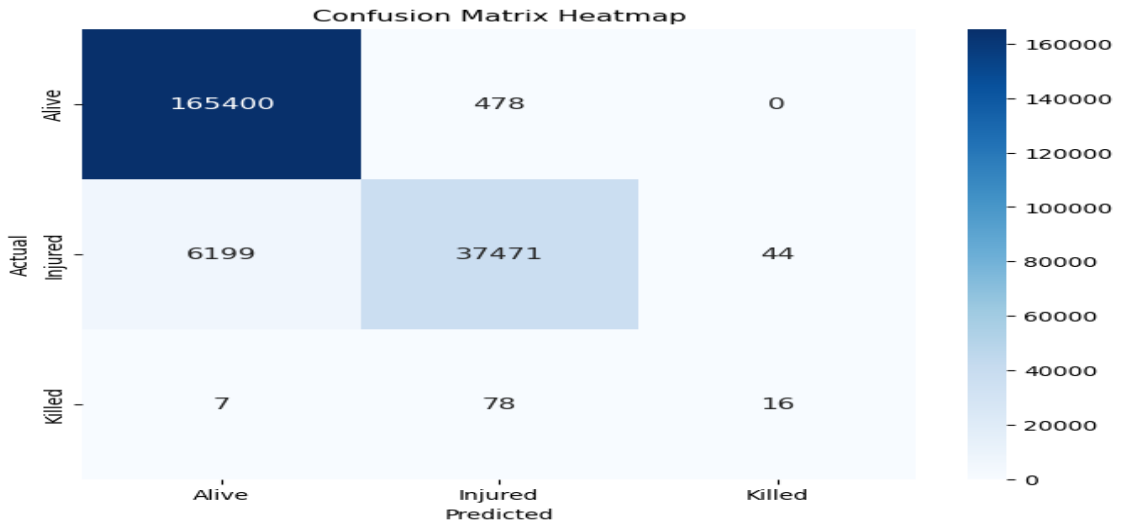

Figure 12: Confusion matrix of knn and decision tree as base models

From the above figure 12 this model has poor performance in predicting the minority class variables such as injured and killed. It incorrectly predicted 6200 values of injured as alive. The precision and recall score calculated for the minority values are 0.27 and 0.16 which shows the model's poor performance in identifying imbalanced data even after applying some oversampling techniques such as SMOTE.

6.1.3 Voting classifier with random forest and decision tree as base models

Random forest and decision tree are the base models used in the third combination. The accuracy score for this combination is 98.93%. Both the precision and recall values of this model are high for the minority class compared to other models. This is evident that this model performs well on predicting minority class. This model has robust performance on predicting the majority class since the value of precision and recall is 0.99.

6.2 Case Study 2: Stacking Approach

This case study is stacking approach where the models knn, random forest and decision tree are shuffled between meta learner and base learner models. Random state is assigned as 42 for all the models for reproducibility of the results. The cross validation split value is given as 5 where the data is split into five folds. For each of the folds the models are trained on cv1 and the predictions are made using the remaining folds. The process is done for all the folds and the predictions from the all the folds are aggregated. The aggregated predictions from the base learners are given as input to the meta learner model.

6.2.1 Stacking using random forest classifier as meta learner

In this section random forest classifier is used as the meta learner and base learners are shuffled between the other models. The evaluation score for the stacking combinations

are in the figure 13 which shows all the evaluation metrics for the models. This section has random forest classifier as the meta learner and knn as the base model. The accuracy score for this model is 98.03%. The base model knn is trained with 3 neighbors and the number of estimators is given as 100 for the meta learner. Stacking classifier library is used. The recall and precision score for alive is predicted high and the killed category precision and recall score are 0.05 and 0.07 which is low compared to other categories. The other combination which has decision tree as the base model's accuracy is 98.94% which is better compared to the previous one. Also, the recall and precision scores of the category killed are 0.72 and 0.42 which shows that it performed well in predicting the minority class well than the previous combination. The training time between the models where random forest as the meta learner the one with the base model which has decision tree classifier runs faster and even predicts the minority class better than its peer.

Model	Accuracy	Time(s)	Category	Recall	Precision	F1 Score
Stacking with random forest as meta learner and knn as base model	98.03%	77.05	Alive	0.99	0.99	0.99
			Injured	0.96	0.95	0.95
			Killed	0.07	0.05	0.06
Stacking with random forest as meta learner and decision tree classifier as base model	98.94%	12.25	Alive	0.99	1	0.99
			Injured	0.99	0.97	0.98
			Killed	0.72	0.42	0.53
Stacking with decision tree as meta learner and knn as base model	97.90%	87.26	Alive	0.99	0.99	0.99
			Injured	0.95	0.95	0.95
			Killed	0.11	0.04	0.06
Stacking with decision tree as meta learner and random forest classifier as base model	98.80%	147.43	Alive	0.99	0.99	0.99
			Injured	0.97	0.98	0.97
			Killed	0.82	0.66	0.73
Stacking with knn as meta learner and decision tree classifier as base model	98.86%	5.27	Alive	0.99	0.99	0.99
			Injured	0.98	0.97	0.97
			Killed	0.01	1	0.02
Stacking with knn as meta learner and random forest classifier as base model	98.95%	150.41	Alive	0.99	1	0.99
			Injured	0.99	0.97	0.98
			Killed	0.64	0.83	0.73

Figure 13: Evaluation results of stacking approaches

6.2.2 Stacking using decision tree classifier as meta learner

This section contains decision tree classifier as the meta learner and the other models as base learners. The model with the base learner random forest classifier has the best accuracy comparing between them with 98.80% and is better in predicting the minority class like injured and killed with a recall and precision score of 0.82 and 0.66 which is better than all the stacking combination of models. The below is the feature importance of the base model random forest classifier. The training time is a bit high than its peer but this model has good accuracy in predicting both majority as well as minority classes.

From the above figure the feature importance of the random forest classifier as base model shows that it weighs the bodily injury, emotional status and ejection status as the most important feature in predicting a mortality in motor collision.

6.2.3 Stacking using k-nearest neighbor as meta learner

The final combination is using knn as the meta learner. The best accuracy between the two where the base models random forest and decision tree are interchanged while running the stacking models by using only base model. The highest accuracy score is 98.95% where the random forest classifier as the base model. Also, this combination predicts the minority classes with good accuracy than the other one where the base

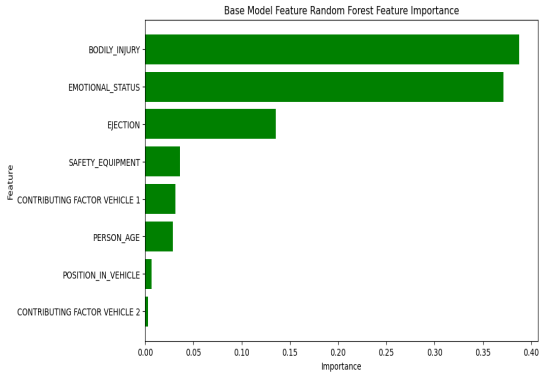

Figure 14: Feature importance of random forest classifier

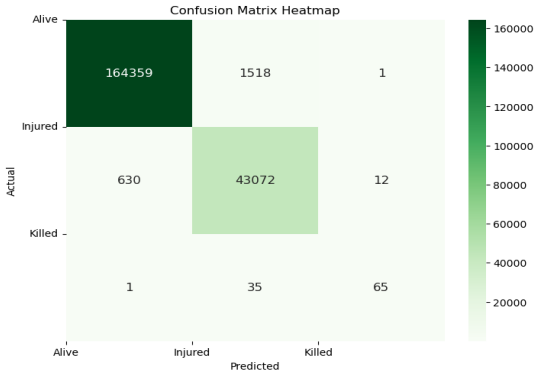

Figure 15: Confusion Matrix of random forest as base and decision tree as meta

learner is decision tree classifier. The base model with decision tree classifier trains faster than the one with random forest base model.

6.3 Discussion

From the above comparison of machine learning approaches of both ensemble and stacking. A series of combination of models is applied for both the approaches where applying of oversampling technique like SMOTE helped in increasing the minority samples of the target data to help the model perform better in training and predicting. In the above combinations where ensemble approach is used the combination with decision tree classifier and random forest classifier has the highest accuracy of 98.93% than the rest of other with a training time of 33.25s. It also predicts the minority class like injured and killed with a good recall and precision score with 97 and 98 for the injured category and 0.76 and 0.68 for the killed category.

The combination of stacking approach with base model of random forest classifier and meta learner with knn has the best accuracy of 98.95%. This model predicts well on the minority class as well on the categories of injured and killed. The precision and recall score of 0.83 and 0.64. From the above approaches the meta model with knn and base learner with random forest performs better in terms of predicting the minority class better than the best model of the ensemble approach with a precision and recall better than the best model of ensemble approach. Also, the feature importance derived from the base models shows the features of bodily injury, emotional status and ejection has the highest weightage in predicting mortality of person.

7 Conclusion and Future Work

This study compared both the ensemble and stacking approaches for predicting the mortality in motor vehicle in new york. Both the approaches well on predicting the mortality of person with high accuracy and performs better in predicting the minority class too. The ensemble approach with random forest and decision tree showed strong predictive capabilities by aggregating the predictions of the weak learners. The stacking approach which trained a meta learner with the base learners predictions as input showed improved precision and recall values of the minority class compared to the ensemble approach. It

showed high accuracy in predicting the minority as well as the majority class. Stacking can leverage the strength of each model in a supportive way.

Future studies could explore by using diverse range of models such as support vector machines, neural networks and advanced classifiers. With the experimentation of diverse range of models enhancing the predictive performance might be possible. Choosing a meta learner model is crucial in stacking applying different meta learners like boosting models and deep learning models may improve results in prediction. Exploring hyperparameter tuning techniques to the meta learner in the model's optimisation. Different datasets containing different region data could help in building a robust model. Applying advanced feature engineering techniques could also help the model by giving more inputs that has significant information in imporving the prediction of the model.

References

Ahmed, S., Hossain, M. A., Ray, S. K., Bhuiyan, M. M. I. and Sabuj, S. R. (2023). A study on road accident prediction and contributing factors using explainable machine learning models: analysis and performance, *Transportation Research Interdisciplinary Perspectives* **19**: 100814.
URL: *https://www.sciencedirect.com/science/article/pii/S2590198223000611*

Bíl, M., Andrášik, R. and Sedoník, J. (2019). A detailed spatiotemporal analysis of traffic crash hotspots, *Applied Geography* **107**: 82–90.
URL: *https://www.sciencedirect.com/science/article/pii/S0143622818309081*

Chandra, S., Kaur, P., Sharma, H., Varshney, V. and Sharma, M. (2021). In-database analysis of road safety and prediction of accident severity, *in* D. Goyal, V. E. Bălaş, A. Mukherjee, V. Hugo C. de Albuquerque and A. K. Gupta (eds), *Information Management and Machine Intelligence*, Springer Singapore, Singapore.

Chawla, N. V. (1970). Data mining for imbalanced datasets: An overview.
URL: *https://link.springer.com/chapter/10.1007/0-387-25465-*$X_4$0

City, N. Y. (2024).
URL: *https://catalog.data.gov/dataset/motor-vehicle-collisions-crashes*

David, H and Wolpert (1992). Stacked generalization, *Neural Networks* **5**(2): 241–259.
URL: *https://www.sciencedirect.com/science/article/pii/S0893608005800231*

Delen, D., Tomak, L., Topuz, K. and Eryarsoy, E. (2017). Investigating injury severity risk factors in automobile crashes with predictive analytics and sensitivity analysis methods, *Journal of Transport Health* **4**: 118–131.
URL: *https://www.sciencedirect.com/science/article/pii/S2214140516302389*

Dogan, A. and Birant, D. (2019). A weighted majority voting ensemble approach for classification, *2019 4th International Conference on Computer Science and Engineering (UBMK)*, pp. 1–6.

Evwiekpaefe, A. E. and Umar, S. (2022). Predicting road traffic crash severity in kaduna metropolis using some selected machine learning techniques.
URL: *https://www.ajol.info/index.php/njt/article/view/225131*

Hussain, S. F. and Ashraf, M. M. (2023). A novel one-vs-rest consensus learning method for crash severity prediction, *Expert Systems with Applications* **228**: 120443.
URL: *https://www.sciencedirect.com/science/article/pii/S0957417423009454*

Islam, M. K., Reza, I., Gazder, U., Akter, R., Arifuzzaman, M. and Rahman, M. M. (2022). Predicting road crash severity using classifier models and crash hotspots, *Applied Sciences* **12**(22).
URL: *https://www.mdpi.com/2076-3417/12/22/11354*

Jian Zhang, Zhibin Li, Z. P. C. X. (2018). Comparing prediction performance for crash injury severity among various machine learning and statistical methods, *IEEE Access* **6**: 60079–60087.

Lloyd, D., Wilson, D., Mais, D., Deda, W. and Bhagat, A. (2015). Reported road casualties great britain: 2014 annual report.
URL: *https://trid.trb.org/View/1375096*

Lord, D. and Mannering, F. (2010). The statistical analysis of crash-frequency data: A review and assessment of methodological alternatives, *Transportation Research Part A: Policy and Practice* **44**(5): 291–305.
URL: *https://www.sciencedirect.com/science/article/pii/S0965856410000376*

Organization, W. H. (2023). Despite notable progress, road safety remains urgent global issue.
URL: *https://www.who.int/news/item/13-12-2023-despite-notable-progress-road-safety-remains-urgent-global-issue*

Qiang Zeng, Helai Huang, X. P. S. W. M. G. (2016). Rule extraction from an optimized neural network for traffic crash frequency modeling.
URL: *https://www.sciencedirect.com/science/article/pii/S0001457516303037*

Santos, K., Dias, J. P. and Amado, C. (2022). A literature review of machine learning algorithms for crash injury severity prediction, *Journal of Safety Research* **80**: 254–269.
URL: *https://www.sciencedirect.com/science/article/pii/S0022437521001584*

Satu, M. S., Ahamed, S., Hossain, F., Akter, T. and Farid, D. M. (2017). Mining traffic accident data of n5 national highway in bangladesh employing decision trees, *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pp. 722–725.

Sharma, B., Katiyar, V. K. and Kumar, K. (2016). Traffic accident prediction model using support vector machines with gaussian kernel, *in* M. Pant, K. Deep, J. C. Bansal, A. Nagar and K. N. Das (eds), *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, Springer Singapore, Singapore, pp. 1–10.

Tang, J., Liang, J., Han, C., Li, Z. and Huang, H. (2019). Crash injury severity analysis using a two-layer stacking framework, *Accident Analysis Prevention* **122**: 226–238.
URL: *https://www.sciencedirect.com/science/article/pii/S0001457518308546*

Wahab, Lukuman, Jiang and Haobin (2019). A comparative study on machine learning based algorithms for prediction of motorcycle crash severity.