Traffic Violation Arrests Using Machine Learning Approaches

MSc Research Project
MSc Data Analytics

Akanksha Shephali
Student ID: x23137720

School of Computing
National College of Ireland

Supervisor: Furqan Rustam

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Akanksha Shephali
Student ID:	X23137720
Programme:	MSc in Data Analytics
Year:	2024-2025
Module:	MSc Research Project
Supervisor:	Furqan Rustam
Submission Due Date:	29/01/2025
Project Title:	Traffic Violation Arrests Using Machine Learning Approaches
Word Count:	8809
Page Count:	24

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:	AKANKSHA SHEPHALI
Date:	29th January 2025

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):	

Traffic Violation Arrests Using Machine Learning Approaches

Akanksha Shephali
x23137720

Abstract

Offenses related to the roads are deemed a huge problem that affects public safety as well as the systems of using the roads. Traffic violation arrest prediction can help improve the implementation of the law due to proper planning and utilization of available resources. Conventional solutions to this problem involve either using people to go through the data sets and trying to identify the relevant patterns manually or designing a set of rules that can do the same thing but is not scalable. In this paper, different ML techniques will be discussed and examined such as Random Forest, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, and Linear Regression. Deep Neural Decision Forest, Multilayer perceptron, Recurrent neural network (RNN), LSTM, and Gated Recurrent Unit (GRU) are used as the deep learning approaches. Thus, in the data preprocessing, feature scaling was applied, and categorical features were encoded by using one hot encoding. Confusion matrix, accuracy, precision, recall, F1 score, and AUC ROC charts have been adopted with variance. Among the above-said ML models, Random Forest outperformed all the other models with an accuracy of 88% and the highest Recall and F1 Score of 80.21% and 34.15% respectively. However, GRU performed best among the deep learning models with an accuracy of 96.14% and Recall and F1 Score being 34.04% and 40.44%. The results presented here demonstrate that both types of models can help predict arrests for traffic violations which could aid police work by predicting when an arrest is likely to occur though it’s important to balance accuracy with transparency.

1 Introduction

Traffic violations leading to accidents are still considered a serious problem in the health sphere for people worldwide according to WHO’s 2023 Global Status Report on Road Safety. While there was a slight decline in the total annual rate to 1.19 million, traffic-relationship deaths remain to be one of the biggest sources of death among people within the 5 to 29 years age bracket indicating that the problem remains rampant. The latest WHO report shows that especially in LMICs, vulnerable road users share a large load; more than one-third of all road fatalities involve pedestrians, cyclists, and motorcyclists Behboudi et al. (2024). Li et al. (2021) state that a study was done for three months and 21,525 violations were detected at the intersections under the AES system. In urban regions, arrests for traffic offenses may be dangerous to the roads' well-being and proper traffic flow. Even though there are certain traffic rules, laws, and regulations that have been put in place there is a high range of traffic violations and hence accidents and many cases of injuries, driving under the influence of alcohol/ substance, the extreme being deaths. Perhaps Law enforcement can afford better participation in enhancing the whole process and resource allocation at the same time while researching the trends and patterns of traffic violations. Consequently, proper Traffic management and compliance measures should be observed to reduce and or manage cases of traffic accidents.

Li et al. (2021) state that AES is a type of traffic monitoring technology that runs constantly and captures different violations including; running a red light as well as speeding. It produces historical data at a large scale involving time, space, and environmental dimensions. AES improves prediction because it allows for the analysis of traffic behavior patterns in addition to user behavior; however, it suffers from low acceptance by the public and the perception of unfairness. Common traffic violation information was collected mainly by traffic enforcement cameras as shown in a study on Hohhot city in China. More than 13,000 violation records such as RLR and aggregated records of WWD were captured within the three months. The enforcement cameras even described how vehicles operated, the state of the road, and temporal characteristics. However, this approach has some disadvantages, for example, geographic coverage is limited, there are limited types of violations recorded in the data and limited environmental information Chuanyun et al. (2020).

The two promising solutions in traffic prediction are machine learning and deep learning techniques. Enhanced models like multilayer perceptron, Recurrent neural networks, and more developed LSTMs have much higher capabilities to read temporal and spatial patterns in traffic data Yashan et al. (2020). Current research shows how these techniques not only enhance predictive ability but also the applicability of the models to respond to shape-shifting traffic formations. For instance, deep neural networks have been used in traffic flow predictions, as have been seen to offer better accuracy coupled with scalability Behboudi et al. (2024). Such a change proves the increasing importance of AI-based approaches to traffic safety and policing Sayed et al. (2023).

1.1 Research Motivation

The application of learning models to predict traffic violation arrests meets a distinct societal need for traffic safety decreasing the violations and the incidents that accompany them. As per Behboudi et al. (2024), currently used enforcement strategies are costly and have low elasticity, while machine learning models use big and diverse data for prediction. Machine and deep learning methods allow tracking of car traffic in real time and provide law enforcement agencies with a basis for action. Nisha et al. (2023) state that there is immense possibility for contributing to a decrease in rates of traffic accidents and an increase in transport organization effectiveness, and, as a result, minimization of potential transport casualties and less losses in the economy. As the number of data sources from sensors and automated enforcement systems grows, learning models present a revolutionary way of traffic management. I have decided to use 5 models for machine learning and 5 models for deep learning and then compare their accuracy rates. According to my research, the current deep neural decision forest, multilayer perceptron has not been developed for predicting arrests due to traffic violations or any similar kind of traffic violation dataset. This gave me motivation to go ahead and use these models along with other learning models for my project.

1.2 Research Objective

Research Question: How does the accuracy of Machine Learning models compare to Deep Learning Models performed for traffic violation arrests?

The objective of this research is to forecast traffic violation arrests utilizing machine learning techniques and compare them with deep learning strategies in terms of how accurately they can predict arrest outcomes. According to traditional machine learning algorithms including

Logistic Regression, Linear Regression, random forests, the support vector machine, and state-of-the-art deep learning approaches including Deep neural decision forest, LSTM, multilayer perceptron, GRU, and recurrent neural networks, this study intends to determine which methodology yields the best estimates of traffic violations that lead to arrests. The specific objectives of the study are to estimate the effects of driver specifics, traffic situations, and different types of offenses on the model's performance. The intended purpose is to increase the effectiveness of traffic law enforcement practices, raise the level of ensured population safety and thus suggest toward the efficiency of police utilization of resources.

The structure of the thesis is as follows: Part 2 covers Related work, Part 3 states the methodology part, Part 4 details the Design Specification, Part 5 gives a detailed description of Implementation and Part 6 discusses the evaluation. At last, Part 7 covers the conclusion and future work.

2 Related Work

For traffic violation datasets, Deep learning, and machine learning approaches are commonly applied in this area of research. In this literature review, we will give a brief on the current knowledge on the identification of whether the given individual was arrested for traffic violation using either deep learning or machine learning plans. The goal of the research is to examine the strengths and weaknesses of the current models with the view of fostering endorsement of techniques and new developments in the arrest of traffic violators. Finally, our goals will contribute to the increased protection of roads and the advanced use of law enforcement.

2.1 Machine Learning Approaches

The study by Li et al. (2021) states that traffic violations at intersections involving signals do a good job of establishing antecedent conditions such as the design of roads, adverse weather, and characteristics of the vehicles. Strengths are very strong enforcement camera data and innovative models such as the random-effects logit models. In this research, it is found that the ProWSyn oversampling method has performed better than the traditional SMOTE technique. The AUC value of the random forest model is 0.91. However, they have some limitations to date, for instance, the policies may have less geographical coverage, and failure to capture some violations of law like failure to wear a seat belt. These gaps limit the generalization of the results and need bigger datasets that contain added information and larger geographic and violation ranges. Our work seeks to address the above gaps and offer integrated traffic safety answers.

Emre (2021) study employs Naïve Bayes, SVM, and KNN machine learning algorithms as well as spatial analysis with the IDW interpolation technique to analyze heavy vehicle speed violations in Turkey. It effectively does identify violations and at the same time suggests measures to prevent such violations. Strengths are combined spatial and machine learning and detailed classification. However, the use of the local data reduces the transferability of the results. They indicate that the emphasis should be put on the implementation of policies and the extension of the model in terms of the scope of data that should be used for its application.

This paper by Hossain et al. (2024) aims to analyze hit-and-run incidents involving pedestrians with the help of XGBoost and binary logistic regression on 8502 crashes that occurred in Louisiana between 2015 and 2019. In the end, the XGBoost model resulted in 78.38% accuracy and found variables such as high-speed roads, low light, and pedestrian actions to be important. Specific factors confirmed by logistic regression with odds ratios include increased risks for older pedestrians and crossing the high-speed zones. While XGBoost has prediction power logistic regression has interpretability. Some of the main limitations include a lack of driver data as well as unmeasured variables. Future work can involve the collection of larger datasets and the development of more complicated models that combine both numerical and categorical predictions as well as give improved insight into the variables. Some safety measures that can be proposed are lighting improvements and high-speed area patrols.

The paper by Narayanan et al. (2023) states that traffic control and dispatching of emergency vehicles are an important area which is dealt with in traffic control systems. The analyzed paper brings a new approach to combining FCM clustering and SVM classification for priority settings for emergency vehicle routes. The dataset employed real-time information on traffic density derived from lane-specific cameras and achieved 97% identification of the least congested lanes. In essence, the proposed method greatly minimizes delays due to the occurrence of signal lane adjustment with an adaptive signal. Nonetheless, the approach seems to achieve high accuracy, but it hurts computational efficiency while clustering large data sets and lacks scalability when applied to very dynamic systems. Future work can minimize these discrepancies by improving the speed of computation and enlarging real-time flexibility.

2.2 Deep Learning Approaches

Alrayes et al. (2023) use an approach known as “Deep Neural Decision Forests” which combines representation learning with the decision tree classification demonstrating comparable accuracy on MNIST and ImageNet datasets. The advantages of the algorithm are end-to-end optimization and scalability, and it offers robust stochastic routing. However, the approach is constrained by the computational cost, model interpretability, and sensitivity to hyperparameters. Still, these problems show some unsolved drawbacks related to the lack of accuracy, scalability, and efficiency in the hybrid model; therefore, more investigation is needed to improve decision-tree-based neural networks for more efficient and less costly models.

The paper by Mingze (2024) designs the hierarchical CNN, LSTM network-based method for predicting driver traffic violations considering both temporal and spatial aspects. The strengths are increased accuracy due to the attention mechanism and spatial-temporal modules. However, the use of such data leads to biases and computation complexity hinders scalability. Still, since the proposed dataset outperforms other models, generalization of the dataset is a limitation. Concerning that, future work needs to increase the number of samples, reduce biases, and scale down models to be implementable in real-time to make the model more usable in real traffic safety systems.

The reviewed paper by Farhad Mortezapour Shiri et al. (2024) examines CNN, Simple RNN, LSTM, Bi-LSTM, GRU, and Bi-GRU performance on three datasets: The approaches used were IMDB for sentiment analysis, ARAS for human activity recognition and Fruit-360 for image classification. While applying CNNs has produced remarkable performance on image classification with an overall accuracy of 99.69 percent on the Fruit-360 dataset, the use of their ability to capture spatial features played into this realization. GRUs outperformed ARAS because temporal dependence could be modeled efficiently using it. On the other hand, recurrent models such as RNNs and LSTMs fared poorly with image data as they couldn’t handle spatial features. For that reason, the study pointed out that CNNs were the most suitable for their datasets in terms of accuracy and efficiency when compared to GRUs. Some drawbacks include LSTMs and Bi-LSTMs computationally expensive and recurrent models that poorly handle image data sets. Further research could investigate other architectures, for example, the use of CNN–

LSTM to advance spatial and temporal representation. The study implies the requirement for the targeted model improvement and the search for new composite solutions.

2.3 Traffic Violation Datasets

Archana (2023) review focuses on the risks of recidivism, risky behavior, and crash risks associated with drivers with traffic violation records. Strengths comprise reliable data synthesis from various studies together with the proper appraisal of interventions such as the alcohol-interlock programs. Some of the limitations include different legal definitions from one country to another as well as targeting only high-income countries. This serves to imply a positive intervention outcome, though effects reduce post-intervention; these parades glitches related to sustainable interventions. More studies should be conducted to enhance our knowledge of driver training and prevention with plural strategies.

The literature by Ben (2023) also noted that drivers with traffic violations are most likely to re-offend, perform other risky behaviors, and be involved in crashes. Strengths are seen in the fact the 25 studies used different methodologies; alcohol-interlock programs, for instance, can be seen to decrease violations during periods of activity. However, limitations comprise variability in long-term results, and the variation of detailed laws in different nations, lack affair constricted to non-alcohol violating and teens driven autos. A view derived from this review is that there is a need for multiple theoretical and methodological strategies in combating recidivism. Research that has been conducted on hit-and-run pedestrian crashes identifies proximal factors that include this lighting, pedestrian behavior, alcohol influence, and regions that allow high speeds. Strengths comprise large and valid data extracted from the Tamil Nadu RADMS database and emphasizing demographic and environmental profiles. However, limitations such as insufficient driver details, describing locations only by a grid, and omitting multiple-vehicle accidents reduce outcomes. Thus, the review for this paper by Sathish (2020) underlines the importance of geospatial studies and the enhancement of existing laws to tackle hit-and-run categories in the Indian car milieu as well as hitches in ensuring driver responsibility and complementary safety installations.

The field involving traffic offense identification has changed with computer vision and deep learning technologies. The paper by Nikhil et al. (2023) describes the system of red-light violation and overboard pillion riding detection in video streams originating from surveillance cameras using YOLOv7. It uses the MSCOCO dataset for red light skipping and Google images for overboard which has been annotated. Key metrics include a 93% detection accuracy for red-light skipping and a map of (0.5:0.95) for overboard. Despite its effectiveness, some of the limitations are: it identifies only static vehicles and specific violation types. The scenarios in the future may include dynamic scenes together with other violation types to increase realism.

2.4 Traffic Violation Arrest Due to Age and Race Characteristics

The paper by Yan (2021) examines contextual racial effects on police traffic-stop arrests using hierarchical logistic regression. Thus, the paper identifies that an increased level of racial diversity corresponds to a lower arrest rate and, vice versa, areas with high Black or minority populations have a higher arrest rate. Indeed, its strength involves the use of a multilevel framework and data from various sources, but a major shortcoming is that it relies on data from only one city and there might also be selection bias due to the choice of variables included and those excluded. Implications of current research reveal a strong relevance to race within policing, calling for further, integrated wide zone analyses and legislative changes.

Similarly, the study by Arafa et al. (2020) discusses the age factors in driving behaviors of non-professional drivers in the South Egypt region. It also allows for more meaningful trends to be featured, for instance, the under-30 category was observed to engage in riskier behaviors in that, they have been observed not to use seat belts, and drive while drowsy and compromised, while the over 70 years vehicle users rated better violation scores. There is a certain bias in the use of self-administered questionnaires and the study is not nationwide. This study emphasizes the lack of effectiveness of existing measures and underlines the need for specific programs oriented toward enhancing the traffic safety of young drivers.

The pretextual traffic stop and race study conducted by Makofske (2020) on Louisville culminated in the discovery of elevated arrests whenever the police claimed solely “failure to signal.” Black motorists suffered these stops more than their counterparts: such stops that occurred during daylight bore the stamp of bias. Body-worn cameras are known to have affected both the ethnic apprehension differential, and the public pretext stop rate, and both were only observed in the short-term. However, there is a lack of comparable national data which restricts the degree of generalization. This leaves the need for change across the whole system to make changes within the police force to have balance in the black people being arrested in traffic.

The research by Saville et al. (2024) has raised issues on the partiality of traffic police, noting racial and sexual balance. This is done using machine learning models employing a database of more than 600,000 traffic stops from Montgomery County, Maryland. The work looks at race and gender as sensitive parameters used in predictive systems and discusses ways of developing ethical, superior-performing AI systems. It shows that race and gender are important predictors but could be dropped hence reducing social bias without a significant reduction in accuracy. The following research proposals should be considered with the goal of identifying and handling each officer’s bias.

Category	Study	Methodology/Tools	Strengths	Limitations	Future Directions
Machine Learning	Li et al. (2021)	Random-effects logit models, ProWSyn oversampling	Strong camera data, innovative modeling, AUC = 0.91	Limited geographical coverage, incomplete violation capture (e.g., seatbelt violations)	Use larger datasets with expanded geographic and violation range
	Emre (2021)	Naïve Bayes, SVM, KNN, IDW interpolation	Combines spatial and ML techniques, detailed classification	Results limited to local data	Expand the model scope and improve data generalizability
	Hossain et al. (2024)	XGBoost, logistic regression	High prediction accuracy (78.38%), interpretable logistic regression results	Lack of driver data, unmeasured variables	Develop comprehensive models combining numeric and categorical data
	Narayanan et al. (2023)	FCM clustering, SVM	High lane identification accuracy (97%)	Computational inefficiency, lack of scalability	Improve computational speed and flexibility for real-time application
Deep Learning	Alrayes et al. (2023)	Deep Neural Decision Forests	Scalable, robust routing	High computational cost, sensitivity to hyperparameters	Enhance efficiency, accuracy, and interpretability of decision-tree-based models
	Mingze (2024)	CNN-LSTM	Accurate spatiotemporal modeling	Data bias, high computational complexity	Increase dataset size, reduce bias, improve scalability
	Farhad et al. (2024)	CNN, RNN, LSTM, GRU	High CNN accuracy for images (99.69%)	Poor recurrent model performance on image data, the computational cost of LSTMs	Explore hybrid architecture like CNN-LSTM
Traffic Violation Datasets	Archana (2023)	Literature review, intervention programs	Synthesis of reliable data from diverse studies	Legal and demographic variability among countries	Conduct studies to enhance interventions and driver training
	Ben (2023)	Review of 25 studies	Identifies risks of recidivism and intervention outcomes	Variability in outcomes and definitions between countries	Pursuing diverse theoretical and methodological strategies
	Sathish (2020)	Tamil Nadu RADMS database analysis	Strong data emphasis on demographic/environmental profiles	Insufficient driver detail, location descriptions limited to grid-based approaches	Develop geospatial approaches and legal enhancements
	Nikhil et al. (2023)	YOLOv7	High detection accuracy (93% for red light skipping)	Limited to static scenarios and specific violations	Include dynamic scenes and broader violation types
Demogra-phics	Yan (2021)	Hierarchical logistic regression	Multilevel data framework	Reliance on data from one city	Conduct broader zone analyses and integrate data to address racial disparities
	Arafa et al. (2020)	Survey-based analysis	Insights into risky behavior trends by age group	Bias in self-reported data, limited regional coverage	Develop safety programs for younger drivers
	Mackofske (2020)	Observational study	Identifies racial bias in traffic stops	Lack of national-level comparative data	Implement systemic changes to reduce racial disparities in policing
	Saville et al. (2024)	ML-based bias analysis	Demonstrates the potential for bias mitigation through AI	Dependence on sensitive attributes (race, gender)	Develop ethical AI systems and address officer bias

Table 1: Summary Table (Literature Review)

3 Methodology

The following sub-section gives a detailed description of the method used in this study. The main goal is to forecast traffic violation-related arrests with the help of machines and deep learning technologies. The methodology steps are as follows starting with data preprocessing, defining features and target variables, categorical data, resampling augmenting and reducing the data availability, standardizing the data, and finally training and evaluating different sets of machines as well as deep learning models.

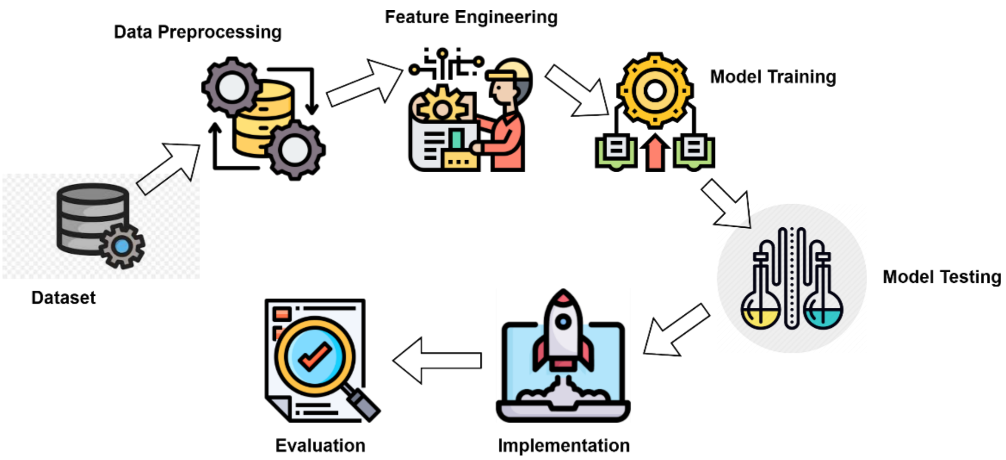


Fig 1: Methodology

The above methodology diagram is an illustration of the flow to be used for the analysis and modeling of the data. The process known as data cleaning takes place at the first step in which useful data is collected and processed to eliminate noise from the data collected. Afterward, training and validation are done to facilitate accurate predictions before using a model for real-life purposes. The process is cyclical and is followed up through numerous test or optimization loops.

3.1 Data Overview

This study has a dataset in the form of 65000 records with 7 columns. Basically, each row gives details of traffic violations, including the date, time of the stop, demography of the driver, and geographical location. The representation of the columns is presented in Table 2 below.

Data Attribution	Details
stop_date	It is the date of violation
stop_time	It is the time of the violation
driver_gender	Gender of violators
driver_age	The exact age of the violators
driver_race	Race of violators
violation	Cause of violations
search_conducted	Whether the search is conducted or not
stop_outcome	Outcome of violation
is_arrested	If a person was arrested or not
stop_duration	The detained time duration for violators in mins
drugs_related_stop	Whether an individual was involved in a drug crime or not

Table 2: Data Distribution

3.2 Data Preprocessing Data Loading and Initial Processing

A Python data processing library, pandas, was used to load the traffic violations dataset. The dataset contains 7 columns, including demographic and time-related information such as “stop_date”, “stop_time”, “driver_gender”, and “driver_race”. The dataset was loaded using pandas.read_csv() to create a data frame for analysis. The target variable “is_arrested” was converted into binary format which is 1 for “is_arrested” otherwise 0.

3.3 Visualizations

The visualization in this study provides a comprehensive overview of factors affecting arrests due to traffic violations, highlighting key relationships.

3.3.1 Traffic Violation Arrests Based on Race

The graph below represents traffic violations committed by drivers of different races. It employs pd.crosstab in Python to categorize the violation by the given race categories in this case providing the frequency of each violation kind. The horizontal axis refers to distinct driver races while the vertical axis reflects the entire violations. Each bar is further divided by segments and every color in the bar corresponds to the different types of violation. The chart's goal is to find out how different counts of violations are up to race. This means that the stacked format makes it easy to compare from one race to another while at the same time showing what comprises violations in every race.

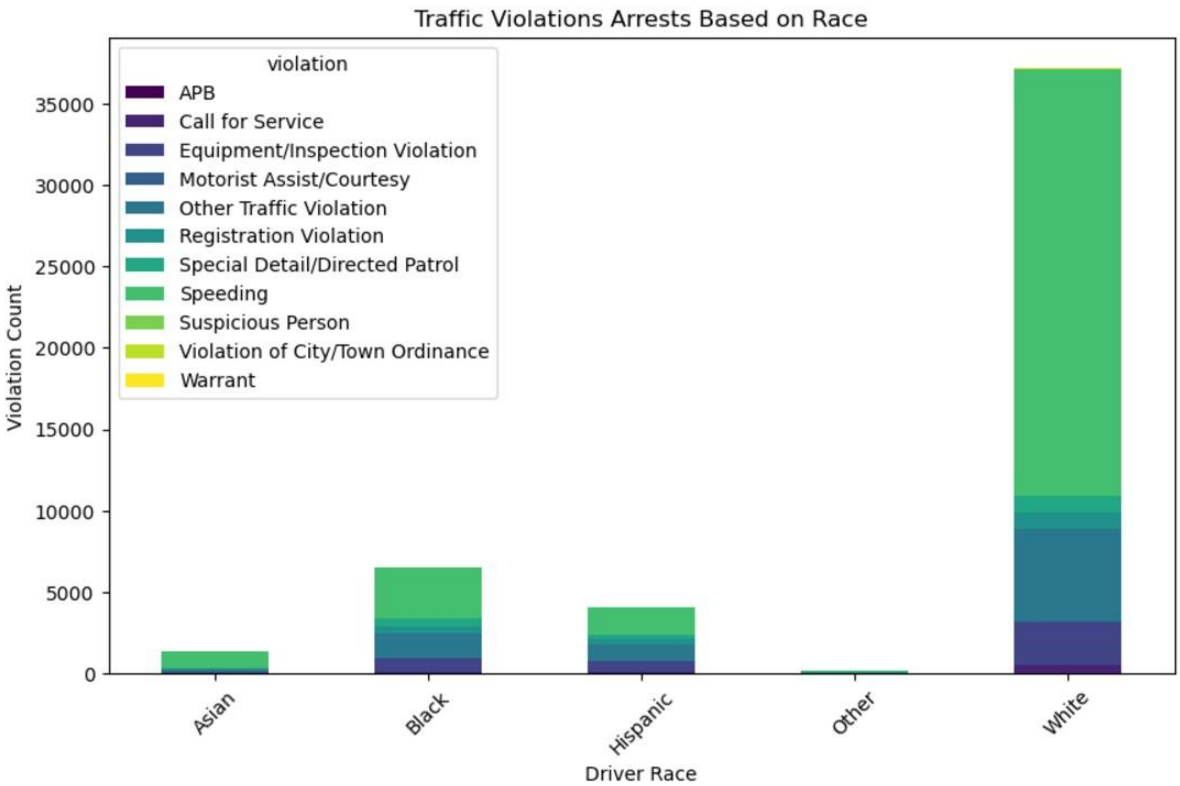


Fig 2: Traffic Violations Arrests Based on Race graph

3.3.2 Arrests Before and After Search Conduct

The graph displayed below compares arrests with the conduct of searches. A count plot is used in the x-axis: those who were arrested = 1, not arrested = 0. The color discriminates between those who had a search done (yes, no). On the y-axis, one has the number of people in each category. This is achieved through enabling comparison of arrests before and after searches, shown to demonstrate the effect of search activities on the arrests.

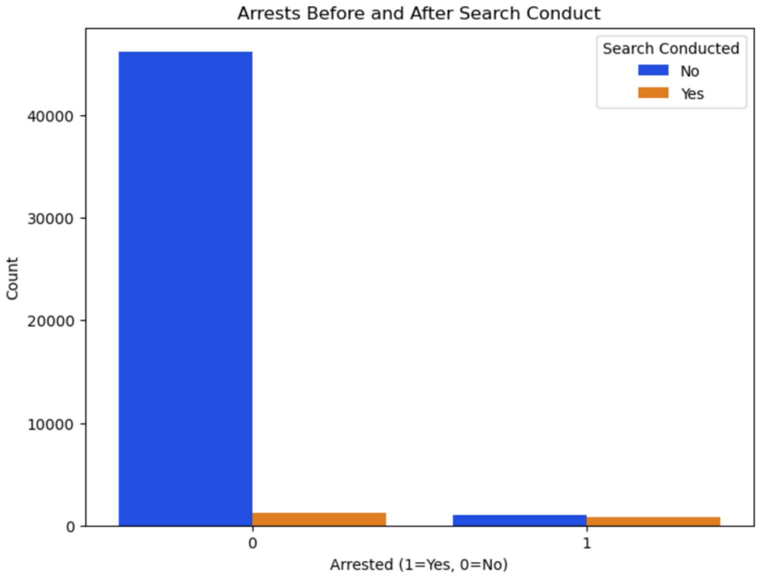
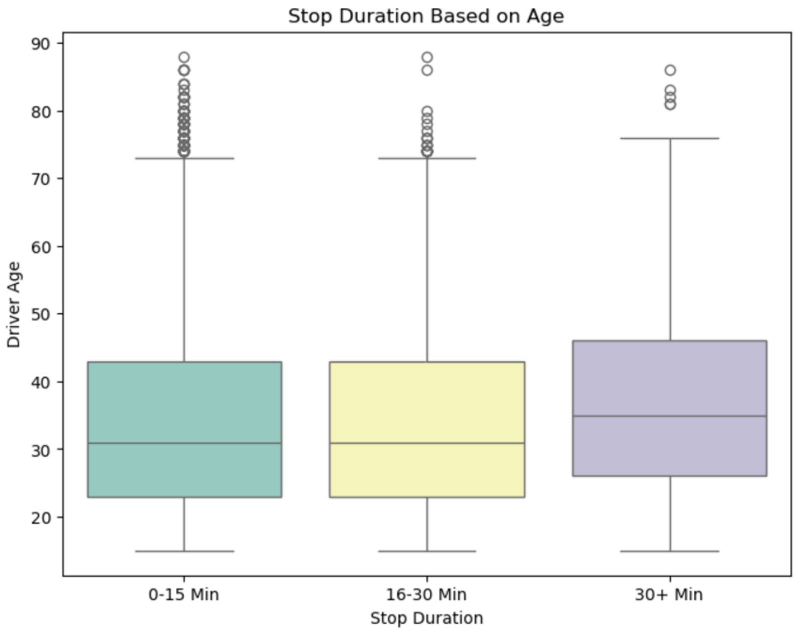


Fig 3: Arrests Before and After Search Conduct graph

3.3.3 Stop Duration Based on Age

The variable on the horizontal axis of the box plot represents driver age while the vertical axis represents stop duration. The x-axis scale is measured on the time spent on the traffic stops, and the y-axis is on the ages of the drivers. Each box is the IQR of the stop duration of different driver ages. The horizontal line within each box represents the median number of minutes each age group spends at a stop and the lower and upper bars denote the spread of stop times, not including outliers. Outliers are shown as data points. The plot assists in determining trends of stop duration in comparison to the age range of students thereby pointing out whether there exists a considerable variation between different age brackets.

Fig 4: Stop Duration Based on Age Graph



3.4 Categorical Conversion

The target variables were brought into a more suitable format for the model by converting them to categorical format with one hot encoding using “pd.get_dummies” function. This flexibility was particularly crucial for training frequency-based models which rely on binary, or categorical outputs, such as neural networks.

3.5 Data Splitting

Pre-processing of data was done using the train test split function in Scikit-learn where the dataset was split into training (80 %) and test (20 %). A random state was used to ensure the repeatability of the results outcomes in the case of the implementation of the experiments. As stated by Jahin et al. (2024), it is a usual practice used in the evaluation of implementations of machine learning models as a further assessment of performance.

3.6 Standardization

Standardization was done using “StandardScaler” from the Scikit-learn library to scale features. In general, standardization of functions involves the process of obtaining mean and standardizing to unit deviation. This step ensures that all factors contributed equally towards the build of the model and helped to converge the algorithms used to train models.

3.7 Data Reshaping

The machine and deep learning models needed the data to be put in the format that was preferred by the program. Some of the available data fields were categorical, which means they have to be encoded: the one-hot encoding was used; some columns were split into a set of meaningful sub columns (stop_date into stop_year, stop_month, stop_day, for example). They further enabled the preprocessing of the data, so that the feature representations of the data would be in the best form for the modeling algorithms.

3.8 Model Training and Evaluation

The following measures such as accuracy, precision as well as recall were employed to determine the effectiveness of various machine learning as well as deep learning models that were trained to estimate the number of arrests as a result of traffic violation.

- **LSTM (Long Short-Term Memory)**: LSTM is an RNN variant specifically that aims to learn long-term dependencies in sequences of time or of characters. In contrast with traditional RNNs, LSTM reduces the issue of vanishing gradient by including the employing gate controlling the transmission flow (Input, forget, and output gates) making the LSTMs ideal for the current specific applications such as speech recognition, language modeling, and time series analysis as per AIML (2023).

- **Deep Neural Decision Forest (DNDF)**: Alrayes et al. (2023) state that Deep Neural Decision Forest integrates decision forest with deep neural networks and allows the efficient training of the model while at the same time maintaining the decision tree interpretability of the model. This type of model is optimal for cases where the results need a high degree of accuracy and where the need for decision justification is high, for example, in financial foreseen or diagnostics.

- **RNN (Recurrent Neural Network)**: RNNs are specifically neural networks that are used for sequential data where the outputs of one step become fed into the next step. They are used in speech recognition and text generation techniques, but the problem of long-term dependencies is with these, which is solved by using LSTMs and GRUs as suggested by AIML (2023).

- **GRU (Gated Recurrent Unit)**: They are a simplified model of LSTMs that excludes one or two of the gates necessary to enhance computational operations. They function similarly to LSTMs for modeling long-range dependencies in sequences and are used commonly in situations that require real-time because of faster training as stated by AIML (2023).

- **MLP (Multilayer Perceptron)**: MLPs are simple forms of neural networks with input layer, hidden layer, and output layer. They are good for functions such as image classification, and regression and need careful hyperparameter tuning when it comes to things like the number of layers and neurons per layer to avoid overfitting.

- **Random Forest**: Yan (2024) states that Random forests on the other hand is an ensemble learning technique that puts together many decision trees and uses them to offer an improved estimate. It is less sensitive to overfitting and is applicable to classification and regression problems, such as customer churn and image classification of diseases.

- **Logistic Regression**: Logistic Regression is a model of statistical classification that finds application in binary classification problems. It tends to predict probabilities by logarithmic function; thus, it is most suitable in a situation like spam filtering or disease prognosis.

- **Linear Regression**: Linear Regression is among the simplest algorithms useful for regression problems; the relationship between input variables and the output variable is determined to be a linear function. It is mostly applied in the prediction and evaluation of risks.

- **KNN (K-Nearest Neighbors)**: KNN is an algorithm that does not rely on any assumptions and assigns a data point to a class to the most often occurring class among its k number of nearest neighbors. For such applications as image classification, it is easy and straightforward but can be highly time-consuming for big data.

- **SVM (Support Vector Machine)**: SVM is a strong classifier that determines the best hyperplane through which class boundaries of different feature vectors can be drawn. It works well especially where the input vectors have many features, it is widely used in areas such as image identification as well as text categorization.

3.9 Evaluation Metrics

The assessment of the overall performance of the models was done using accuracy scores by Scikit Learn with the help of classification report metrics. The following metrics precision, recall, and F1 scores provide an overall evaluation of the effectiveness of this model proposed. According to these criteria, models were evaluated to identify which out of the models in question offered the best probabilities of ‘arrests’ for traffic violations. The systematic approach to data processing, machine, and deep learning model training, variable identification, and performance testing is guaranteed with this approach.

4. Design Specification

This section elaborately describes the architecture employed in the prediction of arrest due to traffic violations, a classification problem. We also have deep learning models multilayer perceptron and Deep Neural Decision Forests and a traditional Machine learning Model which is a Random Forest. Performance is assessed to illustrate why these models worked and to recommend which are better suited to certain tasks.

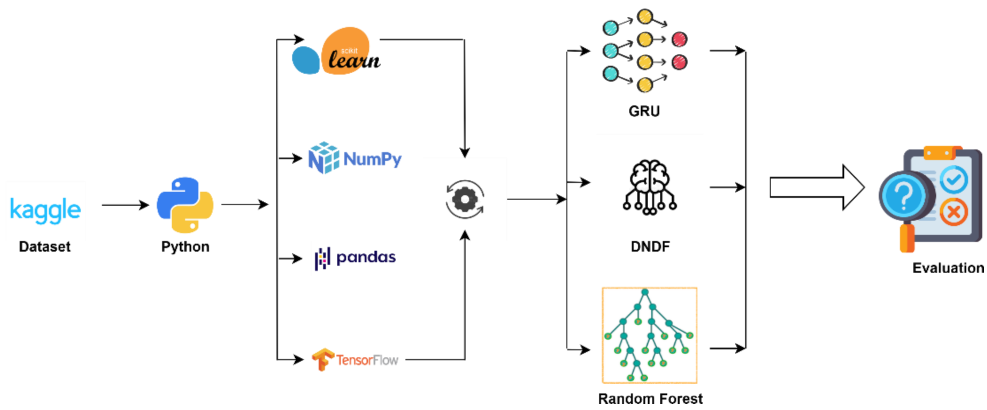


Fig 5: Design Specification diagram

4.1 Gated Recurrent Unit (GRU)

GRU is a subtype of RNN developed specifically for the analysis of sequences including time series or text inputs. It avoids the issues of implementation of traditional RNNs such as vanishing gradient thus making it effective in the capturing of long-time dependencies. GRUs are less complex than LSTMs, but they do not decrease effectiveness while using elements of the network.

The GRU takes an input vector and a memory from each time step of a sequence and produces an output at each step. This intermediate state helps the model carry forward from the previous steps of the sequence which in other words means that the model can remember the context while working on the data. GRUs use two main gates to regulate how information flows:

- Reset Gate: The first sets the tolerance level of which of the past information to discard. If the reset gate is close to zero, the GRU "resets" and therefore does not take much consideration with the past values for the step.

- Update Gate: This means that it balances old and new information as it comes to a decision about how much of the hidden state to keep as it is, and how much to update based on current inputs.

A candidate activation is the input-level potential for new memory, computed by its reset gate and the current input. This makes certain that the GRU pays attention to the right aspects of data. The last hidden state is a result of a linear combination of the last hidden step and the candidate activation. This decided the proportions and made it possible for the GRU to selectively update its memory through the update gate. The modified hidden state is either fed to the next timeframe in the sequence or extracted for the prediction of results as given by Datatechnotes (2024).

4.2 Deep Neural Decision Forest (DNDF)

Alrayes et al. (2023) state that the Deep Neural Decision Forest (DNDF) is a part of a new breed of decision forests that are based on Deep Neural Networks, the main advantage of using this model is that the feature extraction is done together as well as the decision in combination with decision forests. On jobs such as predicting arrests from traffic offenses, Deep Neural Decision Forest is useful because it captures interactions in the data.

In the proposed framework, the neural network component can be considered as a feature extractor. Other layers are Dense layers (ReLU activation) which transform inputs to features in which significant patterns are easy to identify. In this context, driver-specific information such as the driver's age, and gender, and contextual information such as the number of times the car has stopped, and the type of violation are converted into features. After configuration, extracted features are given to a decision forest, such as a Random Forest. Here, the tree-based model of feature interactions is properly addressed, and the area is divided into decision regions for classification (e.g., arrest or no arrest).

We can see that Deep Neural Decision Forest can handle both numerical and categorical data fast after data preprocessing, for example, one hot encoding. Therefore, neural layers automatically acquire feature extraction, and the decision forest is well-suitable for interpretability as well as overfitting issues in high dimensions. In the case of arrest prediction, Deep Neural Decision Forest outperforms because neural networks identify subtle and complex patterns different from linear patterns while the decision tree focuses on a cascaded decision-making approach. This is especially important in modeling the relationship between the interaction of violation types in conjunction with demographic attributes and officer behavior.

4.3 Random Forest

Random Forest is an ensemble learning method formulated during its training process that comprises a set of decision trees and the outcome is the mean of the responses submitted by members of such a forest. Here’s how its architecture works:

- Bootstrapped Datasets: Random Forest forms multiple subsets of the training data set with the bootstrapping samples (samples with replacement). This makes certain that each decision tree is trained on a slightly different data set from the rest.
- Decision Trees: Each tree is an individual classifier that partitions the data into feature and threshold space to minimize the impurity of the resulting groups (for classification, this is the Gini index; for regression, variance).
- Feature Selection: After each split, a random sample of features is used for splitting, thereby avoiding the problem of trees being too much alike and increasing hypothesis complexity to improve the model.
- For classification, the forest predicts the class with the majority vote across all trees.
- For regression, the prediction is the average of the tree outputs.
- Parallelism: Random Forest is highly parallelizable given that the decision trees are constructed independently of one another.
- Randomization: The randomness in data sampling and feature selection eliminates cases of over-fitting and guarantees accuracy in the results produced.

Random Forest handles both numerical and categorical data which is very useful for tasks such as arrest prediction. Thus, the ability to model non-linearity and feature interactions guarantees good results with datasets containing many features and complex relations between them, like driver characteristics and violation circumstances as stated by Lacherre (2024).

5. Implementation

The implementation involves several approaches based on machine learning and deep learning to forecast arrests for traffic violations. It starts with data preprocessing, including data cleaning and data transformation, and then goes sequentially to model training by logistic regression, linear regression, SVM, KNN, Random Forest, Deep Neural Decision Forest, multilayer perceptron, RNN, GRU, and LSTM. The performance of each model is assessed to identify differences and choose the optimal strategy.

5.1 Tools and Technologies

Task/Component	Library Used	Description
Data Manipulation	Pandas	For data cleaning, preprocessing, and feature engineering.
Numerical Computations	NumPy	For handling arrays and performing numerical operations efficiently.
Visualization	Seaborn, Matplotlib	To create insightful graphs and plots for exploratory data analysis (EDA).
Machine Learning Models	Scikit-learn	Implemented Logistic Regression, Linear Regression, Random Forest, SVM, and KNN.
Deep Learning Models	TensorFlow/Keras	Used to develop and train Deep Neural Decision Forest, MLP, RNN, GRU, and LSTM architectures.

Table 3: Tools and Technologies Used

5.2 Process

5.2.1 Data Preparation

The traffic violations dataset was imported by pandas using the function pandas.read_csv() which effectively imports the structured data in a DataFrame format. As part of data preparation, data cleaning, and feature engineering, columns such as country_name and search_type were deleted to reduce noise in the data since it was an empty row. Categorical data was also then quantized into numerical forms. For the variables such as driver_gender, the map () function fits the data to binary codes, on the other hand, for violation and stop_duration which are multi-class variables, the data was fit to the one-hot encoding method as adopted by the machine learning algorithms.

Further, stop_date and stop_time fields were also converted to other features that accommodated temporal structures, including stop_year, stop_month, and stop_day. Those were further transformed by StandardScaler which makes the data into a normal distribution with a mean of 0 and a standard deviation of 1. This preprocessing step was particularly significant in enhancing the learning performance as well as the fast convergence of deep learning.

Below is the summary table for all the hyperparameters used in my research work:

Model	Hyperparameters	Notes
Random Forest Classifier	- n_estimators: [200]	Grid search used for tuning.
	- min_samples_leaf: [40, 60, 100, 150, 200]	
	- max_depth: [3, 5, 10, 15, 20]	
	- max_features: [0.05, 0.1, 0.15, 0.2, 0.25]	
K-Nearest Neighbors (KNN)	- n_neighbors: 5	Fixed hyperparameter; no grid tuning specified.
Support Vector Machine (SVM)	- kernel: 'linear'	No further tuning specified.
	- probability: True	
Logistic Regression	- max_iter: 1000	No other tuning grid provided.
Linear Regression	No hyperparameter tuning specified.	Default configuration used.
Deep Neural Network (Feature Extractor)	- Layers:	Used as a feature extractor.
	Dense(128, activation='relu')	
	Dense(64, activation='relu')	
	Dense(32, activation='relu')	
	- n_estimators: 100	
	- random_state: 100	
	- class_weight: 'balanced'	
Multi-Layer Perceptron (MLP)	- Layers:	Neural network for binary classification.
	Dense(128, activation='relu')	
	Dense(64, activation='relu')	
	Dense(32, activation='relu')	
	Dense(1, activation='sigmoid')	
	- optimizer: adam	
	- loss: binary_crossentropy	
	- epochs: 10	
	- batch_size: 32	
Recurrent Neural Network (RNN)	- Layers:	Designed for sequential data.
	SimpleRNN(50, activation='relu')	
	Dense(32, activation='relu')	
	Dense(1, activation='sigmoid')	
	- optimizer: adam	
	- loss: binary_crossentropy	
	- epochs: 10	
	- batch_size: 32	
Long Short-Term Memory (LSTM)	- Layers:	Handles long-term dependencies in data sequences.
	LSTM(50, activation='relu')	
	Dense(32, activation='relu')	
	Dense(1, activation='sigmoid')	
	- optimizer: adam	
	- loss: binary_crossentropy	
	- epochs: 10	
	- batch_size: 32	
Gated Recurrent Unit (GRU)	- Layers:	Alternative to LSTM with simpler
	GRU(50, activation='relu')	
	Dense(32, activation='relu')	
	Dense(1, activation='sigmoid')	
	- optimizer: adam	
	- loss: binary_crossentropy	
	- epochs: 10	
	- batch_size: 32	

Table 4: Hyperparameters Table

5.2.2 Model Training

- **Machine Learning Models:**

    1. Models like Logistic Regression, SVM, KNN, and Random Forest were trained using train_test_split() for splitting data into training and test sets.

2. Hyperparameter tuning for Random Forest was performed using GridSearchCV, testing parameters like n_estimators, max_depth, and min_samples_leaf.

- **Deep Learning Models:**

1. Multilayer Perceptron: Constructed with two hidden layers with density values of ReLU and a single sigmoid layer for binary predictions.

2. RNN, GRU, LSTM: Recurrent layers were applied to time series data analysis, and the models included in the study were sequential.

3. Deep Neural Decision Forest: Integrated neural feature extraction with decision trees, using TensorFlow for the neural part of the combination and Scikit-learn's Random Forest as the decision part.

5.2.3 Model Evaluation

Classification and model-specific measures were used in the assessment of the developed models. Measures like accuracy and performance, and testing methods like precision, recall, and F1 values of the studied models highlighted how well the data was classified and the performance of the models with imbalanced data. Area Under Curve (AUC), being a model-specific measure of binary classification, represented a trade-off between sensitivity and specificity. Using confusion matrices and ROC curves the performance of the models could be evaluated clearly. In confusion matrices, true and false positive and negative results were presented by giving the number of correct and wrong predictions. This made it easier to look at classification errors from a different perspective. These were accompanied by more comprehensive classification reports that provided improved evaluation of the given metrics such as precision, recall, and F1-Score for each class to understand the efficiency of the models in predicting arrests.

6. Evaluation

In the evaluation phase of this study, various machine learning and deep learning models were used to evaluate the performance of arrests relating to traffic violations. The evaluation process included comparison operations using measures like accuracy, precision, recall, F1-score, the use of a confusion matrix, and ROC curve.

6.1 Case Study 1: Machine Learning Models

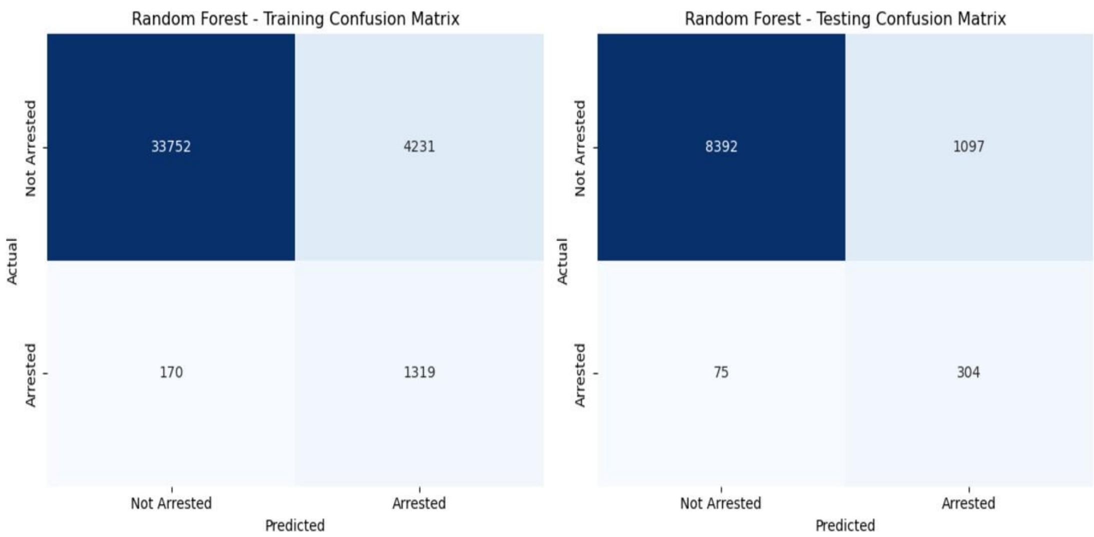
Among all 5 ML models employed in this study, it seems that the Random Forest Model is particularly suitable for this study because it can prevent overfitting and can also accommodate the interaction effects of driver demographics and violation contexts. Specific features such as ensemble learning and bootstrap sampling enabled it to deal with nonlinear relationships between the variables and come up with high levels of precision and recall. Its performance was fine-tuned by using the Hyperparameter tuning via GridSearchCV. In binary classification, Logistic Regression results showed its simplicity and robustness in conjunction with interpretability. Yet, it was less accurate than Random Forests due to difficulties identifying both intricate and nonlinear relationships. In high-dimensionality feature space, SVM (Support Vector Machine) was much more effective and gave more correct decision boundaries. While it performed well, it had high computational costs, as well as dependency on parameter adjustment, and was slower than ensemble methods. The results of the KNN (K-Nearest Neighbors) algorithm were fairly good but testing data size was an issue during the prediction step in terms of computation. Linear Regression was mainly developed to be used for regression analysis and it turns out that it is not very suitable for this type of binary classification. The models were compared using the metrics as shown in the table below:

Model	Test Accuracy	Precision	Recall	F1 Score	Training Time (s)	Prediction Time (s)
Random Forest	0.881	0.217	0.802	0.342	444.09	0.37
KNN	0.961	0.375	0.008	0.016	0.03	7.74
SVM	0.961	0.333	0.003	0.005	641.24	1.86
Logistic Regression	0.962	0.532	0.198	0.288	0.8	0.01
Linear Regression	0.962	0.569	0.087	0.151	0.17	N/A

Table 5: Classification report for ML models

The confusion matrices pointed out the true positive and false positive of each model, while the ROC curves presented the problem-solution space graphically. The performance of the ML models in this research has been assessed using these measures. For Random Forest, high recall on the "arrested" cases reveals the ability of this classifier to predict accurately the positive situations. Therefore, using Logistic Regression, it was found that the accuracy would be moderate between precision and, thus could be implemented practically.

A high number of handling instances, low recall, and imbalance, all impacted KNN's performance as a learning technique. High accuracy as indicated by SVM is a positive sign showing the ability of the system to classify classes while low recall for minority classes was observed. Linear Regression had reasonable accuracy and lower precision than its non-classification-specific design. A high AUC presented by ROC curves demonstrates the discriminative ability of Random Forest and Logistic Regression.



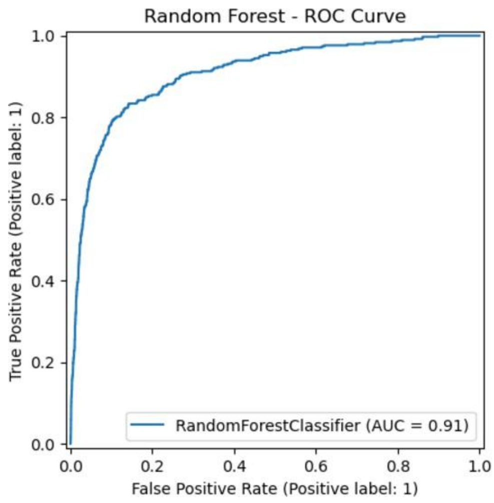


Fig 6: Confusion Matrix and ROC curve for Random Forest Model

6.2 Case Study 2: Deep Learning Models

Deep Neural Decision Forest (DNDF) achieves 96.32% test accuracy, as well as promising precision and recall on the training set. However, it does not fare so well with unseen data because it overfits the training data. Tying to recall for the “Arrested” class during the testing is (13.19%), demonstrating its incapability to identify the minority class cases. This problem is due to unequal class distribution where the majority class is ‘Not Arrested’ which apart from increasing precision and accuracy, reduces recall. Thus, while exhibiting great potential for practice performance during training, the Deep Neural Decision Forest needs to be improved to provide class predictions.

The multilayer perceptron has a precision of (64.46%), which is the highest of the models, and its F1 score (31.20%) is higher than that of the Deep Neural Decision Forest, showing a good level of recall. It has high density, makes it easy to train, and generates good features which makes it better in classifying the cases in the minority. Unfortunately, as seen, its recall (20.58%) is still low as an issue with imbalanced data set persistence. Although not a perfect algorithm, multilayer perceptron is quite easy to implement, and more importantly, is very efficient for real-world applications where training sessions are relatively long. Recurrent Neural Networks are the efficient strategy for dealing with sequential data, meeting temporal dependencies in most of the cases of the majority class. But the recognition of the “Arrested” class is extremely low (2.11%), and the F1 score of them is (4.08%). Due to the vanishing gradient problem, their performance is poor and thus they fail to capture long-term dependencies efficiently. These limitations are overcome in Long Short-Term Memory (LSTM) models that lead to a recall rate of (13.45%) and an F1 score of (21.79%). Long Short-Term Memory are great at learning sequences, but they are not as great with training times and perform poorly for this dataset.

Among all the compared models, the Gated Recurrent Unit (GRU) works best in recall, with a value of 34.04%, and is slightly higher in the F1 score of 40.44%, thereby demonstrating a proper balance between precision and recall. It takes less time to train than Long Short-Term Memory and has nearly equivalent accuracy. Still, its results: precision (49.80%) and accuracy (96.14%) are slightly lower than results obtained by Deep Neural Decision Forest and multilayer perceptron but contain potential for improvement. Compared to the above-mentioned four models, GRU has the best comprehensive performance, and multilayer perceptron is a more practical model for fast training. RNNs are slow, and Long Short-Term Memory, in addition to Deep Neural Decision Forest, needs more improvements regarding the unsuccessful generalization.

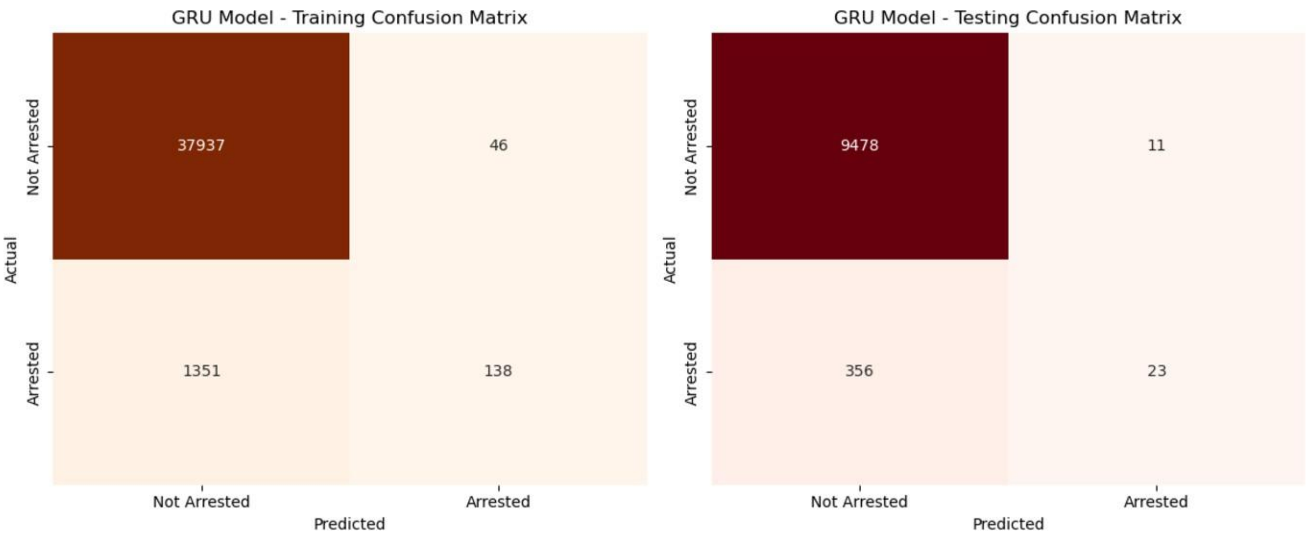
Model	Test Accuracy	Precision	Recall	F1 Score	Training Time (s)	Prediction Time (s)
Deep Neural Decision Forest	0.963	0.595	0.132	0.216	13.4	0.55
GRU	0.961	0.498	0.34	0.404	179.06	13.12
MLP	0.965	0.645	0.206	0.312	10.78	0.4
LSTM	0.963	0.573	0.135	0.218	167.58	9.97
RNN	0.962	0.615	0.021	0.041	90.95	6.68

Table 6: Classification Report for DL Models

The Deep Neural Decision Forest (DNDF) comes up with a confusion matrix with high true negatives compared to true positives because of the program's struggle in detecting "Arrested" cases. In the ROC curve, this type of model has high specificity but low sensitivity because of the low recall of this model. Likewise, the multilayer perceptron and its true negatives are high, but the precision is a bit higher than the Deep Neural Decision Forest model. In terms of the ROC curve, it seems to have a better balance of precision and recall still, it is inclined towards the majority class.

The situation deteriorates in the case of "Arrested" cases which represent only a minority in the RNN where most of the predictions are put in the "Not Arrested" bucket. This is apparent in the large and highly skewed ROC curve accompanied by little area under the curve due to poor recall. In evaluation, both the RNN and Long Short-Term Memory models encounter similar issues, yet the Long Short-Term Memory model shows small enhancements in terms of precision & recall. Nevertheless, using the receiver operating characteristic curve reveals only a slight increase in accuracy without improving the category separation.

Nonetheless, the Gated Recurrent Unit (GRU) shows the highest recall among the models and a better distribution across the confusion matrix. The AUC-ROC curve of GRU is slightly larger than those of all the other models, mainly due to the higher recall in cost of the slightly lower precision. In conclusion, GRU is the most auspicious classifier here.



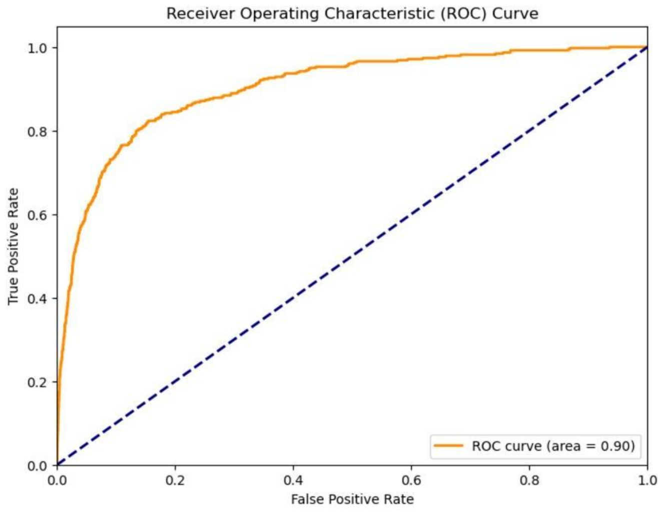


Fig 7: Confusion Matrix and ROC curve for GRU Model

6.3 Discussion

DL models, particularly GRU, bridged the gap between ML and DL approaches, outperforming all other models in a recall by 34.04% and an F1 score of 40.44% for the "arrested" class. Random Forest was the best model with 80.21% on test data mainly because of well predicting the "arrested" cases and being less sensitive to the problems of insufficient data. For this type of dataset and type of classification problem. In this case, Logistic Regression offered interpretable results with both a high precision of 53.19% and a high recall of 19.79%. However, due to its linear characteristics, the model failed to address intricate patterns of the network. Precision was high for SVM specifically with balanced data however unlike the Deep model, the "arrested" class was highly impacted because of the high computational requirements of SVM and chosen parameters tuning. It was also found that KNN had low computational efficiency and poor recall 0.79%, presumably because of the enormous number of records and non-transparent distributions of the classes used in the study. Linear Regression achieved a reasonable level of accuracy, but it lacked precision and recall because Linear Regression was not able to perform well in binary classification. The multilayer perceptron DL model mostly presented the best precision of 64.46% among the DL models; however, its recall of 20.58% was equally poor. While Deep Neural Decision Forest attained the highest accuracy 96.32%, it was overtaken by overfitting and a low recall value for minority cases 13.19%. Long Short-Term Memory retained a certain amount of sequential information though slightly better than RNN in recall at 13.45% and F1 score of 21.79%. The results also indicate that RNN has the lowest recall, 2.11% and F1 score 4.08%. In general, the results of the study showed that the ML models were more accurate and feasible for predicting arrests for traffic violations.

7 Conclusion and Future Work

The present research shows that the use of machine learning (ML) and deep learning (DL) can be used to predict arrests for traffic violations. Based on the evaluation results, Random Forest was deemed as the most accurate algorithm for imbalanced data and the best performance in terms of precision and recall. When compared with other DL models, GRU offered the highest recall and F1-score which other studies can apply to rectify the minority case detection issue. The study points to timely and effective estimation models as critical in improving the delivery of traffic law enforcement as well as the allocation of resources. Nevertheless, difficulties like overfitting and advanced data balancing of the model still exist, which indicates the potential for improvement in the future use of such experiments.

7.1 Future Work

- Improving Recall for Minority Classes: The feature resampling process offers some promise for recall for minority classes, but more research could investigate other resampling strategies or cost-sensitive learning methods that might enhance recall with respect to minority classes.

- Integrating Contextual Data: The extension of the model datasets by using real-time environmental information like, weather conditions or traffic congestion can further polish the prediction.

- Advanced DL Techniques: The transformer and its variants were investigated as a remedy for overfitting and enhancing generality for low-sized datasets and rare cases.

- Real-Time Deployment: Extending these models for real traffic monitoring for the early prevention of crime by law enforcement agencies.

- Enhancing Public Perception: Calling for researchers to conduct surveys and investigate problems relating to equality and personal data protection in AI-trafficked systems.

References

Behboudi, N., Moosavi, S. and Ramnath, R. (2024). *Recent Advances in Traffic Accident Analysis and Prediction: A Comprehensive Review of Machine Learning Techniques*. [online] arXiv.org. Available at: https://arxiv.org/abs/2406.13968.

Sahal, N., Preethi, D., & Singh, D. (2023). *Autonomous traffic prediction: A deep learning-based framework for smart mobility*. Proceedings on Engineering Sciences, 5(S1), pp. 35-46. Available at: https://doi.org/10.24874/PES.SI.01.005

Li, Y., Liu, L., Yuan, J., Lu, J. and Abdel-Aty, M. (2021). Analysis and prediction of intersection traffic violations using automated enforcement system data. 162, pp.106422–106422. doi:https://doi.org/10.1016/j.aap.2021.106422.

Hossain, A., Sun, X., Hasan, A.S., Jalayer, M. and Codjoe, J. (2024). Comprehensive Investigation of Pedestrian Hit-and-Run Crashes: Applying XGBoost and Binary Logistic Regression Model. *Transportation Research Record Journal of the Transportation Research Board*. doi:https://doi.org/10.1177/03611981241262315.

Kuşkapan, E., Çodur, M.Y. and Atalay, A. (2021). Speed violation analysis of heavy vehicles on highways using spatial analysis and machine learning algorithms. *Accident Analysis & Prevention*, 155, p.106098. doi:https://doi.org/10.1016/j.aap.2021.106098.

Narayanan, K.S.S. and Saravanan, K., 2020. Traffic Signal Data for Emergency Vehicles using C- Means and SVM Classification. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 9(4), pp.983–987. DOI: 10.35940/ijitee.D1306.029420.

Alrayes, F.S., Zakariah, M., Driss, M. and Boulila, W. (2023). Deep Neural Decision Forest (DNDF): A Novel Approach for Enhancing Intrusion Detection Systems in Network Traffic Analysis. *Sensors*, [online] 23(20), p.8362. doi:https://doi.org/10.3390/s23208362.

Wang, M. and Li, N. (2024). A Hierarchical Network-based Method for Predicting Driver Traffic Violations. *IEEE Access*, [online] 12, pp.121280–121290. doi:https://doi.org/10.1109/access.2024.3450935.

Shiri, F.M., Perumal, T., Mustapha, N. and Mohamed, R. (2024). A Comprehensive Overview and Comparative Analysis on Deep Learning Models. *Journal on Artificial Intelligence*, 6(1), pp.301–360. doi:https://doi.org/10.32604/jai.2024.054314.

Kaur, A., Williams, J., Feiss, R., Rose, D.-M., Zhu, M. and Yang, J. (2023). Subsequent risky driving behaviors, recidivism and crashes among drivers with a traffic violation: A scoping review. *Accident Analysis & Prevention*, 192, pp.107234–107234. doi:https://doi.org/10.1016/j.aap.2023.107234.

Laoula, B., Elfahim, O., Marouane El Midaoui, Youssfi, M. and Bouattane, O. (2023). Traffic violations analysis: Identifying risky areas and common violations. *Heliyon*, 9(9), pp.e19058–e19058. doi:https://doi.org/10.1016/j.heliyon.2023.e19058.

Zhang, Y. and Zhang, L. (2021). Racial characteristics of areas and police decisions to arrest in traffic stops: multilevel analysis of contextual racial effects. *Policing: An International Journal*, ahead-of-print(ahead-of-print). doi:https://doi.org/10.1108/pijpsm-11-2020- 0176.

Arafa, A., Saleh, L.H. and Senosy, S.A. (2020). Age-related differences in driving behaviors among non-professional drivers in Egypt. *PLOS ONE*, 15(9), p.e0238516. doi:https://doi.org/10.1371/journal.pone.0238516.

Makofske, M. (2020). *Pretextual Traffic Stops and Racial Disparities in their Use*. [online] MPRA Paper. Available at: https://ideas.repec.org/p/pra/mprapa/100792.html [Accessed 12 Dec. 2024].

Saville, K., Berger, D. and Levman, J. (2024). Mitigating Bias Due to Race and Gender in Machine Learning Predictions of Traffic Stop Outcomes. *Information*, [online] 15(11), p.687. doi:https://doi.org/10.3390/info15110687.

Fu, C. and Liu, H. (2020). Investigating influence factors of traffic violations at signalized intersections using data gathered from traffic enforcement camera. *PLOS ONE*, 15(3), p.e0229653. doi:https://doi.org/10.1371/journal.pone.0229653.

Oliveira, D.D., Rampinelli, M., Tozatto, G.Z., Andreão, R.V. and Müller, S.M.T. (2021). Forecasting vehicular traffic flow using MLP and LSTM. *Neural Computing and Applications*, 33(24), pp.17245–17256. doi:https://doi.org/10.1007/s00521-021-06315-w.

Lacherre, J., José Luis Castillo-Sequera and Mauricio, D. (2024). Factors, Prediction, and Explainability of Vehicle Accident Risk Due to Driving Behavior through Machine Learning: A Systematic Literature Review, 2013–2023. *Computation*, 12(7), pp.131–131. doi:https://doi.org/10.3390/computation12070131.

Datatechnotes.com. (2024). *Sequence Prediction with GRU Model in PyTorch*. [online] Available at: https://www.datatechnotes.com/2024/05/sequence-prediction-with-gru-model-in.html [Accessed 12 Dec. 2024].

AIML.com (2023). *Compare the different Sequence models (RNN, LSTM, GRU, and Transformers)*. [online] AIML.com. Available at: https://aiml.com/compare-the-different-sequence-models-rnn-lstm-gru-and-transformers/.

Yan, Y., Guo, L., Li, J., Yu, Z., Sun, S., Xu, T., Zhao, H. and Guo, L. (2024). Hybrid GRU–Random Forest Model for Accurate Atmospheric Duct Detection with Incomplete Sounding Data. *Remote Sensing*, [online] 16(22), pp.4308–4308. doi:https://doi.org/10.3390/rs16224308.