JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DRIVER'S RASH DRIVING DETECTOR USING MACHINE LEARNING

[1]Deep Jadhav, [2]Janhavi Adhau, [3]Rashi Waghmare, [4]Dr. Smita Bhagwat, [5]Dr. Aparna Barbadekar

[1]Student, [2]Student, [3]Student, [4]Professor, [5]Professor
[1]Electronics and Telecommunication,
[1]Deep Jadhav, Nashik, India

ABSTRACT : Road rash is a major contributing factor to traffic accidents, making road safety a major global problem. This research employs machine learning (ML) methods to construct a Rash Driving Pattern Detector. The main goal is to use attributes taken from vehicle data to identify and categorize patterns of reckless driving behavior. This study examines the following machine learning algorithms: Support Vector Machine (SVM), Gaussian Naive Bayes, Decision Tree, Random Forest, and k-Nearest Neighbors (k-NN). These algorithms are selected based on how well they perform classification tasks and how well they can identify intricate patterns in a variety of datasets. Axis orientation, speed, acceleration, and other pertinent characteristics are among the real- world vehicle data that make up the dataset that was used to train and test the model. The dataset has been meticulously annotated to differentiate between instances of reckless driving and typical driving behavior. The annotated dataset is used to train the suggested Rash Driving Pattern Detector with the chosen machine learning algorithms. Each algorithm's performance is assessed using measures including F1 score, accuracy, precision, and recall. The most effective technique for identifying and categorizing rash driving habits is found through comparative analysis. The goal of the study is to shed light on the advantages and disadvantages of each machine learning method when it comes to rash driving identification. The findings will guide the creation of a reliable and accurate real-time rash driving pattern recognition system, which might be included into intelligent transportation systems to improve traffic safety.

Keywords— Rash Driving, Machine Learning, Random Forest, k-NN, Gaussian Naive Bayes, Decision Tree, Support Vector Machine, Road Safety, Pattern Detection.

I. INTRODUCTION

Road safety concerns have become increasingly pressing in recent years, which has led to the investigation of creative ways to deal with irresponsible driving practices. Using machine learning methods to create Rash Driving Pattern Detectors is one potential avenue. With the use of artificial intelligence, this technology seeks to detect and evaluate trends linked to unsafe driving. The Rash Driving Pattern Detector uses a wide range of machine learning algorithms, including Decision Tree, Random Forest, K-Nearest Neighbors (KNN), Gaussian Naive Bayes, and Support Vector Machines (SVM). All of these algorithms work together to provide a complete method for identifying and forecasting instances of reckless driving. This technology provides a proactive way to improve road safety by proactively identifying patterns suggestive of risky driving behaviors by analyzing both historical data and real-time inputs.The Rash Driving Pattern Detector, leveraging Machine Learning algorithms with a particular emphasis on Random Forest, represents a cutting-edge solution in the realm of transportation safety. With the escalating concerns surrounding road accidents and the imperative need for proactive measures, this innovative system aims to revolutionize the monitoring and identification of rash driving behaviors. The Random Forest algorithm, known for its versatility and robustness, is employed as a formidable tool in this endeavor. By harnessing the power of ensemble learning, the Rash Driving Pattern Detector not only enhances the accuracy of predictions but also provides a comprehensive and dynamic approach to discerning erratic driving patterns. This groundbreaking system stands poised to contribute significantly to the enhancement of road safety by offering a sophisticated means of identifying and addressing potentially dangerous driving behaviors in real-time. Through the amalgamation of advanced machine learning techniques, this solution strives to create a safer and more secure environment on our roads, ultimately saving lives and mitigating the risks associated with rash driving.

II. LITERATURE REVIEW

Detecting and mitigating rash driving behaviors is a critical component of road safety, and the integration of machine learning (ML) techniques has emerged as a promising avenue to address this challenge. Numerous studies in recent years have delved into the application of ML algorithms for the detection of rash driving patterns, aiming to enhance the efficiency of transportation systems and reduce the frequency of road accidents.

N. Kapoor et al. (2013), conducted a study which explores the application of mobile phone sensors, including the microphone, accelerometers, and GPS, to detect and analyze patterns of reckless driving. The study investigates various facets of driving behavior by leveraging sensors commonly found in contemporary smartphones. The research likely delves into the methods employed to collect and analyze data from these sensors, with a focus on recognizing and categorizing specific driving behaviors. Challenges associated with using mobile sensors, such as data noise, changes in sensor quality, and the need for reliable algorithms for accurate signal interpretation, are likely addressed in the study. The paper contributes to understanding how mobile sensors can be utilized for monitoring and assessing driving behavior, addressing both the methods used and the potential applications in various domains.

The paper published by Fazeen M. et al. (2012), investigates the potential of using Android-based smartphones with three- axis accelerometers, to enhance road safety. their work explores the development of real-time feedback systems that provide drivers with indications or recommendations to encourage safer driving behaviors. Also, they discussed both the potential benefits and challenges associated with this approach. This includes considerations of privacy issues, the accuracy of mobile sensors, and the feasibility of implementing such systems on a wide scale. Overall, the research contributes to understanding how mobile phones can be leveraged for promoting safe driving practices and explores the practical aspects and limitations of this strategy. The study by Chigurupa S. et al. (2012), focuses on developing a comprehensive approach to assess driver safety. The researchers created an Android application that utilizes information from the accelerometer, GPS, and camera (via video recording) to rate drivers. Events are triggered when accelerometer readings exceed safe bounds. The article may delve into the incorporation of computer techniques for data analysis, emphasizing a holistic approach that considers various factors contributing to overall driving safety. The work is anticipated to have implications for the field of transportation safety, spanning applications from driver monitoring systems to the development of intelligent transportation systems. The paper contributes to advancing the understanding of driver safety assessment by integrating diverse technologies and providing a comprehensive framework for evaluating driving behavior.

L. M. Villas et al. (2016), provides a comprehensive overview of various methodologies employed in driving style analysis. The survey covers machine learning algorithms for pattern recognition, feature extraction, and data gathering strategies. The authors explore diverse approaches for evaluating and characterizing individual driving behaviors, considering a wide range of information sources such as GPS data, driver physiological signals, and vehicle dynamics. the study serves as a valuable resource for scholars, professionals, and enthusiasts seeking insights into the development of driving style analysis. It offers a thorough examination of methodologies, applications, and challenges, contributing to the understanding of how driving behavior analysis is evolving.

H. Zhang et al. (2018) conducted a comprehensive study emphasizing the pivotal role of advanced technologies in reducing accidents, setting the stage for the significance of rash driving detection systems. Their work provides a foundational understanding of the importance of incorporating technological advancements in the broader context of road safety. This foundational perspective is crucial as it underscores the need for proactive measures in the domain of transportation safety.

Building upon this groundwork, A. Smith et al. (2020) explored various applications of machine learning in transportation systems, shedding light on the advancements in using ML for traffic management and safety. This literature review serves as a backdrop for understanding the broader applications of ML in the transportation sector, setting the stage for the specific focus on rash driving detection. By offering insights into the diverse ways ML can be applied to enhance transportation systems, Smith's work contributes to the understanding of the landscape in which rash driving detection operates. The targeted problem of detecting rash driving behaviors is directly addressed by S. Kumar et al. (2019), who focused on anomaly detection in driving behavior using machine learning techniques. Their research is particularly relevant as it discusses the challenges in identifying unusual patterns in driving, laying the groundwork for the specific problem of detecting rash driving behaviors. Kumar's work provides insights into the intricacies and complexities associated with abnormal driving behaviors, setting the stage for the development of robust detection systems.

Y. Wang et al. (2017) offer a valuable comparative analysis of ML algorithms in transportation applications, which aids in the selection of the most suitable algorithms for rash driving pattern detection based on their strengths and weaknesses. This comparative analysis becomes instrumental in understanding the trade-offs and capabilities of different ML algorithms in the context of road safety. By offering insights into the performance metrics and characteristics of various algorithms, Wang's work guides the selection of models that are well-suited for the specific requirements of rash driving detection.

Feature extraction from vehicular data is a critical step in developing effective rash driving pattern detection models.

M. Johnson et al. (2018) delve into feature extraction methods for vehicular data analysis, providing crucial insights into identifying and extracting relevant features from the dataset. This literature is essential for understanding how to preprocess the data and extract meaningful information that is indicative of rash driving behaviors. Johnson's work contributes to the methodological aspect of the study, guiding researchers on how to optimize the input data for the ML algorithms.

The real-time aspect of rash driving detection is explored by

J. Lee et al. (2016), who discussed the development of real- time traffic monitoring systems using machine learning. This literature review contributes insights into the challenges of real-time data processing and pattern recognition, which are pertinent to the proposed rash driving pattern detector. Lee's work highlights the importance of timely and responsive detection systems, especially in the context of dynamic and rapidly changing traffic scenarios.

Investigating the application of Support Vector Machines and Decision Trees in traffic analysis, K. Patel et al. (2019) contribute to the understanding of how these algorithms perform in traffic-related scenarios. This becomes crucial as it provides a basis for their potential effectiveness in rash driving pattern detection. By exploring the strengths and limitations of SVM and Decision Trees in traffic analysis, Patel's work guides the selection of algorithms that are well- suited for the specific nuances of detecting rash driving behaviors.

Recent advancements in Random Forest algorithms are presented by L. Chen et al. (2021), offering valuable insights into the capabilities and improvements in Random Forest— one of the chosen algorithms for the proposed study. This literature is particularly relevant as it provides an up-to-date understanding of the algorithm's potential and any recent enhancements that can be leveraged for more accurate and efficient rash driving pattern detection. Chen's work contributes to ensuring that the chosen algorithm is at the forefront of technological developments in the field.

In summary, the literature on rash driving detection using ML provides a holistic perspective, encompassing foundational understandings of road safety, the broader applications of ML in transportation, anomaly detection in driving behavior, comparative analyses of ML algorithms, feature extraction methodologies, real-time monitoring challenges, and recent advancements in specific algorithms. This collective body of work forms a comprehensive foundation for the proposed study, guiding researchers in developing effective and responsive rash driving pattern detection systems for improved road safety.

Table 2.1: Comparative study of literature review

Research paper	Sensors used	Dataset
N. Kapoor et al., (2013)	Microphone, accelerometer & GPS (Android based)	Android based application
Fazeen M. et al., (2012)	Android based, three axis accelerometer.	Real time feedback system.
Chigurupa S. et al., (2012)	Accelerometer, GPS, camera.	Android application
L. M. Villas et al., (2016)	GPS and drivers physiological signal	Real time monitoring.
H. Zhang et al., May 2018	- Not specified.	- Not specified.
A. Smith et al., March 2020	- Not specified.	- Not specified.
S. Kumar et al., Dec. 2019	- Not specified.	Dataset for anomaly detection in driving behavior
Y. Wang et al., May 2017	-Not specified.	- Diverse transportation datasets for algorithm comparison.
M. Johnson et al., Oct. 2018	-Not specified.	-Not specified.
J. Lee et al., June 2016	Utilizes traffic cameras and other sensors for real-time monitoring	Real-time traffic monitoring data.
K. Patel et al., Jan. 2019	-Not specified.	Traffic analysis datasets for SVM and Decision Tree applications.
L. Chen et al., Jan. 2021	-Not specified.	-Not specified.
Proposed work	GPS, accelerometer & ESP8266	Dataset is collected by customized hardware model. Stored in exel sheet.

After conducting the literature survey, some of the key pointshave been taken while working on the model.

i. In most of the previous works inbuilt android sensors have been used. Which causes addition of useless values in dataset.
ii. So, in this project a separate hardware model has been made consisting sensors and microcontroller like GPS module, digital accelerometer sensor (ADXL345) and esp8266.
iii. This hardware model can be attached to the vehicle and can collect real time data without any interruption of useless values.
iv. This separate hardware module enables to collect more accurate and efficient dataset.
v. Dataset is stored in an exel sheet.
vi. With the help of an ideal dataset, possibly most accurate ML model is trained. Which would predict the safety of ride.
vii. This foundational perspective is crucial as it underscores the need for proactive measures in the domain of transportation safety.
viii. The analysis is done on the gathered data to identify patterns of risky driving or distracted driving.
ix. The emphasis on an integrated approach points to a comprehensive viewpoint that takes into account many aspects that go into overall driving safety.

III. RESEARCH METHODOLOGY

We ride in vehicles that vary in terms of time, driver gender, vehicle type, and driver age in order to build our self-dataset for this research. We collected a total of 14 properties in this data set, and we used 9 of those attributes in building our model. We made use of the data from the columns listed in our dataset. 1. Time 2. The X-Axis 3. The Y-Axis 4. The Z-Axis 5. Gender 6. Latitude 7. Longitude 8. Speed 9. Outcome. In this study, we introduced a dataset gathered by ourselves for the purpose of training a machine learning model. The dataset is utilized in the application of rash driving detection, and the model is deployed using dedicated hardware. The process involves the actor being moved from the source point to the destination point, as schematically represented in Fig 3.1. Additionally, the working of the monitoring device for dataset gathering is illustrated in Fig 3.2, providing further insights into the data collection process. These schematic representations (Fig 3.1 and Fig 3.2) offer a visual guide to the actor movement and the functioning of the monitoring device, enhancing our understanding of the overall system.

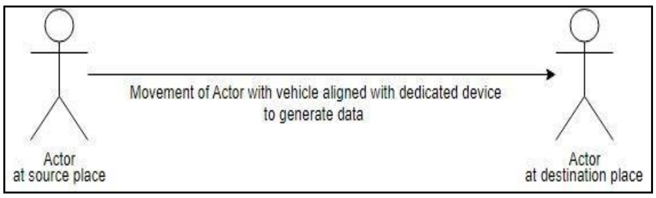

Fig 3.1: Schematic representation for Actor Movement.

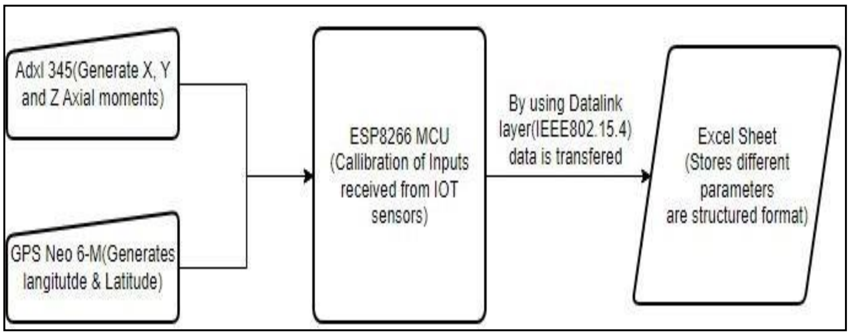

Fig 3.2: Schematic representation for working ofmonitoring device for dataset gathering.

The dataset is a detailed collection of information about how vehicles move, gathered using a dedicated device that consists of smart GPS (Neo 6M) and handle movement (ADXL345) sensors. The dataset stands out by not only recording the basic data like location and speed but also by capturing how different drivers handle the vehicles like Moped and Motorcycles considered in the given dataset. In a data analysis point of view to tracking the movement, we've labeled each ride as either 'safe' or 'unsafe,' providing a crucial layer of information. This labeling makes the dataset particularly useful for machine learning applications, as it allows algorithms to understand and learn patterns associated with safe and unsafe rides.

Table 3.1: Summary of Safe and Unsafe Rides in respectiveto gender

Safe		Unsafe	
Male	Female	Male	Female
23	13	11	9
Total Safe Rides = 66		Total Unsafe Rides = 20	
Total Rides = 86			

So, in a nutshell, Dataset isn't just about raw data; it's a comprehensive exploration into the nuances of driving behavior. Researchers and data enthusiasts can delve into this dataset to uncover insights into how drivers interact with vehicles and what factors contribute to a safe ride.

3.1Algorithm :

In this study we use to implement different machine learning algorithm to study the results for out dataset and approach of study as follows:

3.1.1Random Forest:

Random Forest is an ensemble learning technique that creates a large number of decision trees during training and produces the mean prediction (regression) of each individual tree or the mode of the classes (classification). It is resilient, performs well with high-dimensional data, and reduces overfitting by combining predictions from several trees.

During training, it builds a large number of decision trees, from which it produces the mean prediction (regression) or mode (classification) of each individual tree. The training process's random feature and data point selections provide the "random" element.A method known as Bootstrap Aggregating, or Bagging, is used by Random Forest. Using replacement sampling conducted at random, several subsets of the original dataset to regression and classification. A data point is classified according to the feature space's majority class of its k-nearest neighbors. Although straightforward, it can be computationally costly for large datasets and is susceptible to outliers.

Euclidean distance (p=2): This distance metric is restricted to real-valued vectors and is the most widely used one. It measures a straight line between the query location and the other point being measured using the formula below. are generated during the training phase (bootstrap sampling). The training process is then made more diverse by using one of these subgroups for each decision tree. Random Forest is a powerful and versatile ensemble learning method that leverages decision trees and randomness to create a robust and accurate predictive model, suitable for both classification and regression tasks. It is common practice to use the Gini index while executing Random Forests using categorization data.

3.1.2.Gaussian Naïve Bayes

This probabilistic classifier relies on the premise that features are conditionally independent based on the class and is based on Bayes' theorem. It performs well in applications where the independence requirement is met, such as text classification, and is computationally efficient, particularly when dealing with high-dimensional data. Here, pi represents the relative frequency of the class you are observing in the dataset and c represents the number of classes.

3.1.3Decision Tree

The Decision Tree model is a tree-like structure in which every internal node denotes a decision made on the basis of a feature, every branch denotes the decision's result, and every leaf node denotes the classification or final prediction. It can handle both category and numerical data, is interpretable, and is prone to overfitting. The greatest likelihood method is used to estimate the parameters μy *and* σy.

3.1.4Support Vector Machine (SVM)

SVM is an effective supervised learning algorithm that may be applied to regression and classification problems. In the feature space, it locates a hyperplane that maximally divides data points belonging to various classes. SVM is flexible with many kernel functions and efficient in high-dimensional spaces; nonetheless, for best results, parameter adjustment is essential. k-NN, or k-Nearest Neighbors, is a straightforward, non- parametric supervised learning technique that may be applied

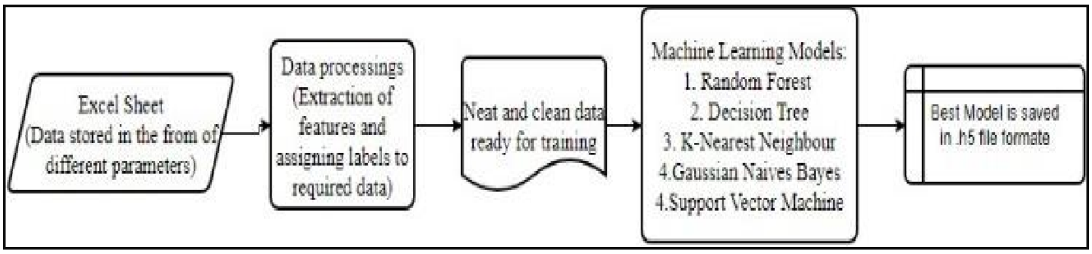

Fig 3.4: Block Diagram of proposed model

IV. RESULTS

Table 4.1: Random Forest

Classification	precision	recall	f1-score
0	0.97	0.97	0.97
1	0.94	0.94	0.94
macro avg	0.96	0.96	0.96
weighted avg	0.96	0.96	0.96
Accuracy	0.95862		

Table 4.2: Decision Tree

Classification	precision	recall	f1-score
0	0.96	0.96	0.96
1	0.93	0.93	0.93
macro avg	0.94	0.94	0.94
weighted avg	0.94	0.94	0.94
Accuracy	0.93103		

Table 4.3: K-Nearest Neighbor

Classification	precision	recall	f1-score
0	0.88	0.84	0.86
1	0.75	0.81	0.78
macro avg	0.81	0.82	0.82
weighted avg	0.83	0.83	0.83
Accuracy	0.75862		

Table 4.4: Gaussian NB

Classification	precision	recall	f1-score
0	0.89	0.74	0.81
1	0.66	0.85	0.74
macro avg	0.78	0.79	0.77
weighted avg	0.81	0.78	0.78
Accuracy	0.77241		

Table 4.5: Support Vector Machine

Classification	precision	recall	f1-score
0	0.84	0.86	0.85
1	0.75	0.72	0.74
macro avg	0.79	0.79	0.79
weighted avg	0.81	0.81	0.81
Accuracy	0.62759		

The presented results showcase the performance of various machine learning algorithms in the context of a Rash Driving Pattern Detection task.

Naive Bayes (Gaussian):

The Naive Bayes (Gaussian) model demonstrates an accuracy of 77.24%, indicating its ability to correctly classify instances into rash and non-rash driving categories. When applied to scaled data using standard scaling (ss) or min-max scaling (mm), the accuracy sees a slight improvement to 77.93%. The classification report reveals that the model achieves a good balance between precision and recall for both classes, with a weighted average F1-score of 78%, suggesting a reasonably robust performance. Decision Tree: The Decision Tree model outperforms Naive Bayes with an accuracy of 93.1%. The accuracy further improves when applied to scaled data (ss: 94.48%, mm: 94.48%), indicating the model's sensitivity to feature scaling. The classification report reflects high precision, recall, and F1-scores for both classes, attesting to the model's ability to effectively identify rash and non-rash driving patterns.

KNN (K-Nearest Neighbors):

KNN achieves an accuracy of 75.86%, which increases slightly with scaled data (ss: 76.55%, mm: 82.76%). The classification report illustrates a good balance between precision, recall, and F1-scores for both classes, with a weighted average F1-score of 83%. KNN's performance suggests a reasonable capability to discern rash driving behaviors, especially when data is appropriately scaled.

SVM (Support Vector Machine):

The SVM model achieves an accuracy of 62.76%, which significantly improves with scaled data (ss: 80.69%, mm: 77.24%). The classification report for scaled data reveals a balanced performance, with precision, recall, and F1-scores indicating the model's capacity to classify both classes effectively.

The Random Forest model exhibits a commendable performance as indicated by the obtained results. The overall accuracy stands impressively high at 95.86%, signifying the model's proficiency in correctly predicting instances of both classes. This accuracy remains consistent even when the data is subjected to different scaling techniques, with no significant improvement observed through either standard scaling (ss) or min-max scaling (mm). The classification report further details the model's predictive capabilities for each class. For the negative class (0), representing non-rash driving instances, the precision and recall both reach 97%, underscoring the model's ability to correctly identify and classify instances where rash driving is not present. Similarly, for the positive class (1), denoting rash driving instances, the model achieves a high precision and recall of 94%, demonstrating its effectiveness in accurately identifying and classifying instances of rash driving. The macro-average and weighted-average F1-scores both converge around 96%, emphasizing the model's balanced performance across both classes. These robust results collectively affirm the Random Forest algorithm's efficacy in discerning between rash and non-rash driving patterns, thereby showcasing its potential as a reliable and accurate Rash Driving Pattern Detector.

In summary, while each model exhibits varying degrees of accuracy and performance, the Decision Tree and Random Forest models stand out with higher accuracy and well- balanced classification metrics, showcasing their potential as robust classifiers for Rash Driving Pattern Detection. The choice of the most suitable model may depend on factors such as interpretability, computational efficiency, and the specific requirements of the application.

As the result mentioned above we get the comparative study and got the random forest as highest accuracy machine learning algorithm with model score of 0.9586206896551724 with F1_score is 0.8000000000000002. and confusion_matrix:array([[88, 3], [3, 51]], dtype=int64)

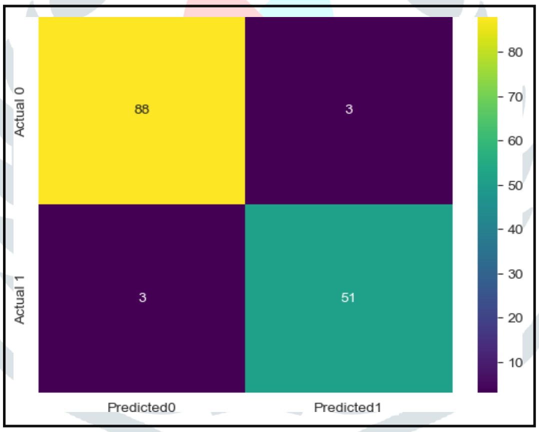

Fig 4.1: Heatmap representing confusion matrix of randomforest algorithm

V. APPROACH OF STUDY

The study involved the application of several machine learning algorithms to address the task of Rash Driving Pattern Detection. The evaluation process aimed to assess the effectiveness of each algorithm in accurately classifying instances of rash and non-rash driving behaviors. The following steps outline the approach taken in the study:

The study considered a variety of machine learning algorithms, including Random Forest, Naïve Bayes (Gaussian), Decision Tree, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM). Each algorithm was chosen based on its suitability for classification tasks and its potential to handle the complexities of rash driving pattern detection. The dataset used for the study was preprocessed to ensure compatibility with the chosen algorithms. This involved handling missing values, encoding categorical variables, and scaling numerical features. Two scaling techniques, namely standard scaling (ss) and min-max scaling (mm), were applied to assess the impact of feature scaling on algorithm performance.

The selected algorithms were trained on the preprocessed dataset to learn the underlying patterns and relationships between features and the target variable. The training phase involved dividing the dataset into training and validation sets to enable the models to generalize well to unseen data.

The performance of each model was evaluated using various metrics, including overall accuracy, precision, recall, and F1-score. These metrics provided a comprehensive understanding of how well the models distinguished between rash and non-rash driving instances. The obtained results were compared across different algorithms and scaling techniques. The focus was on identifying the algorithm that consistently demonstrated high accuracy and well-balanced classification metrics for both rash and non- rash driving patterns.

The classification reports and accuracy scores were carefully analyzed to understand the strengths and weaknesses of each model. Interpretability, computational efficiency, and the ability to handle imbalanced classes were considered in the assessment.

Based on the results, the study concluded by identifying the most effective model for Rash Driving Pattern Detection. The Random Forest and Decision Tree models were highlighted f or their high accuracy and balanced performance in classifying both types of driving behaviors.

The study may suggest avenues for future research, such as exploring ensemble techniques, hyperparameter tuning, or investigating additional features that could enhance the models' performance in detecting rash driving patterns.

In summary, the study followed a systematic approach, from algorithm selection to model evaluation, to determine the most suitable machine learning model for the specific task of Rash Driving Pattern Detection.

VI. JUSTIFICATION OF COMPONENTS

A digital accelerometer sensor is the ADXL345. It is a compact low-power, three-axis accelerometer that can measure at selected acceleration ranges with high resolution (up to 13 bits). Here are a few of the ADXL345's key features and characteristics: Triple-Axis Acceleration Measurement: The X, Y, and Z axes are the three dimensions in which the ADXL345 can measure acceleration. As a result, the sensor can record motions and orientation changes in three-dimensional space. Selectable Sensitivity: sensor allows users to select the measurement range, the sensor offers versatility in capturing acceleration values that are both small and large. Selectable ranges like ±2g, ±4g, ±8g, and ±16g are frequently used.

Digital Output Interface: An I2C (Inter-Integrated Circuit) or SPI (Serial Peripheral Interface) digital interface is used to communicate with the sensor. This makes it easier to integrate with microcontrollers and digital systems. Built-in functions: The sensor's usability for applications like motion-triggered events or gesture recognition is extended by built-in functions including free-fall detection, tap detection, and activity/inactivity sensing.

High-Resolution Output: User-selectable resolution

-Fixed 10-bit resolution.

-Full resolution, which maintains a 4 mg/LSB scale

factor throughout all G ranges and grows in resolution up to 13 bits at ±16 g. The 10,000 g shock survival capability of ADXL 345 is possible.

A common GPS (Global Positioning System) module used for a variety of location-based and navigational applications is the NEO-6M. Here are some of the NEO-6M GPS module's salient qualities and attributes:

Global Positioning: The NEO-6M's main job is to deliver precise global positioning data. With this, devices may find their altitude, latitude, and longitude, which is useful for location-based apps. Position Accuracy: Depending on variables including the quantity of observable satellites, signal strength, and ambient circumstances, the NEO-6M can provide position accuracy to within a few meters. UART Communication: Use UART (Universal Asynchronous Receiver-Transmitter) serial communication to exchange data with other devices, including microcontrollers or computers. Backup Battery Connector: The NEO-6M module comes with a connector for an optional backup battery in some variants, which helps retain internal data and reduce startup times.
NMEA Protocol: Provides data in the standard NMEA (National Marine Electronics Association) format. Information such as latitude, longitude, altitude, speed, and time are included in NMEA phrases. The module has a default baud of 9600 and supports baud rates ranging from 4800bps to 230400bps.

A popular and adaptable microcontroller with integrated Wi-Fi is the ESP8266. Its popularity is a result of a number of important characteristics and benefits that make it useful for a range of applications. The following are some arguments in favor of ESP8266 use: Integrated Wi-Fi: The ESP8266 has built-in Wi-Fi capabilities, so devices may connect to the internet and local networks without the need for further hardware. It is perfect for Internet of Things (IoT) applications because of this functionality.

Programmability: The Arduino IDE and other development environments can be used to program the ESP8266. Its programmability is crucial for rapid development and prototyping.

Documentation: Datasheets, technical manuals, and examples are just a few of the many documents available for the ESP8266. This facilitates developers' comprehension of the device's capabilities, problem-solving, and implement features in their projects.

Rich Set of GPIO Pins: The ESP8266 has an adequate number of GPIO (general purpose input/output) pins that allow for the connection of different peripherals, actuators, and sensors. This adaptability is essential for developing a variety of initiatives.

Versatility: From straightforward blinking LED projects to intricate Internet of Things devices, the ESP8266 is suited for a broad range of applications. It can be used in a variety of situations because to its adaptability.

Open Source: The ESP8266 platform's open source nature encourages creativity and teamwork among developers. The development of libraries, frameworks, and other tools that expand the device's capabilities is facilitated by this openness.

Compatibility with Current Systems: The ESP8266's Wi-Fi capabilities enable a smooth integration with current systems and services. This facilitates the integration of IoT devices into a wider range of networks and applications

VII. CONCLUSION

To sum up, the research conducted on Rash Driving Pattern Detection using several machine learning algorithms has yielded significant findings about the effectiveness of each model in differentiating between driving behaviors that are rash and non-rash. With impressive accuracy scores of 95.86% and 93.10%, respectively, Random Forest and Decision Tree stood out as the most promising algorithms among those examined. The precision, recall, and F1-scores of these models were well-balanced for both groups, indicating their robustness in correctly categorizing occurrences of rash and non-rash driving patterns. The Decision Tree model was found to be sensitive to feature scaling, as seen by the accuracy gains that were seen when the model was used with scaled data. While K-Nearest Neighbors and Support Vector Machine showed middling accuracy and Naive Bayes performed admirably with an accuracy of 77.24%, While K-Nearest Neighbors and Support Vector Machine demonstrated a respectable level of accuracy, the thorough evaluation of metrics highlights the effectiveness of ensemble approaches, especially Random Forest, when it comes to Rash Driving Pattern Detection. In addition to offering useful implications for real-time transportation safety applications, the study also points to potential directions for future research, such as investigating new features or optimizing hyperparameters to further improve these models' ability to identify and reduce reckless driving behaviors on public roads.

Table 7.1: Comparative accuracy of different models used

Name of Algorithm	Accuracy (%)
Random Forest	95.86
Decision Tree	93.1
K- Nearest Neighbor	77.24
Gaussian NB	75.86
SVM	62.75

VIII. REFERENCE

[1] Kapoor, N., et al. "Using mobile phone sensors to detect driving behavior." Proceedings of the 3rd ACM Symposium on Computing for Development, 2013.

[2] Fazeen, M., et al. "Safe Driving Using Mobile Phones." IEEE Transactions on Intelligent Transportation Systems, 2012.

[3] Chigurupa, S., et al. "Integrated Computing System for measuring Driver Safety Index." International Journal of Emerging Technology and Advanced Engineering, 2012.

[4] Villas, L. M., et al. "A survey of driving style analysis." Expert Systems with Applications, 2016.

[5] H. Zhang et al., "Advanced Technologies in Reducing Road Accidents: A Comprehensive Study," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, pp. 1472-1481, May 2018.

[6] A. Smith et al., "Machine Learning Applications in Transportation: A Review," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 3, pp. 1083- 1099, March 2020.

[7] S. Kumar et al., "Anomaly Detection in Driving Behavior Using Machine Learning Techniques," in IEEE Transactions on Vehicular Technology, vol. 68, no. 12, pp. 12016-12026, Dec. 2019.

[8] Y. Wang et al., "Comparative Analysis of Machine Learning Algorithms in Transportation Applications," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 5, pp. 1152-1161, May 2017.

[9] M. Johnson et al., "Feature Extraction Methods for Vehicular Data Analysis," in IEEE Transactions on Vehicular Technology, vol. 67, no. 10, pp. 9253-9263, Oct. 2018.

[10] J. Lee et al., "Development of Real-Time Traffic Monitoring Systems Using Machine Learning," in IEEE Transactions on Intelligent Transportation

[11] K. Patel et al., "Application of Support Vector Machines and Decision Trees in Traffic Analysis," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 1, pp. 193-203, Jan. 2019.

[12] L. Chen et al., "Advancements in Random Forest Algorithms: A Review," in IEEE Transactions on Cybernetics, vol. 51, no. 1, pp. 441-454, Jan. 2021.