information

Article

Smart Collaborative Intrusion Detection System for Securing Vehicular Networks Using Ensemble Machine Learning Model

Mostafa Mahmoud El-Gayar [1,2,*], Faheed A. F. Alrslani [3,*] and Shaker El-Sappagh [4,5]

[1] Department of Information Technology, Faculty of Computers and Information, Mansoura University, Mansoura 35516, Egypt
[2] Department of Computer Science, Arab East Colleges, Riyadh 11583, Saudi Arabia
[3] Department of Information Technology, Faculty of Computing and Information Technology, Northern Border University, Rafha 91911, Saudi Arabia
[4] Faculty of Computer Science and Engineering, Galala University, Suez 435611, Egypt; shaker@skku.edu
[5] Information Systems Department, Faculty of Computers and Artificial Intelligence, Benha University, Banha 13518, Egypt
* Correspondence: mostafa_elgayar@mans.edu.eg (M.M.E.-G.); f.alrslani@nbu.edu.sa (F.A.F.A.)

Abstract: The advent of the Fourth Industrial Revolution has positioned the Internet of Things as a pivotal force in intelligent vehicles. With the source of vehicle-to-everything (V2X), Internet of Things (IoT) networks, and inter-vehicle communication, intelligent connected vehicles are at the forefront of this transformation, leading to complex vehicular networks that are crucial yet susceptible to cyber threats. The complexity and openness of these networks expose them to a plethora of cyber-attacks, from passive eavesdropping to active disruptions like Denial of Service and Sybil attacks. These not only compromise the safety and efficiency of vehicular networks but also pose a significant risk to the stability and resilience of the Internet of Vehicles. Addressing these vulnerabilities, this paper proposes a Dynamic Forest-Structured Ensemble Network (DFSENet) specifically tailored for the Internet of Vehicles (IoV). By leveraging data-balancing techniques and dimensionality reduction, the DFSENet model is designed to detect a wide range of cyber threats effectively. The proposed model demonstrates high efficacy, with an accuracy of 99.2% on the CICIDS dataset and 98% on the car-hacking dataset. The precision, recall, and f-measure metrics stand at 95.6%, 98.8%, and 96.9%, respectively, establishing the DFSENet model as a robust solution for securing the IoV against cyber-attacks.

Keywords: Internet of Vehicles; intrusion detection system; Dynamic Forest-Structured Ensemble Network; cybersecurity

check for updates

Citation: El-Gayar, M.M.; Alrslani, F.A.F.; El-Sappagh, S. Smart Collaborative Intrusion Detection System for Securing Vehicular Networks Using Ensemble Machine Learning Model. *Information* **2024**, *15*, 583. https://doi.org/10.3390/info15100583

Academic Editor: Jian Tang

Received: 8 August 2024
Revised: 13 September 2024
Accepted: 18 September 2024
Published: 24 September 2024

1. Introduction

Industry 4.0 has propelled the importance of the Internet of Things (IoT) to the forefront of technological innovation. This innovation is especially evident in the sectors of intelligent vehicles (IVs), where the fusion of vehicle-to-everything (V2X) technology, IoT networks, and inter-vehicle networks has brought forth intelligent connected vehicles (ICVs). As a result, a comprehensive and intricate network for vehicle communication has been established. In recent years, the proliferation of Internet of Vehicles (IoV) systems has exposed significant vulnerabilities, necessitating robust intrusion detection systems (IDSs) tailored for this complex environment. Traditional IDS solutions, while foundational, often struggle with high false positive rates, limited scalability, and suboptimal performance against sophisticated or novel attacks. Recognizing these challenges, this study introduces the Dynamic Forest-Structured Ensemble Network (DFSENet), a novel approach designed to transcend the limitations of conventional methods. DFSENet aims to enhance detection accuracy while minimizing false negatives significantly—a critical issue in current IDS frameworks. By integrating a state-of-the-art deep learning architecture with the Synthetic Minority

Over-sampling Technique (SMOTE) to address data imbalances, DFSENet is engineered to improve the precision of threat detection across varying attack vectors, thereby reducing the typical high false positive rates associated with existing systems. This manuscript will detail the innovative aspects of DFSENet, emphasizing its capability to adapt and respond to the evolving landscape of cyber threats in the IoV. Additionally, the study is motivated by the increasing frequency and complexity of cyber-attacks in IoV environments, underlining the urgent need for more sophisticated and dynamic IDS solutions. By bridging the identified research gap through advanced machine learning techniques and a refined data-handling strategy, DFSENet stands as a pivotal advancement in IDS technology, promising substantial improvements in system reliability and security efficacy. As the Internet of Vehicles (IoV) continues to evolve, it increasingly becomes a target of various cyber threats. These vulnerabilities expose IoV systems to various cyber-attacks, each carrying potentially severe consequences such as data breaches, operational disruptions, and compromised safety. For instance, a typical IoV threat could involve unauthorized access to vehicle communication systems, leading to unauthorized control over vehicle functionalities. Such real-world implications underscore the critical need for robust security measures within these networks. Intrusion detection systems (IDSs) serve as an essential line of defense in response to these escalating security challenges. IDSs are pivotal in identifying and mitigating threats, thereby preserving the integrity and reliability of IoV networks. Their role extends beyond mere detection; they provide the necessary frameworks to initiate prompt responses to security breaches, which is crucial for maintaining the operational efficacy and safety of IoV systems. This study is driven by the need to address the limitations observed in current IDS implementations, particularly their struggle with high false positive rates and inadequate scalability. Our primary objective is to introduce the DFSENet, specifically designed to enhance the detection capabilities of IDSs within IoV environments. DFSENet aims to improve detection accuracy, reduce false negatives, and demonstrate superior scalability and adaptability in facing new and evolving cyber threats. However, this network's complexity also exposes it to potential cyber-attacks, presenting a significant challenge to the Internet of Vehicles (IoV)'s stability and resilience. As integral components of intelligent transportation systems, vehicular networks are increasingly vulnerable to a wide range of cyber-attacks due to their open and distributed infrastructure. Attacks range from passive spying to more active disruptions such as Denial of Service (DoS) attacks, Sybil attacks, and false data dissemination. In DoS attacks, excessive requests overwhelm the network, delaying or neglecting legitimate requests [1–6]. Sybil attacks involve a malicious node impersonating multiple nodes, undermining network functionality and trust. False data propagation spreads inaccurate or deceptive information across the network, potentially causing traffic issues, accidents or life-endangering situations. These cyber threats pose substantial risks to vehicular network safety, efficiency and reliability, highlighting the urgent need for robust and effective security protocols. The potential damage from these cyber-attacks is substantial, ranging from immobilizing vehicles to causing severe accidents. The integration of the IoT amplifies these risks by introducing vulnerabilities to internet-based attacks and malware, which could lead to remote control or the destruction of vehicle systems. This is not just a theoretical risk; real-world incidents, such as the Jeep Cherokee and Tesla hackings, have demonstrated the potential for severe consequences like the loss of life, energy, and significant financial losses. The stability of IVs is also threatened by traditional network attacks such as eavesdropping and sniffing. Attacks on the IoV fall into two principal classifications based on the attacker's objective: attacks between vehicles (Inter-vehicle) and those within a single vehicle (Intra-vehicle) [7–9]. This underscores the urgent need for robust intrusion detection systems (IDSs), which play an essential role in enhancing the cybersecurity of the IoV. However, the limited occurrence of attack instances in typical network traffic data often results in class imbalance, which can adversely impact IDS performance. To manage system and maintenance costs, modern vehicular networks have turned to wireless protocols for data transmission. Vehicular networks are an emerging research field in the realm of smart cities, where vehicles function as nodes in a network, facilitating communication between each other. These networks enhance the quality of life, security, and safety, making them crucial to the evolution of smart cities. Vehicular networks strive to provide advanced communication technologies for vehicles, fostering cooperative relationships through information sharing. However, the spatiotemporal challenges associated with vehicular networks can affect their efficiency in traffic management systems, necessitating effective data analysis [10–18]. This paper introduces an IDS for the IoV based on a DFSENet model. This enables precise multiclass classification and the effective detection of various cyber threats within vehicle communication networks. The paper's main contributions include the following:

- The paper examines different data-balancing methods and their effects on IDS efficacy. A combined mechanism of the SMOTE and random undersampling is utilized to address class imbalance issues and attain balanced class distribution. This approach's efficacy is demonstrated using the CICIDS2017 (CICIDS) dataset. Additionally, Principal Component Analysis (PCA) is employed to reduce feature dimensionality, substantially lessening computational demands.
- The paper presents a DFSENet as the core of the IDS. This network effectively classifies network traffic data from In-Vehicle Networks (IVNs) and external sources. The deep-layered IDS model stacks various machine learning (ML) models sequentially layer by layer, connecting them in an ordered manner. This architecture enhances the ability to precisely and efficiently detect a spectrum of cyber-attacks, safeguarding both IoV systems and intelligent connected vehicles (ICVs) from diverse cyber threats.
- A CIDS architecture is proposed, based on machine learning, that enables information exchange and knowledge sharing within vehicular networks.
- A design principle is also presented to determine the optimal privacy parameter value. This is attained by solving an optimization problem that balances the tradeoff between security for the vehicular network and protecting its privacy.
- The proposed IDS's performance was evaluated using two datasets—the widely accepted CICIDS2017, known as CICIDS, for network intrusion detection, and the car-hacking dataset pertinent to IoV security.

The structure of this manuscript is as follows: Section 2 presents a comprehensive review of existing studies in the realm of intrusion detection for the IoV. Section 3 delves into the research approach employed in our investigation, elaborating on the architecture of the system, the preprocessing of data, and the construction of the intrusion detection model that constitutes the proposed effective IDS. Section 4 presents a detailed evaluation and discussion of the experimental outcomes. Lastly, Section 5 wraps up the paper by succinctly summarizing the key findings from our research and proposing directions for further study.

2. Literature Review

The unique characteristics of the IoV, such as high mobility and predictable node movements, contribute to its security challenges. Due to life-critical safety implications, protection is paramount. A privacy protection mechanism is needed to ensure information sharing and encourage nodes to collaborate. Therefore, a mechanism ensuring privacy protection is critical for safeguarding training data privacy across the network and enabling an efficient Collaborative Intrusion Detection System (CIDS). Differential privacy, as proposed in [9], is a well-established concept able to furnish robust privacy assurance by guaranteeing any single entry change in the dataset produces only a minor alteration to the response distribution of the dataset. Previous work explored IoV threats and demonstrated potential attacks on intelligent vehicle systems like braking interference [19]. In 2013, researchers began analyzing vulnerabilities and attacks on intelligent vehicles, conducting risk analyses [1,3–5]. Subsequent experiments tested attacks and suggested solutions often involving encryption [20,21]. IoV cybersecurity has grown to be increasingly important. As IoV adoption increases along with intelligent vehicles (IVs), so too do communication nodes and the potential exploitation of new security issues. Effective and precise intrusion detection is essential. ML techniques have often developed mature, effective unauthorized access detection through misuse and anomaly detection using decision trees [22], a fog-based DT IDS [23], and CSV-ISVM for training data enlargement [24]. Other research combined frequented Random Forests for network intrusion [25,26]. Significant attention addressed distributed ML-based IDSs utilizing fog models [27–30] and federated learning distributed architectures maintaining user privacy [22–24].

Deep learning (DL), an ML subset, has outperformed traditional ML techniques in NLP and CV in recent years [31]. For example, CNN intrusion detection requires converting network traffic to a matrix for two-dimensional processing [32,33]. Notably, LSTM was used as an FCN sub-module for time series classification, introducing an attention mechanism that improved intrusion detection performance. Others combined DL techniques and SVMs, integrating 1D CAE and OCSVM into a one-stage model demonstrating superior detection and generalization [34]. Another approach used k-means, a DNN and an SVM in two stages, k-means for anomaly detection followed by a DNN and a SVM for intrusion detection, showing effectiveness though some generalization limitations [35]. While the above IDS addressed general networks, research increasingly focuses on IoV-specific IDSs. One study proposed a deep CNN IDS optimizing CAN bus data, achieving excellence on vehicle datasets. Another designed an LSTM auto-encoder IDS detecting network traffic anomalies, providing crucial ITS security. Other work simulated VANET attack scenarios, collecting and analyzing traffic data statistically, though it was less accurate for multiple events [36]. Additionally, neural networks identified denial-of-service attacks on autonomous vehicles communicating via VANETs [37]. Feature dimensionality reduction using PCA constructed low-parameter cyber-attack classifiers on traffic data's multidimensionality complexities [38]. Lastly, ensemble approaches proved effective for imbalanced traffic data classification, with some studies achieving exceptional detection through ensemble technology [39,40]. Studies [22–24,31–35,41,42] design general network IDSs, while [27–30,43] emphasize distributed IDS deployment in the IoT/IoV. Refs. [36–40] primarily focus on IVN cybersecurity. However, current IDSs are commonly limited to binary classification, which is insufficient for multiclass tasks. Additionally, network traffic data's severe imbalance severely challenges multiclass classification. Albers et al. proposed a collaborative IDS with a local IDS on each mobile network node for local security, extendable to address global security via collaboration [44]. Sterne et al. designed a hierarchical IDS using multilevel clustering [45]. Machine learning and data mining have also been explored for IDSs. These techniques enable continual learning, enhancing security knowledge, linking suspicious events, and predicting attacks. Unsupervised methods like clustering categorized normal and anomalous packets [46] using hierarchical [47], K-means [48] clustering. Supervised learning including SVMs detected anomalies using new window kernels [2]. Differential privacy research focuses on machine learning applications [33,34], balancing privacy and performance [22,35]. Interest grows in distributed differential privacy with automated verification frameworks [36] and differentially private constrained optimization algorithms [37], using clouds for differentially private computations [38]. Hikal et al. [49] introduced a lightweight, machine learning-based IDS for the IoT. Employing an ensemble data preprocessing stage to enhance feature selection, the system achieves up to 99.7% detection accuracy and a 30–80 s detection time in combating IoT botnet attacks. Mohammed et al. [50] explored hybrid metaheuristic optimization algorithms, specifically gray wolf optimization and the salp swarm algorithm, to efficiently solve IoT sensors' localization and privacy problems in wireless networks with power and processing capability constraints. Fetooh et al. [51] proposed a novel admin-side method for detecting fake Wi-Fi access points capable of identifying multiple cyber-attacks, including various forms of WI-phishing and the DE-authentication attack. By analyzing frame types and static and dynamic parameters in real-time, the system distinguishes between normal and malicious packets, achieving an average accuracy of 94.40%, a precision of 87.08%, and a specificity of 96.39% across five types of attacks. Zhang et al. [52] leverages XGBoost (Version 2.1.1.) and RF ensemble learning methods, enhanced by Bayesian optimization, to predict the undrained shear strength

(USS) of soft clays from the TC304 database using five soil parameters. Demonstrating superior performance under 5-fold CV, the XGBoost-based model also offers valuable insights into feature importance, thus improving predictability and interpretability in geotechnical parameter estimation. Zhang et al. [53] developed a time-variant reliability analysis method for landslide prevention in the Three Gorges Reservoir Area. The method utilizes XGBoost and LightGBM machine learning algorithms to evaluate landslide failure probabilities efficiently and accurately under varying environmental conditions. Our proposed model advances intrusion detection research by addressing limitations found in prior studies, such as poor generalizability, inefficient high-dimensional data management, and overfitting. By employing a dynamic-depth, tree-based network with a diverse ensemble of base estimators, it enhances accuracy and robustness. Strategic data preprocessing and feature selection reduce dimensionality, cutting down on the computational load. Unlike other models, ours handles imbalanced datasets effectively without requiring data balancing, showcasing its capacity for real-world applicability and superior performance in identifying network threats. Table 1 shows the comparison between the existing methods in the related works.

Table 1. Comparison between existing methods.

	Precision	Recall	F1-Score	Accuracy	# Categories
LSTM	0.954	0.895	0.885	0.893	2
MLP	0.882	0.859	0.868	0.872	2
1D-CNN	0.964	0.906	0.935	0.938	2
DBN	0.897	0.975	0.943	0.946	6

3. Proposed CIDS

In order to guarantee the accurate, streamlined, and detailed categorization of network traffic within vehicular communication systems, thus protecting intelligent connected vehicles (ICVs) in the IoV against cyber threats emanating from external networks and Intra-Vehicle Networks (IVNs), we introduce a potent IDS founded on the principles of DFSENet. Figure 1 presents a detailed breakdown of the IDS workflow. The five IDS components depicted in Figure 1 are (1) intra-vehicle network data collection, (2) data processing, (3) the local IDS engine, (4) detection result output, and (5) privacy preservation. Our aim is to elevate ICV security through implementing a robust IDS efficiently identifying and neutralizing potential threats.

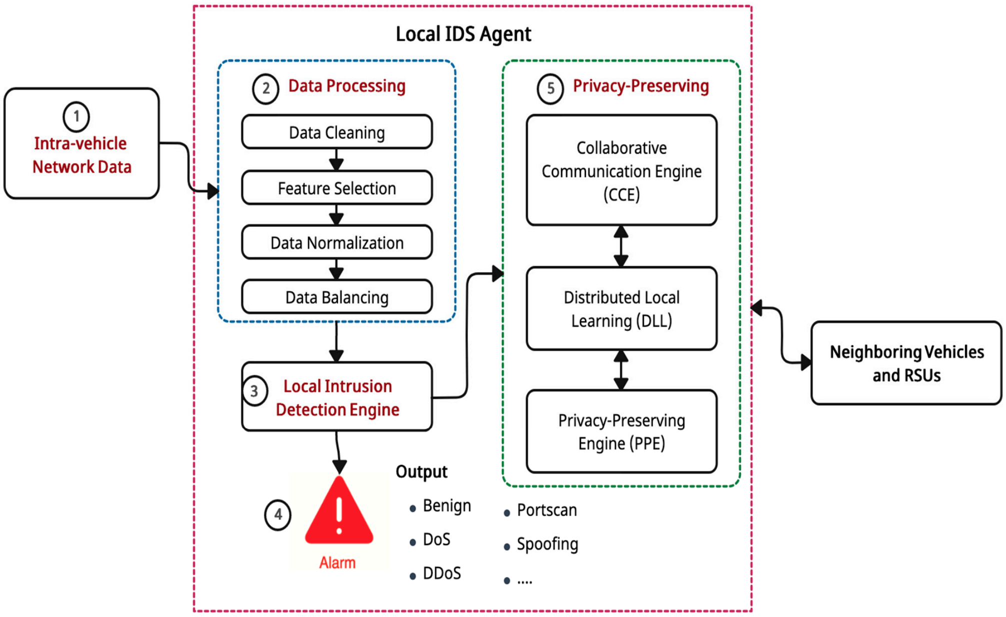

Local IDS Agent
1
Intra-vehicle Network Data
2
Data Processing
Data Cleaning
Feature Selection
Data Normalization
Data Balancing
3
Local Intrusion Detection Engine
4
Alarm
Output
• Benign
• DoS
• DDoS
• Portscan
• Spoofing
• ....
5
Privacy-Preserving
Collaborative Communication Engine (CCE)
Distributed Local Learning (DLL)
Privacy-Preserving Engine (PPE)
Neighboring Vehicles and RSUs

Figure 1. Proposed CIDSs.

3.1. Data Processing

3.1.1. Data Collection

For the training and evaluation of intrusion detection models within any IDS, a substantial volume of data samples is crucial. Adequate records of network traffic, encompassing both normal and malevolent scenarios, are vital for the development of such models. In the case of the IDS we have proposed, the gathered data are segmented into a pair of collections: (1) the training dataset, which is utilized for the purpose of model training, and (2) the testing dataset, which serves to assess the efficacy of the model.

3.1.2. Data Cleaning

To rigorously assess IDS effectiveness, numerous network intrusion detection datasets are available for related studies, including well-recognized datasets like KDD99, NSL-KDD [42], UNSWNB15 [43], and a CIDS [3]. Concurrently, collective research efforts have created dedicated car-hacking datasets furthering automotive safety within the IoV [4]. However, a sequence of data purification procedures is required to optimize these data for model training and evaluation, for example, scrutinizing datasets for missing values, constant values, and other outliers incompatible with model learning. These irregularities require either purification or transformation into useful data. Algorithm 1: Enhanced Real-time VANET Surveillance delineates the structured process we developed to monitor and detect intrusions effectively. Initially, the algorithm preprocesses the incoming VANET system data stream (X), a critical step for ensuring the integrity and usability of data. Subsequent steps involve assessing the need to update the intrusion detection system (IDS) classifier based on the preprocessed data. If an update is required, the IDS classifier is refreshed using a locally stored dataset before proceeding with intrusion detection. The final step involves continuous monitoring where, whether updated or not, the classifier is used to detect potential intrusions and trigger alerts (Y) if anomalies are detected. This algorithm is pivotal for maintaining robust surveillance within VANETs and enhancing security measures in real-time operational environments.

Algorithm 1: Enhanced Real-time VANET Surveillance

INPUT: Real-time VANET System Data Stream (X)
OUTPUT: Intrusion Detection Alerts (Y)

```
START PROCEDURE
//Step 1: Data Preprocessing Stage
Preprocessed_Data = Processing_data(X)

//Check if classifier update is needed
IF NEED_UPDATE(Preprocessed_Data) THEN

//Step 2: Classifier Update Mechanism
Updated_Classifier <- IDS (Load_local_dataset())
Classifier = Updated_Classifier

//Step 3: Local IDS with updated classifier
Alerts = Local_Detection (Preprocessed_Data, Classifier)
    IF Alerts CONTAIN Intrusions THEN
        TRIGGER_ALERT(Y)
     END IF
     ELSE
          //Step 4: Continuous Monitoring with the current
             classifier
          Alerts = Local_Detection (Preprocessed_Data,
                             Classifier)
          IF Alerts CONTAIN Intrusions THEN
               TRIGGER_ALERT(Y)
          END IF
    END IF
END PROCEDURE
```

3.1.3. Feature Selection

Datasets designed for intrusion detection typically comprise a wide array of standard network attributes. Take, for instance, the CICIDS dataset, which is characterized by 78 distinct network attribute features. When datasets possess such a multitude of features, they are often referred to as high-dimensional data. To manage this complexity, feature selection methods are applied to retain crucial attributes while discarding those that are redundant or irrelevant. This technique helps in reducing data duplication and lessens the computational load. In our initial analysis, we evaluated the interrelationships between the 78 features of the CICIDS dataset using specific equations. From the resulting correlation coefficient matrix, we identified and removed 27 features that exhibited high correlation. The progression of these steps is depicted in Figure 2 through heatmaps, which illustrate the correlation between features in the CICIDS dataset at various stages of processing. The correlation coefficient, denoted as $\rho_{u,v}$, measures the linear relationship between two variables u and v. It is defined as follows:

$$\rho_{u,v} = \frac{cov(u,v)}{\sigma_U \sigma_V} \tag{1}$$

where $cov(u, v)$ represents the covariance between the variables. This coefficient is crucial for identifying features in the IoV data that move together, which may indicate underlying patterns or influential relationships.

$$cov(u,v) = \frac{1}{M-1}\sum_{i=1}^{M}(u_i - \overline{u})(v_i - \overline{v}) \tag{2}$$

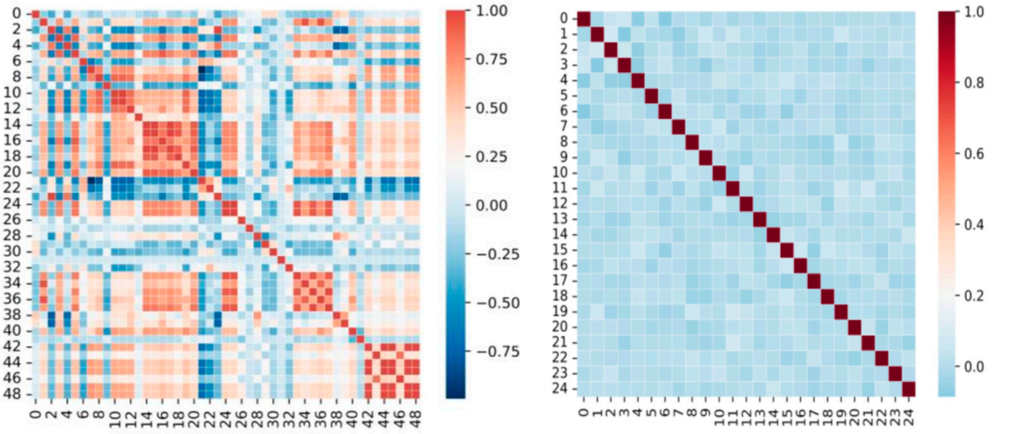
$$\sigma_u = \sqrt{M\sum {u_i}^2 - \left(\sum u_i\right)^2} \tag{3}$$

$$\sigma_v = \sqrt{M\sum {v_i}^2 - \left(\sum v_i\right)^2} \tag{4}$$

$$\overline{u} = \frac{1}{M}\sum_{i=1}^{M} u_i \tag{5}$$

$$\overline{v} = \frac{1}{M}\sum_{i=1}^{M} v_i \tag{6}$$

where u and v represent sample sets, $\overline{u}$ signifies the mean of u, $\overline{v}$ denotes the mean of v, the term σ is used to represent variance, while $cov(u, v)$ indicates the covariance of u and v.



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

Figure 2. Heatmaps Depicting the Feature Correlation in CICIDS data at various processing stages: (**a**) the heatmap demonstrating data feature correlations following feature selection; (**b**) the heatmap showing data feature correlations following the implementation of PCA.

In Figure 2a—After Feature Selection, this heatmap shows the correlation between different network traffic features after a feature selection process has been applied. Feature selection aims to reduce the number of input variables for modeling, focusing on those most relevant to detecting intrusions. This map helps identify which features retain strong associations with each other and are potentially significant for the subsequent analytical steps. In Figure 2b—After Implementing PCA (Principal Component Analysis), the second heatmap depicts the correlation between features after the transformation by PCA, a statistical technique used to emphasize variation and bring out strong patterns in a dataset. PCA reduces the dimensionality of the data by transforming the original variables into a new set of variables (principal components), which are uncorrelated and ordered so that the first few retain most of the variation present in all of the original variables.

3.1.4. Data Normalization

Normalizing data with many dimensions is an essential part of the data preprocessing phase. Unstandardized data in such high-dimensional spaces can lead to heavier computational burdens on machine learning algorithms, thereby hampering the speed and efficacy of their training processes. In this research, we have implemented a quantile transformation to achieve data normalization. This method uses non-linear transformations to render the data robust to outlier influences. Fundamentally, it calculates a mapping function that aligns well with the input variables, effectively converting the values to a uniform distribution ranging from 0 to 1. These values are then processed through specific quantile functions to conform to the desired distribution shape. For a deeper analysis of the distribution of features, we utilize the quantile function, particularly focusing on the inverse quantile function denoted as Q^{-1}. The equation is given as follows:

$$y = Q^{-1}(f(k)) = Q^{-1}(\int_{\infty}^{k} f_k(t)dt) \tag{7}$$

where k denotes the features, $f(k)$ corresponds to the cumulative distribution function of features, and Q^{-1} refers to the inverse quantile function associated with the distribution of the anticipated output values.

3.1.5. Data Balancing

Upon examining open-source IDS datasets, significant data imbalance is evident. For example, NSL-KDD and CICIDS2017 show considerable benign samples at 95% and 90%, with remaining samples consisting of various attacks. This stems from rare attack instances in real-world scenarios, with networks predominantly maintaining regular states. Pronounced imbalance skews classifier performance toward the majority class (benign), challenging minority class classification. Numerous techniques address this issue. Common strategies include undersampling, oversampling, class weight, and sample weight. Random oversampling balances data by randomly duplicating minority samples but risks overfitting. This study uses SMOTE oversampling, generating synthetic minority samples by examining neighbors to augment minority classes. Undersampling also reduces majority instances. The principle aims to balance underrepresented and dominant class distributions by randomly selecting majority instances. This study also considers class and sample weight strategies for adjusting weights, with class weights allocating identical weights to samples of the same class. We investigate benefits of several techniques to intrusion detection models.

3.2. Local Intrusion Detection Engine

3.2.1. Overview

Ensemble learning inherently excels against highly imbalanced challenges, evident by top algorithms in contests like KDDCup, Netflix, and Kaggle using ensembles. Hence, we employ ensemble learning for intrusion detection. Deep learning's recent success stems from multilayer representation learning, incrementally extracting features from original data via MLP, CNNs, DNNs, etc. However, tree-based models typically outperform neural models on tabular data.

3.2.2. Dynamic Forest-Structured Ensemble Network (DFSENet)

Our approach introduces a dynamic-depth, tree-structured network model within the ensemble learning framework. Recognizing the importance of diversity among component learners, we leverage a combination of machine learning models (including Random Forest, Extra Trees, LightGBM, and XGBoost) as base estimators and stack them to create a multi-layered network structure. The model is akin to an 'ensemble of ensembles' with connections between successive layers that may include ensemble estimators like XGBoost and Random Forest. This layered approach, reminiscent of deep learning structures, utilizes diversity to expand the model's depth, guard against overfitting, and improve overall accuracy. In the machine learning domain, diversity is crucial for enhancing the learning process. To further increase the diversity during training, our model integrates the initial inputs, augmenting the variety of data. This method enriches the class probability vectors from preceding layers in a sequential manner, offering rich insights for the layers that follow. Our experimental results indicate that this strategy significantly boosts accuracy. To refine the model and ensure it generalizes well, we apply cross-validation techniques, specifically 5-fold cross-validation, to optimize hyperparameters and guard against overfitting, which could otherwise degrade the model's performance on unseen test data. Specifically, validation samples constituting 20% of input are used in each layer of training to assess current performance (we opt for recall as a validation index). If the current layer metric drops beyond the threshold, training halts without expanding the next forest layer. Thus, the proposed model depth adapts without artificial setting. Figure 3 presents the design of the Dynamic Forest-Structured Ensemble Network, showcasing the framework's systematic approach to integrating multiple decision trees in a dynamic ensemble configuration.

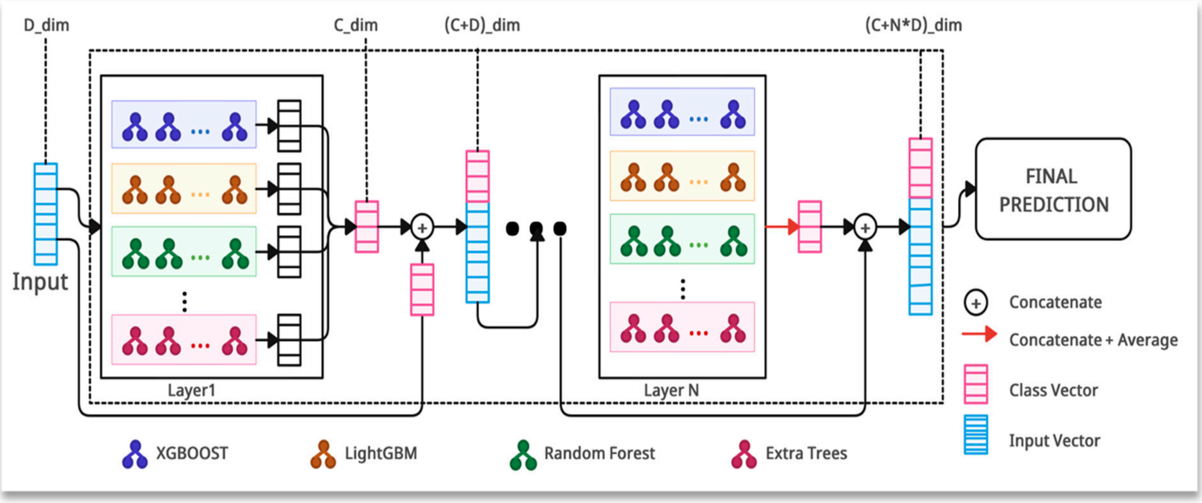


Figure 3. Structure of the DFSENet.

3.2.3. Machine Learning Models

Incorporated models include RF, ET, and XGBoost, constructed on diverse rule-based ensemble decision trees. A decision tree (DT) comprises decision and leaf nodes. Decision nodes signify decision routes while leaf nodes correspond to final predictions. Random Forest orchestrates decision trees utilizing bagging. Feature split point optimality in Random Forest minimizes the Gini index per Equation (8). Extra Trees (ETs) similarly amalgamate decision trees, distinguished by using all samples and randomly selecting features for branching.

$$Gini(k) = 1 - \sum_{i=1}^{n} (\frac{|C_i|}{|k|})^2 \tag{8}$$

where k represents the aggregate number of samples, C_i denotes the count of samples in the i-th class, n is indicative of the total number of classes. XGBoost stands as a gradient-boosting framework that integrates multiple decision trees and is celebrated for its speed, flexibility, and relatively small resource footprint. During each iteration, XGBoost focuses on optimizing the submodel relevant to that specific phase. For example, during the i-th iteration, it merely considers the $f_c(u_i)$.

$$f_c(u_i) = f_{c-1}(u_i) + f_c(u_i) \tag{9}$$

where $f_c(u_i)$ is the current model, $f_{c-1}(u_i)$ represents the model as established in the preceding step. The objective function of XGBoost is composed of a loss function coupled with a regularization term, which serves to constrain the complexity of the model, as depicted in Equation (10). The regularization term is obtained as defined in Equation (11). The ultimate objective function is reached after optimizing the loss and regularization components, as demonstrated in Equation (12). For multi-class classification tasks, we use XGBoost, where the loss function is specified as *softmax*, as delineated in Equation (13).

$$objective_{function} = \sum_{i=1}^{m} L(y_i, \hat{y_l}) + \sum_{i=1}^{t} \vartheta(f_i) \tag{10}$$

$$\vartheta(f_t) = \delta T + \frac{1}{2}\lambda \sum_{j=1}^{T} \omega_j^2 \tag{11}$$

$$objective_{function} = -\frac{1}{2}\sum_{j=1}^{T} \frac{R_j^2}{S_j + \lambda} + \delta T \tag{12}$$

$$softmax(z_i) = \frac{\exp(z_i)}{\sum_j \exp(z_j)} \tag{13}$$

where T represents the number of leaf nodes, ω_j^2 is the L2-norm of the leaf scores, λ and δ are the penalty coefficients. While R and S correspond to the cumulative first and second-order gradient statistics of the loss function, respectively, while z_i is the i-th samples in data z.

4. Experimental Results

All experiments in this study were conducted using the Google Colab system. PyTorch (version 2.4) served as our primary development framework and Python served as the programming language. The graphics processing unit (GPU) used was an NVIDIA T4 @1.59 GHz with 16 GB of memory and 8.1 TFLOPS of compute performance.

4.1. Datasets

For IDSs in connected car environments to effectively detect a wide range of attacks from both IVNs and external networks, we chose the CICIDS and car-hacking datasets to evaluate IDS progress in this study.

4.1.1. CICIDS Dataset

An IDS needs a representative benchmark network dataset for evaluating various IDSs. The first (CICIDS) dataset serves this purpose as a modern flow-based intrusion dataset. Previously, commonly used datasets were KDD99 and NSL-KDD, but in 2016, 11 criteria were published specifying fundamental requirements for a reliable intrusion detection dataset. The CICIDS dataset meets all criteria, positioning it as comprehensive and up to date. It primarily contains benign data over 80% and various attack types. The dataset is divided into 13 subcategories consolidated into six main categories during preprocessing, as depicted in Figure 4. Additionally, we split the CICIDS dataset into training and test sections at a 70/30 ratio for model training and performance evaluation, respectively.

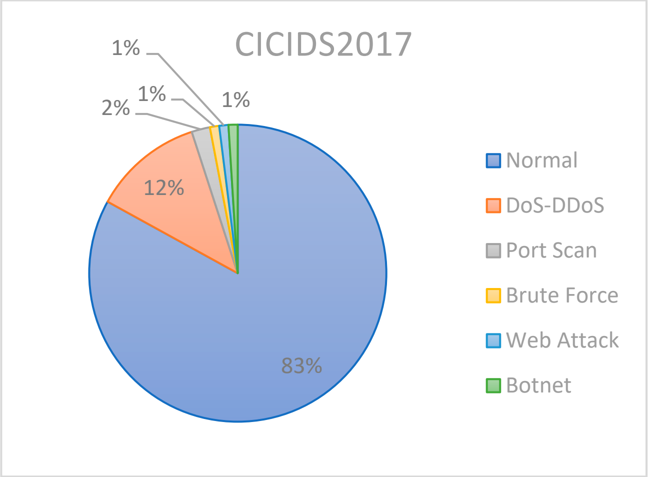


Figure 4. Consolidation of 13 subcategories into six main categories from the original CICIDS2017 dataset during preprocessing.

4.1.2. Car-Hacking Dataset

The second dataset, known as the car-hacking dataset, was created by capturing CAN traffic through the OBD-II port under conditions of a CAN attack. This dataset is characterized by 10 features, including a timestamp, CAN ID, DLC data bytes, CAN packet, and a label indicating Receive/Transmit (R/T). Nonetheless, the DLC and label were deemed to provide limited informative value and thus were discarded during the data cleansing phase. Any anomalies present in the data were also eliminated. With the original label column removed, new appropriate labels were designated. The dataset comprises normal traffic as well as four types of attacks: DDoS, Fuzzy, and Spoofing. Consequently, each data entry was annotated with the name of the attack it related to. The distribution within the dataset is illustrated in Figure 5. It is important to note that the dataset is predominantly composed of attack samples, which make up 95% of the data, making it unnecessary to apply any data-balancing techniques. In addition to cleansing, a data-shuffling method was implemented to ensure a thorough mix of the data. This step was crucial for splitting the dataset into training and testing partitions that were subsequently utilized for developing and validating the intrusion detection model.

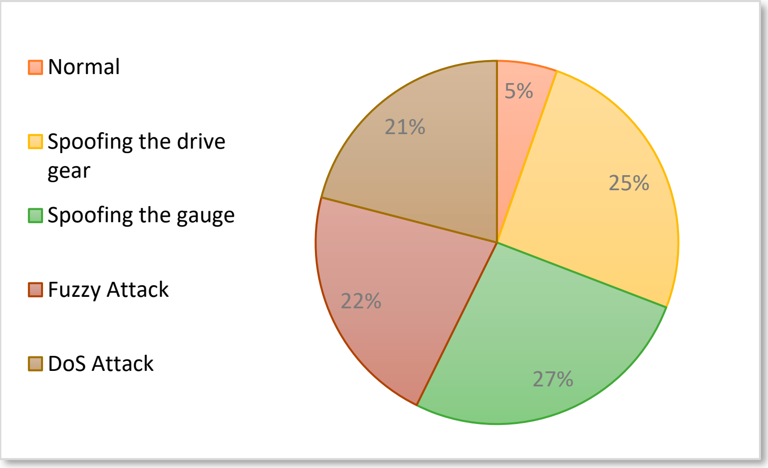


Figure 5. Distribution of the car-hacking dataset, highlighting the preponderance of attack samples at 95%, with no data balancing required.

4.2. Evaluation Metrics

Network intrusion detection involves classifying traffic as normal or an attack. Several machine learning metrics provide insight into a classifier's performance, defined below. Specifically, a binary classifier categorizes instances as positive or negative. Correctly classified instances are true positives (TP) or true negatives (TN). False positives (FP) or false negatives (FN) occur from misclassification. To quantify the efficacy of our IDS, we employ several performance metrics, such as the accuracy metric, which evaluates the overall correctness of the model by comparing the true positives (TP) and true negatives (TN) against all outcomes. Recall measures the model's ability to detect all relevant instances (true positives) out of all actual positives, which is critical for security systems where missing a true threat can be costly, precision assesses the accuracy of the positive predictions made by the model, highlighting its effectiveness in identifying true threats among all detected threats, and the F1-measure combines precision and recall into a single metric by calculating their harmonic mean, providing a balanced view of the model's performance, especially when the classes are imbalanced. As defined in Equation (15), the accuracy measures the proportion of correctly classified data. However, intrusion datasets often exhibit class imbalance, skewing accuracy to classify benign traffic while diminishing the identification of malicious traffic. Therefore, we focus more on precision, recall, and the F1-measure. An effective IDS considers both precision and recall. The F1-measure, capturing precision and recall through harmonic mean, presents a comprehensive evaluation metric. A higher F1-measure indicates greater algorithm classification capability. Besides these metrics, execution time merits assessment to understand the efficiency in processing single samples.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{14}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{15}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$\text{F1} - \text{measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recal}} \tag{17}$$

4.3. Discussion

Our aim is to develop a precise, streamlined, and effective IDS to protect IoV intelligent connected vehicles (ICVs) against various cyber threats. This section empirically validates the model using the datasets from Section 4.1. We assess effectiveness using the stated metrics and analyze results from multiple perspectives. During data preprocessing, we explore benefits of different data-balancing techniques for enhancing IDS performance, applying them to the CICIDS dataset. We equalized the training set's class distribution using diverse methods while leaving the testing set unchanged. Table 2 presents the balanced class distribution of the preprocessed training set.

Table 2. Balanced class distribution of the first dataset (CICIDS) after preprocessing.

Categories	Prior-Balance Adjustment	Post-Balance Adjustment
Normal	1,221,300	700,000
DDoS	192,162	420,000
Botnet	1160	45,000
Web Attack	1272	45,000
Port Scan	34,384	49,000
Brute Force	5131	53,000

We conduct a preliminary analysis of several data-balancing methods previously mentioned, using RF as the basis. The balanced training dataset is employed for model training. In contrast, the original testing set is used for evaluating the model's efficacy. The results of this experiment are visualized in Figure 6.

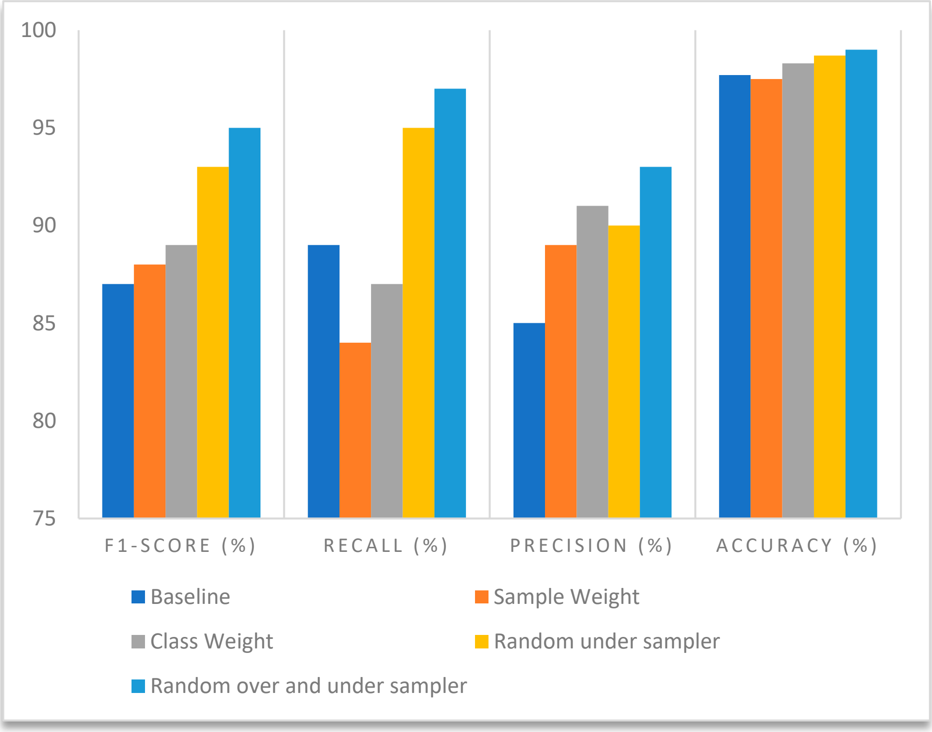


Figure 6. Comparative visualization of the detection performance of the RF model using various data-balancing techniques tested on the original testing set.

Figure 6 compares the detection performance of Random Forest (RF) using different data-balancing techniques. The baseline without balancing yielded an 87% F1-score, 89% recall, 85% precision, and 97% accuracy. Four balancing methods were evaluated: the (1) class weight mechanism, the (2) Random UnderSampler mechanism, the (3) sample weight mechanism, and (4) combining the Random OverSampler and UnderSampler. Figure 7 illustrates that all four methods substantially improved detection performance versus the baseline. F1-score enhancement ranged from 1.34% to 8.08%, and accuracy increased between 1% and 2%. Notably, combining oversampling and undersampling delivered the most impressive results. Consequently, we opted for the combined Random Over and UnderSampler approach to balance the original training set. Table 2 compares the model's detection performance with and without PCA. Our investigation showed that PCA had a minor impact, reducing the F1-score by 0.89%, recall by 1.13%, and precision by 0.73%. However, dimensionality was effectively decreased from 49 to 25 features, significantly reducing computational cost. PCA was selected primarily for its effectiveness in reducing our dataset's dimensionality while retaining the data's most significant variance. This capability is crucial for our analysis because it allows us to simplify the data without substantial loss of information. PCA does not require labels to reduce dimensionality, making it suitable for our initial exploratory data analysis where class labels might not be effectively utilized. PCA scales well with large datasets, which was necessary given the volume of data we needed to process. PCA is a generic, well-understood method that provides a solid baseline for dimensionality reduction, ensuring our results are easily interpretable and comparable within the broader research community. While LDA is also a popular choice for dimensionality reduction, it differs from PCA in several vital aspects: LDA requires class labels to maximize the separability between known categories, making it less flexible than PCA in unsupervised scenarios. LDA is limited to extracting at most C − 1 features (where C is the number of class labels), which can be a significant limitation in datasets with a small number of classes. LDA can overfit in scenarios with very few data points per class, whereas PCA remains general. Table 3 presents the balanced class distribution of the CICIDS training set after preprocessing.

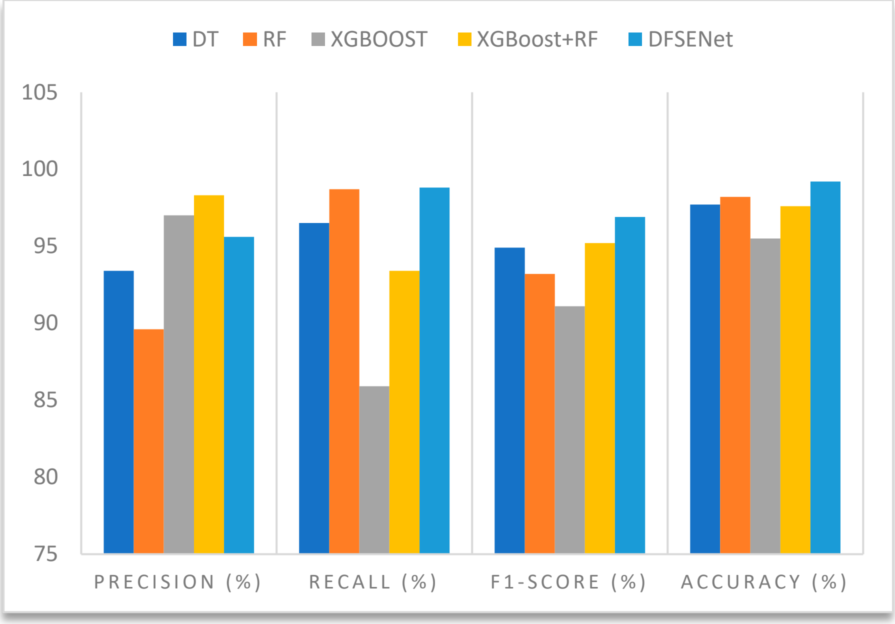


Figure 7. Performance metrics of the optimal base estimators for the IDS model.

Table 3. Balanced class distribution of the CICIDS training set after preprocessing.

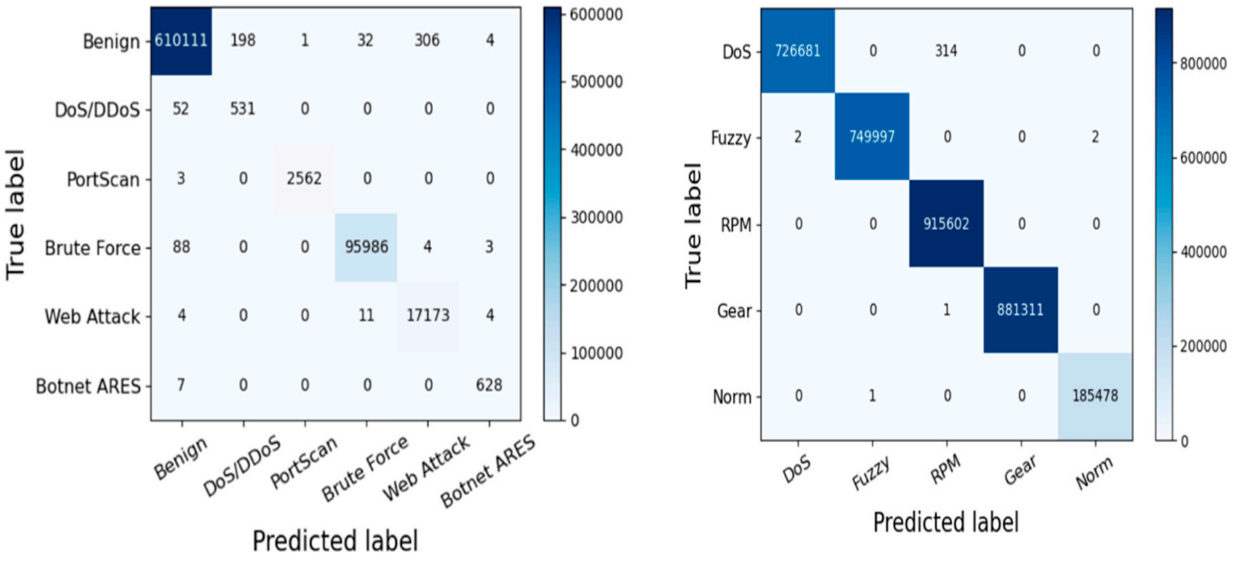
	Precision	Recall	F1-Score	Accuracy	Execution Time (ms)
Without PCA	95.9	98.8	96	99	1.05
With PCA	95.4	98.3	95	98	2.66×10^{-3}

We analyzed various ML models (RF, ET, XGBoost, and LightGBM) using the processed CICIDS dataset to choose the optimal base estimator for the definitive IDS model. The initial four rows show that bagging ensemble models like RF and ET outperform boosting models like XGBoost and LightGBM in precision and speed. Therefore, we selected RF and ET as base estimators. To incorporate diversity, we also included XGBoost. Stacking these varied base estimators without extending layers yielded a 94.77% F1-score, 92.3% recall, 98.66% precision, and 99.88% accuracy. The final row shows that DFSENet outperforms single ML models, with F1-score enhancements of 3.09–7.6%, a recall of 0.26–13.42%, a precision of 0.82–5.76%, and accuracy increases of 0.13–0.35%. Further analysis revealed that deepening structure significantly boosts detection performance. Compared to stacked models, the DFSENet demonstrated a 1.69% higher F1-score, 6.07% increased recall, and 0.02% accuracy enhancement. This strongly suggests successive learning layers significantly contribute to improved performance. We evaluated several models to ascertain their efficacy in terms of precision, recall, F1-score, accuracy, and execution time. The summarized results are presented in Table 4. This table highlights the performance metrics of different models including Decision Trees (DT), Random Forest (RF), XGBoost, a combined model of XGBoost and RF, and DFSENet. Each model's performance was assessed under identical conditions to ensure a fair comparison. Notably, the DFSENet model outperformed others with an accuracy of 99.2% and a notably efficient execution time.

Table 4. Comparative analysis of detection capabilities across ensemble and single ML models with emphasis on DFSENet enhancements.

Model	Precision	Recall	F1-Score	Accuracy	Execution Time (ms)
DT	93.4	96.5	94.9	97.7	1.23×10^{-25}
RF	89.6	98.7	93.2	98.2	1.67×10^{-5}
XGBOOST	97	85.9	91.1	95.5	3.83×10^{-5}
XGBoost + RF	98.3	93.4	95.2	97.6	1.26×10^{-4}
Proposed DFSENet	95.6	98.8	96.9	99.2	2.91×10^{-4}

We generated confusion matrices for our proposed model using the testing portion of both the CICIDS dataset and the car-hacking dataset as shown in Figure 8. The numerical values allocated to each category within these matrices present a granular perspective of the successful and mistaken classifications. The data evidently demonstrates the model's aptitude in distinguishing between various types of network traffic, whether they are benign or malicious.



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

Figure 8. Confusion matrices (**a**) obtained from the CICIDS testing set, and (**b**) obtained from the car-hacking dataset's testing set.

Table 5 presents evaluation metrics in detail for each category within the CICIDS testing set. The results clearly show that the DFSENet delivers outstanding performance across all categories, achieving near-perfect precision, recall, and F1-score. This implies the superb detection of most attack instances. However, the "Botnet" categories exhibit distinctly higher recall but lower precision. Detecting all malicious activity is paramount for network intrusion detection, making high recall critical. Meanwhile, a few false positives (lower precision) are considered less important. Figure 9 illustrates the evaluation metrics for each category within the CICIDS testing set, with a particular focus on the performance of the DFSENet model. The figure clearly demonstrates DFSENet's superior detection capabilities across various metrics.

Table 5. Assessment of model performance on the first (CICIDS) dataset.

	Precision	Recall	F1-Score	Accuracy
Normal	0.984	0.974	0.978	0.992
DDoS	0.963	0.973	0.965	0.991
Web Attack	0.978	0.968	0.973	0.992
Botnet	0.882	0.962	0.922	0.93
Brute Force	0.982	0.986	0.984	0.995
Port Scan	0.979	0.989	0.984	0.995

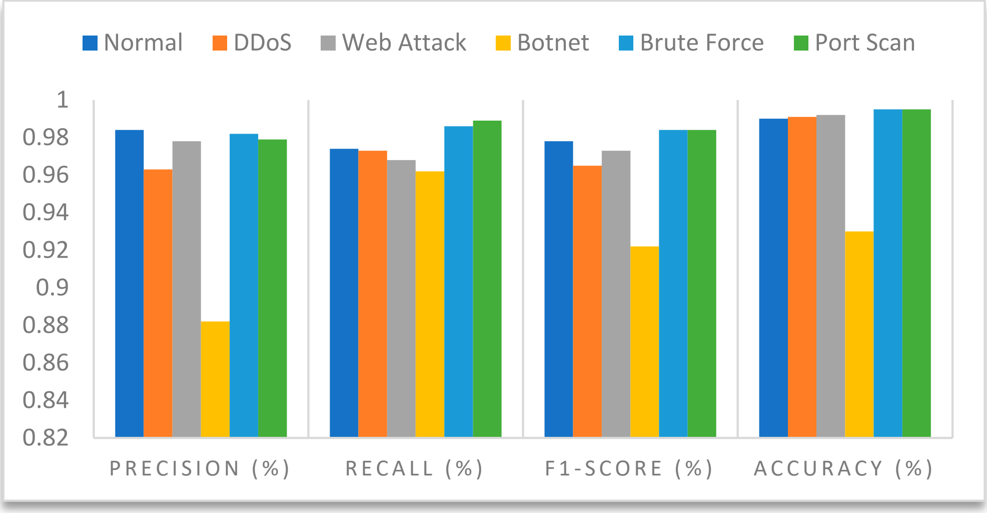


Figure 9. Evaluation metrics for each category in the CICIDS testing set, highlighting DFSENet's superior detection performance with a special note of the 'Botnet' Category's high recall and lower precision.

The results from testing on the car-hacking dataset demonstrate the effectiveness of our intrusion detection model. Table 6 shows that the model achieves almost perfect detection for all categories, with the F1-score approaching 98%. This underscores the outstanding capability of the proposed intrusion detection system (IDS) in recognizing malicious activity within In-Vehicle Networks (IVNs). Table 6 presents a comprehensive view of the proposed IDS performance when evaluated using the car-hacking dataset.

Table 6. Assessment of model performance on the second (car-hacking) dataset.

	Precision	Recall	F1-Score	Accuracy
Normal	0.984	0.984	0.984	0.98
DoS	0.964	0.986	0.975	0.98
Gear	0.978	0.968	0.973	0.98
Spoofing Gauge	0.972	0.982	0.977	0.98
Fuzzy	0.982	0.986	0.984	0.98

Figure 10 provides a comprehensive overview of the performance of our proposed Intrusion Detection System (IDS) on the car-hacking dataset. This visualization is instrumental in demonstrating the effectiveness of the IDS in identifying and mitigating threats in automotive systems.

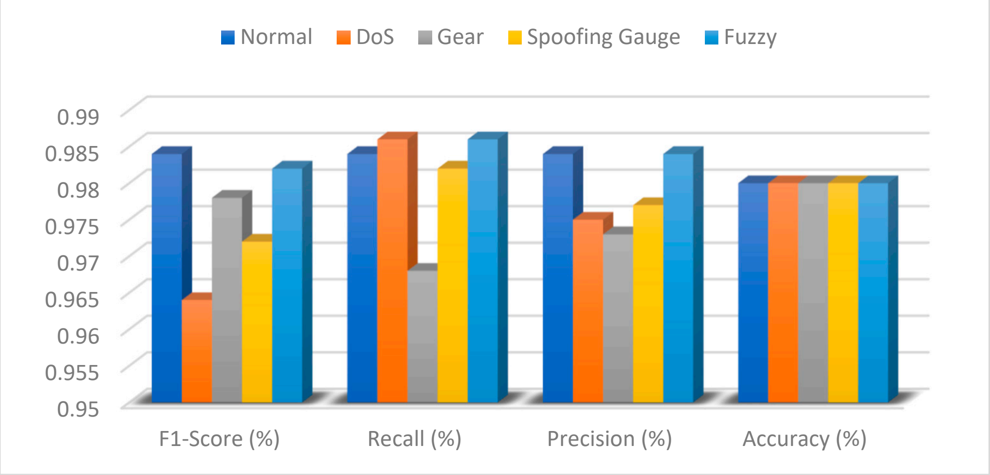


Figure 10. Performance overview of the proposed IDS on the car-hacking dataset.

4.4. Empirical Analysis between the Proposed Model and Related Works

In Table 7, a variety of studies such as those using MLP, 1D-CNN, and LSTM, predominantly focus on binary classification over multiclass classification. Within this subset, the 1D-CNN framework stands out, achieving an F1-score peak of 0.939. Our objective extends beyond the binary distinction of normal and anomalous activities to encompass precise categorization across a broader spectrum. When juxtaposed with the approaches listed in Table 6, our technique exhibits superior performance metrics, including precision, recall, and an F1-score that tops at 0.965. In comparison to the DBN model, which also tackles multiclass classification, our method shows an enhancement in the F1-score by 2.5%.

Table 7. Empirical analysis and comparison between existing methods and the proposed model.

	Precision	Recall	F1-Score	Accuracy	Categories
LSTM	0.954	0.895	0.885	0.893	2
MLP	0.882	0.859	0. 868	0.872	2
1D-CNN	0.964	0.906	0.935	0.938	2
DBN	0.897	0.975	0.943	0.946	6
Proposed Model	0.956	0.988	0.969	0.992	6

4.5. Limitation and Future Works

This section discusses several key limitations of our current study. Firstly, the experimental design, while robust, is constrained by the size and diversity of the datasets employed, potentially affecting the universality of our findings. Secondly, while our model shows promising results, it relies heavily on high-quality data input, which may not be as readily available in real-world scenarios. Lastly, our analysis methods, though effective for the scope of this study, may not capture all nuances of complex data interactions, suggesting a need for more sophisticated analytical tools in future studies. The DFSENet is inherently complex due to its multi-layered ensemble structure, which integrates multiple advanced machine learning models like XGBoost, LightGBM, Random Forest, and Extra Trees. This complexity is necessary to handle multiclass classification and address class imbalance effectively. However, it can also affect the DFSENet's scalability in large-scale IoV environments, as each additional layer or model increases the computational load and memory requirements. Deploying such a model in real-world IoV settings might require substantial computational resources, which could be a limitation in environments with restricted resource availability. For future research, we aim to enhance the DFSENet model's adaptability to emerging threats, improve its scalability for growing IoV networks, and ensure its efficiency in high-traffic environments. We also plan to explore the application of our IDS framework across various IoT domains to secure interconnected systems within smart infrastructures. Furthermore, we will investigate collaborative defense strategies that leverage network-wide intelligence for proactive threat mitigation, and rigorously test the system against advanced attack vectors, including AI-driven and zero-day exploits, to maintain robust and forward-thinking vehicular cybersecurity.

5. Conclusions

This study presents a novel intrusion detection system (IDS) for the Internet of Vehicles (IoV) that leverages a DFSENet. The proposed model is a significant contribution to the field of vehicular cybersecurity, offering a sophisticated multilayered approach to detect and classify a spectrum of cyber threats. By incorporating advanced data-balancing techniques and feature reduction through Principal Component Analysis (PCA), our model effectively addresses the challenges of class imbalance and high-dimensional data, which are prevalent in network traffic datasets. The DFSENet model's architecture, which sequentially stacks multiple machine learning models, represents a breakthrough in enhancing detection accuracy and response time against cyber-attacks. Empirical results from our extensive experiments using the CICIDS2017 and car-hacking datasets have demonstrated the superiority of the proposed IDS. With an impressive accuracy of 99.2% on the CICIDS dataset and 98% on the car-hacking dataset, along with high precision, recall, and f-measure scores, the DFSENet model has proven to be highly effective. Moreover, it has shown its potential in solving related problems such as reducing false positives and enhancing real-time detection capabilities, which are crucial for the practical deployment of an IDS in the IoV.

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

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this paper are publicly available to everyone and can be accessed at: https://www.unb.ca/cic/datasets/ids-2017.html (accessed on 15 September 2024) and https://ocslab.hksecurity.net/Datasets/car-hacking-dataset (accessed on 15 September 2024).

Conflicts of Interest: The authors declare that they have no conflicts of interest.

References

1. Manale, B.; Tomader, M. A Survey of Intrusion Detection Algorithm in VANET. In Proceedings of the NISS2020: The 3rd International Conference on Networking, Information Systems & Security, ACM International Conference Proceeding Series, Marrakech, Morocco, 31 March–2 April 2020. [CrossRef]
2. Buczak, A.L.; Guven, E. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1153–1176. [CrossRef]
3. Young, C.; Zambreno, J.; Olufowobi, H.; Bloom, G. Survey of automotive controller area network intrusion detection systems. *IEEE Des. Test* **2019**, *36*, 48–55. [CrossRef]
4. Liang, J.; Sheikh, M.S.; Wang, W. A Survey of Security Services, Attacks, and Applications for Vehicular Ad Hoc Networks (VANETs). *Sensors* **2019**, *19*, 3589. [CrossRef] [PubMed]
5. Sharma, S.; Kaul, A. A survey on Intrusion Detection Systems and Honeypot based proactive security mechanisms in VANETs and VANET Cloud. *Veh. Commun.* **2018**, *12*, 138–164. [CrossRef]
6. Dong, S.; Su, H.; Xia, Y.; Zhu, F.; Hu, X.; Wang, B. A Comprehensive Survey on Authentication and Attack Detection Schemes That Threaten It in Vehicular Ad-Hoc Networks. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 13573–13602. [CrossRef]
7. Ullah, S.; Khan, M.A.; Ahmad, J.; Jamal, S.S.; e Huma, Z.; Hassan, M.T.; Pitropakis, N.; Arshad; Buchanan, W.J. HDL-IDS: A Hybrid Deep Learning Architecture for Intrusion Detection in the Internet of Vehicles. *Sensors* **2022**, *22*, 1340. [CrossRef]
8. Injadat, M.; Moubayed, A.; Nassif, A.B.; Shami, A. Multi-Stage Optimized Machine Learning Framework for Network Intrusion Detection. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 1803–1816. [CrossRef]

9. Schwenker, F. Ensemble Methods: Foundations and Algorithms [Book Review]. *IEEE Comput. Intell. Mag.* **2013**, *8*, 77–79. [CrossRef]
10. Kachirski, O.; Guha, R. Effective intrusion detection using multiple sensors in wireless ad hoc networks. In Proceedings of the 36th Annual Hawaii International Conference on System Sciences, HICSS 2003, Big Island, HI, USA, 6–9 January 2003. [CrossRef]
11. Ahmed, H.A.; Hameed, A.; Bawany, N.Z. Network intrusion detection using oversampling technique and machine learning algorithms. *PeerJ Comput. Sci.* **2022**, *8*, e820. [CrossRef]
12. Manderna, A.; Kumar, S.; Dohare, U.; Aljaidi, M.; Kaiwartya, O.; Lloret, J. Vehicular Network Intrusion Detection Using a Cascaded Deep Learning Approach with Multi-Variant Metaheuristic. *Sensors* **2023**, *23*, 8772. [CrossRef]
13. Goncalves, F.; Ribeiro, B.; Gama, O.; Santos, A.; Costa, A.; Dias, B.; Macedo, J.; Nicolau, M.J. A Systematic Review on Intelligent Intrusion Detection Systems for VANETs. In Proceedings of the 2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, Dublin, Ireland, 28–30 October 2019; Volume 2019. [CrossRef]
14. Nie, L.; Ning, Z.; Wang, X.; Hu, X.; Cheng, J.; Li, Y. Data-Driven Intrusion Detection for Intelligent Internet of Vehicles: A Deep Convolutional Neural Network-Based Method. *IEEE Trans. Netw. Sci. Eng.* **2020**, *7*, 2219–2230. [CrossRef]
15. Gad, A.R.; Nashat, A.A.; Barkat, T.M. Intrusion Detection System Using Machine Learning for Vehicular Ad Hoc Networks Based on ToN-IoT Dataset. *IEEE Access* **2021**, *9*, 142206–142217. [CrossRef]
16. Arya, M.; Sastry, H.; Dewangan, B.K.; Rahmani, M.K.I.; Bhatia, S.; Muzaffar, A.W.; Bivi, M.A. Intruder Detection in VANET Data Streams Using Federated Learning for Smart City Environments. *Electronics* **2023**, *12*, 894. [CrossRef]
17. Karthiga, B.; Durairaj, D.; Nawaz, N.; Venkatasamy, T.K.; Ramasamy, G.; Hariharasudan, A. Intelligent Intrusion Detection System for VANET Using Machine Learning and Deep Learning Approaches. *Wirel. Commun. Mob. Comput.* **2021**, *2022*, 5069104. [CrossRef]
18. Hassan, F.; Yu, J.; Syed, Z.S.; Ahmed, N.; Al Reshan, M.S.; Shaikh, A. Achieving model explainability for intrusion detection in VANETs with LIME. *PeerJ Comput. Sci.* **2023**, *9*, e1440. [CrossRef]
19. Koscher, K.; Czeskis, A.; Roesner, F.; Patel, S.; Kohno, T.; Checkoway, S.; McCoy, D.; Kantor, B.; Anderson, D.; Shacham, H.; et al. Experimental security analysis of a modern automobile. In Proceedings of the 2010 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 16–19 May 2010; pp. 447–462. [CrossRef]
20. Song, W.; Choi, H.; Kim, J.; Kim, E.; Kim, Y.; Kim, J. Fingerprinting Electronic Control Units for Vehicle Intrusion Detection. 2016. Available online: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/song (accessed on 17 January 2024).
21. Woo, S.; Jo, H.J.; Lee, D.H. A Practical Wireless Attack on the Connected Car and Security Protocol for In-Vehicle CAN. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 993–1006. [CrossRef]
22. Kumar, M.; Hanumanthappa, M.; Kumar, T.V.S. Intrusion Detection System using decision tree algorithm. In Proceedings of the International Conference on Communication Technology Proceedings, ICCT 2012, Chengdu, China, 9–11 November 2012; pp. 629–634. [CrossRef]
23. Peng, K.; MLeung, V.C.; Zheng, L.; Wang, S.; Huang, C.; Lin, T. Intrusion Detection System Based on Decision Tree over Big Data in Fog Environment. *Wirel. Commun. Mob. Comput.* **2017**, *2018*, 4680867. [CrossRef]
24. Chitrakar, R.; Huang, C. Selection of Candidate Support Vectors in incremental SVM for network intrusion detection. *Comput. Secur.* **2014**, *45*, 231–241. [CrossRef]
25. Zhang, H.; Dai, S.; Li, Y.; Zhang, W. Real-time Distributed-Random-Forest-Based Network Intrusion Detection System Using Apache Spark. In Proceedings of the 2018 IEEE 37th International Performance Computing and Communications Conference, IPCCC 2018, Orlando, FL, USA, 17–19 November 2018. [CrossRef]
26. Waskle, S.; Parashar, L.; Singh, U. Intrusion Detection System Using PCA with Random Forest Approach. In Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020, Coimbatore, India, 2–4 July 2020; pp. 803–808. [CrossRef]
27. Diro, A.; Chilamkurti, N. Leveraging LSTM Networks for Attack Detection in Fog-to-Things Communications. *IEEE Commun. Mag.* **2018**, *56*, 124–130. [CrossRef]
28. Samy, A.; Yu, H.; Zhang, H. Fog-Based Attack Detection Framework for Internet of Things Using Deep Learning. *IEEE Access* **2020**, *8*, 74571–74585. [CrossRef]
29. Labiod, Y.; Korba, A.A.; Ghoualmi, N. Fog Computing-Based Intrusion Detection Architecture to Protect IoT Networks. *Wirel. Pers. Commun.* **2022**, *125*, 231–259. [CrossRef]
30. Li, S.; Lu, Y.; Li, J. CAD-IDS: A Cooperative Adaptive Distributed Intrusion Detection System with Fog Computing. In Proceedings of the 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2022, Hangzhou, China, 4–6 May 2022; pp. 635–640. [CrossRef]
31. Latif, S.; e Huma, Z.; Jamal, S.S.; Ahmed, F.; Ahmad, J.; Zahid, A.; Dashtipour, K.; Aftab, M.U.; Ahmad, M.; Abbasi, Q.H. Intrusion Detection Framework for the Internet of Things Using a Dense Random Neural Network. *IEEE Trans. Ind. Inform.* **2022**, *18*, 6435–6444. [CrossRef]
32. Thapa, K.N.K.; Duraipandian, N. Malicious Traffic classification Using Long Short-Term Memory (LSTM) Model. *Wirel. Pers. Commun.* **2021**, *119*, 2707–2724. [CrossRef]

33. Latif, S.; Driss, M.; Boulila, W.; e Huma, Z.; Jamal, S.S.; Idrees, Z.; Ahmad, J. Deep Learning for the Industrial Internet of Things (IIoT): A Comprehensive Survey of Techniques, Implementation Frameworks, Potential Applications, and Future Directions. *Sensors* **2021**, *21*, 7518. [CrossRef]
34. 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]
35. Ma, T.; Yu, Y.; Wang, F.; Zhang, Q.; Chen, X. A hybrid methodologies for intrusion detection based deep neural network with support vector machine and clustering technique. *Lect. Notes Electr. Eng.* **2018**, *422*, 123–134. [CrossRef]
36. Zaidi, K.; Milojevic, M.B.; Rakocevic, V.; Nallanathan, A.; Rajarajan, M. Host-Based Intrusion Detection for VANETs: A Statistical Approach to Rogue Node Detection. *IEEE Trans. Veh. Technol.* **2016**, *65*, 6703–6714. [CrossRef]
37. Alheeti, K.M.A.; Donald-Maier, K.M. Intelligent intrusion detection in external communication systems for autonomous vehicles. *Syst. Sci. Control Eng.* **2018**, *6*, 48–56. [CrossRef]
38. Zhao, R.; Gui, G.; Xue, Z.; Yin, J.; Ohtsuki, T.; Adebisi, B.; Gacanin, H. A Novel Intrusion Detection Method Based on Lightweight Neural Network for Internet of Things. *IEEE Internet Things J.* **2022**, *9*, 9960–9972. [CrossRef]
39. Yang, L.; Moubayed, A.; Hamieh, I.; Shami, A. Tree-based intelligent intrusion detection system in internet of vehicles. In Proceedings of the 2019 IEEE Global Communications Conference, GLOBECOM 2019—Proceedings, Waikoloa, HI, USA, 9–13 December 2019. [CrossRef]
40. Chen, Z.; Simsek, M.; Kantarci, B.; Djukic, P. All Predict Wisest Decides: A Novel Ensemble Method to Detect Intrusive Traffic in IoT Networks. In Proceedings of the 2021 IEEE Global Communications Conference, GLOBECOM 2021—Proceedings, Madrid, Spain, 7–11 December 2021. [CrossRef]
41. Khalvati, L.; Keshtgary, M.; Rikhtegar, N. Intrusion Detection based on a Novel Hybrid Learning Approach. *J. AI Data Min.* **2018**, *6*, 157–162. [CrossRef]
42. Canbay, Y.; Sagiroglu, S. A hybrid method for intrusion detection. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015, Miami, FL, USA, 9–11 December 2016; pp. 156–161. [CrossRef]
43. Singh, P.; Kaur, A.; Aujla, G.S.; Batth, R.S.; Kanhere, S. DaaS: Dew Computing as a Service for Intelligent Intrusion Detection in Edge-of-Things Ecosystem. *IEEE Internet Things J.* **2021**, *8*, 12569–12577. [CrossRef]
44. Albers, P.; Camp, O.; Percher, J.; Jouga, B.; Mé, L.; Puttini, R. Security in Ad Hoc Networks: A General Intrusion Detection Architecture Enhancing Trust Based Approaches. In Proceedings of the Wireless Information Systems, 1st International Workshop on Wireless Information Systems, WIS 2002 in Conjunction with ICEIS 2002, Ciudad Real, Spain, 2–3 April 2002.
45. Sterne, D.; Balasubramanyam, P.; Carman, D.; Wilson, B.; Talpade, R.; Ko, C.; Balupari, R.; Tseng, C.-Y.; Bowen, T.; Levitt, K.; et al. A general cooperative intrusion detection architecture for MANETs. In Proceedings of the 3rd IEEE International Workshop on Information Assurance, IWIA 2005, College Park, MD, USA, 23–24 March 2005; pp. 57–70. [CrossRef]
46. Blowers, M.; Williams, J. Machine learning applied to cyber operations. *Adv. Inf. Secur.* **2014**, *55*, 155–175. [CrossRef]
47. Horng, S.J.; Su, M.-Y.; Chen, Y.-H.; Kao, T.-W.; Chen, R.-J.; Lai, J.-L.; Perkasa, C.D. A novel intrusion detection system based on hierarchical clustering and support vector machines. *Expert Syst. Appl.* **2011**, *38*, 306–313. [CrossRef]
48. Muda, Z.; Yassin, W.; Sulaiman, M.N.; Udzir, N.I. Intrusion detection based on K-Means clustering and Naïve Bayes classification. In Proceedings of the 2011 7th International Conference on Information Technology in Asia, Sarawak, Malaysia, 12–13 July 2011. [CrossRef]
49. Hikal, N.A.; Elgayar, M.M. Enhancing IoT Botnets Attack Detection Using Machine Learning-IDS and Ensemble Data Preprocessing Technique. In *Internet of Things—Applications and Future*; Lecture Notes in Networks and Systems; Ghalwash, A., El Khameesy, N., Magdi, D., Joshi, A., Eds.; Springer: Singapore, 2020; Volume 114.
50. Mohammed; Jassim, R.; Abed, E.A.; Elgayar, M.M. Comparative study between metaheuristic algorithms for internet of things wireless nodes localization. *Int. J. Electr. Comput. Eng.* **2022**, *12*, 660–668.
51. Haytham Tarek Mohammed, F.; El-Gayar, M.M.; Aboelfetouh, A. Detection Technique and Mitigation Against a Phishing Attack. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2021**, *12*, 2021. [CrossRef]
52. Zhang, W.; Wu, C.; Zhong, H.; Li, Y.; Wang, L. Prediction of undrained shear strength using extreme gradient boosting and random forest based on Bayesian optimization. *Geosci. Front.* **2020**, *12*, 469–477. [CrossRef]
53. Zhang, W.; Wu, C.; Tang, L.; Gu, X.; Wang, L. Efficient time-variant reliability analysis of Bazimen landslide in the Three Gorges Reservoir Area using XGBoost and LightGBM algorithms. *Gondwana Res.* **2023**, *123*, 41–53. [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.