electronics

MDPI

Article

A Study on Machine Learning-Enhanced Roadside Unit-Based Detection of Abnormal Driving in Autonomous Vehicles

Keon Yun [1,†], Heesun Yun [1,†], Sangmin Lee [1], Jinhyeok Oh [1], Minchul Kim [1,*], Myongcheol Lim [1], Juntaek Lee [2], Chanmin Kim [2], Jiwon Seo [2] and Jinyoung Choi [3]

[1] Pentasecurity, Incorporated, 9F, 115, Yeouigongwon-ro, Yeongdeungpo-gu, Seoul 07241, Republic of Korea; kyun@pentasecurity.com (K.Y.); hsyun@pentasecurity.com (H.Y.); smlee@pentasecurity.com (S.L.); jhoh@pentasecurity.com (J.O.); mclim@pentasecurity.com (M.L.)
[2] Korea Automotive Technology Institute, 4F, 94, Cheongna emerald-ro, Seo-gu, Incheon 22739, Republic of Korea; jtlee@katech.re.kr (J.L.); cmkim@katech.re.kr (C.K.); jwseo@katech.re.kr (J.S.)
[3] Xbrain, Incorporated, 5F, 168, Yeoksam-ro, Gangnam-gu, Seoul 06248, Republic of Korea; freddie@xbrain.team
* Correspondence: minchul@pentasecurity.com
† These authors contributed equally to this work.

Abstract: Ensuring the safety of autonomous vehicles is becoming increasingly important with ongoing technological advancements. In this paper, we suggest a machine learning-based approach for detecting and responding to various abnormal behaviors within the V2X system, a system that mirrors real-world road conditions. Our system, including the RSU, is designed to identify vehicles exhibiting abnormal driving. Abnormal driving can arise from various causes, such as communication delays, sensor errors, navigation system malfunctions, environmental challenges, and cybersecurity threats. We simulated exploring three primary scenarios of abnormal driving: sensor errors, overlapping vehicles, and counterflow driving. The applicability of machine learning algorithms for detecting these anomalies was evaluated. The Minisom algorithm, in particular, demonstrated high accuracy, recall, and precision in identifying sensor errors, vehicle overlaps, and counterflow situations. Notably, changes in the vehicle's direction and its characteristics proved to be significant indicators in the Basic Safety Messages (BSM). We propose adding a new element called *linePosition* to BSM Part 2, enhancing our ability to promptly detect and address vehicle abnormalities. This addition underpins the technical capabilities of RSU systems equipped with edge computing, enabling real-time analysis of vehicle data and appropriate responsive measures. In this paper, we emphasize the effectiveness of machine learning in identifying and responding to the abnormal behavior of autonomous vehicles, offering new ways to enhance vehicle safety and facilitate smoother road traffic flow.

Keywords: autonomous vehicle; roadside unit; abnormal driving; machine learning

Citation: Yun, K.; Yun, H.; Lee, S.; Oh, J.; Kim, M.; Lim, M.; Lee, J.; Kim, C.; Seo, J.; Choi, J. A Study on Machine Learning-Enhanced Roadside Unit-Based Detection of Abnormal Driving in Autonomous Vehicles. *Electronics* **2024**, *13*, 288. https://doi.org/10.3390/electronics13020288

Academic Editor: Sergio Busquets-Monge

Received: 13 December 2023
Revised: 3 January 2024
Accepted: 5 January 2024
Published: 8 January 2024

1. Introduction

Autonomous vehicles represent an innovative technology that operates without direct human intervention, autonomously navigating and ensuring safe passage to destinations. This innovation significantly enhances the safety and efficiency of transportation systems, offering substantial societal benefits, including the reduction of traffic accidents. As concerns about autonomous vehicles increase, advancing this technology becomes paramount. In response to these developments, our research is focused on advancing autonomous driving technologies. This includes exploring the transition from WAVE/LTE to 5G for faster communication, enhancing the processing capabilities of edge computing, and delving into specific aspects of security like cybersecurity and data protection.

In this paper, we simulate complex and diverse scenarios that autonomous vehicles may encounter in real road environments in real time and analyze these scenarios through a machine learning algorithm designed to detect vehicle abnormalities. Figure 1 illustrates the use of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. These methods, along with other vehicle-related communications, are collectively referred to as vehicle-to-everything (V2X). The On-Board Diagnostics (OBD) and On-Board Units (OBU) function as communication tools within the vehicle. The Basic Safety Messages (BSM), generated by the vehicle, are transmitted through these tools. BSMs carry critical data about the vehicle's dynamic state and other essential safety details, playing a pivotal role in facilitating effective communication between vehicles and between vehicles and infrastructure. This cooperation, enabled by the BSMs, allows for rapid response during driving. The V2I system exchanges data with vehicles through devices such as WAVE/LTE modems and edge computers installed in the infrastructure.

However, as autonomous vehicles gain prominence, it is equally important to acknowledge and address the underlying issues that emerge in their wake. In particular, failures in autonomous vehicle systems can lead to significant safety risks, potentially resulting in serious accidents and endangering human lives. A variety of issues threaten safe driving, including communication problems including communication delays, sensor errors, OBD and OBU failures, navigator errors, software and hardware malfunctions, power system issues, engine and mechanical failures, environmental factors, and cybersecurity concerns. These problems can directly affect not only the safety of the vehicle but also the lives of drivers and others. We discuss three abnormal behaviors due to these factors: (1) sensor problems, (2) overlap, and (3) reverse driving.

Accordingly, we are researching an abnormal behavior detection system based on machine learning. Research using machine learning identifies problems in the abnormal vehicle and provides immediate help from the outside through RSU. This research is important because it can ensure the safety of passengers and improve the safety and reliability of autonomous vehicles by monitoring abnormal driving vehicles and taking necessary measures.

This paper is organized as follows: Section 2 introduces related works and Section 3 describes the background of machine learning. In Section 4, we simulate three abnormal behaviors such as sensor problems, overlap, and reverse driving, and in Section 5, we experiment with a method of detecting abnormal behavior using machine learning based on BSM, which is the simulation result in Section 4. Finally, Section 6 is the conclusion of this paper.

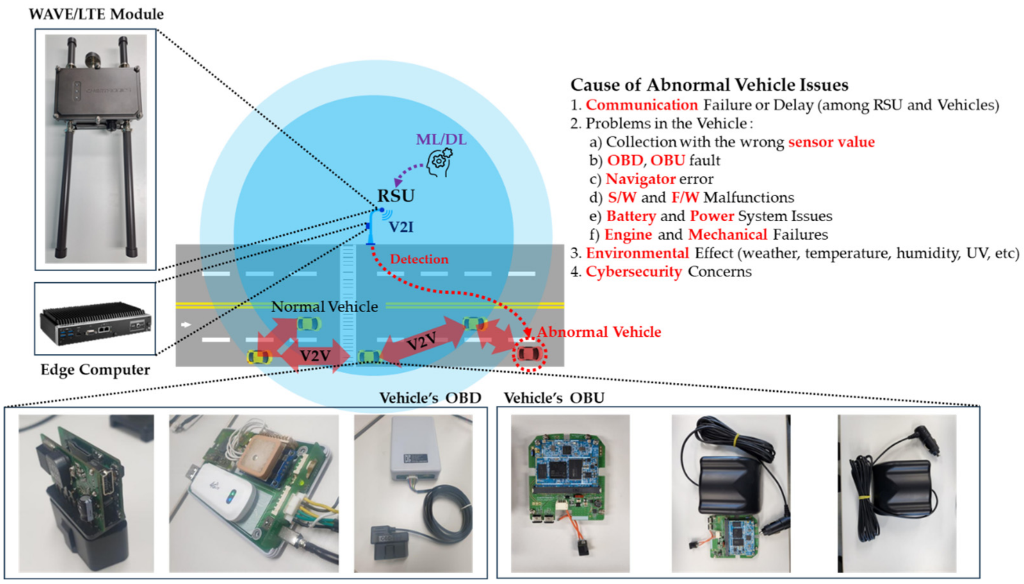


Figure 1. Abnormal vehicle concerns within V2X communication systems.

2. Related Works

Advances in sensors and driving algorithms for autonomous vehicles emphasize the importance of anomaly detection to ensure passenger safety and data reliability. Research on anomaly detection is broadly divided into two main approaches. The first is a data-centric approach based on the vehicle's sensor data, and the second is a method that utilizes images captured by external cameras.

The data-centric approach utilizes data generated inside the vehicle to detect abnormalities for self-recovery or transmits this data to the outside to determine whether an abnormality exists in an external system [1–4]. Image-based approaches identify vehicles through image-processing technology and detect abnormal behavior based on this [5–9]. In this paper, image-based approaches were excluded because it is difficult to process image data on edge computers. This decision is due to the processing power and resource limitations of edge computers. Our goal is to detect abnormal behavior through the communication of data generated within the vehicle with the RSU.

Generally, data-based anomaly detection uses a rule-based method. However, it is difficult for these systems to respond quickly and flexibly to the complex and dynamic environment of autonomous vehicles. Because it is close-to-impossible to define in advance all abnormal situations that may occur in the real environment, real-time response is difficult. In contrast, machine learning-based approaches have the potential to make decisions and respond in real time, similar to humans. Vu et al. [10] proposed a method to detect abnormal conditions using machine learning in an IoT environment. This method utilizes unsupervised learning in an IoT environment where various data are generally unlabeled and mixed. Ryan et al. [11] proposed a methodology to evaluate the risks of autonomous vehicles. In this research, normal driving patterns were modeled using CNN, and the operational risk of autonomous vehicles was quantified by applying a GP-based anomaly detection method. Alladi et al. [12] proposed a DNN-based anomaly detection framework to detect unknown abnormalities in VANETs.

In addition to these approaches, recent research has expanded the understanding of vehicle communication and machine learning in traffic systems. Bifulco et al. [13] investigate decentralized cooperative crossing at unsignalized intersections, demonstrating the effectiveness of vehicle-to-vehicle communication in mixed traffic flows. This research provides valuable insights into the role of communication technologies in improving traffic management and safety. Lu et al. [14] offer a comprehensive review of real-time performance-focused localization techniques for autonomous vehicles. Their work highlights the importance of precision in localization methods and their impact on the safety and efficiency of autonomous driving. Furthermore, Chakraborty et al. [15] explore the development of novel machine learning frameworks for secure communication in Vehicle Ad hoc Networks (VANETs), emphasizing the role of machine learning in enhancing the security and efficiency of smart transportation systems.

3. Background about Machine Learning

Machine learning is the research of computer algorithms that automatically improve through experience. Based on input data, human intervention is minimized. Data patterns are derived and prediction models are created. As big data technology develops, it becomes easier to collect large amounts of diverse learning data, accelerating the development of machine learning technology. These advances in machine learning technology enable the recognition and prediction of patterns in large amounts of data and complex environments that are difficult for humans to process.

Machine learning consists of a learning stage and a prediction stage. In the learning stage, a model is created by training the algorithm with data, and in the prediction stage, the data is applied to the model created to predict the result. Figure 2 shows the machine learning process.

Machine learning is classified into Supervised Learning and Unsupervised Learning depending on the learning method. These are described in detail in the subsection.

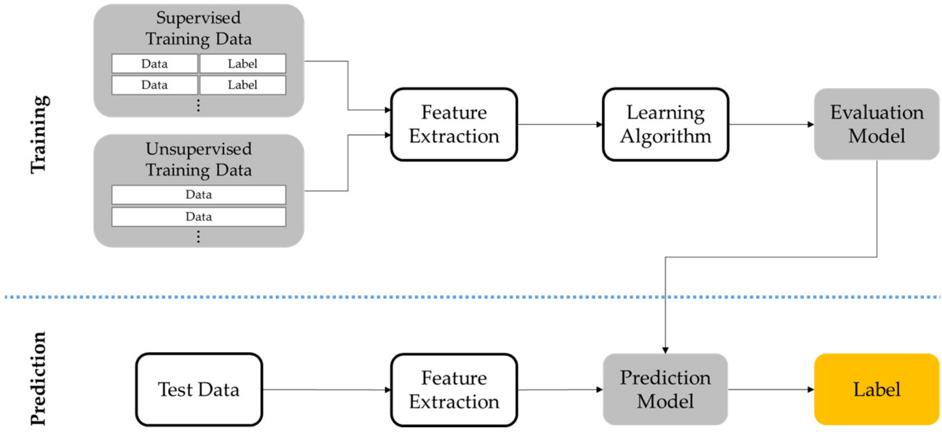


Figure 2. Training and prediction in machine learning.

3.1. *Supervised Learning*

Supervised learning is a method of learning of a model using labeled training data. Because this method has high prediction accuracy and applies to a variety of real-world problems, it is widely used in both research and practical applications. Supervised learning algorithms can be divided into classification and regression, and each algorithm has its strengths and weaknesses.

There are various algorithms in supervised learning, as shown in Table 1. Machine learning algorithms each have their own advantages and disadvantages, and understanding these is essential for choosing an appropriate algorithm. Classification algorithms, such as logistic regression [16–18], support vector machines (SVM) [18–23], decision trees [18,19,24], and Naive Bayes [17–19,23], are effective for categorical results and are used in a variety of applications. However, they have a risk of overfitting, their performance may vary on imbalanced data, and require fine-tuning. Regression algorithms, such as linear regression [25,26], Ridge [27], Lasso [27,28], and ElasticNet [27,29,30], are effective in predicting continuous outcomes and are easy to understand and simple to implement. However, they are sensitive to outliers, risk overfitting, and are limited to linear relationships. Ensemble methodologies, such as Random Forest [17–19,31], Gradient Boosting [31,32], and Adaboost [33,34], combine the strengths of individual models to improve accuracy and reduce volatility. However, they are more complicated to tune, more computationally expensive, and have a risk of overfitting in noisy data. Recently developed deep neural networks [19,35], convolutional neural networks [17,36–38], and recurrent neural networks [19,39,40] show high performance in complex tasks, can model nonlinear relationships, and effectively process large-scale data. However, they require significant computational resources, are susceptible to overfitting, and require large amounts of training data.

Table 1. Advantages and disadvantages of supervised learning algorithms.

Learning Method	Learning Algorithm	Advantages	Disadvantages
Classification	Logistic Regression, SVM, Decision Trees, Naive Bayes	1. Effective for categorical outputs 2. Wide range of applications 3. Well-established algorithms	1. Overfitting risk 2. Requires fine-tuning 3. Varying performance on imbalanced data
Regression	Linear Regression, Ridge, Lasso, ElasticNet	1. Effective for continuous outputs 2. Simple to understand and implement 3. Basis for more advanced techniques	1. Sensitive to outliers 2. Overfitting risk 3. Limited to linear relationships
Ensemble Methods	Random Forest, Gradient Boosting, AdaBoost	1. Improved accuracy 2. Reduced variance 3. Combines strengths of individual models	1. More complex to tune 2. Increased computational cost 3. Risk of overfitting with noisy data
Recent Advancements	Deep Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks	1. High performance on complex tasks 2. Ability to model non-linear relationships 3. Effective in handling large-scale data	1. Requires significant computational resources 2. Prone to overfitting 3. Requires large amounts of training data

3.2. Unsupervised Learning

Unsupervised Learning is a machine learning method that performs learning based on unlabeled data. Unlike Supervised Learning, it has no or limited labels, so it is mainly used to group similar data or detect outliers by learning the structure of the data. Unsupervised learning methods can be seen using various methods for grouping, as shown in Figure 3.

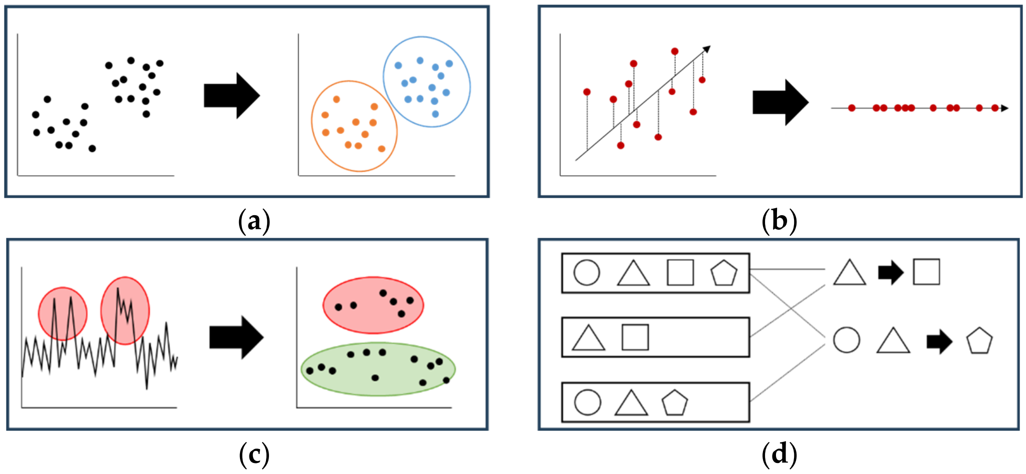


Figure 3. Grouping methods of unsupervised learning. (**a**) Clustering; (**b**) Dimensionality Reduction; (**c**) Abnormal Detection; (**d**) Association Rule Learning.

Table 2 shows the advantages or disadvantages of unsupervised learning algorithms. Clustering algorithms consist of k-means [19,31–45], DBSCAN [21], hierarchical clustering [46–48], and spectral clustering [49–51], and provide a simple structure, widely used methodology, and diversity according to various needs. However, they have disadvantages, such as spherical cluster assumption, sensitivity to scale, and difficulty processing complex data structures. Anomaly detection algorithms include One-class SVM [52,53], Isolation Forest [42,54], and Local Outlier Factor [55,56], which demonstrate effective operation in high-dimensional spaces and suitability for various anomaly types. However, they have the disadvantage of missing complex outliers and being sensitive to parameters. Dimensionality reduction algorithms include PCA [22,23], t-SNE [43], Autoencoders [19], UMAP [44,57], and MiniSom [45,58,59], which provide advantages such as reduced data complexity, ease of visualization, and capturing nonlinear relationships. However, they have disadvantages such as high computational intensity and the need to select appropriate dimensions. Association rule learning includes Apriori [60,61], Eclat [61], and FP-Growth [62,63], and provide relationship discovery, efficiency in large-scale datasets, and suitability for itemset size. They have disadvantages such as the amount of rules, sensitivity to noise and outliers, and the need for parameter tuning. In recent development trends, deep clustering [64,65], anomaly detection using GANs [66], and deep reinforcement learning [19,67,68] are attracting attention. These algorithms provide the advantages of utilizing deep learning, adaptability to complex patterns, and suitability for large-scale and dynamic data, but have disadvantages such as large data requirements, computational intensity, and complex implementation.

In this way, identifying the advantages and disadvantages of each unsupervised learning algorithm is an important criterion for selecting and applying an effective algorithm.

Table 2. Advantages and disadvantages of unsupervised learning algorithms.

Learning Method	Learning Algorithm	Advantages	Disadvantages
Clustering	k-means, DBSCAN, Hierarchical Clustering, Spectral Clustering	1. Simplicity 2. Widespread use 3. Variety for needs	1. Spherical cluster assumption 2. Scale sensitivity 3. Complexity struggle
Abnormal Detection	One-class SVM, Isolation Forest, Local Outlier Factor	1. High-dimensional space effectiveness 2. Suitability for anomaly types	1. Complex anomaly missing 2. Parameter sensitivity
Dimensionality Reduction	PCA, t-SNE, Autoencoders, UMAP, MiniSom	1. Data complexity reduction 2. Visualization facilitation 3. Non-linear relationship capture	1. Computational intensity 2. Dimension selection necessity
Association Rule Learning	Apriori, Eclat, FP-Growth	1. Relation discovery 2. Efficiency in large datasets 3. Suitability for itemset sizes	1. Rule quantity 2. Noise and outlier sensitivity 3. Parameter tuning requirement
Recent Advancements	Deep Clustering, GANs for Anomaly Detection, Deep Reinforcement Learning	1. Deep learning leverage 2. Adaptability to complex patterns 3. Suitability for large-scale and dynamic data	1. Large data requirement 2. Computational intensity 3. Complex implementation

4. Abnormal Vehicle's BSM with Simulation Tool, CANoePro

The experimental environment configuration is shown in Figure 4 and is designed for the simulation of vehicle communication systems. The experimental setup consisted of a high-performance desktop computer, CANoePro S/W (CANoe 15.4.35) and H/W [69], and an edge computing device for data processing and analysis in the RSU enclosure. On a desktop computer, it is designed using CANoePro, which simulates V2V and V2I. Using this computer, vehicle communication protocols are created and tested in various road and traffic scenarios. In this experiment, the message format of the vehicle communication protocol is defined as shown in Table 3, and real-time data transmission and reception are experimented with. Edge computing devices play a central role in RSU and are used to process large amounts of data generated from vehicles. Additionally, the device provides network connectivity to support real-time processing and analysis of vehicle data, enabling complex decision-making processes.

In vehicle communication systems, BSM provides information about the dynamic state of the vehicle, which is essential to ensuring road safety. As shown in Table 3, each BSM contains various data elements used for effective communication in V2V and/or V2I, which are defined by the J2735 standard [70].

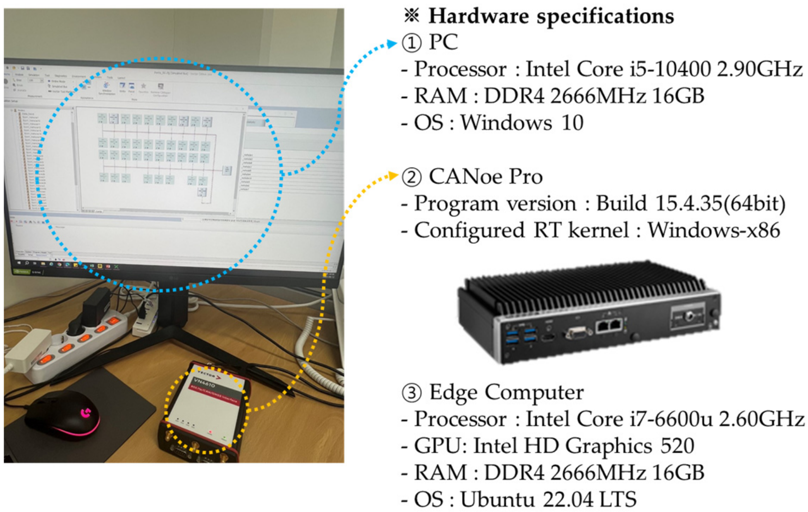


Figure 4. Experiment environments for simulation.

Table 3. J2735 Message frame [70].

Message Parameter	Message Type	Data Frame	Data Element
packet_time	BSM Part 1		
Mac_address			
Message_count			MsgCount
BSM_lat			Latitude
BSM_lon			Longitude
BSM_elev			Elevation
BSM_accuraacy_semiMajor		PositionalAccuracy	semiMajorAxisAccuracy
BSM_accuraacy_semiMinor			semiMinorAxisAccuracy
BSM_accuraacy_orientation			SemiMajorAxisOrientation
BSM_speed(m/s)			speed
BSM_heading			Heading
BSM_angle			SteeringWheelAngle
BSM_accelset_lat		AccelerationSet4Way	Acceleration
BSM_accelset_lon			Acceleration
BSM_accelset_vert			VerticalAcceleration
BSM_accelset_yaw			YawRate
BSM_breakes_WheelBrakes		BrakeSystemStatus	BrakeAppliedStatus
BSM_breakes_traction			TractionContralStatus
BSM_breakes_antiLockBreakes			AntiLockBrakeStatus
BSM_breakes_scs			StabilityControlStatus
BSM_breakes_breakBoost			BrakeBoostApplied
BSM_breakes_auxBrakes			AuxiliaryBrakeStatus
BSM_size_width		VehicleSize	VehicleWidth
BSM_size_length			VehicleLength
Event_message	BSM Part 2	VehicleSafetyExtensions	eventHazardLights eventStopLineViolation eventABSactivated eventTractionControlLoss eventStabilityControlactivated eventHazardousMaterials eventReserved1 eventHardBraking eventLightsChanged eventWipersChanged eventFlatTire eventDisabledVehicle eventAirBagDeployment ***linePosition***

The Packet_time property included in BSM Part 1 indicates the exact time when the message was transmitted. Mac_address is used to uniquely identify the vehicle. Message_count assigns sequential numbers to messages within the message stream to prevent data loss and/or duplication. Geographic location is expressed in 1/10 micro-degree units through latitude and longitude through BSM_lat and BSM_lon, and BSM_elev represents altitude based on the WGS-84 [71] ellipsoid. Position accuracy is further provided by properties such as BSM_accuracy_semiMajor and BSM_accuracy_semiMinor, which provide the margin of error for position along the vehicle's major and minor axes, respectively.

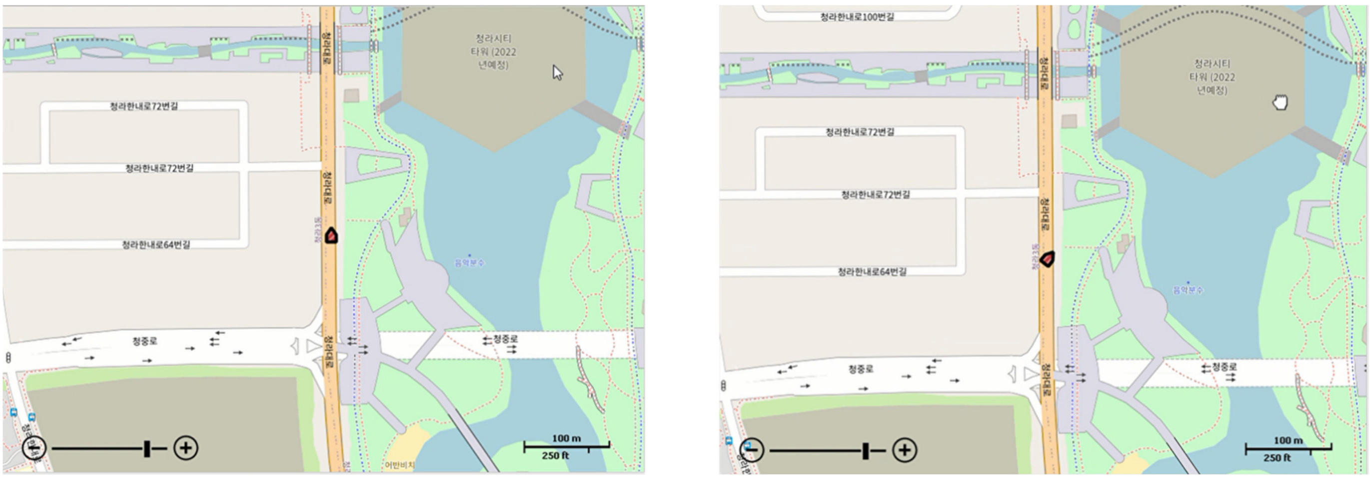
BSM_speed and BSM_heading represent the vehicle's speed in meters per second (m/s) and the increasing direction clockwise, with north as 0 degrees. SteeringWheelAngle, denoted as BSM_angle, provides information indicating the direction of the vehicle wheels.

BSM Part 2 expands on safety-related aspects, containing data on various vehicle events and anomalies in VehicleSafetyExtensions. For example, Event_message contains flags for events such as eventHardBraking, which indicates hard braking, or eventAirBagDeployment, which indicates airbag deployment, which is a safety-critical situation. Through these BSMs, vehicles communicate their immediate operational status, including mechanical, environmental, and behavioral aspects, to surrounding vehicles and infrastructure. We propose to add *linePosition* message to this BSM Part 2 rather than just known information.

As Figure 1 shows the factors that can cause abnormal behavior, we consider three major scenarios and simulate them to apply in Section 5.

In the case of autonomous vehicles, many sensors are installed to ensure the safety of passengers. The first scenario, which can occur frequently in these vehicles, is when there is an error in the value received from the sensor. The second case considered vehicles driving in the same lane overlapping due to communication problems. The third scenario is a counterflow vehicle, which is a scenario that occurs due to various factors. For example, when a counterflow vehicle occurs in a construction area. An emergency vehicle's reverse driving can also occur. The reason we chose these three scenarios is because they occur frequently around us.

The first scenario is shown in Figure 5. This is a case of frequently operating the steering wheel while driving the vehicle on a straight road. This is a case where the vehicle cannot drive straight, but it can occur due to a failure of the vehicle's steering sensor.



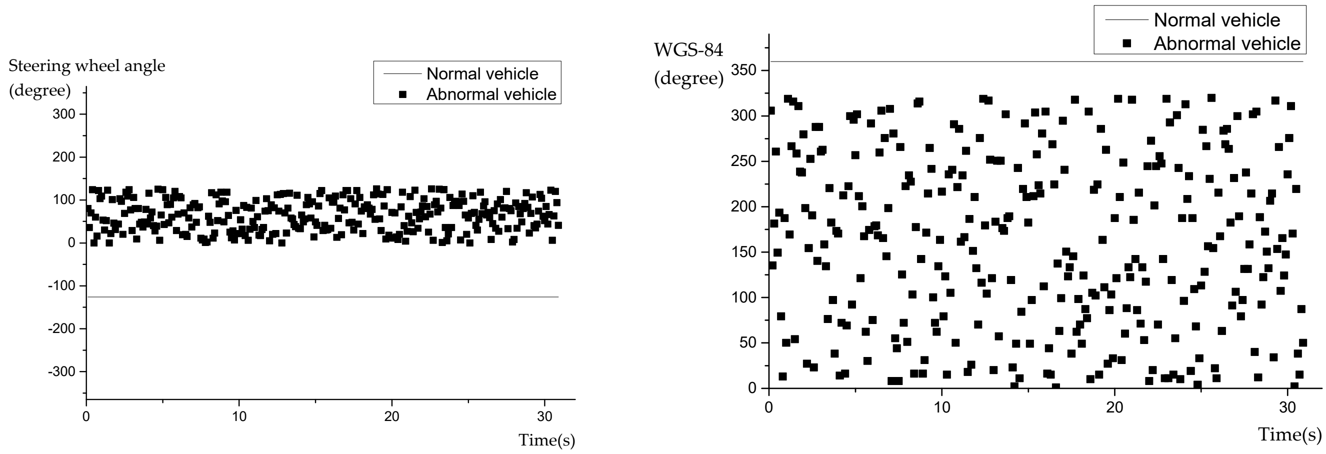
(**a**) (**b**)

Figure 5. First scenario: appearance of wrong sensor values. (**a**) Normal vehicle driving; (**b**) Abnormal vehicle driving with wrong data.

In first scenario, the steering wheel data is abnormal, so the messages about steering wheel operation in BSM are shown in Figure 6a,b. Figure 6a is the steering wheel angle, and Figure 6b is the information containing the WGS-84 coordinate system information over time. In Figure 6a,b, it can be seen at a glance that a problem has occurred in the sensor in the case of an abnormal vehicle.

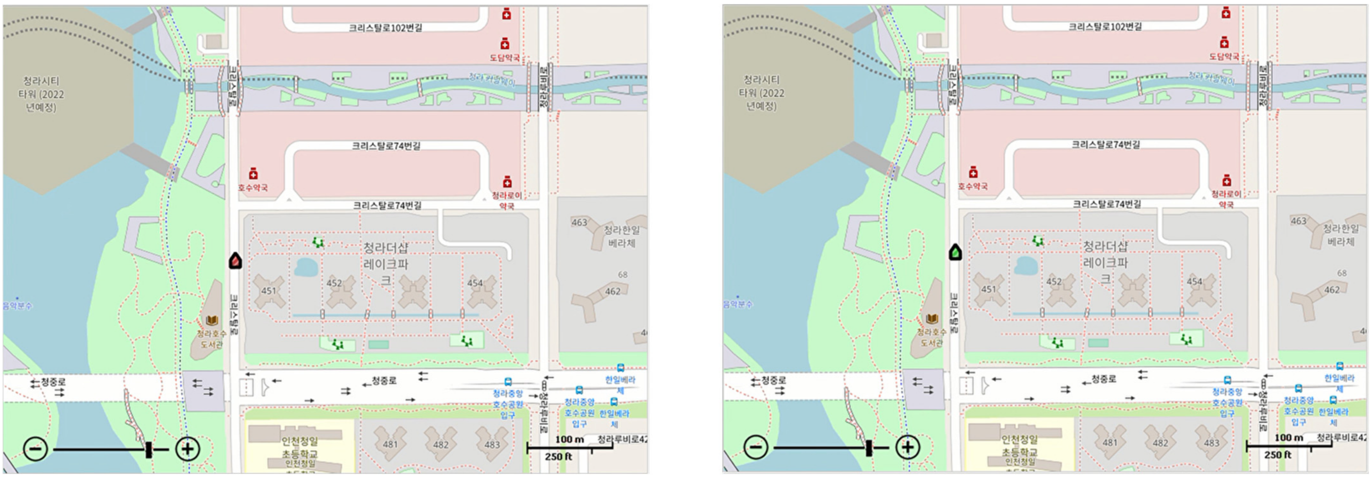
First scenario, the demonstration of abnormal vehicle is available at https://youtu.be/Vn67SosNBCM (Video S1: accessed on 15 December 2023) with additional multimedia.

The second scenario is depicted in Figure 7. The problem cannot be resolved using data alone, as the data matches perfectly with BSM from both vehicles. In such instances, if the messages received from the RSU are identical for both vehicles, they can be processed through conditional statements. For instance, if two vehicles transmit identical location information, abnormal behavior could be detected by the RSU. Consequently, a message to readjust the RTC (Real Time Clock) in their systems could be sent by the RSU.



(**a**) (**b**)

Figure 6. Compare normal vehicle and abnormal vehicle in first scenario. (**a**) Steering wheel angle message in BSM; (**b**) WGS+84 message in BSM.



(**a**) (**b**)

Figure 7. Second scenario: appearance of overlap with two vehicles. (**a**) first vehicle; (**b**) second vehicle.

Second scenario, the demonstration of overlap with two vehicles is available at https://youtu.be/H9Pv0DYVi_E (Video S2: accessed on 15 December 2023) with additional multimedia

In the third scenario, as illustrated in Figure 8, the situation appears normal when viewed through the BSM, similar to the second scenario. However, the addition of the *linePosition* information to BSM Part 2 data, as proposed in Table 3, enables the detection of abnormalities without relying on GPS data calculations. The *linePosition* element can be readily computed using cameras installed in autonomous vehicles.

Third scenario, the demonstration of counterflow vehicle is available at https://youtu.be/6bR8BxJc6_M (Video S3: accessed on 15 December 2023) with additional multimedia

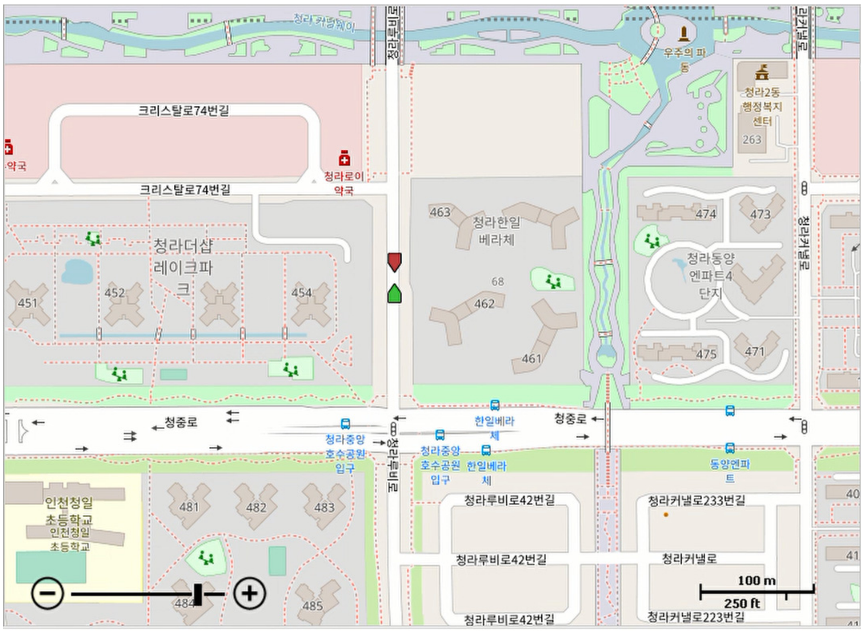


Figure 8. Third scenario: appearance of counterflow vehicle.

5. Detection Abnormal Vehicle with Machine Learning

In Section 5, we experiment with a method of detecting abnormal vehicle behavior using machine learning. In this experiment, the data generated by the simulator in Section 4 is tested on an edge computer. The edge computer used is an industrial PC equipped with a low-power i7-6600u CPU, 16 GB DDR4 RAM, and 512 GB SSD, suitable for the environment inside the RSU enclosure. This specification is apt for handling machine learning computations using the CPU, and GPU-based machine learning methods demanding high computing resources are excluded from this paper. Among various machine learning algorithms, Oneclass-SVM, K-Means, HDBSCAN, and Minisom are employed to detect abnormal vehicle behavior. In the case of HDBSCAN, based on DBSCAN, it adapts more flexibly to varying densities, making it better suited for clustering complex data sets. The performance of these algorithms is evaluated in terms of accuracy, recall, and precision for three scenarios: sensor problems, overlap, and counterflow vehicles. The experimental values used are the vehicle's GPS information (aka. lat and lon in BSM), speed, and angle according to time.

In the sensor problem, Oneclass-SVM shows an accuracy of 47.9%, recall of 47.7%, and precision of 44.2%. K-Means does not produce significant results, but HDBSCAN and Minisom achieve 100% accuracy, recall, and precision, respectively. In counterflow, Oneclass-SVM records an accuracy of 51.5%, a recall rate of 23.1%, and a precision of 19.5%. K-Means shows low results, and HDBSCAN shows 65% accuracy, 40.3% recall, and 10.1% precision. Minisom shows the highest performance with 66.7% accuracy, 50% recall, and 93.8% precision. In an overlap, Oneclass-SVM shows an accuracy of 51.5%, a recall rate of 33.7%, and a precision of 47.1%. K-Means and Minisom achieve 100% accuracy, recall, and precision, and HDB-SCAN achieves 41% accuracy and 36.1% recall.

These results confirm that Minisom consistently shows higher performance than other algorithms in sensor problems, overlap, and counterflow situations. This suggests that Minisom is a very effective tool for detecting abnormal behavior. In particular, the BSM heading change amount is found to be a significant feature in sensor problems, and the BSM heading value is found to be a significant feature in vehicle overlap. In situations where vehicles overlap, there is no significant difference between normal driving data and abnormal driving data. The results of this analysis are detailed in Table 4.

Table 4. Detected abnormal vehicle with machine learning.

Algorithm	**Oneclass-SVM**			**K-Means**			**HDBSCAN**			**Minisom**		
Abnormal	Accuracy	Recall	Precision	Accuracy	Recall	Precision	Accuracy	Recall	Precision	Accuracy	Recall	Precision
Sensor problem	47.9	47.7	44.2	50	0	0	100	100	100	100	100	100
Overlap	51.5	23.1	19.5	66.7	0	0	65	40.3	10.1	66.7	50	93.8
Counterflow	51.5	33.7	47.1	100	100	100	41	36.1	100	100	100	100

In our research, we investigated the integration of a new *linePosition* element into Minisom. Remarkably, with the addition of *linePosition*, Minisom's ability to detect overlapping vehicles improved significantly, achieving 100% accuracy, recall, and precision. This finding highlights the potential of enhancing BSM for more effective abnormal behavior detection. However, we acknowledge the practical challenges of updating all vehicles to accommodate the new BSM. Therefore, to reflect the current state of vehicle technology and ensure the broad applicability of our results, we presented the findings in Table 4 without incorporating the *linePosition* element.

When such issues arise, the RSU is capable of prompt intervention. For the first scenario, it is feasible to dismiss anomalies detected in the vehicle's data and inform other vehicles to prepare accordingly. Moreover, the RSU can alert that a malfunction has occurred in an abnormal vehicle, and in urgent cases, it can inform the police and road management authorities to safely halt the vehicle. In the second scenario, if special situations like construction sites are registered with the road management office, the map information can be updated with relevant data and communicated to other vehicles. In cases of aberrant behaviors, the RSU can notify the police and assist in the immediate resolution of issues caused by such behaviors. The third scenario deals with situations where both vehicles exhibit abnormal behavior. In this case, the RSU can advise both vehicles to synchronize their system time. Communication failures can be addressed by prompting the use of the vehicle's built-in watchdog timer. This timer is a crucial safety feature within the vehicle's system that triggers a reset if the system becomes unresponsive, ensuring continuous and reliable functionality. In other instances, the application of machine learning to the RSU's edge computer assists autonomous vehicles in making independent decisions. Our research contributes to ensuring passenger safety and facilitating smoother traffic flow on the roads.

6. Conclusions

We emphasize the importance of abnormal behavior detection according to the development of autonomous vehicle sensors and driving algorithms, research data-centric approaches, and machine learning-based abnormal behavior detection methods. In this paper, we present an approach to identifying abnormal behavior by utilizing communication between the vehicle and RSU, excluding image-based processing. By simulating V2X communication, we analyzed the abnormal behavior of autonomous vehicles through vehicle communication protocols, and based on this, experiments are conducted on an edge computer that will be installed in the actual RSU enclosure.

In this paper, we applied various machine learning algorithms to detect abnormal behavior in autonomous vehicles, focusing on scenarios involving sensor problems, counterflow, and vehicle overlap. Notably, the Minisom algorithm emerged as a consistently high performer, demonstrating its effectiveness in these complex scenarios. For instance, in sensor problem scenarios, Minisom achieved an accuracy of 100%, significantly outperforming OneClass-SVM, which showed an accuracy of 47.9%, recall of 47.7%, and precision of 44.2%. Similarly, in counterflow and overlap scenarios, Minisom again demonstrated superior performance with accuracy, recall, and precision all reaching 100%, while HDBSCAN and OneClass-SVM showed varied inferior results.

Furthermore, we propose two main enhancements to our system. First, we recommend adding the *linePosition* element to BSM Part 2 to simplify calculations and expedite the detection of abnormal behavior. Second, we suggest the application of machine learning on edge computers within RSUs to enable immediate action in scenarios such as sensor errors, counterflow, and vehicle overlap, allowing for autonomous decisions.

We present the possibility that, in the event of a breakdown of an autonomous vehicle, an external system can monitor the vehicle's status and take necessary actions. We provide a new direction for identifying vehicle interior problems from the outside and, if necessary, taking appropriate measures to ensure the safety of passengers. This provides an opportunity to respond immediately through an external system when a breakdown in an autonomous vehicle is not directly recognized by passengers. This approach contributes to increasing the reliability and safety of autonomous vehicles and will become an important standard for future research and development.

Supplementary Materials: The following are available online at https://youtu.be/Vn67SosNBCM, Video S1: 1. Sensor problem. The following are available online at https://youtu.be/H9Pv0DYVi_E, Video S2: 2. Overlap. The following are available online at https://youtu.be/6bR8BxJc6_M, Video S3: 3. Counterflow.

Author Contributions: Conceptualization, K.Y., H.Y. and M.K.; methodology, K.Y.; software, J.L.; validation, K.Y., H.Y. and J.S.; formal analysis, J.O.; investigation, S.L.; resources, C.K., J.S. and J.C.; data curation, J.O. and S.L.; writing—original draft preparation, K.Y. and H.Y.; writing—review and editing, M.K.; visualization, H.Y.; supervision, M.L.; project administration, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00304, Development of Total Security Platform to Protect Autonomous Car and Intelligent Traffic System Under 5G Environment, 100%).

Data Availability Statement: Data are contained within the article and Supplementary Materials.

Conflicts of Interest: Author Keon Yun, Heesun Yun, Sangmin Lee, Jinhyeok Oh, Minchul Kim and Myongcheol Lim were employed by the company Pentasecurity. Author Jinyoung Choi was employed by the company Xbrain. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Wu, D.; Tu, S.Z.; Whalin, R.W.; Zhang, L. Adaptive Individual-Level Cognitive Driving Anomaly Detection Model Exclusively Using BSMs. *Vehicles* **2023**, *5*, 1275–1293. [CrossRef]
2. Ranaweera, M.; Seneviratne, A.; Rey, D.; Saberi, M.; Dixit, V.V. Detection of anomalous vehicles using physics of traffic. *Veh. Commun.* **2021**, *27*, 100304. [CrossRef]
3. Zamouche, D.; Aissani, S.; Omar, M.; Mohammedi, M. Highly efficient approach for discordant BSMs detection in connected vehicles environment. *Wirel. Netw.* **2023**, *29*, 189–207. [CrossRef]
4. Yang, Z.; Ying, J.; Shen, J.; Feng, Y.; Chen, Q.A.; Mao, Z.M.; Liu, H.X. Anomaly Detection Against GPS Spoofing Attacks on Connected and Autonomous Vehicles Using Learning From Demonstration. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 9462–9475. [CrossRef]
5. Humayun, M.; Ashfaq, F.; Jhanjhi, N.Z.; Alsadun, M.K. Traffic management: Multi-scale vehicle detection in varying weather conditions using yolov4 and spatial pyramid pooling network. *Electronics* **2022**, *11*, 2748.
6. Farid, A.; Hussain, F.; Khan, K.; Shahzad, M.; Khan, U.; Mahmood, Z. A Fast and Accurate Real-Time Vehicle Detection Method Using Deep Learning for Unconstrained Environments. *Appl. Sci.* **2023**, *13*, 3059. [CrossRef]
7. Guo, D.; Wang, Y.; Zhu, S.; Li, X. A Vehicle Detection Method Based on an Improved U-YOLO Network for High-Resolution Remote-Sensing Images. *Sustainability* **2023**, *15*, 10397. [CrossRef]
8. Soe, M.T.; Min, A.Z.; Kyaw, H.T.; Paing, M.M.; Htet, S.M.; Aye, B. Abnormal Behavior Detection in Real-time for Advanced Driver Assistance System (ADAS) using YOLO. In Proceedings of the 2022 IEEE Symposium on Industrial Electronics & Applications (ISIEA), Langkawi Island, Malaysia, 16–17 July 2022; pp. 1–6.
9. Sankaranarayanan, M.; Aggarwal, M.; Mala, C. Semi-automatic Vehicle Detection System for Road Traffic Management. In Proceedings of the 3rd International Conference on Artificial Intelligence: Advances and Applications: ICAIAA 2022; Springer Nature Singapore: Singapore, 2023; pp. 303–314.
10. Vu, L.; Nguyen, Q.U.; Nguyen, D.N.; Hoang, D.T.; Dutkiewicz, E. Learning latent representation for IoT anomaly detection. *IEEE Trans. Cybern.* **2020**, *52*, 3769–3782. [CrossRef]

11. Ryan, C.; Murphy, F.; Mullins, M. End-to-end autonomous driving risk analysis: A behavioural anomaly detection approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1650–1662. [CrossRef]
12. Alladi, T.; Gera, B.; Agrawal, A.; Chamola, V.; Yu, F.R. DeepADV: A deep neural network framework for anomaly detection in VANETs. *IEEE Trans. Veh. Technol.* **2021**, *70*, 12013–12023. [CrossRef]
13. Bifulco, G.N.; Coppola, A.; Petrillo, A.; Santini, S. Decentralized Cooperative Crossing at Unsignalized Intersections via Vehicle-to-Vehicle Communication in Mixed Traffic Flows. *J. Intell. Transp. Syst.* **2022**, 1–26. [CrossRef]
14. Lu, Y.; Ma, H.; Smart, E.; Yu, H. Real-time Performance-focused Localization Techniques for Autonomous Vehicle: A Review. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 6082–6100. [CrossRef]
15. Chakraborty, R.; Kumar, S.; Awasthi, A.; Suneetha, K.; Rastogi, A.; Jethava, G. Machine Learning Based Novel Frameworks Developments and Architectures for Secured Communication in VANETs for Smart Transportation. *Soft Comput.* **2023**, 1–11. [CrossRef]
16. Nusinovici, S.; Tham, Y.C.; Yan, M.Y.C.; Ting, D.S.W.; Li, J.; Sabanayagam, C.; Wong, T.Y.; Cheng, C.-Y. Logistic regression was as good as machine learning for predicting major chronic diseases. *J. Clin. Epidemiol.* **2020**, *122*, 56–69. [CrossRef]
17. Bhushan, S.; Alshehri, M.; Keshta, I.; Chakraverti, A.K.; Rajpurohit, J.; Abugabah, A. An experimental analysis of various machine learning algorithms for hand gesture recognition. *Electronics* **2022**, *11*, 968. [CrossRef]
18. Alzahrani, R.J.; Alzahrani, A. Security analysis of ddos attacks using machine learning algorithms in networks traffic. *Electronics* **2021**, *10*, 2919. [CrossRef]
19. Gupta, C.; Johri, I.; Srinivasan, K.; Hu, Y.-C.; Qaisar, S.M.; Huang, K.-Y. A systematic review on machine learning and deep learning models for electronic information security in mobile networks. *Sensors* **2022**, *22*, 2017. [CrossRef] [PubMed]
20. Huang, Y.; Doh, I.-J.; Bae, E. Design and validation of a portable machine learning-based electronic nose. *Sensors* **2021**, *21*, 3923. [CrossRef]
21. Khalid, M.J.; Irfan, M.; Ali, T.; Gull, M.; Draz, U.; Glowacz, A.; Sulowicz, M.; Dziechciarz, A.; AlKahtani, F.S.; Hussain, S. Integration of discrete wavelet transform, DBSCAN, and classifiers for efficient content based image retrieval. *Electronics* **2020**, *9*, 1886. [CrossRef]
22. Khan, M.A.H.; Thomson, B.; Debnath, R.; Motayed, A.; Rao, M.V. Nanowire-based sensor array for detection of cross-sensitive gases using PCA and machine learning algorithms. *IEEE Sens. J.* **2020**, *20*, 6020–6028. [CrossRef]
23. Waskle, S.; Parashar, L.; Singh, U. Intrusion detection system using PCA with random forest approach. In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2–4 July 2020; pp. 803–808.
24. Bukhari, S.N.H.; Webber, J.; Mehbodniya, A. Decision tree based ensemble machine learning model for the prediction of Zika virus T-cell epitopes as potential vaccine candidates. *Sci. Rep.* **2022**, *12*, 7810. [CrossRef] [PubMed]
25. Liang, D.; Frederick, D.A.; Lledo, E.E.; Rosenfield, N.; Berardi, V.; Linstead, E.; Maoz, U. Examining the utility of nonlinear machine learning approaches versus linear regression for predicting body image outcomes: The US Body Project I. *Body Image* **2022**, *41*, 32–45. [CrossRef] [PubMed]
26. Baturynska, I.; Martinsen, K. Prediction of geometry deviations in additive manufactured parts: Comparison of linear regression with machine learning algorithms. *J. Intell. Manuf.* **2021**, *32*, 179–200. [CrossRef]
27. García-Nieto, P.J.; García-Gonzalo, E.; Paredes-Sánchez, J.P. Prediction of the critical temperature of a superconductor by using the WOA/MARS, Ridge, Lasso and Elastic-net machine learning techniques. *Neural Comput. Appl.* **2021**, *33*, 17131–17145. [CrossRef]
28. Chen, D.-L.; Cai, J.-H.; Wang, C.C. Identification of key prognostic genes of triple negative breast cancer by LASSO-based machine learning and bioinformatics analysis. *Genes* **2022**, *13*, 902. [CrossRef]
29. Johnsen, T.K.; Gao, J.Z. Elastic net to forecast COVID-19 cases. In Proceedings of the 2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT), Sakheer, Bahrain, 20–21 December 2020; pp. 1–6.
30. Zhang, Y.; Dai, Y.; Wu, Q. An accelerated optimization algorithm for the elastic-net extreme learning machine. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 3993–4011. [CrossRef]
31. Sandhu, A.K.; Batth, R.S. Software reuse analytics using integrated random forest and gradient boosting machine learning algorithm. *Softw. Pract. Exp.* **2021**, *51*, 735–747. [CrossRef]
32. Shrivastav, L.K.; Jha, S.K. A gradient boosting machine learning approach in modeling the impact of temperature and humidity on the transmission rate of COVID-19 in India. *Appl. Intell.* **2021**, *51*, 2727–2739. [CrossRef]
33. Ileberi, E.; Sun, Y.; Wang, Z. Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost. *IEEE Access* **2021**, *9*, 165286–165294. [CrossRef]
34. Chen, S.; Shen, B.; Wang, X.; Yoo, S.-J. A strong machine learning classifier and decision stumps based hybrid adaboost classification algorithm for cognitive radios. *Sensors* **2019**, *19*, 5077. [CrossRef]
35. Lieskovská, E.; Jakubec, M.; Jarina, R.; Chmulík, M. A review on speech emotion recognition using deep learning and attention mechanism. *Electronics* **2021**, *10*, 1163. [CrossRef]
36. Atik, I. Classification of electronic components based on convolutional neural network Architecture. *Energies* **2022**, *15*, 2347. [CrossRef]
37. Chien, J.-C.; Wu, M.-T.; Lee, J.-D. Inspection and classification of semiconductor wafer surface defects using CNN deep learning networks. *Appl. Sci.* **2020**, *10*, 5340. [CrossRef]

38. Bisen, D.; Lilhore, U.K.; Manoharan, P.; Dahan, F.; Mzoughi, O.; Hajjej, F.; Saurabh, P.; Raahemifar, K. A Hybrid Deep Learning Model Using CNN and K-Mean Clustering for Energy Efficient Modelling in Mobile EdgeIoT. *Electronics* **2023**, *12*, 1384. [CrossRef]
39. Zou, Y.; Lv, J. Using recurrent neural network to optimize electronic nose system with dimensionality reduction. *Electronics* **2020**, *9*, 2205. [CrossRef]
40. Xiong, J.; Yu, D.; Liu, S.; Shu, L.; Wang, X.; Liu, Z. A review of plant phenotypic image recognition technology based on deep learning. *Electronics* **2021**, *10*, 81. [CrossRef]
41. Awad, F.H.; Hamad, M.M. Improved k-means clustering algorithm for big data based on distributed smartphoneneural engine processor. *Electronics* **2022**, *11*, 883. [CrossRef]
42. Laskar, M.T.R.; Huang, J.X.; Smetana, V.; Stewart, C.; Pouw, K.; An, A.; Chan, S.; Liu, L. Extending isolation forest for anomaly detection in big data via K-means. *ACM Trans. Cyber-Phys. Syst. TCPS* **2021**, *5*, 1–26. [CrossRef]
43. Ullah, B.; Kamran, M.; Rui, Y. Predictive modeling of short-term rockburst for the stability of subsurface structures using machine learning approaches: T-SNE, K-Means clustering and XGBoost. *Mathematics* **2022**, *10*, 449. [CrossRef]
44. Hozumi, Y.; Wang, R.; Yin, C.; Wei, G.-W. UMAP-assisted K-means clustering of large-scale SARS-CoV-2 mutation datasets. *Comput. Biol. Med.* **2021**, *131*, 104264. [CrossRef]
45. Benbrahim Ansari, O. Geo-marketing segmentation with deep learning. *Businesses* **2021**, *1*, 51–71. [CrossRef]
46. Morales, F.; García-Torres, M.; Velázquez, G.; Daumas-Ladouce, F.; Gardel-Sotomayor, P.E.; Gómez-Vela, F.; Divina, F.; Vázquez Noguera, J.L.; Sauer Ayala, C.; Pinto-Roa, D.P. Analysis of electric energy consumption profiles using a machine learning approach: A Paraguayan case study. *Electronics* **2022**, *11*, 267. [CrossRef]
47. Jasiński, M.; Sikorski, T.; Leonowicz, Z.; Borkowski, K.; Jasińska, E. The application of hierarchical clustering to power quality measurements in an electrical power network with distributed generation. *Energies* **2020**, *13*, 2407. [CrossRef]
48. Yin, Z.; Zhang, B. Construction of Personalized Bus Travel Time Prediction Intervals Based on Hierarchical Clustering and the Bootstrap Method. *Electronics* **2023**, *12*, 1917. [CrossRef]
49. Ahmadi, M.; Taghavirashidizadeh, A.; Javaheri, D.; Masoumian, A.; Ghoushchi, S.J.; Pourasad, Y. DQRE-SCnet: A novel hybrid approach for selecting users in federated learning with deep-Q-reinforcement learning based on spectral clustering. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 7445–7458. [CrossRef]
50. Berahmand, K.; Mohammadi, M.; Faroughi, A.; Mohammadiani, R.P. A novel method of spectral clustering in attributed networks by constructing parameter-free affinity matrix. *Clust. Comput.* **2022**, 1–20. [CrossRef]
51. Janani, R.; Vijayarani, S. Text document clustering using spectral clustering algorithm with particle swarm optimization. *Expert Syst. Appl.* **2019**, *134*, 192–200. [CrossRef]
52. Binbusayyis, A.; Vaiyapuri, T. Unsupervised deep learning approach for network intrusion detection combining convolutional autoencoder and one-class SVM. *Appl. Intell.* **2021**, *51*, 7094–7108. [CrossRef]
53. Wang, Z.; Cha, Y.-J. Unsupervised deep learning approach using a deep auto-encoder with a one-class support vector machine to detect damage. *Struct. Health Monit.* **2021**, *20*, 406–425. [CrossRef]
54. Ahmed, S.; Lee, Y.; Hyun, S.-H.; Koo, I. Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2765–2777. [CrossRef]
55. You, L.; Peng, Q.; Xiong, Z.; He, D.; Qiu, M.; Zhang, X. Integrating aspect analysis and local outlier factor for intelligent review spam detection. *Future Gener. Comput. Syst.* **2020**, *102*, 163–172. [CrossRef]
56. Alghushairy, O.; Alsini, R.; Soule, T.; Ma, X. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cogn. Comput.* **2020**, *5*, 1. [CrossRef]
57. Grollemund, V.; Chat, G.L.; Secchi-Buhour, M.-S.; Delbot, F.; Pradat-Peyre, J.-F.; Bede, P.; Pradat, P.-F. Development and validation of a 1-year survival prognosis estimation model for Amyotrophic Lateral Sclerosis using manifold learning algorithm UMAP. *Sci. Rep.* **2020**, *10*, 13378. [CrossRef] [PubMed]
58. Lim, S.J.; Seo, J.; Seid, M.G.; Lee, J.; Ejerssa, W.W.; Lee, D.-H.; Jeong, E.; Chae, S.H.; Lee, Y.; Son, M. Clustering micropollutants and estimating rate constants of sorption and biodegradation using machine learning approaches. *Npj Clean Water* **2023**, *6*, 69. [CrossRef]
59. Wang, C. Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach. *Inf. Process. Manag.* **2022**, *59*, 103085. [CrossRef]
60. Sornalakshmi, M.; Balamurali, S.; Venkatesulu, M.; Navaneetha Krishnan, M.; Ramasamy, L.K.; Kadry, S.; Manogaran, G.; Hsu, C.-H.; Muthu, B.A. Hybrid method for mining rules based on enhanced Apriori algorithm with sequential minimal optimization in healthcare industry. *Neural Comput. Appl.* **2020**, *34*, 10597–10610. [CrossRef]
61. Mohapatra, D.; Tripathy, J.; Mohanty, K.K.; Nayak, D.S.K. Interpretation of optimized hyper parameters in associative rule learning using eclat and apriori. In Proceedings of the 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8–10 April 2021; pp. 879–882.
62. Rozi, F.; Sukmana, F. Unsupervised Machine Learning Using Fp-Growth in Service and Maintenance of Asset Management. *Int. J. Artif. Intell. Res.* **2022**, *6*. [CrossRef]
63. Shawkat, M.; Badawi, M.; El-ghamrawy, S.; Arnous, R.; El-desoky, A. An optimized FP-growth algorithm for discovery of association rules. *J. Supercomput.* **2022**, *78*, 5479–5506. [CrossRef]
64. Bo, D.; Wang, X.; Shi, C.; Zhu, M.; Lu, E.; Cui, P. Structural deep clustering network. In Proceedings of the Web Conference 2020, Taipei, Taiwan, 20–24 April 2020; pp. 1400–1410.

65. Peng, X.; Zhu, H.; Feng, J.; Shen, C.; Zhang, H.; Zhou, J.T. Deep clustering with sample-assignment invariance prior. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 4857–4868. [CrossRef]
66. Di Mattia, F.; Galeone, P.; De Simoni, M.; Ghelfi, E. A survey on gans for anomaly detection. *arXiv Preprint* **2019**, arXiv:1906.11632.
67. Dong, H.; Dong, H.; Ding, Z.; Zhang, S. *Deep Reinforcement Learning*; Springer: Berlin/Heidelberg, Germany, 2020; ISBN 978-981-15-4094-3.
68. Wang, H.-N.; Liu, N.; Zhang, Y.-Y.; Feng, D.-W.; Huang, F.; Li, D.-S.; Zhang, Y.-M. Deep reinforcement learning: A survey. *Front. Inf. Technol. Electron. Eng.* **2020**, *21*, 1726–1744. [CrossRef]
69. Vector. CANoe. Available online: https://www.vector.com/kr/ko/products/products-a-z/software/canoe (accessed on 13 December 2023).
70. SAE International. J2735_202309: V2X Communications Message Set Dictionary, SAE International, 2023. Available online: https://www.sae.org/standards/content/j2735_202309/ (accessed on 13 December 2023).
71. Jaemyeong, K.; Yunsoo, C.; Hasu, Y.; Wonjong, L. Optimal Space Interpolation Method for Continuous Marine Vertical Datum Based on WGS-84 Ellipsoid. *Sens. Mater.* **2019**, *31*, 3917–3930. [CrossRef]