

Article

# Driving Style and Traffic Prediction with Artificial Neural Networks Using On-Board Diagnostics and Smartphone Sensors

Ghaith Al-refai <sup>1,\*</sup> , Mohammed Al-refai <sup>2</sup>  and Ahmad Alzu'bi <sup>2</sup> <sup>1</sup> Department of Mechatronics Engineering, German Jordanian University, Amman 11180, Jordan<sup>2</sup> Department of Computer Science, Jordan University of Science and Technology, Irbid 22110, Jordan; mnalrefai@just.edu.jo (M.A.-r.); agalzubi@just.edu.jo (A.A.)

\* Correspondence: ghaith.alrefai@ju.edu.jo

**Abstract:** Driving style and road traffic play pivotal roles in the development of smart cities, influencing traffic flow, safety, and environmental sustainability. This study presents an innovative approach for detecting road traffic conditions and driving styles using On-Board Diagnostics (OBD) data and smartphone sensors. This approach offers an inexpensive implementation of prediction, as it utilizes existing vehicle data without requiring additional setups. Two Artificial Neural Network (ANN) models were employed: the first utilizes a forward neural network architecture, while the second leverages bootstrapping or bagging neural networks to enhance detection accuracy for low-labeled classes. Support Vector Machine (SVM) is implemented to serve as a baseline for comparison. Experimental results demonstrate that ANNs exhibit significant improvements in detection accuracy compared to SVM. Moreover, the neural network with bagging model showcases enhanced recall values and a substantial improvement in accurately detecting instances belonging to low-labeled classes in both driving style road traffic.

**Keywords:** deep learning; neural networks; bagging; traffic and driving style prediction; smart cities; on-board diagnostics



**Citation:** Al-refai, G.; Al-refai, M.; Alzu'bi, A. Driving Style and Traffic Prediction with Artificial Neural Networks Using On-Board Diagnostics and Smartphone Sensors. *Appl. Sci.* **2024**, *14*, 5008. <https://doi.org/10.3390/app14125008>

Academic Editor: Roberto Carballedo

Received: 20 April 2024

Revised: 4 June 2024

Accepted: 5 June 2024

Published: 8 June 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The sector of transportation and road safety has transformed in recent years due to the integration of sensor technologies into cars. Advanced smartphone sensors and On-Board Diagnostics (OBD) systems [1] are examples of in-vehicle sensors that have enabled researchers and the automobile industry to collect vast amounts of data about traffic patterns, road conditions [2], and driver behavior [3]. With this abundance of data, predictive models can be created to analyze driving habits, evaluate traffic volumes, and provide up-to-date information on road infrastructure.

In-vehicle sensor data can be collected from many sources. The Engine Control Unit (ECU) [4] provides data on engine speed, RPM, fuel consumption, and other engine parameters that can indicate aggressive or fuel-efficient driving styles. OBD offers a standardized interface to access various vehicle data, including speed, acceleration, and braking patterns, which are crucial for assessing driving style. Wheel speed sensors track the rotational speed of each wheel, allowing for the detection of wheel slip and potentially hazardous situations.

AI algorithms, in conjunction with in-vehicle data, including OBD and smartphone data, have been used in various fields. For instance, Singh et al. [5] presented a scalable vehicle CO<sub>2</sub> emission prediction model utilizing vehicle OBD-II port data in emission prediction. The proposed method employs a Long Short-Term Memory (LSTM) model based on a Recurrent Neural Network (RNN) to estimate the vehicle's CO<sub>2</sub> emissions from real-time in-vehicle sensor data. Based on OBD data and machine learning, Rivera-Campoverde et al. [6] offered an estimation of pollutant emissions in real driving situations.

OBD data are also used in predicting a vehicle's trajectory. For instance, Xing et al. [7] presented a technique that allows a vehicle's trajectory to be accurately and individually predicted based on a limited number of inter-vehicle communication signals, such as the vehicle's acceleration and speed, determined by an unsupervised clustering approach. The integration of GPS and OBD data was suggested by Xiao et al. [8] as a feasible solution for large-scale car trajectory collection.

OBD data are commonly used in fuel economy predictions. Yaw et al. [9] proposed a model to improve fuel consumption monitoring databases based on mobile phone terminals and OBD installed in taxis. Nurcahya et al. [10] proposed an approach for fuel consumption forecasting based on OBD data using a multivariate time series method. In-vehicle sensor data can be utilized in vehicle health estimation and vehicle preventive maintenance prediction as proposed in [11,12].

Several machine learning techniques, such as decision trees and random forests, can be used to predict road conditions and driving styles from in-vehicle data [13,14]. However, among traditional machine learning algorithms, Support Vector Machines (SVM) are recognized for their high accuracy in classification tasks [15]. SVM identifies the optimal hyperplane that separates data points of different classes by maximizing the margin between the support vectors of these classes. Additionally, SVM employs kernel functions to project the data into a higher-dimensional space, enabling the creation of an effective decision boundary. Therefore, SVM is used as a reference algorithm to provide a baseline for result comparison with our proposed deep learning algorithms.

Cameras and LiDAR are advanced sensors that offer visual and spatial data of the surrounding environment, enabling the detection of vehicles, pedestrians, and other road users, which is essential for understanding traffic flow and potential congestion [16,17]. The primary challenge facing computer vision approaches in driver behavior monitoring and road traffic analysis lies in their high implementation costs, which stem from the necessity of vision sensors like cameras and LiDAR, as well as powerful computing systems capable of handling heavy-duty computer vision tasks such as Convolutional Neural Networks (CNNs). This study addresses this challenge by introducing a cost-effective system that leverages in-vehicle data from smartphones and OBD sensors to accurately predict driving styles and traffic conditions. It achieves this by employing Artificial Neural Networks with architectures designed to maintain high accuracy even under conditions of imbalanced classes while also being conducive to onboard system implementation. These networks are used to forecast driving styles as aggressive or normal and to detect road traffic conditions classified as high, medium, or low traffic.

The subsequent sections of this paper are structured as follows: Section 2 reviews recent literature addressing similar challenges; Section 3 outlines the research methodology by explaining the prediction model system, dataset, and the ANN algorithms; Section 4 discusses the training techniques implemented to improve detection performance; Section 5 presents the results of the detection process and compares the deep learning model against SVM as a reference algorithm; and Section 6 concludes and summarizes the research findings and proposes avenues for future research.

## 2. Related Work

Detecting driver behavior plays a crucial role in enhancing road safety and optimizing transportation systems. Numerous studies have tackled this challenge by leveraging vehicle sensors and smartphone technology. Najah [18] presented a system that utilizes the vehicle's CAN-Bus as a source of sensory data and a smartphone as the processing unit to detect road artifacts and monitor driver behavior using machine learning. Zhang et al. [19] developed a window-based SVM model to classify drivers based on OBD data. Nirmali et al. [20] presented a vehicular data acquisition and analytics system for real-time driver behavior monitoring, anomaly detection, and alerting by utilizing a K-means clustering algorithm. Al-refai et al. [21] proposed machine learning approaches, including decision trees, random forests, and SVM, to detect road conditions, traffic conditions, and driving behavior using

OBD data and smartphone data. Kim et al. [22] proposed a system that segments in-vehicle CAN frames and evaluates each segment with a scoring method to detect driving style. Shaikh et al. [23] provided a machine learning model based on XGBoost and an ANN model to detect driving behavior using an Android smartphone connected to the OBD bus. Lattanzi and Freschi [24] proposed SVM and feed-forward neural networks to detect driving behavior. Hermawan and Husni [25] presented a literature survey of recent research related to driving behavior, including how to obtain data from onboard diagnostics (OBD-II), analyze data, model data, and evaluate models or systems.

Numerous methods exist for monitoring road traffic conditions using computer vision. These models require vision sensors such as cameras and radar. For instance, Liu et al. [26] proposed a two-tier edge computing-based model that uses video data to analyze road traffic conditions and speed under different weather conditions. Sajib et al. [27] proposed a video-based approach for road traffic monitoring by utilizing a Haar feature-based Adaboost classifier. Guo et al. [28] proposed an enhanced YOLOv3 model to detect slow-moving or stopped trains at highway-railroad grade crossings. Reddy et al. [29] proposed a deep learning CNN for predicting traffic status as dense traffic or low traffic. Ranyal et al. [30] provided a review of the latest road condition monitoring using computer vision and artificial intelligence.

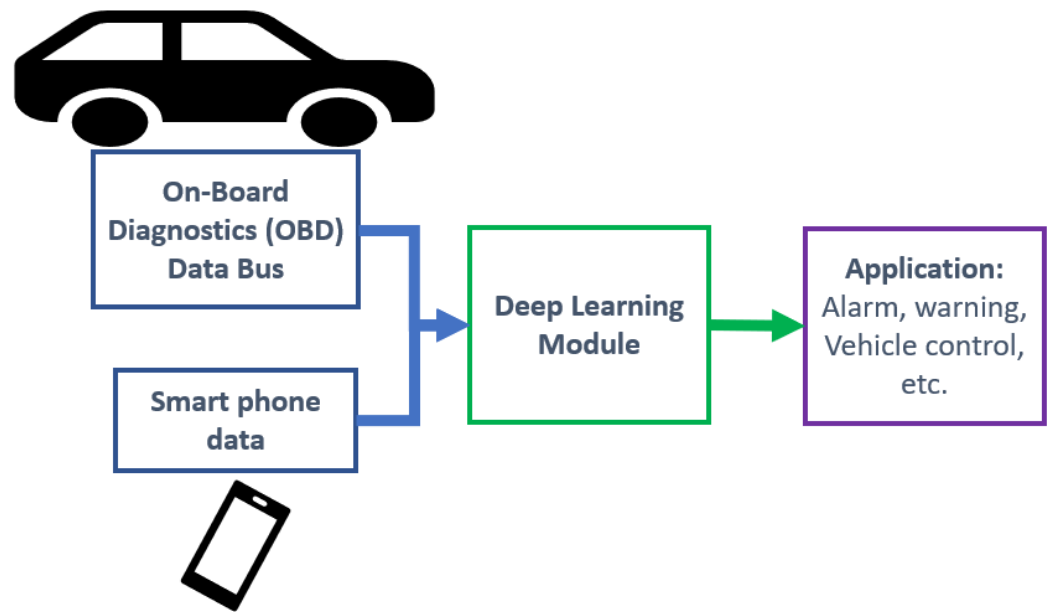
Road traffic can also be predicted using smartphone and in-vehicle sensor data. Vig and Aggarwal [31] employed Mel Frequency Cepstral Coefficients (MFCCs) and Wavelet Packet Transform (WPT) features to monitor road traffic based on smartphone sensor data. Similar approaches were proposed to detect road conditions using smartphone data in [32,33]. Chugh et al. [34] conducted a survey on approaches that use smartphone sensor data to detect road traffic conditions. Utilizing smartphone and in-vehicle sensor data for estimating road traffic conditions requires less computational power compared to computer vision approaches, yet it still offers reasonably accurate estimations of road conditions. This advantage enables the design of cost-effective monitoring systems.

In this study, we introduce a deep learning model based on neural networks to forecast driving styles and traffic conditions using in-vehicle data sourced from OBD buses and smartphone sensors. The system employs deep learning methodologies to develop a highly accurate detection model capable of handling imbalanced data labeling across various classes, given the uneven distribution of labels. Moreover, the system is designed to minimize computational demands, facilitating implementation in embedded systems and enabling integration within vehicles themselves.

### 3. Methodology

#### 3.1. Prediction Model Architecture

To deploy the suggested system, a data logging tool must be utilized to gather data from both the vehicle bus and the smartphone. Numerous affordable commercial OBD scanner tools are accessible for collecting data from the vehicle bus. Additionally, phone sensor data can be transmitted via Bluetooth to the processing module. The collected data are then fed into the deep learning module, which comprises trained neural network models designed to forecast driver behavior and road traffic conditions. Subsequently, the model predictions can be integrated into vehicle applications, such as driver alarms and warnings regarding aggressive driving styles, as well as alerts about traffic congestion. Furthermore, traffic road predictions can be fused with the navigation system to offer optimized routes for the driver. Figure 1 presents the block diagram for the proposed system, showing the data path from the vehicle and smart phone to the deep learning module and then to the vehicle application level.



**Figure 1.** The block diagram for the proposed system, which shows the system components and the data flow through the system

### 3.2. Dataset

The traffic conditions, driving behavior, and road surface dataset [35] was gathered from two vehicles: a Peugeot 207 1.4 HDi and an Opel Corsa 1.3 HDi. Data acquisition was conducted through the car's On-Board Diagnostics (OBD) system and the sensors integrated into the user's smartphone. The dataset comprises 14 input features listed below:

- Altitude change, calculated over 10 s;
- Current speed value, which is the average speed in the last 60 s;
- Speed variance in the last 60 s;
- Speed variation for every second of detection;
- Longitudinal acceleration;
- Engine load, expressed as a percentage;
- Engine coolant temperatures in degree Celsius;
- Manifold Air Pressure (MAP), a parameter used by the internal combustion engine used to compute the optimal air/fuel ratio;
- Revolutions Per Minute (RPM) of the engine;
- Mass Air Flow (MAF) Rate measured in g/s—this reading is used by the engine to set fuel delivery and spark timing;
- Intake Air Temperature (IAT) at the engine entrance;
- Vertical acceleration;
- Average fuel consumption, calculated as liters per 100 km.

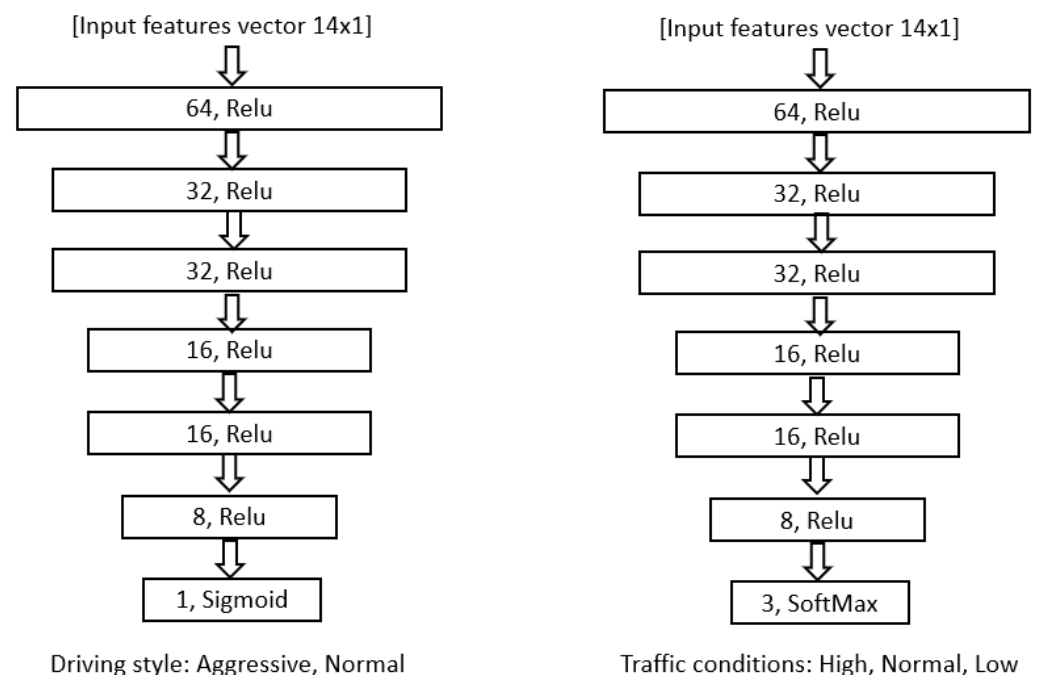
The dataset is labeled to solve three classification problems: driving style, characterized by either aggressive or normal behavior; traffic conditions, categorized as high, low, or normal; and road conditions, distinguished as full of holes, smooth, or medium. This study predominantly concentrates on the detection of driving style and traffic conditions. As depicted in Table 1, the distribution of data exhibits notable imbalances, such as only 12% of the driving style labels being classified as aggressive. Similarly, the occurrences of high and normal traffic conditions are fewer compared to low traffic conditions. The imbalanced dataset was addressed during the algorithm training phase, as elaborated in the subsequent section.

**Table 1.** Dataset size, feature numbers, and number of labels for each class.

Dataset Size	Input Features	Traffic Conditions Labels	Driving Style Labels	Road Conditions Labels
24,957 data array	14 features	18,769 low traffic 3171 normal traffic 3017 high traffic	22,089 normal driving 2868 aggressive driving	3249 full of holes 15,242 smooth 6466 even conditions

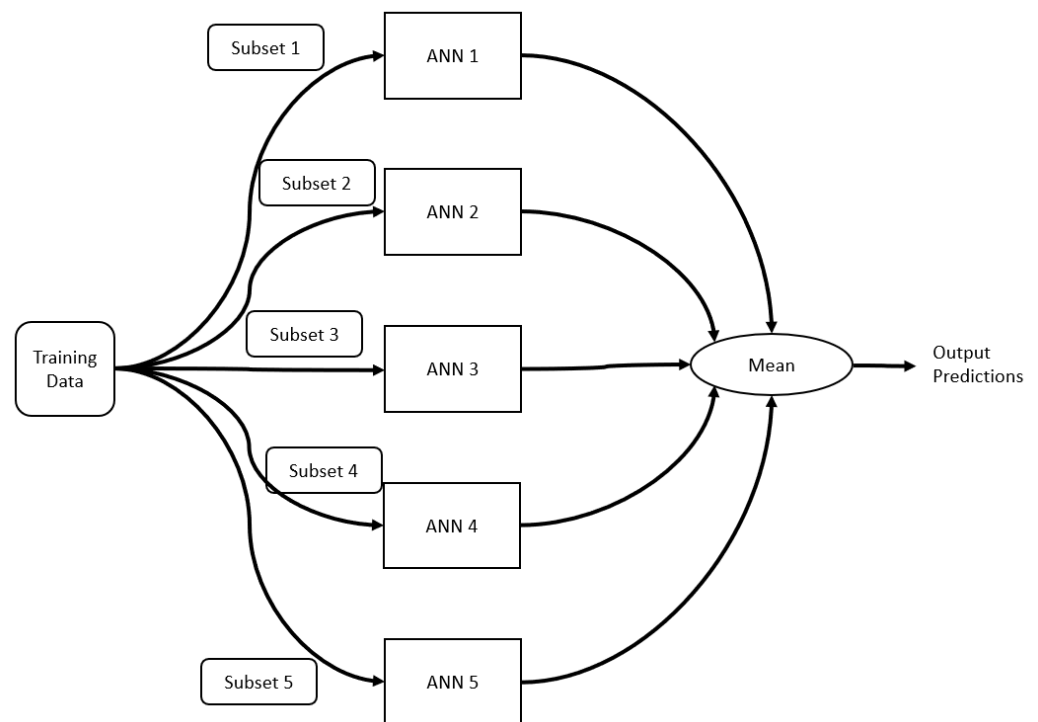
### 3.3. Deep Learning Backbone

In this study, two types of artificial neural network (ANN) models were employed. The first model follows the conventional feed-forward neural network architecture. This network comprises 14 inputs to match the input features of the dataset. The model is structured with seven layers, encompassing both the input and output layers. Driving style is a binary classification problem, with each input data instance is categorized as either aggressive or normal driving style. Thus, the Sigmoid function is employed with one neuron in the output layer to generate the probability of each class. The sigmoid function yields a value ranging from 0 to 1, indicating the probability of the input data belonging to a specific class. On the other hand, in road traffic prediction, the scenario involves three classes: low, medium, and high traffic. Consequently, the Softmax function and three neurons are utilized in the last layer to produce the probability of each class. The softmax function is utilized in multi-class problems because it normalizes the output of a set of multiple neurons into a probability distribution over multiple classes [36]. This normalization property ensures that the sum of the probabilities for all classes equals one, making it suitable for interpreting the output as probabilities. Figure 2 shows the ANN architecture used in the classification problems of driving style and road traffic.

**Figure 2.** The ANN architecture of the models utilized in the classification of driving style and road traffic.

We also employed an ANN model using Bootstrap Aggregating, commonly known as bagging. In this method, the training dataset is split into five subsets, each of which is used to train a separate ANN model. The final prediction is obtained by averaging the predictions from these five models. Bagging is employed to mitigate the risk of overfitting, particularly given the imbalanced nature of our dataset. Further elucidation on the bagging technique is provided in the results section. Figure 3 illustrates the architecture of the ANN

model incorporating the bagging approach. The displayed five ANN models in Figure 3 have the same architecture of the ANN described in Figure 2.



**Figure 3.** The ANN model’s architecture incorporates the bagging technique, where the final prediction is determined by averaging the outcomes of the five trained models.

#### 4. Training Process

To facilitate the training and evaluation of the neural network models, the data are partitioned into three distinct categories: training, validation, and testing [37]. The training dataset is employed during the training phase to adjust the algorithm’s weights and compute the loss function. Validation data are utilized to assess the system’s accuracy and loss following each epoch of training. This validation dataset serves as a metric for evaluating the model’s ability to generalize predictions to unseen data and identifies potential issues such as overfitting or underfitting. The testing dataset is employed to assess the performance of the final models post-training.

The dataset was partitioned into 80% for training and 20% for testing. Within the training dataset, 20% was allocated for model validation. This division resulted in 15,972 training instances, 3993 validation instances, and 4992 testing instances. The dataset was randomly divided among the categories to prevent any bias towards a specific class.

The training dataset for driving style comprises 17,686 instances labeled as normal driving and 2267 instances labeled as aggressive driving. Regarding traffic conditions, there are 14,972 instances labeled as low traffic, along with 2402 and 2591 instances labeled as high and medium traffic, respectively. It is notable that there is a significant imbalance in the distribution of labels, particularly observed in the normal driving style and low traffic conditions categories. This imbalance is influenced by both driver behavior and the specific traffic conditions on the roads from which the dataset originates.

The initial training of the proposed ANN algorithms utilized the original dataset without adjustments. However, it became evident that the validation loss exceeded the training loss and failed to converge with an increasing number of epochs, suggesting a failure to generalize predictions to unseen data; this phenomenon is known as overfitting. To address the overfitting issue, we applied random upsampling to the low-labeled classes. This technique involves duplicating instances from the minority class randomly to balance the class distribution within the training dataset. By doing so, we aimed to mitigate system

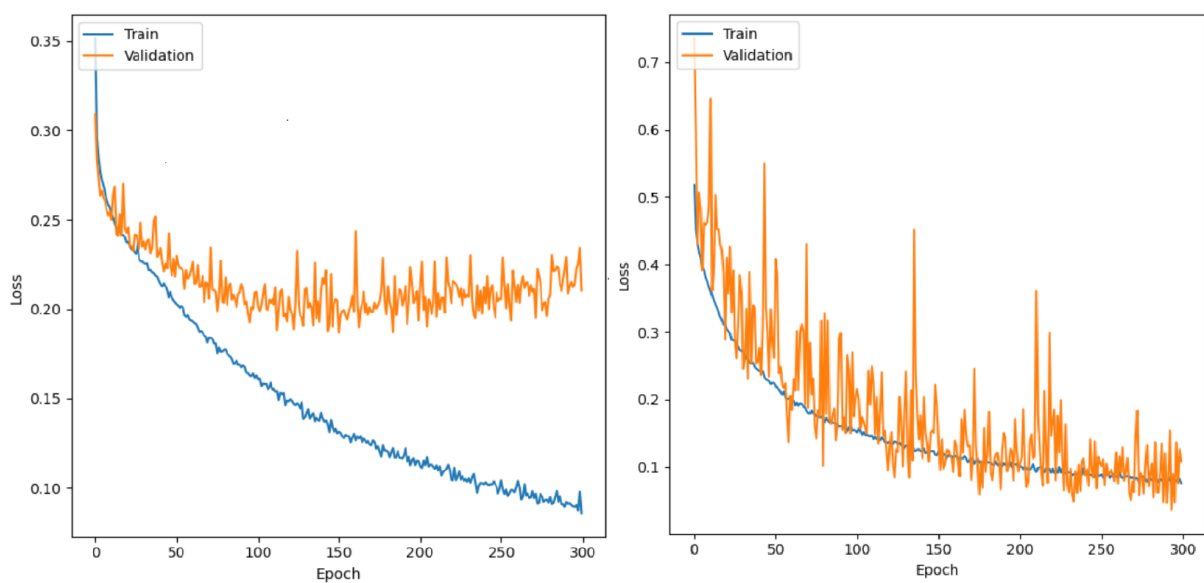
overfitting. The choice of random upsampling was made to retain the information present in the original dataset while addressing class imbalance. Table 2 displays the sample counts for each label before and after the resampling process. Training the model with the balanced dataset substantially narrowed the gap between training and validation loss, effectively resolving the overfitting problem. Figure 4 illustrates the training and validation loss for driving style model, highlighting the presence of overfitting in the system in the imbalanced data as seen in Figure 4a, where the model failed to generalize accurate prediction on the validation data. Figure 4b confirm that overfitting issue is handled by balancing the training dataset.

Figure 5 illustrates the training and validation loss for the driving style for ANN with Bagging model post-dataset resampling. To enhance system accuracy and bolster resilience against overfitting, dropout layers with a drop rate of 0.2 were integrated into the 4th, 5th, and 6th layers of both the ANN model and the ANN with bagging. The effects of dropout layers on detection outcomes are detailed in the results section.

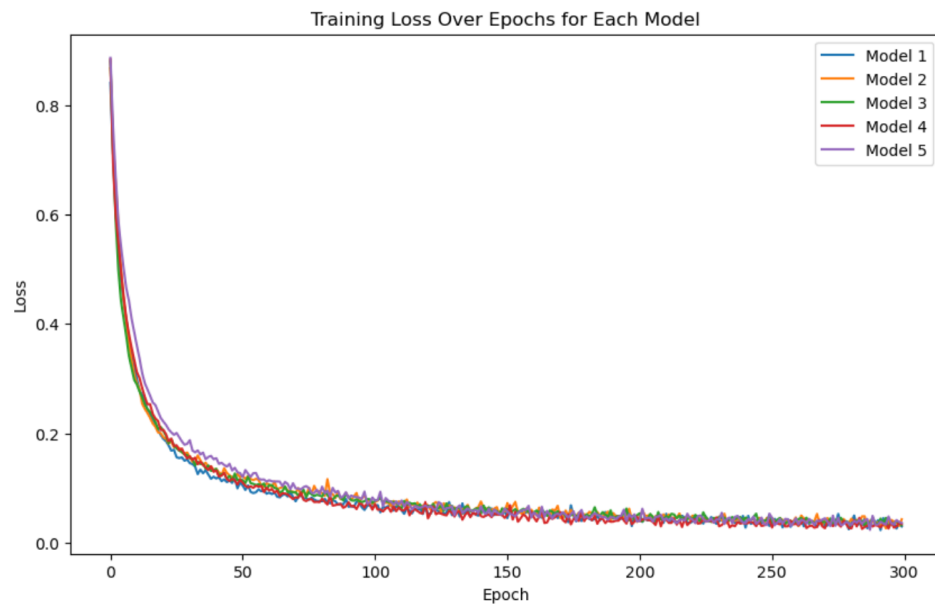
The system underwent training on a computer equipped with a 12th Gen Intel(R) Core i7 processor clocked at 1.70 GHz, and 8GB of RAM. The training process involved 300 epochs. The batch size and learning rate were adjusted by random variation. The most optimal results were obtained with a learning rate of 0.01 and a batch size of 30. Dropout layers were introduced in the early, middle, and final layers. The most favorable outcomes were observed when implementing dropout with a rate of 0.2 in the last three layers. The model was trained using the Adam optimization algorithm, and the cross-entropy was utilized as the loss function. The Adam optimizer adjusts the learning rates based on estimates of the moments of the gradients. This adaptive learning rate helps Adam converge faster and more reliably in practice, especially in tasks with noisy or sparse gradients [38].

**Table 2.** Training dataset label distribution before and after resampling.

	Aggressive Driving	Normal Driving	Low Traffic	Medium Traffic	High Traffic
Original label counts	2267	17,686	14,972	2591	2402
After resampling	17,686	17,686	14,972	14,972	14,972



**Figure 4.** Training loss with epoch. (a) The model training loss with epoch results using the imbalanced dataset. (b) The model training loss with epoch results with the resampled balanced dataset.



**Figure 5.** This figure shows the model training loss with epoch results for the deep neural network with bagging.

## 5. Results

In this section, we discuss the outcomes of our implemented ANN models, presenting their performance through confusion matrices, precision, recall, accuracy, and F1 score. The confusion matrix provides a detailed view of the system's performance, delineating the counts of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) for each class. Additionally, the other metrics utilized in our analysis are defined by the following Equations [39]:

$$Precision = \frac{1}{N} \sum_{i=1}^N \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{1}{N} \sum_{i=1}^N \frac{TP}{TP + FN} \quad (2)$$

$$Accuracy = \frac{1}{N} \sum_{i=1}^N \frac{TP}{TP + TN + FP + FN} \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

where  $N$  is the number of classes. As the percentage of accurately anticipated positive instances among all expected positive instances, precision quantifies the accuracy of positive predictions. Recall measures the percentage of accurately anticipated positive events among all actual positive instances [40]. The F1 score is especially helpful when there is an imbalance between the classes because it integrates recall and precision into a single statistic, offering a balance between the two.

The outcomes of the ANN models are illustrated without incorporating the dropout layers, then with the dropout, and finally with the bagging technique. Furthermore, SVM model with Gaussian Kernel is trained and tested alongside the ANN models using the same datasets, serving as a benchmark to evaluate the effectiveness of ANN approaches against traditional machine learning methods.

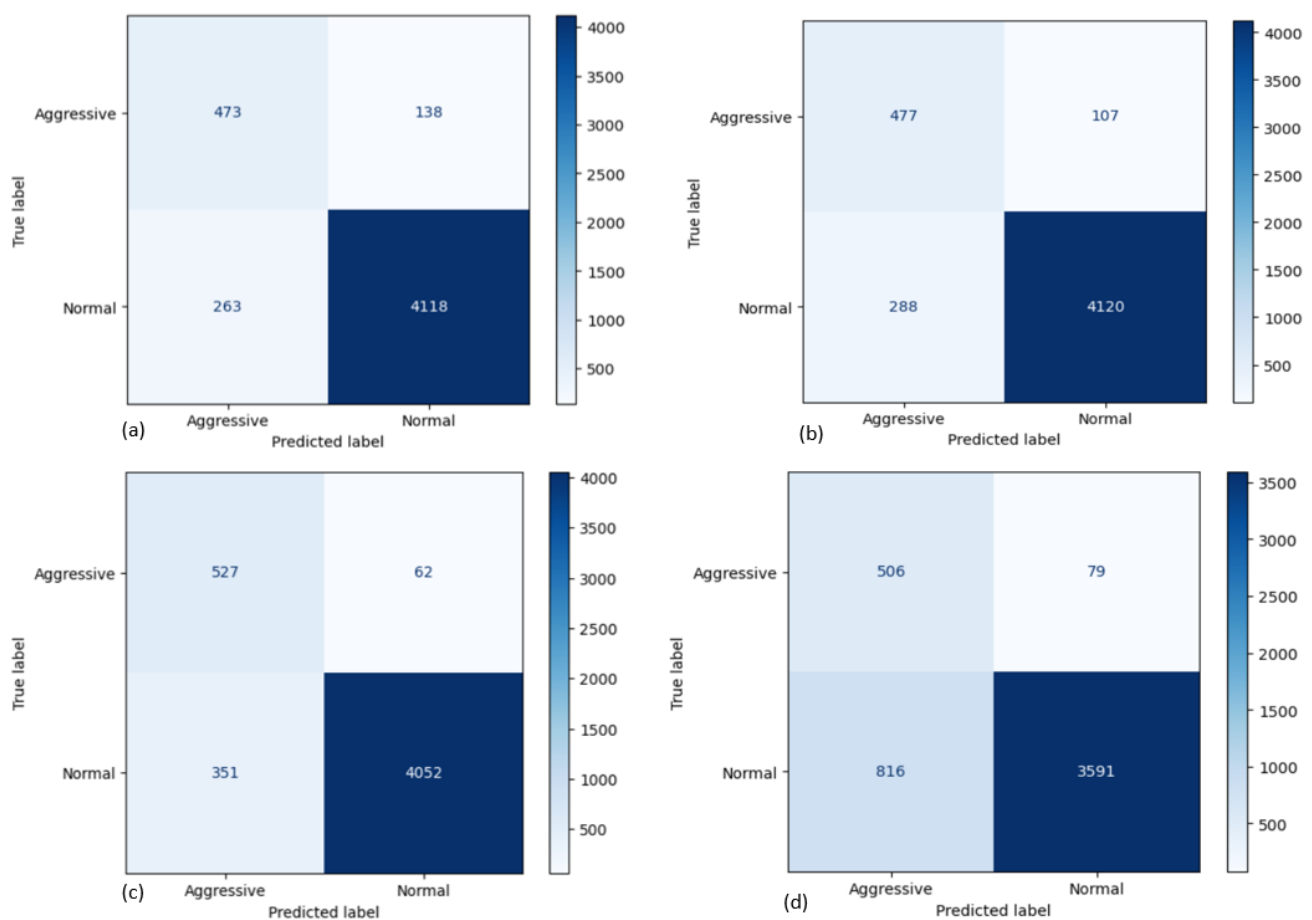
The results reveal that in the predominant categories of normal driving style and low traffic conditions, all three ANN models outperformed SVM. However, for the less common class labels encompassing aggressive driving, medium, and high traffic, SVM demonstrated



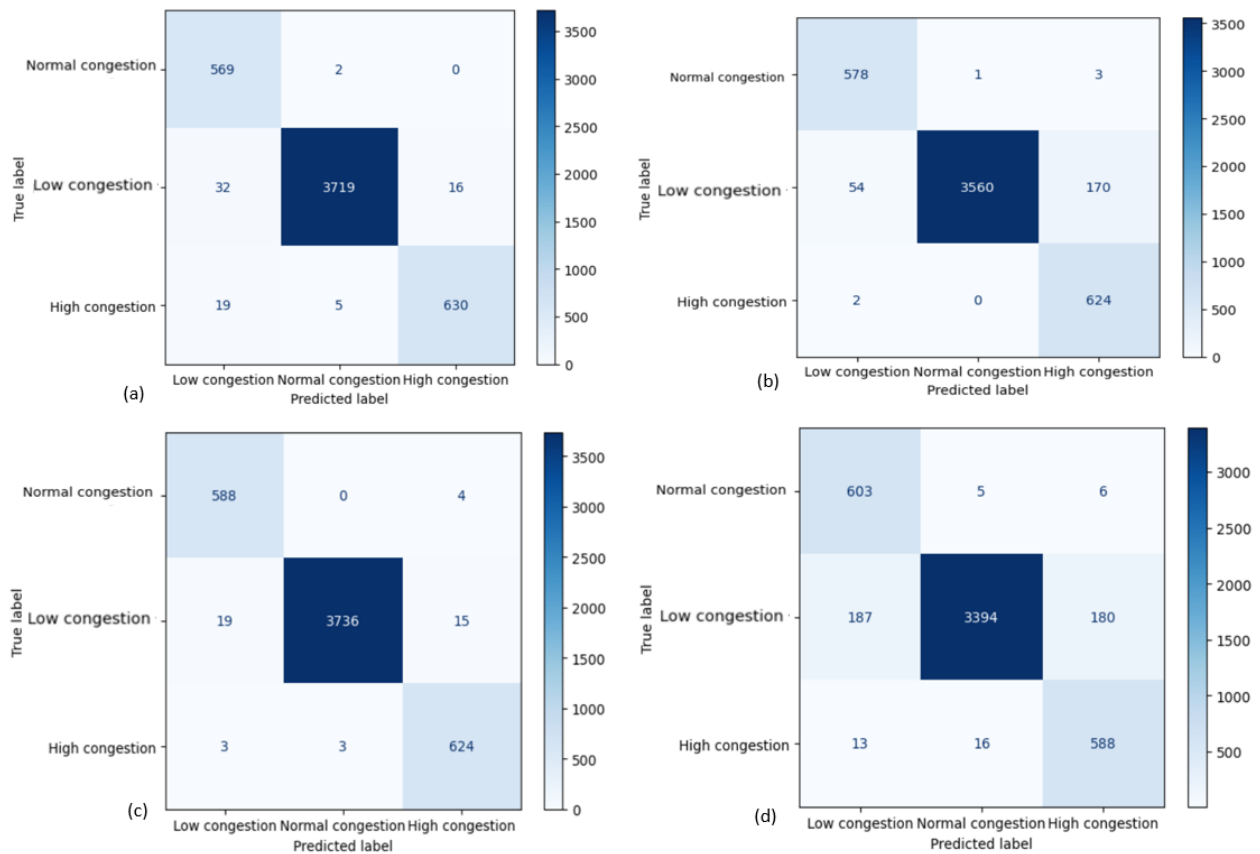
superior performance compared to ANN model without dropout. Remarkably, integrating dropout layers boosted the accuracy of detecting low-labeled classes, although SVM retained a slight edge. Notably, the ANN model employing bagging exhibited the most resilient performance in predicting low-labeled classes. Figures 6 and 7 show the confusion matrix for all the ANN models and SVM model.

The ANN models show higher precision values when compared to SVM. Among them, the ANN model with bagging stands out for its superior recall in predicting driving style and road conditions. This improvement in recall values implies a reduction in false negatives, especially within the low-labeled classes. Interestingly, the ANN model without dropout exhibits weaker recall scores in predicting driving style, underscoring the benefits of dropout and bagging in enhancing detection within imbalanced datasets.

The F1 score, which combines precision and recall, offers a balanced evaluation of model performance, particularly in the presence of imbalanced classes. The results reveal that the ANN model with bagging achieves the highest F1 scores in both driving style and road condition predictions, with values of 0.85 and 0.98, respectively. In contrast, SVM exhibits significantly lower precision values compared to the ANN models, although it demonstrates competitive recall values compared to the ANN model without dropout. SVM also records the lowest accuracy and F1 score among the implemented algorithms, underscoring the superiority of the ANN model over SVM as a conventional machine learning technique. Tables 3 and 4 provide a detailed overview of the precision, recall, accuracy, and F1 score for the implemented models in predicting driving style and road traffic conditions.



**Figure 6.** Driving style prediction results. (a) The confusion matrix using the ANN model without dropout. (b) The confusion matrix using ANN model with dropout. (c) The confusion matrix using the ANN with bagging. (d) The confusion matrix using SVM.



**Figure 7.** Traffic road conditions prediction results. (a) The confusion matrix using the ANN model without dropout. (b) The confusion matrix using the ANN model with dropout. (c) The confusion matrix using ANN with bagging. (d) The confusion matrix using SVM.

**Table 3.** Precision, recall, accuracy, and the F1 score values for driving style prediction.

Model	Precision	Recall	Accuracy	F1 Score
The proposed ANN	0.86	0.81	0.92	0.83
The proposed ANN with dropout	0.8	0.88	0.92	0.84
The proposed ANN with bagging	0.79	0.91	0.92	0.85
SVM (Baseline)	0.68	0.84	0.82	0.71

**Table 4.** Precision, recall, accuracy, and the F1 score values for road traffic condition prediction.

Model	Precision	Recall	Accuracy	F1 Score
The proposed ANN	0.96	0.98	0.99	0.97
The proposed ANN with dropout	0.9	0.98	0.95	0.97
The proposed ANN with bagging	0.98	0.99	0.99	0.98
SVM (Baseline)	0.83	0.95	0.92	0.88

### 6. Conclusions and Future Work

In this research, we utilized OBD and smartphone sensor data along with deep learning models to classify driving styles as aggressive or normal, and road traffic conditions as low, medium, or high. By utilizing OBD and smartphone sensor data instead of computer vision methods, we were able to implement prediction models in a more cost-effective manner. We experimented with various configurations of ANN models for class predictions, using SVM as a benchmark for comparison. The ANN architecture consisted of seven layers, and we trained three different configurations: the first without dropout, the second with dropout in the last three layers, and the third employing the bagging technique. In the

latter approach, the dataset was split to train five ANN models, and the final classifications were determined by averaging the predictions from all models.

The dataset used for training the algorithms exhibited an imbalance, reflecting the inherent nature of driving styles and road traffic during data collection. This led to overfitting issues when training the ANN model, as it struggled to generalize predictions on validation data. To address this, the training dataset was balanced using a resampling approach. Despite this imbalance, the ANN models displayed superior accuracy and precision compared to the SVM model. However, the SVM model demonstrated higher recall values than the ANN model without dropout. The incorporation of dropout and bagging techniques notably improved the system’s recall, particularly for the imbalanced low-labeled classes. Among the ANN models, the bagging approach consistently outperformed the others, as well as the SVM model, across all metrics, especially in terms of recall and F-score. This study underscores the superiority of ANN models over traditional machine learning techniques such as SVM. Figures 8 and 9 show the histograms for detection results for driving style and traffic conditions classifications.

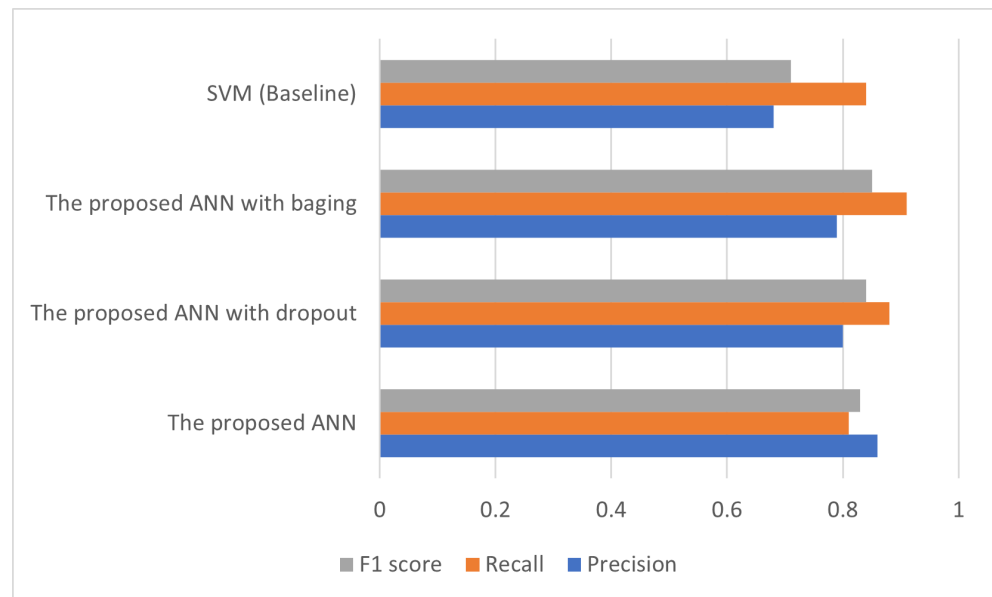


Figure 8. Precision, recall, and F1 score results of driving style classification.

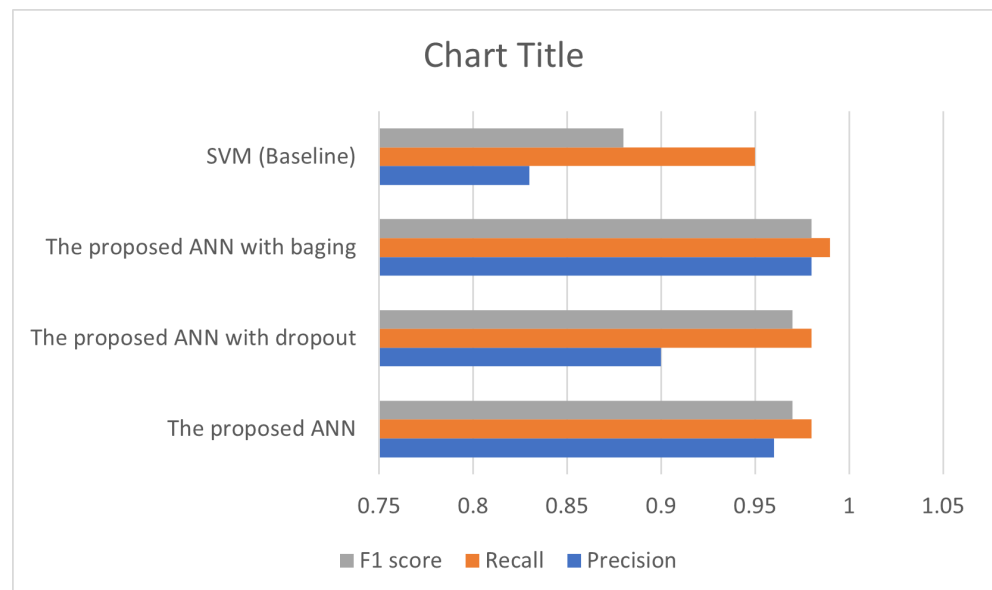


Figure 9. Precision, recall, and F1 score results of traffic conditions classification.

The utilization of OBD data and smartphone sensor data holds potential for addressing a wider array of road prediction challenges, including emissions, fuel consumption, and road infrastructure predictions such as identifying potholes. Incorporating data from vision sensors like cameras and radars could further enhance detection accuracy for these models. Moreover, more sophisticated deep learning methodologies leveraging time sequence data, such as recurrent neural networks (RNN) and Long Short-Term Memory (LSTM), could be employed with OBD and smartphone sensor data to tackle more complex prediction tasks.

**Author Contributions:** G.A.-r. developed the detection system architecture, defined the system elements, and designed the ANN models. He also implemented the training and testing models and contributed to the related work and conclusion. A.A. contributed to data analysis, sections structure, methodology, and results analysis. M.A.-r. contributed to the related work section, results, discussion, and conclusion. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset is available on Kaggle: <https://www.kaggle.com/datasets/gloseto/traffic-driving-style-road-surface-condition>, the data was accessed on 4 April 2024.

**Acknowledgments:** We would like to extend our sincere appreciation to the authors for their invaluable contributions to this research work. Their expertise, dedication, and collaborative spirit have been instrumental in shaping the outcomes of this study.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial Neural Networks
DNN	Deep Neural Networks
RNN	Recurrent Neural Networks
CNN	Convolutional Neural Networks
LSTM	Long Short-Term Memory
OBD	On-Board Diagnostics
SVM	Support Vector Machine
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
CAN	Controller Area Network
YOLO	You Only Look Once
ECU	Engine Control Unit
RPM	Revolution Per Minutes
LiDAR	Light Detection Furthermore, Ranging
MAP	Manifold Air Pressure
MAF	Mass Air Flow
IAT	Intake Air Temperature

## References

1. Baltusis, P. *On Board Vehicle Diagnostics. No. 2004-21-0009*; SAE Technical Paper; SAE International: Warrendale, PA, USA, 2004.
2. Aguilar, C.; Jesús, J.; Carrillo, J.A.C.; Fernandez, A.J.G.; Acosta, E.C. Robust road condition detection system using in-vehicle standard sensors. *Sensors* **2015**, *15*, 32056–32078. [[CrossRef](#)] [[PubMed](#)]
3. Malik, M.; Nandal, R. A framework on driving behavior and pattern using On-Board diagnostics (OBD-II) tool. *Mater. Today Proc.* **2023**, *80*, 3762–3768. [[CrossRef](#)]

4. Tullio, C.; Passeronge, C.; Lavagno, L.; Jurecska, A.; Damiano, A.; Sansoè, C.; Sangiovanni-Vincentelli, A.; Sangiovanni-Vincentelli, A. A case study in embedded system design: An engine control unit. In Proceedings of the 35th Annual Design Automation Conference, San Francisco, CA, USA, 15–19 June 1998; pp. 804–807.
5. Mukul, S.; Dubey, R.K. Deep learning model based CO<sub>2</sub> emissions prediction using vehicle telematics sensors data. *IEEE Trans. Intell. Veh.* **2021**, *8*, 768–777.
6. Rivera-Campoverde, N.D.; Muñoz-Sanz, J.L.; del Valle Arenas-Ramirez, B. Estimation of pollutant emissions in real driving conditions based on data from OBD and machine learning. *Sensors* **2021**, *21*, 6344. [[CrossRef](#)] [[PubMed](#)]
7. Xing, Y.; Lv, C.; Cao, D. Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles. *IEEE Trans. Veh. Technol.* **2019**, *69*, 1341–1352. [[CrossRef](#)]
8. Xiao, Z.; Li, P.; Havyarimana, V.; Hassana, G.M.; Wang, D.; Li, K. GOI: A novel design for vehicle positioning and trajectory prediction under urban environments. *IEEE Sens. J.* **2018**, *18*, 5586–5594. [[CrossRef](#)]
9. Yao, Y.; Zhao, X.; Liu, C.; Rong, J.; Zhang, Y.; Dong, Z.; Su, Y. Vehicle fuel consumption prediction method based on driving behavior data collected from smartphones. *J. Adv. Transp.* **2020**, *2020*, 1–11. [[CrossRef](#)]
10. Nurcahya, S.; Erfianto, B.; Setyorini, S. Forecasting fuel consumption based-on OBD II data. *Indones. J. Comput.* **2022**, *7*, 93–102.
11. Vasavi, S.; Aswarth, K.; Sai Durga Pavan, T.; Anu Gokhale, A. Predictive analytics as a service for vehicle health monitoring using edge computing and AK-NN algorithm. *Mater. Today Proc.* **2021**, *46*, 8645–8654. [[CrossRef](#)]
12. Shivakarthik, S.; Krishnanjan Bhattacharjee, M.; Mithran, S.; Mehta, S.; Kumar, A.; Rakla, L.; Aserkar, S.; Shah, S.; Komati, R. Maintenance of automobiles by predicting system fault severity using machine learning. In *Sustainable Communication Networks and Application, Proceedings of the ICSCN 2020, Erode, India, 6–7 August 2020*; Springer: Singapore, 2021; pp. 263–274.
13. De Ville, B. Decision trees. *Wiley Interdiscip. Rev. Comput. Stat.* **2013**, *5*, 448–455. [[CrossRef](#)]
14. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
15. Suthaharan, S. Support vector machine. In *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*; Springer: New York, NY, USA, 2016; pp. 207–235.
16. Zhang, Z.; Zheng, J.; Xu, H.; Wang, X. Vehicle detection and tracking in complex traffic circumstances with roadside LiDAR. *Transp. Res. Rec.* **2019**, *2673*, 62–71. [[CrossRef](#)]
17. Hasanujjaman, M.; Chowdhury, M.Z.; Jang, Y.M. Sensor fusion in autonomous vehicle with traffic surveillance camera system: detection, localization, and AI networking. *Sensors* **2023**, *23*, 3335. [[CrossRef](#)] [[PubMed](#)]
18. AbuAli, N. Advanced vehicular sensing of road artifacts and driver behavior. In Proceedings of the 2015 IEEE Symposium on Computers and Communication (ISCC), Larnaca, Cyprus, 6–9 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 45–49.
19. Zhang, C.; Patel, M.; Buthpitiya, S.; Lyons, K.; Harrison, B.; Abowd, G.D. Driver classification based on driving behaviors. In Proceedings of the 21st International Conference on Intelligent User Interfaces, Sonoma, CA, USA, 7–10 March 2016; pp. 80–84.
20. Nirmali, B.; Wickramasinghe, S.; Munasinghe, T.; Amalraj, C.R.J.; Dilum Bandara, H.M.N. Vehicular data acquisition and analytics system for real-time driver behavior monitoring and anomaly detection. In Proceedings of the 2017 IEEE International Conference on Industrial and Information Systems (ICIIS), Peradeniya, Sri Lanka, 15–16 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–6.
21. Al-refai, G.; Elmoaqet, H.; Ryalat, M. In-vehicle data for predicting road conditions and driving style using machine learning. *Appl. Sci.* **2022**, *12*, 8928. [[CrossRef](#)]
22. Kim, B.; Baek, Y. Sensor-based extraction approaches of in-vehicle information for driver behavior analysis. *Sensors* **2020**, *20*, 5197. [[CrossRef](#)] [[PubMed](#)]
23. Shaikh, M.K.; Palaniappan, S.; Ali, F.; Khurram, M. Identifying Driver Behaviour through Obd-Ii Using Android Application. *Palarch's J. Archaeol. Egypt/Egyptol.* **2020**, *17*, 13636–13647.
24. Lattanzi, E.; Freschi, V. Machine learning techniques to identify unsafe driving behavior by means of in-vehicle sensor data. *Expert Syst. Appl.* **2021**, *176*, 114818. [[CrossRef](#)]
25. Hermawan, G.; Husni, E. Acquisition, modeling, and evaluating method of driving behavior based on OBD-II: A literature survey. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Bristol, UK, 2020; Volume 879, p. 012030.
26. Liu, G.; Shi, H.; Kiani, A.; Khreishah, A.; Lee, J.; Ansari, N.; Liu, C.; Yousef, M.M. Smart traffic monitoring system using computer vision and edge computing. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 12027–12038. [[CrossRef](#)]
27. Sajib, M.S.R.; Amir-Ul-Haque Bhuiyan, T.M. Computer vision based traffic monitoring and analyzing from on-road videos. *Glob. J. Comput. Sci. Technol.* **2019**, *19*, 19–24.
28. Guo, F.; Wang, Y.; Qian, Y. Computer vision-based approach for smart traffic condition assessment at the railroad grade crossing. *Adv. Eng. Informatics* **2022**, *51*, 101456. [[CrossRef](#)]
29. Reddy, P.H.; Manjunath, M.; Rohith, M.; Reddy, N.M.; Satyanarayana, A. Deep CNN Model for Condition Monitoring of Road Traffic: An Application Of Computer Vision. *Turk. J. Comput. Math. Educ. (TURCOMAT)* **2023**, *14*, 1362–1370. [[CrossRef](#)]
30. Ranyal, E.; Sadhu, A.; Jain, K. Road condition monitoring using smart sensing and artificial intelligence: A review. *Sensors* **2022**, *22*, 3044. [[CrossRef](#)]
31. Vij, D.; Aggarwal, N. Smartphone based traffic state detection using acoustic analysis and crowdsourcing. *Appl. Acoust.* **2018**, *138*, 80–91. [[CrossRef](#)]
32. Allouch, A.; Koubâa, A.; Abbes, T.; Ammar, A. Roadsense: Smartphone application to estimate road conditions using accelerometer and gyroscope. *IEEE Sens. J.* **2017**, *17*, 4231–4238. [[CrossRef](#)]

33. Bhoraskar, R.; Vankadhara, N.; Raman, B.; Kulkarni, P. Wolverine: Traffic and road condition estimation using smartphone sensors. In Proceedings of the 2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012), Bangalore, India, 3–7 January 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 1–6.
34. Chugh, G.; Bansal, D.; Sofat, S. Road condition detection using smartphone sensors: A survey. *Int. J. Electron. Electr. Eng.* **2014**, *7*, 595–602.
35. Github. Available online: <https://github.com/sisinflab-swot/mafalda> (accessed on 1 April 2024).
36. Sharma, S.; Sharma, S.; Athaiya, A. Activation functions in neural networks. *Towards Data Sci.* **2017**, *6*, 310–316. [[CrossRef](#)]
37. Joseph, V.R.; Vakayil, A. SPlit: An optimal method for data splitting. *Technometrics* **2022**, *64*, 166–176. [[CrossRef](#)]
38. Bock, S.; Weiß, M. A proof of local convergence for the Adam optimizer. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–8.
39. Powers, D.M.W. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv* **2020**, arXiv:2010.16061.
40. Su, L.T. The relevance of recall and precision in user evaluation. *J. Am. Soc. Inf. Sci.* **1994**, *45*, 207–217. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.