World Electric Vehicle Journal

Article

An Intelligent Attack Detection Framework for the Internet of Autonomous Vehicles with Imbalanced Car Hacking Data

Samah Alshathri [1], Amged Sayed [2,3,*] and Ezz El-Din Hemdan [4,5]

[1] Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; sealshathry@pnu.edu.sa
[2] Industrial Electronics and Control Engineering Department, Faculty of Electronic Engineering, Menoufia University, Menouf 32952, Egypt
[3] Department of Electrical Energy Engineering, College of Engineering & Technology, Arab Academy for Science Technology & Maritime Transport, Smart Village Campus, Giza 12577, Egypt
[4] Structure and Materials Research Lab, Prince Sultan University, P.O. Box 66833, Riyadh 11586, Saudi Arabia; ezzeldinhemdan@el-eng.menofia.edu.eg
[5] Department of Computer Science and Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf 32952, Egypt
* Correspondence: amgad.mahmoud@el-eng.menofia.edu.eg

Abstract: The modern Internet of Autonomous Vehicles (IoVs) has enabled the development of autonomous vehicles that can interact with each other and their surroundings, facilitating real-time data exchange and communication between vehicles, infrastructure, and the external environment. The lack of security procedures in vehicular networks and Controller Area Network (CAN) protocol leaves vehicles exposed to intrusions. One common attack type is the message injection attack, which inserts fake messages into original Electronic Control Units (ECUs) to trick them or create failures. Therefore, this paper tackles the pressing issue of cyber-attack detection in modern IoV systems, where the increasing connectivity of vehicles to the external world and each other creates a vast attack surface. The vulnerability of in-vehicle networks, particularly the CAN protocol, makes them susceptible to attacks such as message injection, which can have severe consequences. To address this, we propose an intelligent Intrusion detection system (IDS) to detect a wide range of threats utilizing machine learning techniques. However, a significant challenge lies in the inherent imbalance of car-hacking datasets, which can lead to misclassification of attack types. To overcome this, we employ various imbalanced pre-processing techniques, including NearMiss, Random over-sampling (ROS), and TomLinks, to pre-process and handle imbalanced data. Then, various Machine Learning (ML) techniques, including Logistic Regression (LR), Linear Discriminant Analysis (LDA), Naive Bayes (NB), and K-Nearest Neighbors (k-NN), are employed in detecting and predicting attack types on balanced data. We evaluate the performance and efficacy of these techniques using a comprehensive set of evaluation metrics, including accuracy, precision, F1_Score, and recall. This demonstrates how well the suggested IDS detects cyberattacks in external and intra-vehicle vehicular networks using unbalanced data on vehicle hacking. Using k-NN with various resampling techniques, the results show that the proposed system achieves 100% detection rates in testing on the Car-Hacking dataset in comparison with existing work, demonstrating the effectiveness of our approach in protecting modern vehicle systems from advanced threats.

Keywords: car hacking; Internet of Vehicles; imbalanced data; intrusion detection systems; RANDOM over-sampling (ROS); NearMiss; machine learning

check for updates

Citation: Alshathri, S.; Sayed, A.; Hemdan, E.E.-D. An Intelligent Attack Detection Framework for the Internet of Autonomous Vehicles with Imbalanced Car Hacking Data. *World Electr. Veh. J.* **2024**, *15*, 356. https://doi.org/10.3390/wevj15080356

Academic Editor: Jiangtao Li

Received: 22 July 2024
Revised: 27 July 2024
Accepted: 29 July 2024
Published: 8 August 2024

1. Introduction

Recently, machine learning models have had numerous applications across various fields, including healthcare, finance, transportation, and industries [1–4]. Some common applications of machine learning include image recognition, fraud detection, and cybersecurity. Cybersecurity is essential to detect attacks on the Internet of Things (IoT) and Industrial IoT systems [5–7]. Multiple learning methods are evaluated using dataset attack detection scenarios in various applications such as SDN, IoT, and Cloud. The increased reliance on computers and the internet has made cybersecurity a growing source of concern. Building effective Intrusion Detection Systems is necessary to identify cyberattacks, and the foundation for doing so is the ability to analyze traffic flow data swiftly and efficiently, referred to here as cybersecurity data in modern communication systems.

Modern cars have evolved into network-controlled vehicles, including Autonomous vehicles, vehicle-to-grid, and grid-to-vehicle, as a result of the rapid growth of information technology and IoV technologies [8]. Intra-vehicle networks (IVNs) and external networks are common components of IoV systems. The primary system of IVNs that permits communication between Electronic Control Units (ECUs) to carry out activities and implement functionality is a Controller Area Network (CAN) bus. In contrast, External vehicle connections enable communication linking intelligent cars and other IoV units, such as roadside infrastructure, road users, and roadside units [9–11].

Due to its many advantages over conventional cars, autonomous electric vehicles (AEVs) are the future of transportation [8]. AEVs use cutting-edge technologies such as radar, cameras, satellite navigation systems, artificial intelligence, and the internet to navigate roads safely and effectively without requiring human intervention. They connect with other vehicles, infrastructure, and cloud systems via a variety of wireless technologies, including 4G and 5G networks, to enable real-time information transmission, allowing autonomous vehicles to monitor and adapt to their environment, thereby increasing road safety, traffic management, and the overall driving experience. Additionally, features like automated parking, remote control, and real-time traffic alerts add to the convenience of AEVs. Due to their reliance on remote control and internet connectivity, AEVs may present security problems [12–19]. These threats include software flaws, sensor spoofing, remote exploitation of software systems and wireless communication protocols, and unauthorized access to vital vehicle functions. Car companies must adopt safe connectivity practices to reduce intrusion behavior, as shown in Figure 1.

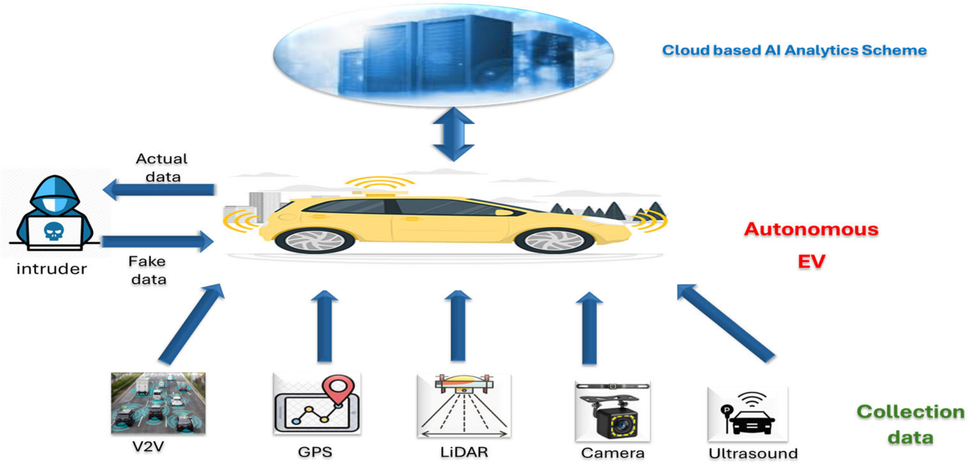

Figure 1. Intrusion System Analysis for Autonomous Car in Smart City.

Autonomous vehicles, which are sometimes referred to as self-driving automobiles, have enormous growth potential in the automotive industry, as shown in Figure 2. Several automakers are developing self-driving vehicles, such as Tesla, Audi, Waymo, General Motors, Mercedes-Benz, and Nissan [20,21]. Therefore, self-driving technology has been a major research topic recently. Because it can revolutionize transportation, it is regarded with both excitement and caution. Among the many benefits of self-driving cars is their ability to decrease human error and increase traffic efficiency. By providing a means of independent transportation that eliminates the need for driving, these cars can also benefit older people and those with visual or hearing problems [22–26].

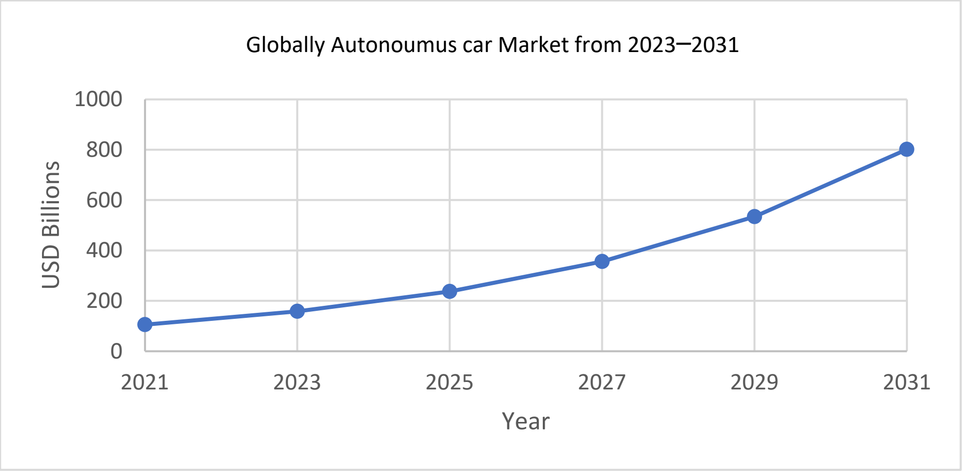

Figure 2. The forecast for the global market of self-driving cars with increase 25% yearly.

Because it enables more advanced and focused attacks on automobiles, machine learning has grown in popularity as a technique in the car hacking industry. Machine learning algorithms can find trends and weaknesses in automotive systems through the analysis of vast volumes of data that human analysts might not see right away. This may result in the creation of more potent hacking strategies, such as making use of common vulnerabilities in hardware or software. Furthermore, malware that is more sophisticated and resistant to detection by conventional security measures can be created via machine learning. Automakers and cybersecurity experts must keep abreast of the most recent developments to defend against the growing threat posed by machine learning-based automotive hacking. To safeguard contemporary industrial control systems from sophisticated severe attacks, it is important to provide an attack Detection System in automobile systems employing machine learning. On balanced datasets, however, machine learning models complete classification. Unfortunately, car-hacking datasets are unbalanced by nature. Therefore, imbalanced intrusion data classes are a considerable challenge due to the imbalanced class distribution deceiving algorithms from precisely assigning a trivial attack class. This work aims to resolve the imbalanced car-hacking data problem using different handling imbalanced pre-processing procedures and then to decide and predict the car attack. The experimental results proved that the performance of the proposed framework is assessed based on different metrics of accuracy, precision, and recall in comparing prevailing algorithms.

Machine learning models perform best when the number of samples in each class is roughly equal. This is so since many algorithms aim to minimize errors and increase accuracy. However, in the case of imbalanced data, one can still predict the majority class with a relatively high degree of accuracy, but it will not be able to capture the minority class, which is typically the main reason for building the model in the beginning. There are many methods of handling imbalanced data, as follows:

- NearMiss: A method of under-sampling. Using a distance will equalize the majority class with the minority class rather than resampling the minority class;
- Random over-sampling (ROS): A commonly used method for handling highly imbalanced datasets. It entails including additional instances from the minority class;
- Tomek links: Data can be cleaned up or under-sampled using Tomek connections. Tomek bases his data-cleaning method on the oversampled training set.

In brief, this paper presents an efficient Car Attack detection framework for handling imbalanced data problems in modern Internet of Autonomous Vehicles (IoV) systems. The proposed framework combines the ML method and imbalanced data techniques to fully increase system accuracy of attack detection and classification. The performance of the proposed framework is evaluated through a case study of the Car-Hacking dataset for car attack detection in the IoV applications in the context of a smart city. The essential contributions of this work can be summarized as follows:

- An efficient car attack detection framework for Vehicle systems is developed to help an asset in real-time intrusion detection. The proposed framework utilizes a combination of imbalanced techniques and Machine Learning models such as LR, LDA, NB, and k-NN effectively. The proposed hybrid IDS framework is endorsed using a car hacking dataset;
- From the experimental results, different resampling techniques such as NearMiss, ROS, and Tom-Links are used for pre-processing the car hacking dataset, which improves in converting imbalanced datasets into balanced datasets;
- The suggested IDS has shown 100% detection rates on the Car-Hacking dataset, a widely used benchmark IoV security dataset using k-NN with ROS and TomLinks Resampling Techniques. This demonstrates how well the suggested IDS detects cyberattacks in external and intra-vehicle networks using unbalanced data on vehicle hacking with imbalanced car hacking data.

The remainder of this paper is planned as follows. Section 2 presents the Materials and methods that are used, while Section 3 presents the details of the proposed attack prediction in the IoV systems framework. The proposed High-Level IDS-IoV Framework for Car Attacks Prediction is provided in Section 4, while Section 5 discusses the used dataset, evaluation metrics, and the analysis of the results. Finally, the conclusions of this paper are given in Section 6.

2. Materials and Methods

2.1. Overview of Car Hacking

Recently, vehicle system security has gradually become a key matter due to the expanded reliance on vehicle systems particularly with the advance of the Internet of Vehicles (IoV) in smart cities. In [10], their effort aims to defend both internal and external vehicular networks by creating an IDS that can identify different forms of threats. Figure 3 depicts the design of an IDS-protected vehicle as well as the usual assault scenario. By transmitting malicious traffic packets, Cybercriminals can use wireless interfaces to launch external assaults on external vehicle networks and the On-Board Diagnostics II (OBD II) interface to launch internal attacks on IVNs. Thus, both IVNs and external networks should use the recommended IDS.

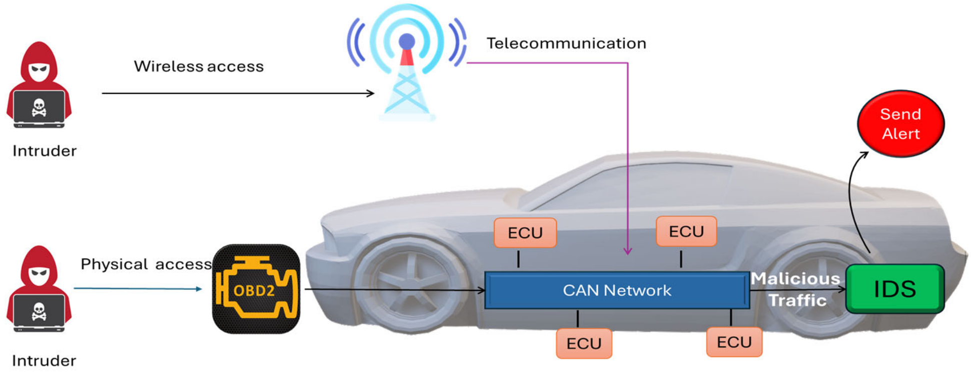

Figure 3. The IDS-protected vehicle architecture.

On the Car-Hacking dataset and the CICIDS2017 dataset, two commonly used public benchmark IoV security datasets, the suggested IDS has exhibited above 99.25% detection rates and F1_Scores in the trials. This demonstrates how well the suggested IDS works to identify cyberattacks in both internal and external vehicular networks [10]. To detect and predict car attacks in smart city systems, it is prudent that we build efficient Car Attack

Detection Systems to efficiently analyze and detect car attack data. The development of CNN-based IDS for automotive networks was the subject of several earlier publications.

A report [11] reveals that the vehicle industry has experienced a significant increase in cyberattacks on cars in the past decade. The frequency of attacks has increased by more than 200% within three years, with most attacks carried out remotely. There is an estimated loss for the auto industry that could reach $505 billion by 2024. These findings highlight the urgent need for robust cybersecurity measures in the automotive industry to protect against cyber threats and ensure the safety and privacy of drivers and passengers.

Mehedi et al. [8] presented the PLeNet approach for deep transfer learning-based in-vehicle network intrusion detection. On the car hacking dataset, the P-LeNet model received a good F1_Score above 97%. To solve several time-series data analytics difficulties, Authors in [12] suggested a one-dimensional CNN (1D-CNN) based IDS for intra-vehicle intrusion detection. To identify attacks on IVNs, Song et al. [13] suggested a deep CNN (DCNN) based IDS model utilizing reduced InceptionResnet. The Car-Hacking dataset demonstrates the great accuracy of the DCNN model. To detect security risks against in-vehicle networks early on, the study in [14] suggests using a deep learning architecture called DeepSecDrive. Lightweight, effective, and comprehensible units are used in the design of the framework to improve feature extraction and interpretability. The experimental results demonstrate that DeepSecDrive outperforms current state-of-the-art detection approaches in terms of effectiveness and durability against real-world IVN attacks. Khan et al. [15] introduced a multi-stage intrusion detection framework to identify intrusions from ITSs with a low rate of false alarms. Similarly, Ashraf et al. [16] presented a deep learning-based Intrusion Detection System (IDS) specifically designed for ITS to detect suspicious network activity in In-Vehicles Networks (IVN), vehicles-to-vehicles (V2V) communications, and vehicles-to-infrastructure (V2I) networks. The vulnerability of autonomous vehicles to hacking has been acknowledged by legal experts and authors. The paper [17] discusses the development of self-driving cars and self-parking systems that enhance safety and convenience using advanced electronics, information communication, and function control technologies. Additionally, it proposes a research methodology based on the Enhanced Security Model in Self-Driving Cars to defend against hacking attacks on smart cars, focusing on Availability Attacks, Man-in-the-middle Attacks, Imperial Password Use, and Inclusive Access Control attacks.

In IoT edge device implementations, a federated learning-based intrusion detection system (FL-IDS) is presented in [18] to improve the security of automotive networks. By adopting local learning, where devices only communicate model updates with an aggregate server, the FL-IDS system preserves data privacy. After that, the server creates an improved detection model. Additionally, the FL-IDS system includes a detection model that uses deep learning and machine learning classifiers to achieve overall accuracy of 94% and 99%, as well as loss of 0.28 and 0.009 for the Car-Hacking and NSL-KDD datasets, respectively. The efficacy of an intrusion detection system that combines the LightGBM algorithm for multiclass attack categorization in automotive CAN networks with an artificial neural network for feature extraction is demonstrated in [19], which achieves accuracy, precision, recall, F1_Scores, and an AUC-ROC score of 99.99%. A clear and organized comparison of the related works, highlighting their main advantages and disadvantages, is tabulated and explored in Table 1.

Proposing an IoV-based Intrusion Detection System (IoV_IDSs) using machine learning is an essential need to protect modern vehicle systems from advanced severe attacks. Nevertheless, machine learning models accomplish classification nicely on balanced datasets. Unluckily, car-hacking datasets are naturally imbalanced. Therefore, imbalanced class distribution causes algorithms to incorrectly label the small attack class, which makes imbalanced car-hacking classification a severe issue. This paper aims to resolve the imbalanced car hacking data problem using different imbalanced pre-processing procedures such as NearMiss, ROS, and TomLinks and then to decide and predict the kind of attack using ML models such as LR, LDA, NB, and k-NN.

Table 1. Advantages and disadvantages of related work on vehicle system security.

Study	Approach	Advantages	Disadvantages
[10]	IDS for both internal and external vehicular networks using CNN-based IDS	High detection rate (above 99.25%) on Car-Hacking and CICIDS2017 datasets—comprehensive protection for internal and external networks	Potential complexity in implementing CNN-based IDS—may require significant computational resources
[11]	Analysis of cyber-attack trends in the automotive industry	Highlights the increasing frequency and impact of cyberattacks—emphasizes the need for robust cybersecurity measures	Does not propose a specific IDS solution—mainly descriptive, lacking actionable recommendations
[8]	PLeNet approach using deep transfer learning for in-vehicle network intrusion detection	High F1_Score (above 97%) on Car-Hacking dataset—effective use of transfer learning for improved performance	Focused solely on in-vehicle networks—may not address external threats comprehensively
[12]	1D-CNN-based IDS for intra-vehicle intrusion detection	Effective for time-series data analytics—high accuracy in identifying IVN attacks	Limited to intra-vehicle network protection—may not scale well to larger datasets or external threats
[13]	Deep CNN (DCNN) using reduced InceptionResnet for IVN attack detection	High accuracy with Car-Hacking dataset—advanced model architecture for improved detection	Potentially high computational requirements—focused on IVN, not addressing external networks
[14]	DeepSecDrive: deep learning architecture for early IVN attack detection	Lightweight, effective, and interpretable—outperforms state-of-the-art detection approaches	Limited to in-vehicle networks—may require specialized knowledge for implementation
[15]	Multi-stage IDS framework for ITSs with low false alarm rate	Low rate of false alarms—multi-stage approach enhances detection accuracy	Complexity in multi-stage implementation—focused on ITSs, not broader vehicular networks

2.2. Car-Hacking Dataset

The Car-Hacking dataset includes DoS attacks, fuzzy attacks, spoofing the drive gear, and spoofing the RPM gauge. Figure 4 shows the class distribution of the car hacking_5% dataset. The detailed description of the dataset can be as follows [13]:

1. DoS Attack: CAN ID messages ‘0000’, which are the most common messages, are injected every 0.3 ms;
2. Fuzzy Attack: Every 0.5 ms, injecting messages with completely random CAN ID and DATA values;
3. Spoofing Attack (RPM/gear): Sending specific CAN ID messages pertaining to gear and RPM data every millisecond. The data attributes are Timestamp, CAN ID, DLC, DATA{0}, DATA{1}, DATA{2}, DATA{3}, DATA{4}, DATA{5}, DATA{6}, DATA{7}, Flag.
    1. Timestamp: recorded time (s);
    2. CAN ID: identifier of CAN message in HEX (ex. 043f);
    3. DLC: number of data bytes, from 0 to 8;
    4. DATA {0~7}: data value (byte);
    5. Flag: T or R, T represents an injected message, while R represents a normal message.

To examine the proposed framework, several performance metrics, such as accuracy, precision, recall, and F-measure, are used. These metrics are calculated from the confusion matrix for attack detection as illustrated in Table 2 as follows [27].

Table 2. Confusion Matrix (CM) for Attack Detection.

	Predicted Normal	Predicted Attack
Actual Normal	True Positive (TP)	False Negative (FN)
Actual Attack	False Positive (FP)	True Negative (TN)

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1\_score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

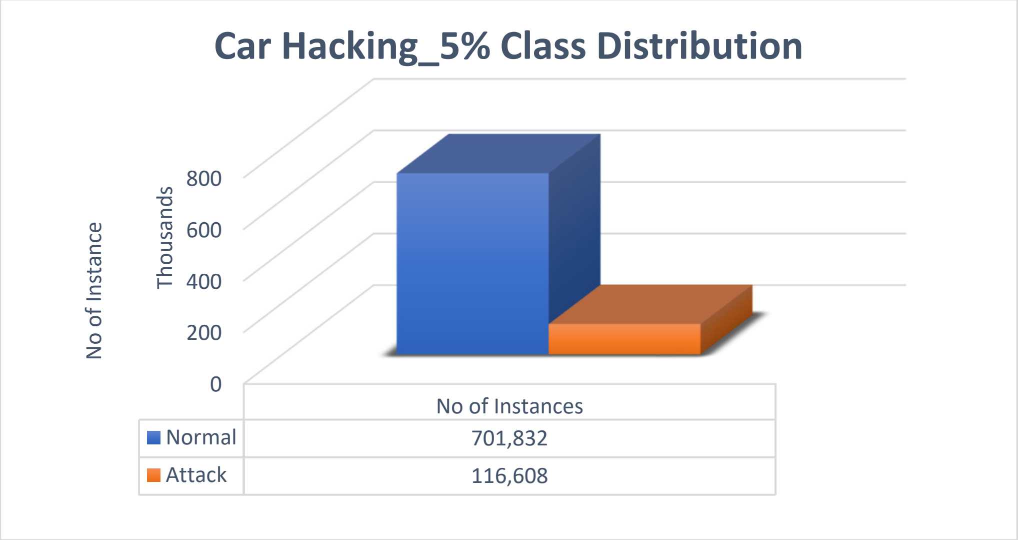

Figure 4. Class distribution for Car Hacking_5% dataset.

3. Proposed Attack Detection Framework

In this work, an effective methodology for automatically determining the state of assaults on vehicle systems is presented in this paper. The suggested intrusion architecture is illustrated in Figure 5 utilizing various machine learning techniques implemented on the unbalanced datasets, which are the automobile hacking dataset in IoV systems. The suggested framework includes several crucial steps to accomplish the car attacking detection system as follows:

- Phase 1: Data Preparation and Pre-processing

In this phase, the focus shifts to addressing the imbalance present in the industrial intrusion data through the utilization of resampling techniques. Specifically, methods such as NearMiss, ROS, and TomLinks are employed to rebalance the distribution of data instances across different classes. By implementing these techniques, the aim is to mitigate the challenges posed by imbalanced datasets, where certain classes are significantly underrepresented compared to others. NearMiss focuses on selecting a subset of majority class samples that are closest to minority class instances; ROS randomly replicates minority class samples to match the size of the majority class, while TomLinks identifies and removes overlapping instances between classes. Through the application of these resampling techniques, the imbalance within the data is effectively addressed, laying a solid foundation for subsequent processing and analysis within the proposed framework.

- Phase 2: Training and Testing of Selected Model

The pre-processed dataset is split 70–30 to begin the training phase of choosing and fine-tuning a specific machine learning algorithm. Random selections of training car hacking data are then sub-sampled for the machine learning algorithms, and performance evaluation metrics are applied to estimate the suggested framework.

- Phase 3: Car Attack Detection and Classification

In this phase, the testing car attack data are fed to the tuned machine learning algorithms to detect all the input real data into one of two types: Normal (negative) or attack (positive). To conclude, the comprehensive performing evaluation for every single machine learning algorithm will be evaluated based on classification-based performance metrics such as accuracy, precision, recall, and F1_Score.

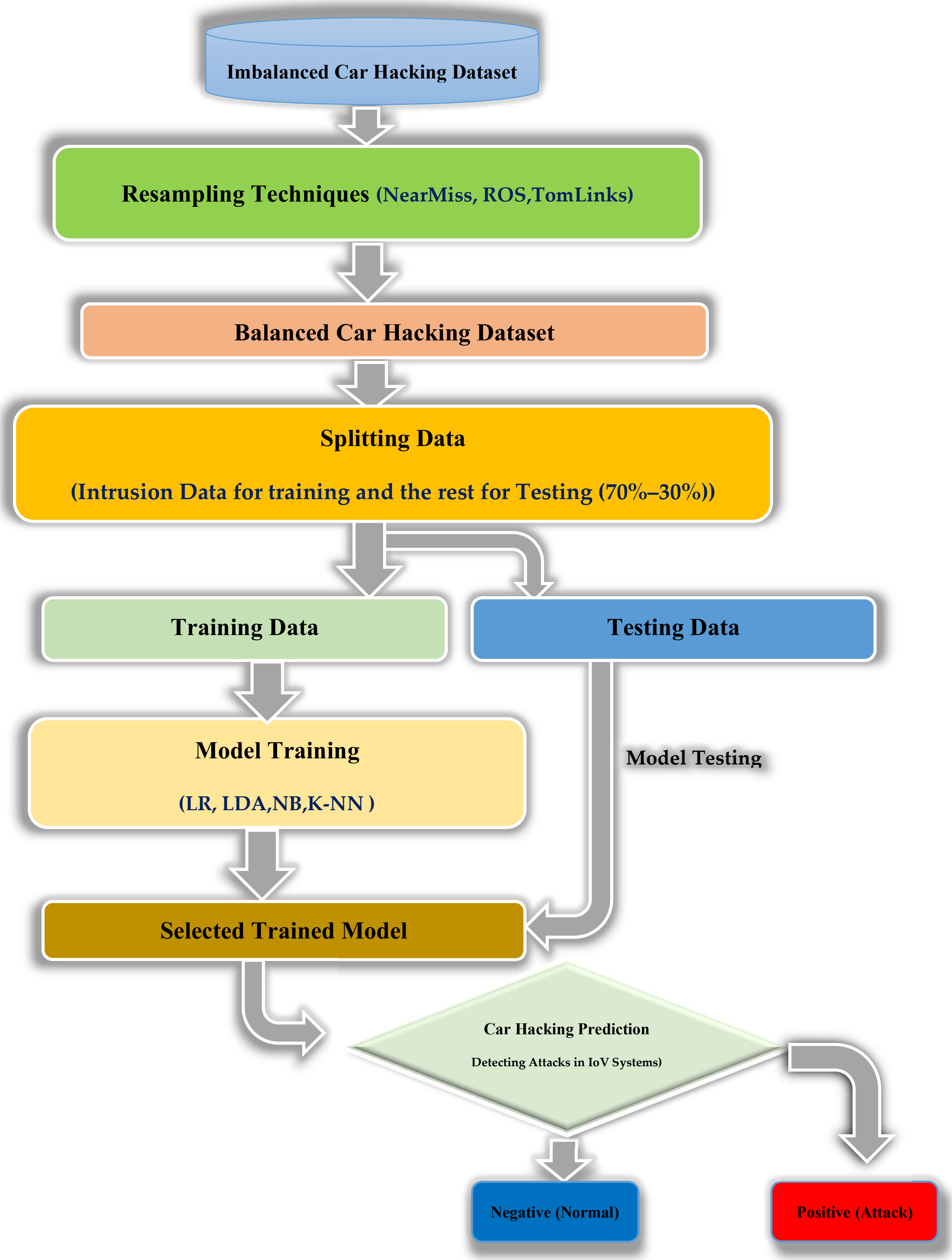
Imbalanced Car Hacking Dataset
Resampling Techniques (NearMiss, ROS,TomLinks)
Balanced Car Hacking Dataset
Splitting Data
(Intrusion Data for training and the rest for Testing (70%–30%))
Training Data
Testing Data
Model Training
(LR, LDA,NB,K-NN)
Model Testing
Selected Trained Model
Car Hacking Prediction
Detecting Attacks in IoV Systems)
Negative (Normal)
Positive (Attack)

Figure 5. Proposed framework for car hacking prediction on the IoVs systems.

The detailed pseudocode of the final proposed IDS Framework, where this framework is structured to follow the three specified phases, ensuring the intrusion data are properly handled, the model is trained and tested, and the results are evaluated effectively, can be as in Algorithm 1:

```
Algorithm 1. Car Hacking Detection Framework using ML
Input: Raw car hacking dataset (D) with features (X) and target (Y)
Output: Classify Car Hacking in IoVs
1. Import Libraries:
- Import necessary data manipulation libraries (e.g., pandas, numpy)
- Import resampling techniques (NearMiss, RandomOverSampler, TomekLinks)
2. Load Dataset: Load the raw car hacking dataset (D) into a DataFrame
3. Split Features and Target: Split (D) into features (X) and target (Y)
4. Apply Resampling Techniques: (NearMiss, RandomOverSampler, TomekLinks)
5. Split Dataset: Use 'train_test_split' to divide into a training set and a testing set. (Parameters:
test_size = 0.3, random_state = 42)
6. Train/Test Classifier: Fit the classifier on the training/testing data
7. Evaluate Model: Generate evaluation metrics
End of Algorithm
```

From the algorithm above, the data are pre-processed and then the resampling technique library is utilized for the class imbalance in the dataset. These techniques are used to balance the dataset. Then, the dataset is loaded into a DataFrame for processing and analysis. After that, the resampling technique is applied to handle the imbalanced data in the dataset. The machine learning is trained on the dataset, and then the assessment metrics are used to evaluate the proposed algorithm. This pipeline provides a comprehensive approach to handling a car hacking dataset, from initial data loading and pre-processing to model evaluation, ensuring that the model is trained on balanced data and its performance can be reliably assessed.

4. Suggested High-Level IDS-IoV Framework for Car Attack Prediction

The suggested framework for car attack prediction is built based on resampling algorithms and machine learning to generate possible attacks for real-time IoV purposes. The suggested frame employs a blend of Resampling Techniques (RT) and Machine Learning (ML) effectively. At this juncture, we offer a high-level IDS-IoV Framework for attack predictions.

The proposed methodology aims to help vehicle security engineers anticipate automotive assaults in real-world circumstances. This may be accomplished by using the Internet of Vehicles (IoV), Cloud Computing, and machine learning for real-time monitoring and safeguarding of vehicle connections; thus, this advice informs the effectiveness of protection. The proposed framework comprises three crucial stages that work together to achieve system objectives. Each stage provides a distinct goal and action that works in harmony with the others. The three phases of the recommended framework are shown in Figure 6 as follows:

- Stage 1: Vehicle systems, such as vehicle hacking systems, transmit data collection and acquisition to the cloud system for upcoming processing activities on cloud servers for the analytics process;
- Stage 2: After the data were gathered, they were sent to cloud servers for processing and storage so that they could be organized for the study of dangerous online activity data. From there, it was possible to forecast automobile or vehicle attacks, as shown more simply in Figure 3;
- Stage 3: To check for any threats in the context of a smart city, vehicle security experts employ a cloud-based dashboard support and monitoring system. The engineers will

have access to the cloud-based AI prediction system's information, and they will be able to make appropriate judgments about real-time vehicle attack scenarios.

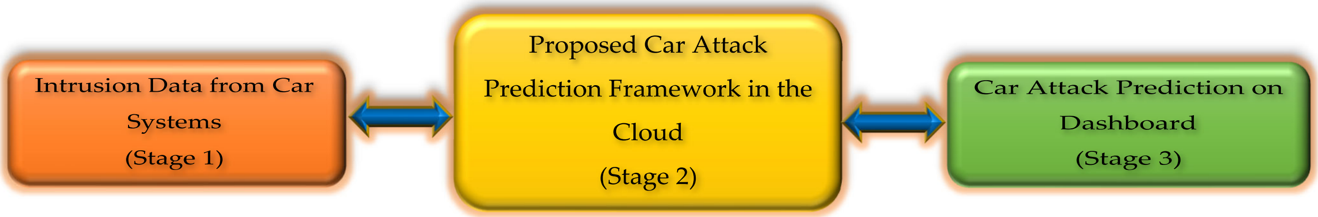
Intrusion Data from Car Systems (Stage 1)
Proposed Car Attack Prediction Framework in the Cloud (Stage 2)
Car Attack Prediction on Dashboard (Stage 3)

Figure 6. Stages of suggested car hacking detection over the cloud system.

From the above scenario, the designed method merges the power of cloud computing and AI to monitor and predict vehicle attacks in a smart city. By collecting data from various sources, processing them in the cloud, and using AI to identify patterns and potential threats, the system can provide real-time insights to security experts. This allows for quick and effective responses to potential threats, enhancing the overall security of the smart city.

5. Results Analysis and Assessment

The obtained results for applying machine learning methods for car hacking data. For each type of car attack. Determining if machine learning may be used as a discriminator of malicious car attacks is the aim of this work. It is possible to measure feasibility in terms of powerful recall and precision. It is capable of precisely categorizing both acceptable and fraudulent information with a low rate of misclassifications. To make sure that the comparatively high number of normal cases does not distort our overall performance findings, we compute precision, recall, and F1_Score for each class.

A training set (70%) and a testing set (30%) were created for the automobile hacking dataset. Both the training and test datasets underwent pre-processing and standardization. Several ML models, including LR, LDA, NB, and k-NN, were then trained using a resampled version of the training dataset. The ML models were evaluated using the test dataset. Different classifications were carried out for each dataset using the following resampling technique combinations, such as NearMiss, ROS, and TomLinks.

5.1. Experimental Results

The experimental results presented in Table 3 and Figure 7 provide insights into the performance of machine learning (ML) models when applied to the Car-Hacking Dataset without utilizing any resampling methods.

Table 3. ML only for the Car-Hacking Dataset without resampling techniques.

Algorithm	Accuracy	Precision	Recall	F1_Score
LR	97	97	90	93
LDA	88	81	63	67
NB	87	73	72	73
K-NN	100	100	100	100

Similarly, Table 4 and Figure 8 offer a comparative analysis of the results obtained without employing resampling techniques for both normal (R) and attack (T) instances. These initial evaluations serve as a baseline for understanding the effectiveness of the ML models in their raw form without any pre-processing techniques. In contrast, Tables 5–7 along with Figures 9–11, showcase the outcomes achieved by incorporating various resampling methods such as NearMiss, ROS, and Tom-Links, respectively. By integrating these resampling techniques with the ML models, a hybrid approach is formed, aiming to enhance the classification performance across all tested detection metrics.

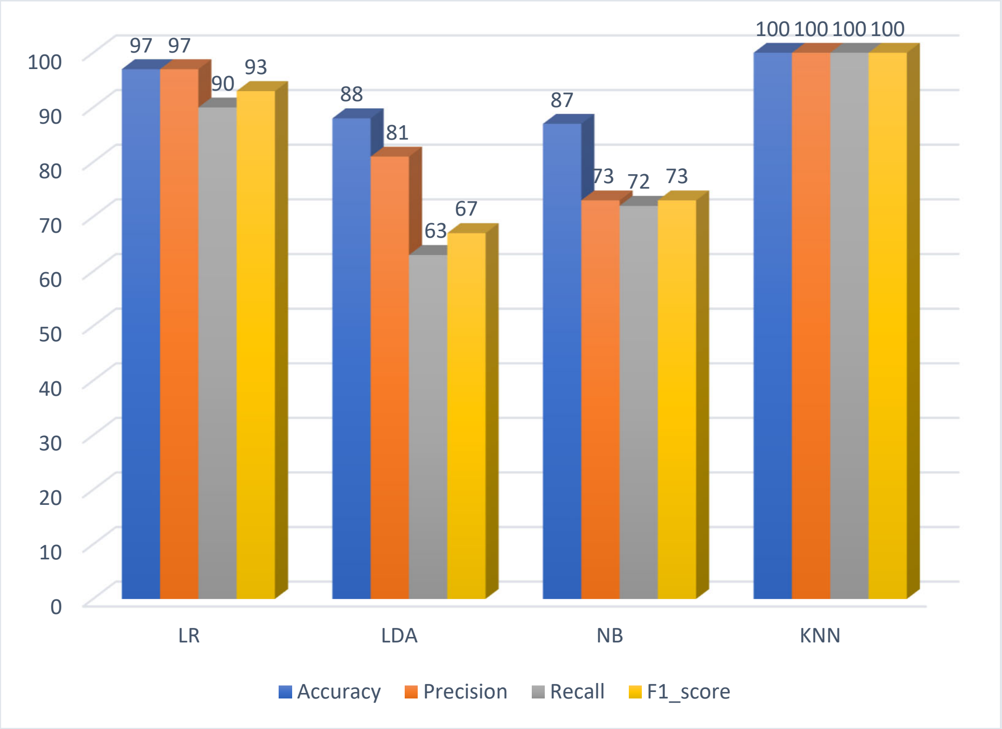

Figure 7. Results without resampling techniques.

Table 4. ML for the Car-Hacking Dataset for Normal (R) and Attack (T) without resampling techniques.

Algorithm	Class	Precision	Recall	F1_Score
LR	R	97	100	98
	T	97	81	89
LDA	R	89	98	93
	T	73	27	40
NB	R	92	93	92
	T	55	52	53
K-NN	R	100	100	100
	T	100	100	100

Table 5. NearMiss-ML for the Car-Hacking Dataset.

Algorithm	Accuracy	Precision	Recall	F1_Score
LR	88	89	88	88
LDA	84	84	84	84
NB	74	74	74	74
K-NN	90	92	90	90

Table 6. ROS-ML for Car-Hacking Dataset.

Algorithm	Accuracy	Precision	Recall	F1_Score
LR	83	83	83	83
LDA	83	83	83	83
NB	68	72	68	67
K-NN	100	100	100	100

Table 7. TomekLinks-ML for the Car-Hacking Dataset.

Algorithm	Accuracy	Precision	Recall	F1_Score
LR	97	97	90	93
LDA	88	81	63	67
NB	87	73	72	73
K-NN	100	100	100	100

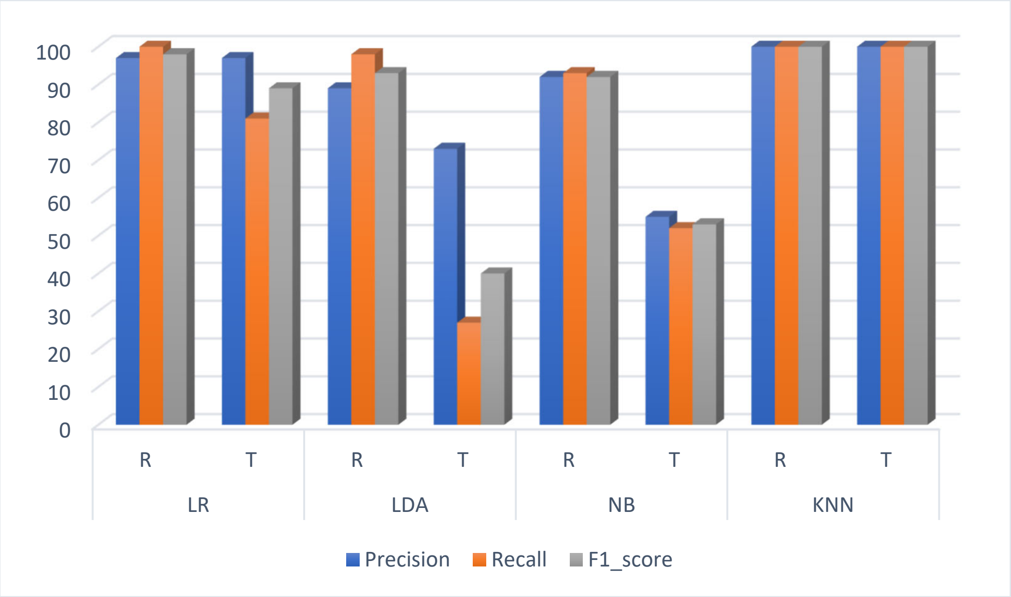

Figure 8. Results without resampling techniques for Normal (R) and Attack (T).

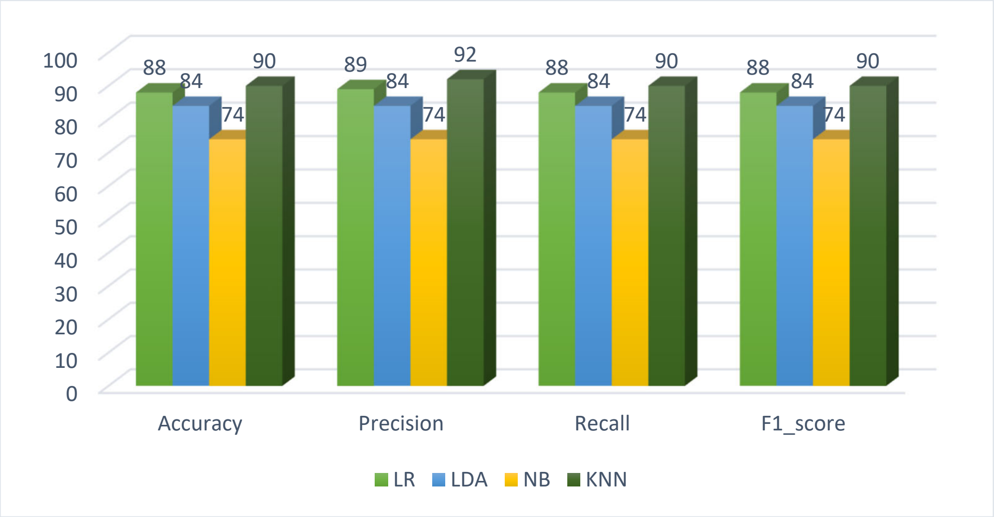

Figure 9. Results with the NearMiss technique.

Notably, the results highlight improvements in classification accuracy, precision, recall, and F1_Score, indicating the efficacy of leveraging resampling methods to mitigate imbalanced data challenges. From the comprehensive analysis of the experimental outcomes, it becomes evident that the hybridization of ML models with resampling techniques significantly enhances the effectiveness of car-attack detection systems. These findings underscore the importance of employing resampling methods, especially in scenarios involving imbalanced data, to ensure robust and efficient detection mechanisms in both intra-vehicle and external vehicular networks.

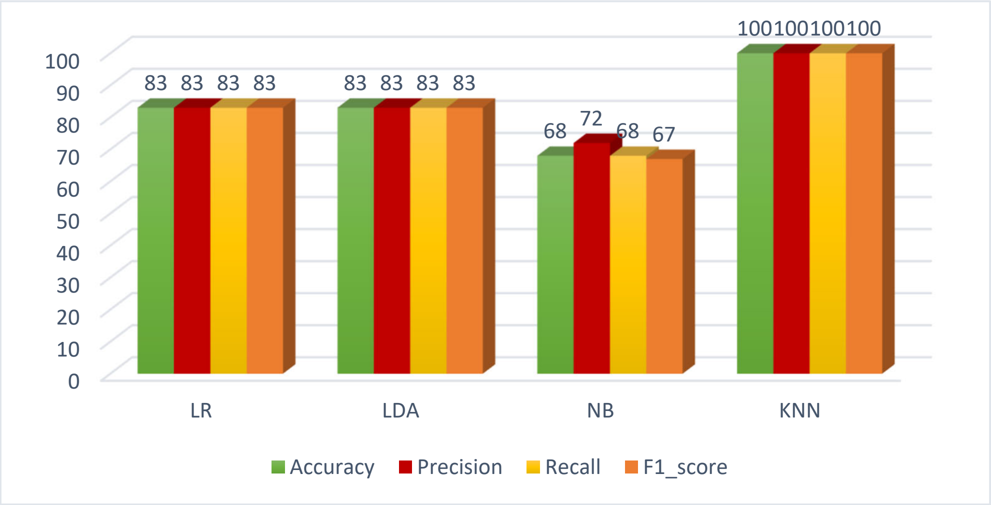

Figure 10. Results with the ROS technique.

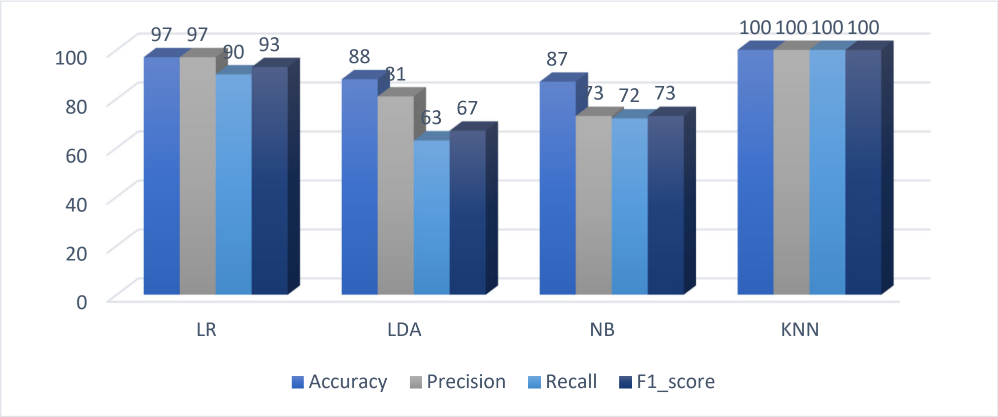

Figure 11. Results with the TomekLinks technique.

By harnessing the synergies between ML algorithms and resampling techniques, researchers and practitioners can develop more reliable and accurate intrusion detection systems tailored to address the unique challenges posed by imbalanced datasets. Additionally, Tables 8–10, alongside Figures 12–14, present the results obtained with employing various resampling techniques for normal (R) and attack (T) instances, providing further context for comparing the performance improvements achieved through the integration of resampling methods. These comparative analyses serve to reinforce the importance and effectiveness of incorporating resampling techniques in enhancing the classification performance of ML models in the context of car-attack detection.

Table 8. NearMiss-ML for the Car-Hacking Dataset for Normal (R) and Attack (T).

Algorithm	Class	Precision	Recall	F1_Score
LR	R	97	78	86
	T	82	97	89
LDA	R	88	78	83
	T	80	89	85
NB	R	100	80	89
	T	83	100	91
K-NN	R	100	80	89
	T	83	100	91

Table 9. ROS-ML for the Car-Hacking Dataset for Normal (R) and Attack (T).

Algorithm	Class	Precision	Recall	F1_Score
LR	R	85	81	83
	T	82	86	84
LDA	R	85	80	82
	T	81	86	83
NB	R	81	47	60
	T	63	89	74
K-NN	R	100	100	100
	T	100	100	100

Table 10. TomekLinks-ML for the Car-Hacking Dataset for Normal (R) and Attack (T).

Algorithm	Class	Precision	Recall	F1_Score
LR	R	97	100	98
	T	97	81	89
LDA	R	89	98	93
	T	73	27	40
NB	R	92	93	92
	T	55	52	53
K-NN	R	100	100	100
	T	100	100	100

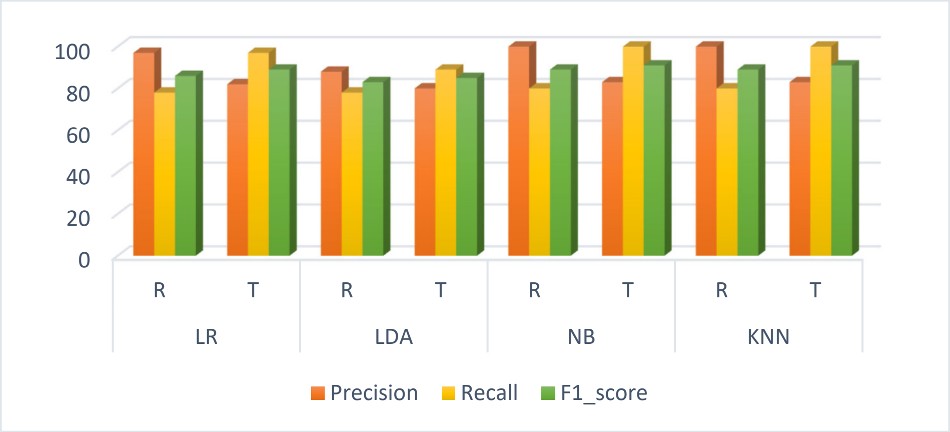

Figure 12. Results with the NearMiss technique for Normal (R) and Attack (T).

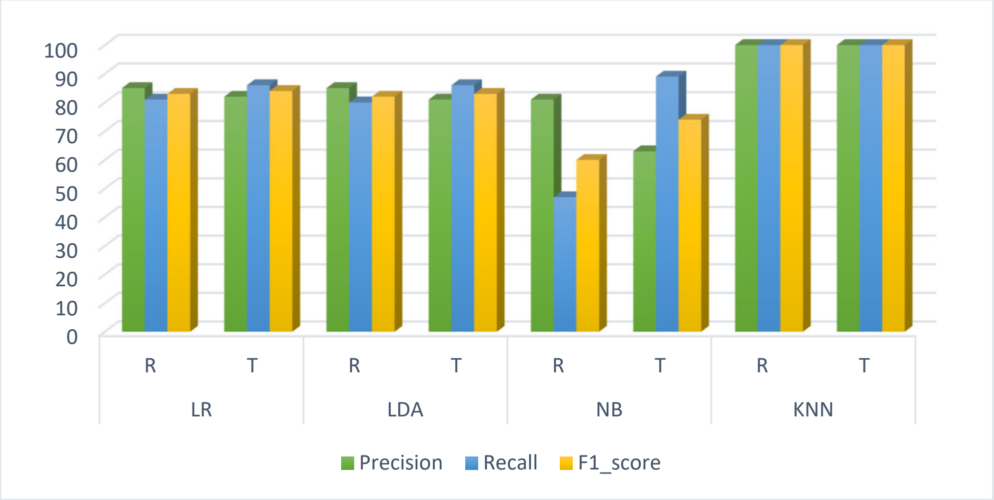

Figure 13. Results with the ROS technique for Normal (R) and Attack (T).

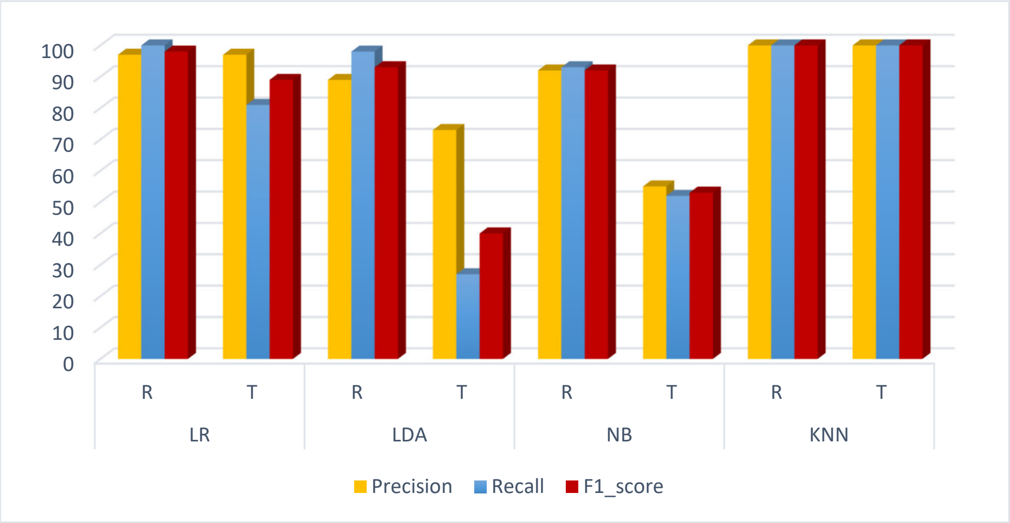

Figure 14. Results with the TomekLinks technique for Normal (R) and Attack (T).

Tables 11–14 and Figures 15–18 present a comprehensive comparison of the examined ML models when utilizing all tested resampling techniques, with a focus on accuracy, precision, recall, and F1_Score. The results unequivocally demonstrate that the strategic application of resampling techniques in conjunction with ML models significantly enhances hacking detection efficiency, yielding high accuracy, precision, recall, and F1_Score results. Notably, both the employed ML models and resampling methods exhibit efficient performance when dealing with the car hacking dataset, which is characterized by highly imbalanced data. The use of resampling techniques is particularly crucial in addressing class imbalances, a common challenge in machine learning. By employing techniques such as NearMiss, ROS, and Tome-Links, it is possible to create a more balanced training set, which can lead to improved model performance.

Table 11. Accuracy for ML models with different resampling techniques.

Algorithm	Without Resampling	NearMiss	ROS	TomekLinks
LR	97	88	83	97
LDA	88	84	83	88
NB	87	74	68	87
K-NN	100	90	100	100

Table 12. Precision for ML models with different resampling techniques.

Algorithm	Without Resampling	NearMiss	ROS	TomekLinks
LR	97	89	83	97
LDA	81	84	83	81
NB	73	74	72	73
K-NN	100	92	100	100

Table 13. Recall for ML models with different resampling techniques.

Algorithm	Without Resampling	NearMiss	ROS	TomekLinks
LR	90	88	83	90
LDA	63	84	83	63
NB	72	74	68	72
K-NN	100	90	100	100

Table 14. F1_Score for ML models with different resampling techniques.

Algorithm	Without Resampling	NearMiss	ROS	TomekLinks
LR	93	88	83	93
LDA	67	84	83	67
NB	73	74	67	73
K-NN	100	90	100	100

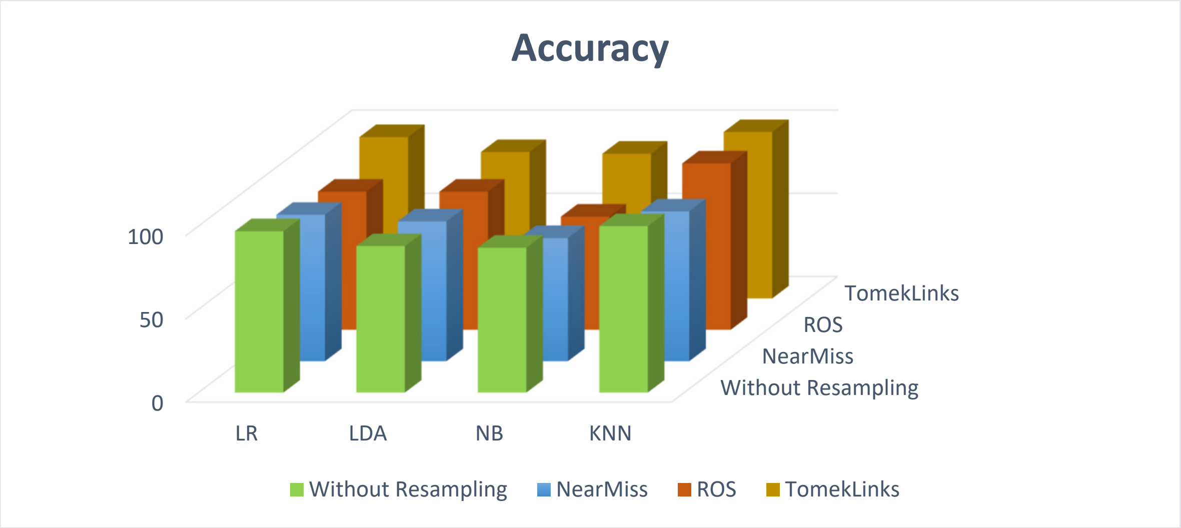

Figure 15. Accuracy with different resampling techniques.

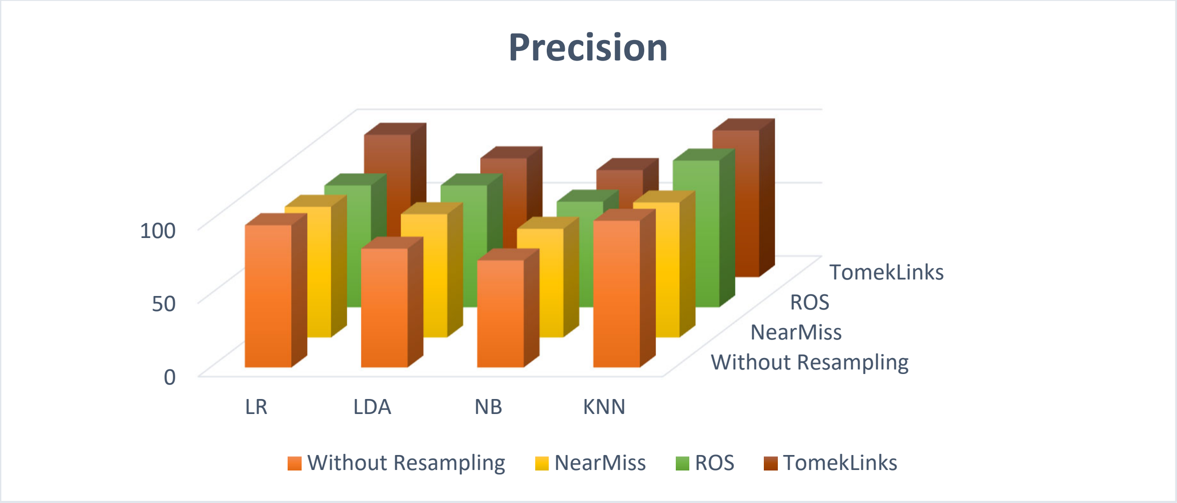

Figure 16. The precision with different resampling techniques.

Table 11 compares the performance of algorithms (LR, LDA, NB, K-NN) with different resampling techniques. Without resampling, LR performs best at 97, while K-NN maintains perfect performance across all techniques except NearMiss. Tomek Links generally maintain or slightly improve performance, especially for LR, LDA, and NB. NearMiss typically reduces performance across all algorithms, while ROS can significantly enhance K-NN but tends to lower performance for others.

Table 12 compares the performance of algorithms (LR, LDA, NB, K-NN) with different resampling techniques. Without resampling, LR performs best at 97, while K-NN maintains perfect performance across all techniques except NearMiss. Tomek Links generally help maintain performance, especially for LR and K-NN. NearMiss typically reduces performance for LR and K-NN but improves it for LDA and NB, while ROS significantly enhances K-NN but tends to lower performance for others.

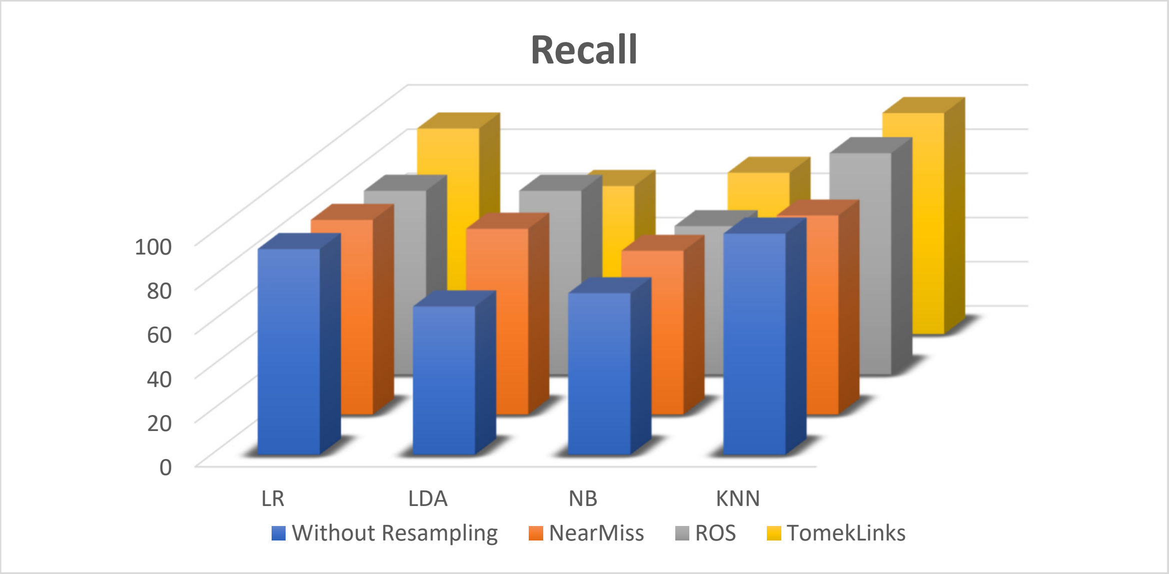

Figure 17. Recall with different resampling techniques.

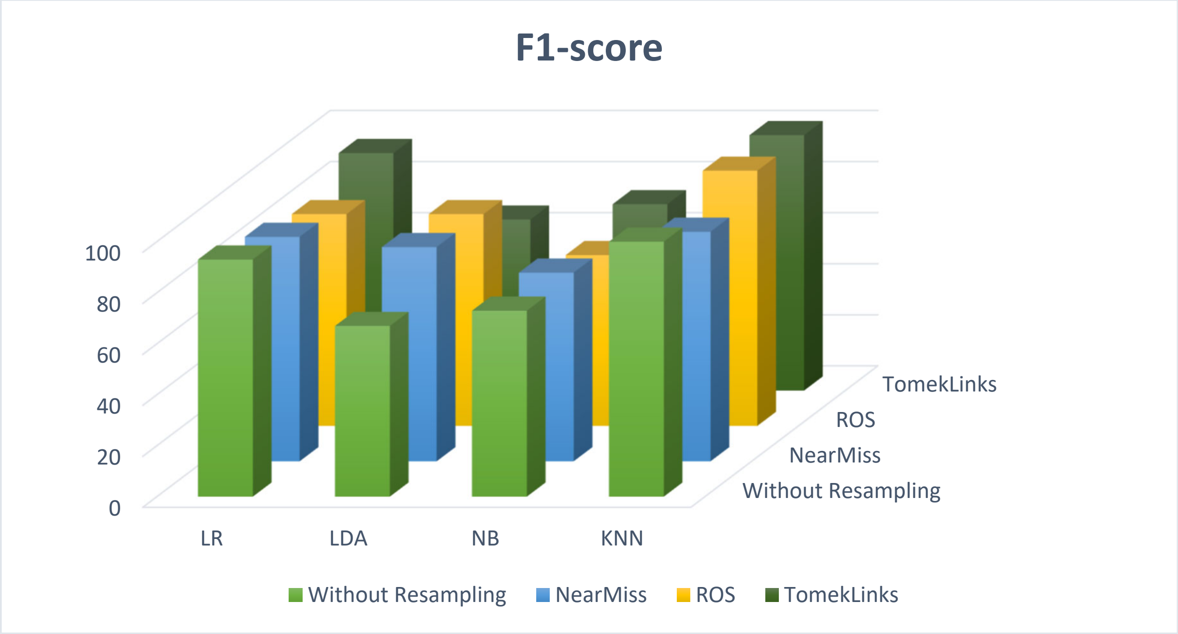

Figure 18. F1_Score with different resampling techniques.

Tables 13 and 14 compare the performance of algorithms (LR, LDA, NB, K-NN) under different resampling techniques. In Table 13, LR scores 90 without resampling, and K-NN achieves perfect scores except with NearMiss. Tomek Links maintains performance for LR, NB, and K-NN but not for LDA. NearMiss improves performance for LDA and NB but reduces it for LR and K-NN. ROS significantly boosts K-NN but lowers performance for the other algorithms. In Table 14, LR scores 93 without resampling, while K-NN maintains perfect scores except with NearMiss. Tomek Links maintains performance for LR, NB, and K-NN but not for LDA. NearMiss improves performance for LDA and NB but reduces it for LR and K-NN. ROS significantly enhances K-NN but decreases performance for LR, LDA, and NB.

5.2. Comparative Results

The quantitative comparison shown in Table 15 clearly shows that the suggested system performs noticeably better than the current systems. By significantly outperforming the previous systems in terms of accuracy, precision, recall, and F1_Score, the suggested method achieves an astounding 100% performance. From the results, the proposed system achieves an accuracy, precision, recall, and F1_Score of 100%, outperforming the state-of-the-art methods. In contrast, the system proposed in [8] achieves an accuracy of 98.10%, which is 1.9% lower than our system. Similarly, the systems proposed in [11,12] achieve accuracies of 99.96% and 99.93%, respectively, which are still 0.04% and 0.07% lower than our system, as clarified in Figure 19. These results indicate that our proposed system is more effective in achieving high accuracy and reliability.

Table 15. A comparative study of the proposed system with existing work.

Work	Accuracy	Precision	Recall	F1_Score
[8]	98.10	98.14	98.04	97.83
[11]	99.96	99.94	99.63	99.80
[12]	99.93	99.84	99.84	99.91
Proposed System	**100**	**100**	**100**	**100**

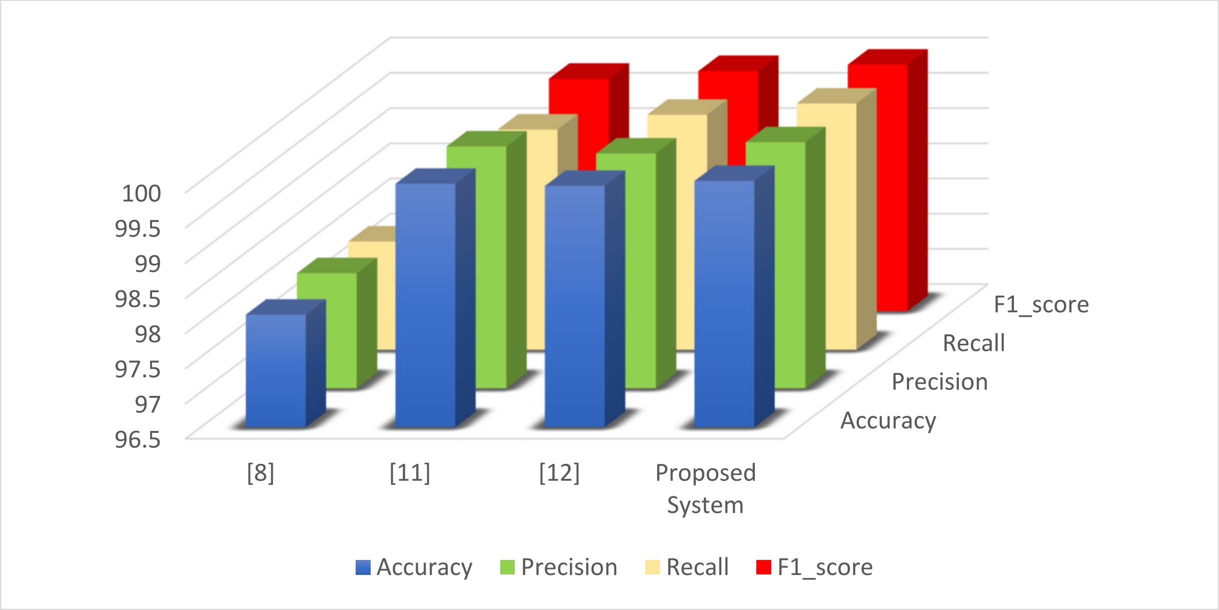

Figure 19. Comparison between the proposed system and current work.

The quantitative comparison also reveals that our system exhibits a notable enhancement in precision, recall, and F1_Score. Specifically, our system achieves a precision and recall of 100%, which is 1.86% and 1.96% higher than the system proposed in [8], respectively. Compared to the systems proposed in [11,12], our system shows an improvement of 0.06% and 0.16% in precision and 0.37% and 0.16% in recall, respectively. These results demonstrate that our proposed system can provide more accurate and reliable results, making it a superior choice for the detection of car hacking in IoV systems.

5.3. Limitations

While the current study shows promising results, there are several limitations and directions for future work to further enhance the security of vehicular networks and the efficacy of the proposed IDS. These include:

- Investigating the Application of Other Models: Explore the use of other conventional models, such as Random Forest (RF) and Support Vector Machine (SVM), as well as more sophisticated machine learning models like deep learning and ensemble learning techniques, to improve detection accuracy and adaptability to new types of attacks;
- Real-Time Detection Capabilities: Develop and implement real-time detection capabilities to ensure timely identification and mitigation of car hacking attempts, potentially integrating edge computing to reduce latency;
- Integration with Other Security Measures: Explore the integration of the IDS with other security measures, such as anomaly detection systems, blockchain for secure data sharing, and advanced encryption techniques to create a comprehensive security framework.

6. Conclusions and Future Scope

Recently, the realm of vehicle security has increasingly become a major concern due to the increased reliance on vehicle systems, especially with the advance of the Internet of Autonomous Vehicles (IoV) in the smart city. To detect car attacks, it is prudent that we build efficient car attack Detection Systems to efficiently analyze and detect car hacking data. There is an inherent problem with most car hacking data where the data are highly imbalanced; that is, there is a disproportionately large amount of good or normal traffic data and, in most cases, very few attack instances. Consequently, this study resolved the imbalanced car-hacking data problem using different handling resampling pre-processing procedures and then decided and predicted the car attack using machine learning models such as LR, LDA, NB, and k-NN. The experimental results proved that the performance of the proposed framework is assessed based on different metrics of accuracy, precision, and recall in comparing prevailing algorithms. Using k-NN with various Resampling Techniques, the suggested IDS has shown 100% detection rates in testing on the Car-Hacking dataset, which is a popular available benchmark IoV security dataset. It demonstrates the efficacy of the suggested IDS for cyberattack detection in external and intra-vehicle vehicular networks with unbalanced data on car hacking. In future work, several directions can be explored to further enhance the security of vehicular networks and the efficacy of the proposed IDS. These include the following:

- Investigate the application of other conventional models, such as RF and SVM, besides more sophisticated machine learning models, such as deep learning and ensemble learning techniques, to improve detection accuracy and adaptability to new types of attacks;
- Develop and implement real-time detection capabilities to ensure timely identification and mitigation of car hacking attempts, potentially integrating edge computing to reduce latency;
- Explore the integration of the IDS with other security measures, such as anomaly detection systems, blockchain for secure data sharing, and advanced encryption techniques to create a comprehensive security framework.

Author Contributions: Conceptualization and methodology, E.E.-D.H. and A.S.; software, validation and formal analysis E.E.-D.H.; validation; resources and data curation, A.S.; writing—original draft preparation, writing—review and editing, S.A., E.E.-D.H. and A.S.; visualization, E.E.-D.H.; supervision, E.E.-D.H., A.S. and S.A.; project administration and funding acquisition, S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R197), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Acknowledgments: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R197), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

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

References

1. Prakash, J.; Murali, L.; Manikandan, N.; Nagaprasad, N.; Ramaswamy, K. A vehicular network based intelligent transport system for smart cities using machine learning algorithms. *Sci. Rep.* **2024**, *14*, 468. [CrossRef] [PubMed]
2. Hemdan, E.E.D.; El-Shafai, W.; Sayed, A. CR19: A framework for preliminary detection of COVID-19 in cough audio signals using machine learning algorithms for automated medical diagnosis applications. *J. Ambient Intell. Humaniz. Comput.* **2022**, *14*, 11715–11727. [CrossRef] [PubMed]
3. Birchler, C.; Khatiri, S.; Bosshard, B.; Gambi, A.; Panichella, S. Machine learning-based test selection for simulation-based testing of self-driving cars software. *Empir. Softw. Eng.* **2023**, *28*, 71. [CrossRef]
4. Zayed, S.M.; Attiya, G.; El-Sayed, A.; Sayed, A.; Hemdan, E.E.D. An efficient fault diagnosis framework for digital twins using optimized machine learning models in smart industrial control systems. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 69. [CrossRef]
5. Pattnaik, D.; Ray, S.; Raman, R. Applications of artificial intelligence and machine learning in the financial services industry: A bibliometric review. *Heliyon* **2024**, *10*, E23492. [CrossRef]
6. Awotunde, J.B.; Folorunso, S.O.; Imoize, A.L.; Odunuga, J.O.; Lee, C.C.; Li, C.T.; Do, D.T. An ensemble tree-based model for intrusion detection in industrial internet of things networks. *Appl. Sci.* **2023**, *13*, 2479. [CrossRef]
7. Choudhary, V.; Tanwar, S.; Choudhury, T. Evaluation of contemporary intrusion detection systems for internet of things environment. *Multimed. Tools Appl.* **2024**, *83*, 7541–7581. [CrossRef]
8. Mehedi, S.T.; Anwar, A.; Rahman, Z.; Ahmed, K. Deep transfer learning based intrusion detection system for electric vehicular networks. *Sensors* **2021**, *21*, 4736. [CrossRef] [PubMed]
9. Yang, L.; Moubayed, A.; Shami, A. MTH-IDS: A multitiered hybrid intrusion detection system for internet of vehicles. *IEEE Internet Things J.* **2021**, *9*, 616–632. [CrossRef]
10. Yang, L.; Shami, A. A transfer learning and optimized CNN based intrusion detection system for Internet of Vehicles. In Proceedings of the ICC 2022-IEEE International Conference on Communications, Seoul, Republic of Korea, 16–20 May 2022; IEEE: New York, NY, USA, 2022; pp. 2774–2779.
11. Knoon, J. "Curbing Automotive Cybersecurity Attacks", Semiengineering, July 2023. Available online: https://semiengineering.com/curbing-automotive-cybersecurity-attacks/ (accessed on 2 April 2024).
12. Hossain, M.D.; Inoue, H.; Ochiai, H.; Fall, D.; Kadobayashi, Y. An effective in-vehicle CAN bus intrusion detection system using CNN deep learning approach. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; IEEE: New York, NY, USA, 2020; pp. 1–6.
13. Song, H.M.; Woo, J.; Kim, H.K. In-vehicle network intrusion detection using deep convolutional neural network. *Veh. Commun.* **2020**, *21*, 100198. [CrossRef]
14. Ding, W.; Alrashdi, I.; Hawash, H.; Abdel-Basset, M. DeepSecDrive: An explainable deep learning framework for real-time detection of cyberattack in in-vehicle networks. *Inf. Sci.* **2024**, *658*, 120057. [CrossRef]
15. Khan, I.A.; Moustafa, N.; Pi, D.; Haider, W.; Li, B.; Jolfaei, A. An enhanced multi-stage deep learning framework for detecting malicious activities from autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 25469–25478. [CrossRef]
16. Ashraf, J.; Bakhshi, A.D.; Moustafa, N.; Khurshid, H.; Javed, A.; Beheshti, A. Novel deep learning-enabled LSTM autoencoder architecture for discovering anomalous events from intelligent transportation systems. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 4507–4518. [CrossRef]
17. Kim, S.-K.; Jang, E.-S. The Intelligent Blockchain for the Protection of Smart Automobile Hacking. *J. Multimed. Inf. Syst.* **2022**, *9*, 33–42. [CrossRef]
18. Bhavsar, M.; Bekele, Y.; Roy, K.; Kelly, J.; Limbrick, D. FL-IDS: Federated Learning-Based Intrusion Detection System Using Edge Devices for Transportation IoT. *IEEE Access* **2024**, *12*, 52215–52226. [CrossRef]
19. Nabil, N.; Najib, N.; Abdellah, J. Leveraging Artificial Neural Networks and LightGBM for Enhanced Intrusion Detection in Automotive Systems. *Arab. J. Sci. Eng.* **2024**, 1–9. [CrossRef]
20. Trovão, J.P. Advancing Automotive Technologies [Automotive Electronics]. *IEEE Veh. Technol. Mag.* **2024**, *19*, 106-C3. [CrossRef]
21. Candelo, E.; Candelo, E. Innovation and digital transformation in the automotive industry. In *Marketing Innovations in the Automotive Industry: Meeting the Challenges of the Digital Age*; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 155–173.
22. Ahmad, E.; Iqbal, J.; Arshad Khan, M.; Liang, W.; Youn, I. Predictive control using active aerodynamic surfaces to improve ride quality of a vehicle. *Electronics* **2020**, *9*, 1463. [CrossRef]
23. Liang, J.; Li, Y.; Yin, G.; Xu, L.; Lu, Y.; Feng, J.; Shen, T.; Cai, G. A MAS-based hierarchical architecture for the cooperation control of connected and automated vehicles. *IEEE Trans. Veh. Technol.* **2022**, *72*, 1559–1573. [CrossRef]

24. Padmaja, B.; Moorthy, C.V.; Venkateswarulu, N.; Bala, M.M. Exploration of issues, challenges and latest developments in autonomous cars. *J. Big Data* **2023**, *10*, 61. [CrossRef]
25. Wiseman, Y. *"Autonomous Vehicles", Encyclopedia of Information Science and Technology*, 5th ed.; IGI Global: Hershey, PA, USA, 2020; Volume 1, Chapter 1; pp. 1–11.
26. Liang, J.; Tian, Q.; Feng, J.; Pi, D.; Yin, G. A polytopic model-based robust predictive control scheme for path tracking of autonomous vehicles. *IEEE Trans. Intell. Veh.* **2023**, *9*, 3928–3939. [CrossRef]
27. Alshathri, S.; Hemdan, E.E.D.; El-Shafai, W.; Sayed, A. Digital twin-based automated fault diagnosis in industrial IoT applications. *Comput. Mater. Contin.* **2023**, *75*, 183–196. [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.