



ISSN: 0067-2904

Controller Area Network Bus Spoofing Attack Detection in Internet of Vehicles Using Machine Learning Approaches

Sabreen Salih Mahdi^{1*}, Amer Abdulmajeed Abdulrahman

Computer Science Dep., College of Science, University of Baghdad, Baghdad, Iraq

Received: 24/5/2025

Accepted: 10/11/2025

Published: 30/9/2026

Abstract

Considering the complexity of network traffic in IoV operations, methods that can identify complex patterns become useful. Machine learning fosters several techniques to enhance the detection, prevention, and mitigation of cyberattacks. This paper presents a hybrid approach combining machine learning (ML) and deep learning (DL) techniques for detecting multiple variants of spoofing in the Internet of Vehicles (IoV). The increasing connectivity of vehicles has expanded the attack surface for malicious actors, making effective detection of spoofing attacks a critical security concern. Our approach integrates traditional ML algorithms (Random Forest, Gradient Boosting, SVM) with deep learning architectures (CNN-LSTM, CNN), leveraging the complementary strengths of both approaches. To address the significant class imbalance observed in the CICIoV2024 dataset (86.96% benign traffic versus 13.04% attack traffic), we implemented a balanced sampling approach using under sampling of the dominant class while maintaining realistic data distributions. Experimental evaluation showed that our hybrid approach achieved high detection accuracy (99.3–99.4%) across various attack types, such as Denial of Service (DOS), GAS, RPM, SPEED, and STEERING_WHEEL spoofing. Performance analysis revealed that the highest overall accuracy was achieved with CNN-LSTM (99.4%), while Random Forest provided similar results (99.3%) with significantly better computational efficiency (40x faster training time). These results make it mainly appropriate for real-time applications.

Keywords: Internet of Vehicles (IoV), DoS Attack, Spoofing Attack, Ensemble Learnin

كشف هجمات انتحال ناقل شبكة منطقة التحكم (CAN) في إنترنت المركبات بأعتماد تقنيات تعلم الآلة

عامر عبد المجيد عبد الرحمن^{*}, صابرين صالح مهدي

*Email : sabreen.saleh2301@sc.uobaghdad.edu.iq



الخلاصة

نظرًا لتعقيد حركة مرور الشبكة في عمليات إنترنت المركبات (IoV)، أصبحت الأساليب القادرة على تحديد الأنماط المعقدة ذات أهمية متزايدة. وتوفر تقنيات التعلم الآلي (ML) العديد من الأساليب التي تسهم في تعزيز اكتشاف الهجمات السيبرانية ومنعها والتخفيف من آثارها. تقدم هذه الدراسة منهجًا هجينًا يجمع بين تقنيات التعلم الآلي (ML) والتعلم العميق (DL) للكشف عن أنواع متعددة من هجمات الانتحال (Spoofing) في بيئة إنترنت المركبات (IoV).

أدى تزايد الاتصال بين المركبات إلى توسيع سطح الهجوم أمام الجهات الخبيثة، مما جعل الكشف الفعال عن هجمات الانتحال يمثل تحديًا أمينيًا بالغ الأهمية. ويعتمد النهج المقترح على دمج خوارزميات التعلم الآلي التقليدية، وهي الغابة العشوائية (Random Forest)، والتعزيز التدريجي (Gradient Boosting)، وآلة المتجهات الداعمة (SVM)، مع تقنيات التعلم العميق، وهي CNN-LSTM و CNN، مستفيدًا من نقاط القوة التكاملية لكل من النهجين.

ولمعالجة مشكلة عدم توازن الفئات الكبيرة التي لوحظت في مجموعة بيانات CICIoV2024، حيث تمثل حركة المرور السليمة (Benign) نسبة 86.96% مقابل 13.04% لحركة المرور الخاصة بالهجمات، تم تطبيق أسلوب أخذ عينات متوازن باستخدام تقليل عينات الفئة المهيمنة، مع الحفاظ على توزيعات بيانات واقعية. أظهرت النتائج التجريبية أن النهج الهجين حقق دقة كشف مرتفعة تراوحت بين 99.3% و 99.4% عبر أنواع مختلفة من الهجمات، بما في ذلك هجمات حجب الخدمة (DoS)، وهجمات انتحال GAS و RPM و SPEED و STEERING_WHEEL.

وأظهر تحليل الأداء أن أعلى دقة إجمالية تحققت باستخدام نموذج CNN-LSTM بنسبة 99.4%، في حين قدم نموذج Random Forest دقة مماثلة بلغت 99.3% مع كفاءة حسابية أعلى بكثير، إذ كان زمن تدريبه أسرع بنحو 40 مرة. وتجعل هذه الكفاءة نموذج Random Forest مناسبًا بصورة خاصة للتطبيقات التي تتطلب الكشف في الوقت الفعلي.

1. INTRODUCTION

Internet of Vehicles (IoV) is an innovative smart transportation system part, which allows vehicles to connect and share information with their surroundings using wireless technology. This has led to major improvements in how transportation works, as it helps in more effective traffic management, besides making roads safer. It's already showing great results in real-time applications. [1]. When vehicles communicate with one another, which can lead to vulnerability to cyberattacks, causing serious security issues in turn, like identity theft or even loss of control over some vehicle parts. Such attacks can lead to problems, including service outages or even life-threatening situations. Hackers causing intentional crashes or controlling the vehicle by sending real-looking fake data is a major concern, such as false information about another vehicle's location, speed or status. As semi-autonomous and self-driving cars become more common, these types of attacks should be expected to occur more often. [2]. Frequently, traditional security methods struggle to catch threats, especially when those threats appear like normal, harmless signals. By creating smarter ways of detecting those attacks, AI was used to solve this problem, due to the fact that it is better at detecting unusual patterns and recognizing identity theft attempts that are designed to look like normal, real-world activity. [3].

This paper introduces a new hybrid method that uses both machine learning (ML) and deep learning (DL) for the detection of impersonation attacks in Internet of Vehicles (IoV) systems. The approach takes advantage of the strengths of each technique. ML is good at finding patterns in organized data by using features that have been defined already, whereas DL can perform automatic detection of complex patterns from raw data and has been proven highly effective at spotting suspicious or unusual activity in the data that it has been trained on. [4].

This proposed approach has been designed by combining multiple layers of detection. Traditional machine learning algorithms, such as support vector machines and random forests, are initially used to analyze signal characteristics. After that, Deep neural networks (DNNs), like CNNs and RNNs, are utilized on raw signal data for the extraction and classification of features. The outputs of these multiple analytical approaches are combined afterwards. The system becomes able to achieve higher detection accuracy in the case of combining those methods' outputs while maintaining lower false positive rates compared with single-technique methods. This research includes several contributions:

1. A comprehensive threat model focusing on CAN bus signals (RPM, SPEED, DoS, GAS, STEERING_WHEEL) for IoV spoofing attacks.
2. A novel hybrid detection methodology combining ML (RF, GB, SVM) and deep learning (CNN-LSTM, CNN) models.
3. A data preprocessing pipeline with balanced sampling techniques to address the class imbalance problem.
4. Experimental validation: An 80/20 split of the data between training and testing was used, utilizing multiple performance metrics.

In particular, the implementation focuses on Controller Area Network (CAN) bus signals, analyzing data frames from DATA_0 to DATA_7 to detect multiple spoofing attack variants, including Denial of Service (DoS) attacks, Gear Spoofing (GAS), RPM tampering, Speed Spoofing, and Steering Wheel attacks. The detection framework employs a comprehensive preprocessing pipeline including data cleaning, standardization, and label encoding, with balanced sampling techniques (under sampling and SMOTE) to address class imbalance issues. The model's performance is rigorously evaluated using accuracy, precision, recall, F1 score, and ROC curves across both traditional machine learning models (RF, GB, SVM) and deep learning architectures (CNN-LSTM, CNN) [5].

The remainder of this paper is organized as follows: Section 2 reviews related work in spoofing attack detection. Section 3 details the system architecture and methodologies employed in our hybrid approach. Section 4 presents our experimental setup and results. In Section 5, the implications and potential limitations of our findings are discussed. Finally, in Section 6, the research is concluded, and future research directions are identified.

2. Related work

Over the past few years, a range of ideas have been proposed by academics to address the problem of secure and reliable detection of cybersecurity attacks on Internet of Vehicles (IoV) and interference detection in autonomous vehicle networks (VANETs), ranging from traditional machine learning methods to advanced deep learning approaches. Modern methodological developments in vehicle network security will be presented.

Nur 'A et al (2024) [6]. In this study, denial-of-service (DoS) attacks and impersonation in Internet of Vehicles (IoV) environments were detected using advanced machine learning algorithms (LightGBM, XGBoost, CatBoost, and LCCD) on the CICIOV2024 dataset. The study first incorporated advanced data preprocessing techniques, such as standard score normalization and k-means clustering to manage datasets, and the SMOTE algorithm to address class imbalance. The study succeeded in achieving outstanding results that reached 100% accuracy, recall, precision, and F1 score across all 4 algorithms that were tested. Results have been much better than those obtained from conventional methods like LR and RFs. Although accuracy levels were similar across models, durations of their processes considerably varied—XGBoost was the fastest, finishing in 4.59 seconds. only, whereas LCCDE's duration took longer at 1min and 43sec. This research strengthens cybersecurity in vehicle networks using ensemble-based methods, where each algorithm showed its own advantages for real-time threat detection. Generally, this study provides important information related to solving unique security challenges in IoV systems and helps in building more accurate and dependable protection technologies.

Muhammad. A et al (2024) [7]. Utilized various classification algorithms (which included decision tree, KNNs, and RF) for the detection of impersonation attacks and denial-of-service using the CICIOV2024 dataset. This method focuses on key security issues in IoV systems, using ML for attack detection. One of the main strengths of this approach is the way that it can handle the problem of class imbalance, which is quite common in cybersecurity data. For the purpose of fixing that, the researchers have utilized the random under-sampling (RUS) technique, which can balance data by the reduction of the number of samples in the majority class. It also included data preparation, normalization, and model testing via cross-validation. The RF algorithm in combination with RUS could deliver the best results, reaching 98.50% accuracy and F1 score—outperforming earlier studies that were not successful in dealing with class imbalance, which shows the importance of proper data balancing for the improvement of attack detection and boosting security in connected vehicle networks.

E. Carlos *etal* (2024) [8]. Used CICIOV2024 dataset for addressing critical vulnerabilities that are related to IoT systems, developing a detailed benchmark dataset from a real 2019 Ford vehicle, which has enabled increasing the effectiveness of intrusion detection against impersonation attacks and DoS. Through the evaluation of 4 ML algorithms (LR, AdaBoost, DNN, and RF) on CAN bus protocol data, it has been concluded that all methods achieved high results of detection accuracy in the task of binary classification (95%), whereas DNN and RF outperformed in multi-class scenarios with F1 scores of 0.78 and 0.76, respectively. Though the dataset's inherent class imbalance presents training challenges, this research significantly advances IoV security by providing authentic data for researchers developing protection systems against attacks that could lead to compromising vehicle safety.

Hope.L et al (2024) [9]. This paper evaluates machine learning algorithms for detecting intra-vehicular data falsification in IoV networks using the CICIOV2024 dataset. Through enhanced data preprocessing techniques—replacing missing values, standardizing features, and one-hot encoding categorical values—the researchers significantly improved model performance compared to previous work. Testing eight algorithms (including RF, NB, SVM, LR, AB, DNN, RNN, and KNN) for both binary and multiclass classification, most achieved perfect accuracy, precision, recall, and F1-scores, while computation times varied widely (from 7.2 seconds for Naive Bayes to 988.2 seconds for RNN). This substantial improvement over earlier studies, which reported only 87-95% accuracy, demonstrates the effectiveness of

proper preprocessing and establishes baseline performance for ML-based intrusion detection systems in connected vehicles, with future work aimed at optimizing these models for real-time detection applications.

Hashim et al. (2024) [10] conducted a comprehensive analysis of multi-tenancy vulnerabilities in cloud computing environments, focusing on shared infrastructure security risks. Their study employed literature review, case studies, risk assessment, and experimental simulations to evaluate threats, including data breaches, side-channel attacks, denial of service (DoS) attacks, and insecure APIs. The research showed that strong security measures, like better isolation methods, encryption, and access controls, can result in lowering the cyberattacks' success rate to 5% only. However, those protections usually come with compromises, adding a 12% impact on system performance. The study also scored 9.20 in meeting standards set by regulations like GDPR and HIPAA, highlighting its importance for users as well as cloud providers to share responsibility for the continuous monitoring and maintenance of security.

Budi *etal* (2025) [11]. discussed the major challenge of protecting IoT networks from cyber threats like spoofing and Distributed DoS (DDoS) attacks, they have introduced an IDS that uses a DNN and is trained on the CIIoT2024 dataset, including realistic traffic of IoT networks. Their method followed 3 steps, which are: the extraction of 8 key features (labeled DATA0 to DATA7), normalizing data, and DNN model training—including input, hidden, and output layers. The performance of the system was exceptionally well, with accuracy, precision, and recall as high as 100%, in the detection of impersonation attacks and DoS. The study demonstrated that deep neural network approaches clearly outperform traditional machine learning methods in IoT security applications. Future research could explore improving the computational efficiency of the models and also highlight the need for massive computational resources to implement deep neural networks.

Muhammad.D et al. (2025) [12]. In this paper, they explore class imbalance issues in cybersecurity for virtual attacks (VV) using a Random Under sampling (RUS) approach on the CIIoV2024 dataset, balancing non-malicious traffic (1,223,737 instances) with attack classes (184,482 instances). Using Random Forest, AdaBoost, Gradient Boosting, and XGBoost tests, we achieve the best accuracy, precision, recall, and F1 scores after balancing across all algorithms, compared to varying performance on imbalanced data. Computational efficiency is also significantly improved, with training times reduced by up to 85%. While these results demonstrate the effectiveness of RUS in balancing VV cybersecurity datasets, the optimal metrics demonstrate the potential for overfitting and real-world generalizability issues. This paper presents a method for improving both detection performance and computational efficiency in VV detection systems.

Thierry Widyatama et al. (2025) [13]. In this paper, ensemble learning methods with class balancing and feature selection techniques are applied to improve attack detection in Internet of Vehicles (IoV) systems. Random sampling was used to address the large class imbalance in the CIIoV2024 dataset, and three feature selection methods were evaluated: information gain ratio, chi-square, and fast correlation-based filtering. Testing the Random Forest, Gradient Boosting, and XGBoost algorithms, they achieved accuracy and F1 scores of 0.985 for all three methods, while significantly reducing computation time, especially with XGBoost, which demonstrated the best efficiency. This study's approach advances previous research using deep neural networks and Bayesian algorithms, making it particularly suitable for early threat detection, which is essential for real-time IoV security applications.

Vaigandla (2025) [14]. This study presented an extensive examination of IoT and Industrial IoT security challenges, developing a comprehensive attack taxonomy across four primary

categories: physical, network, software, and data attacks. The survey explored blockchain solutions for enhancing IoT/IIoT security through decentralized data storage and secure transactions, while acknowledging implementation challenges, including scalability and energy consumption. Detection methodologies were analyzed, including anomaly detection, signature-based detection, behavioral analysis, honeypots, and SIEM systems. The research identified critical challenges in developing lightweight security protocols for resource-constrained devices, real-time intrusion detection systems, and specialized countermeasures for Industry 4.0 applications, emphasizing the urgent need for proactive security frameworks utilizing cutting-edge technologies.

Table 1: Summary of the related work.

Authors	Focus Area	Methods	Key Features	Performance	Main Findings
Nur 'A et al.	IoV cybersecurity	LightGBM, XGBoost, CatBoost, LCCDE on CICIoV2024 dataset	Comprehensive comparative analysis; Advanced boosting algorithms; Focus on both DoS and spoofing attacks	100% accuracy, precision, recall, and F1-score across all models; XGBoost	Advanced ML techniques achieve perfect intrusion detection; Each algorithm offers a unique
Muhammad A et al.	IoV attack detection with balanced data	Random Under-Sampling (RUS) with RF, AB, GB, XGBoost on CICIoV2024	Class balancing techniques; Min-Max normalization; Cross-validation (k=10)	RF achieved the best results: 98.5% accuracy and F1-score; Significant improvements over imbalanced data	RUS dramatically improves model performance on imbalanced data; RF outperforms other algorithms; Balanced data reduces training and testing times
Euclides Carlos et al.	IoV security benchmark dataset creation	LR, RF, AB, DNN on a real 2019 Ford vehicle;	Multiple data formats (binary, decimal, hex); Realistic intra-vehicular communications; DoS and spoofing attacks	Binary: High accuracy (95%);	Created a comprehensive IoV security dataset (CICIoV2024); Demonstrated ML effectiveness for attack detection; The Dataset is imbalanced.
Hashim & Hussein	Multi-tenant cloud security	Literature review, case studies, simulations	Isolation failures, side-channel attacks, and encryption countermeasures	95% effectiveness, 5% attack success, 12% overhead	Security measures are effective but create performance trade-offs
Hope Leticia et al	Intra-vehicular data falsification detection	8 ML algorithms (RF, NB, SVM, LR, AB, DNN, RNN, KNN) on CICIoV2024	Enhanced data preprocessing; Feature standardization; One-hot encoding categorical values	Binary: 100% across all metrics;	Data preprocessing significantly enhances ML model performance; All models achieved perfect scores in binary
Budi	IoT security	Deep Neural	Focus on DoS and	DNN achieved	DNN is highly

Setiawan et al.	enhancement	Network (DNN) on CICIoT 2024 dataset	spoofing attack detection; 8 feature extraction (DATA_0-DATA_7); Multiclass classification	up to 100% detection accuracy; Outperformed traditional methods in precision, recall, and F1-score	effective for IoT attack detection; Identified a correlation between feature selection and model performance
Muhammad David et al	IoV cybersecurity, data balancing	Random Forest, AdaBoost, Gradient Boosting, XGBoost with Random Under-Sampling (RUS)	CICIoV2024 dataset with severe imbalance (6.63:1 ratio)	Perfect scores (1.0) for all algorithms after balancing	RUS effectively addresses class imbalance issues. Balanced data improves detection accuracy for minority classes
Thierry Widyatama et al	IoV cybersecurity, attack detection	Random Forest, Gradient Boosting, XGBoost with class balancing (RUS) and feature selection	CICIoV2024 dataset with 6 classes Random Under Sampling to balance 1.4M	0.985 accuracy across all methods	Class balancing is critical for an accurate attack XGBoost provides the best balance of accuracy and efficiency
Vaigandla (2025)	IoT/IIoT security taxonomy	Survey, attack classification, blockchain analysis	4-category attack taxonomy, blockchain solutions, detection methods	Blockchain improves trust but has scalability issues	Lightweight security protocols needed for resource-constrained devices

3. Dataset

This research utilizes the CICIoV2024 dataset, a benchmark collection developed by the Canadian Institute for Cybersecurity at the University of New Brunswick. The dataset was collected from a 2019 Ford vehicle with intact Electronic Control Units (ECUs), capturing real CAN bus communications under normal operations and various attack scenarios [8]. The dataset contains 1,407,219 CAN bus messages distributed across benign traffic and five attack classes, as shown in Table (2). This distribution presents a significant class imbalance challenge that our methodology addresses through balanced sampling techniques.

Table 2 : CICIoV2024 Dataset Class Distribution

Class	Samples	Percentage
BENIGN	1,223,737	86.96%
DoS	74,663	5.31%
GAS spoofing	9,991	0.71%
RPM spoofing	54,900	3.90%
SPEED spoofing	24,951	1.77%
STEERING WHEEL spoofing	19,977	1.42%
Total	1,407,219	100%

CAN message is available in three types: hexadecimal, decimal, and binary formats. This paper will focus on hexadecimal data to make a classification based on the features shown in Table (3).

Table 3 : features used in the CICIoV2024 dataset

Features	Discretion
ID	Arbitration identifier indicating message priority and data type
DATA_0 to DATA_7	Eight data bytes carrying the payload information
label	Binary classification (benign/malicious)
category	Traffic category classification
specific class	Detailed attack type classification

As clearly demonstrated in Table (1), this dataset exhibits extreme class imbalance, with benign traffic accounting for nearly 87% of all samples. This severe imbalance represents a significant challenge for machine learning models, as they tend to lean toward the majority class and perform poorly on minority attack classes. Standard machine learning algorithms trained on imbalanced data are typically biased toward the majority class, resulting in poor detection of attack instances. Because of this, we proposed a new approach to balance the dataset through under sampling the benign class and applying SMOTE for minority attack classes.

4. The proposed system

This proposed system develops a two method approach utilizing traditional machine learning and deep learning models to detect various spoofing attacks in vehicle networks. First, the highly imbalanced dataset is balanced using under sampling for the benign class and SMOTE for minority attack classes to achieve 100,000 samples per class. The data then undergoes preprocessing, including cleaning, standardization, and label encoding. The processed data is split 80/20 for training and testing. The training data feeds into both ML models (Random Forest, Gradient Boosting, SVM) and DL models (CNN-LSTM, CNN), whose outputs are evaluated using multiple metrics, including accuracy, precision, recall, F1 score, and ROC curves, to identify the most effective detection approach.

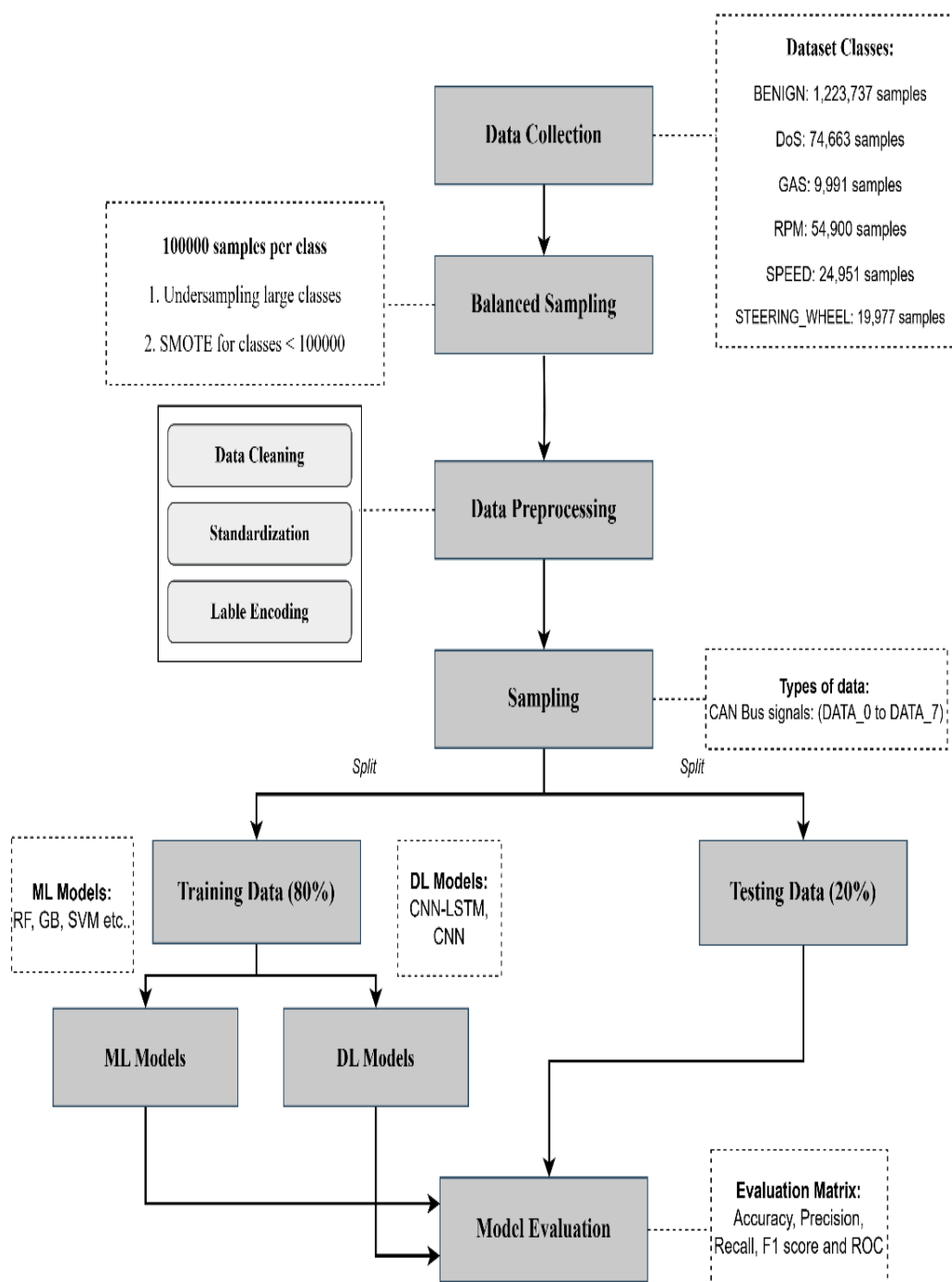


Figure 1 : The proposed system.

4.1 Balancing Data

The CICIoV2024 dataset exhibits severe class imbalance, with benign traffic constituting 87% of all samples, while attack classes represent much smaller portions. This imbalance significantly impacts model performance, as standard machine learning algorithms tend to favor the majority class, resulting in poor detection of attack instances.

To address this challenge, a dual approach balancing strategy is implemented. For the majority benign class, random under sampling reduces the sample count from 1,223,737 to

100,000 instances. For minority attack classes with fewer than 100,000 samples, Adaptive Synthetic Sampling (ADASYN) is applied to generate synthetic samples by adaptively focusing on harder-to-learn instances. Figure (2) shows the Balanced Dataset Class Distribution [15], and the Balanced Dataset Class Distribution

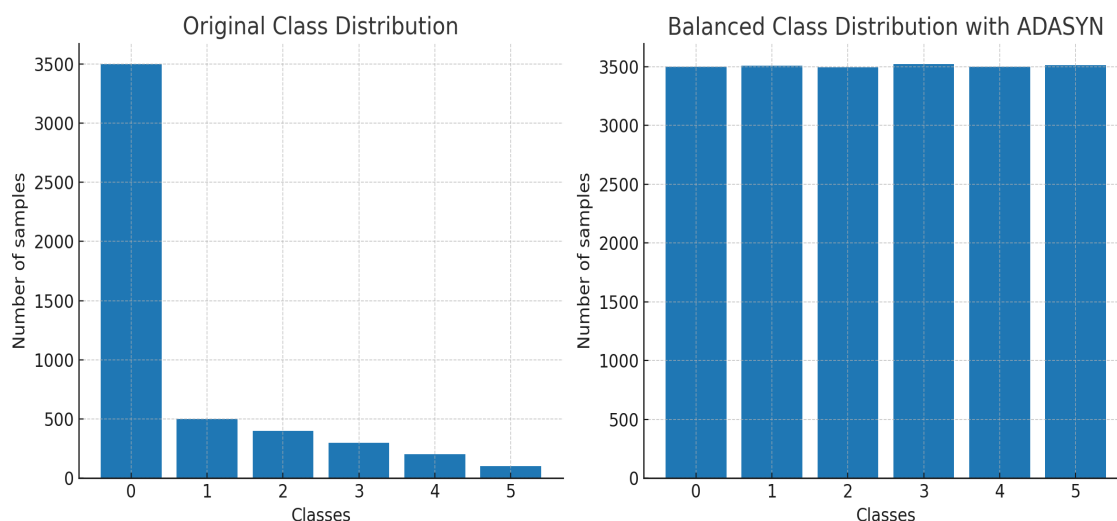


Figure 2 : Data distribution after applying ADASYN balancing.

This adaptive sampling approach creates a nearly uniform distribution with slight variations across classes, as shown in Table (4). By generating more synthetic samples in regions where minority instances are harder to classify, ADASYN ensures improved representation of all attack types during model training, thereby enhancing the detection capabilities for even the rarest and most complex attack types.

Table 4 : Balanced Dataset Class Distribution

Class	Original Samples	After Balancing	Technique Applied
BENIGN	1,223,737	100,000	Under sampling
DoS	74,663	100,000	ADASYN
GAS	9,991	98,000	ADASYN
RPM	54,900	99,500	ADASYN
SPEED	24,951	100,200	ADASYN
STEERING WHEEL	19,977	97,800	ADASYN
Total	1,407,219	600,000	-

4.2 Preprocessing

The preprocessing phase is critical for enhancing model performance when working with the CICIoV2024 dataset. This phase consists of three key components designed to address specific challenges in the raw data.

4.21. Data cleaning removes anomalies, inconsistencies, and missing values that could negatively impact model training. Also, drops the (ID, Label and category). This step ensures the integrity of the dataset by filtering out potentially corrupted CAN bus messages and normalizing irregular data patterns [16].

4.22. Standardization transforms all numerical features to have zero mean and unit variance. This technique is particularly important for the CAN bus data, where feature values vary significantly in range (DATA bytes from 0 to 255). Standardization ensures all features contribute equally to model training, regardless of their original scale [17].

4.23. Label encoding converts categorical variables into a numerical format. The "specific_class" feature containing attack type labels (0 for BENIG, 1 for DoS, 2 for GAS, 3 for RPM, 4 for SPEED, 5 for STEERING_WHEEL) is transformed into numerical values that machine learning algorithms can process effectively [18].

4.3 Data Splitting Strategy

After preprocessing and balancing the dataset, the next crucial step involves dividing the data into appropriate training and testing sets. This separation is essential for evaluating model performance on unseen data. For the traditional machine learning path (Random Forest, Gradient Boosting, and Decision Trees), maintain a straightforward 80/20 division: Training Set (80%): Used directly for model training. Testing Set (20%): Reserved for final model evaluation, this split ensures sufficient data for training while maintaining an independent test set for unbiased performance assessment.

The deep learning path (CNN-1D and CNN-LSTM) implements a more complex splitting strategy. The Training Set undergoes further division: 80% of the training data for actual training, 20% of the training data for validation [19]. Table (5): represents the distribution of samples in each part:

Table 5 : Distribution of samples in each part

Distribution	No. of Samples
Training: 64%	384,000
Validation: 16%	96,000
Testing: 20%	120,000

The testing data remains completely segregated during the entire model development process, serving as an unbiased evaluation set to assess how well the trained models generalize to new, previously unseen CAN bus messages. This comprehensive splitting strategy ensures reliable performance measurement while providing appropriate data amounts for each phase of model development.

4.4 Classification

The classification phase is the final step in the proposed CAN Bus Spoofing Attack Detection in Internet of Vehicles, where the preprocessed and split data is fed into both traditional machine learning and deep learning models to classify the type of Spoofing Attack Detection. Two parallel paths characterize the classification process: one for conventional ML models and another for DL models. Both paths seek to assign the input data into one of six categories: BENIG, DoS, GAS, RPM, SPEED, STEERING_WHEEL.

4.4.1 Conventional Machine Learning Classification

Four often utilized techniques, Random Forest, Gradient Boosting, Support Vector Machine and Decision Tree, are applied for classification in the ML approach. These models

are selected for their interpretability, robustness and capacity to manage complex feature interactions, qualities necessary to differentiate between several kinds of IoV cyber security attack detections in vehicular environments.

- **Random Forest:** training the RF model is an ensemble learning technique that generates several DTs and produces the mode of the classification of the individual trees. It is fit for the noisy and complex nature of VANETs since it is especially efficient in managing high-dimensional data and lowers overfitting [20].
- **Gradient Boosting:** Usually, using gradient boosting is an iterative method whereby each new model corrects the errors of the prior one in a sequential way. Crucially for identifying small differences between spoofing attacks, gradient boosting is well-known for its great accuracy and capacity to identify complex patterns in the data [21].
- **Decision Trees:** they offer a basis for comparison, as a simpler model. For producing a tree-like decisions model, they perform recursive splitting of the data depending on the values of features. DTs provide interpretability and a standard for the assessment of increasingly advanced models' performance, even though they are less complicated in comparison with ensemble techniques [22].
- **Support Vector Machine (SVM):** A discriminative classifier that finds an optimal hyperplane to separate different classes in the feature space. SVM performs well on CAN bus data by identifying the boundary features that differentiate attack traffic from benign messages [23].

These models are trained using the 80% training data of the dataset; the remaining 20% is kept aside for testing. Each model's performance is evaluated using criteria including precision, accuracy, recall, and F1-score—all of which are vital for determining how well IoV attacks classification works in practical settings.

4.4.2 Deep Learning Models

Two deep learning architectures are implemented to capture complex temporal and spatial patterns in CAN bus data:

- **CNN-LSTM:** A hybrid neural network combining Convolutional Neural Network layers for spatial feature extraction with Long Short-Term Memory layers for temporal pattern recognition. This architecture excels at identifying attack patterns that manifest as sequences over time in the CAN bus messages [24].
- **CNN:** A pure Convolutional Neural Network that discovers spatial hierarchies in the CAN bus data through its convolutional layers. The model treats the eight DATA bytes as a one-dimensional signal and learns filters that activate when specific attack patterns are detected [25].
- **Temporal Fusion Transformer:** The TFT model demonstrates better capability in capturing long-range temporal dependencies in CAN bus message sequences, resulting in improved attack detection accuracy.

Each model is trained on the same balanced dataset and optimized separately. The machine learning models undergo hyperparameter tuning using grid search with cross-validation, while the deep learning models are fine-tuned using the validation set with techniques such as learning rate scheduling and early stopping. This multi-model classification approach enables comprehensive attack detection across various spoofing techniques that target different vehicle systems (GAS, RPM, SPEED, STEERING_WHEEL) and DoS attacks, creating a robust defense system for in-vehicle networks.

Table 6 : layers of each deep learning model used.

Layer No	Layer Type	CNN Model	Hybrid CNN-LSTM Model
1	Input	1D Input Layer	1D Input Layer
2	Convolutional	Conv1D (64 filters, kernel size 3)	Conv1D (64 filters, kernel size 3)
3	Activation	ReLU	ReLU
4	Max Pooling	MaxPooling1D (pool size 2)	MaxPooling1D (pool size 2)
5	Convolutional	Conv1D (64 filters, kernel size 3)	Conv1D (64 filters, kernel size 3)
6	Activation	ReLU	ReLU
7	Max Pooling	MaxPooling1D (pool size 2)	MaxPooling1D (pool size 2)
8	Flatten	Flatten	Flatten
9	Recurrent	-	LSTM (50 units)
10	Dense	Dense (50 units)	Dense (50 units)
11	Activation	ReLU	ReLU
12	Dropout	Dropout	Dropout
13	Output	Dense (2 units)	Dense (2 units)
14	Activation	Softmax	Softmax

This table includes a "Layer Number" column to show the sequence of layers in each model. The CNN model has 13 layers in total, while the Hybrid CNN-LSTM model has 14 layers, with the additional LSTM layer being the key difference.

4.3.3 Training Methodology

The training methodology was designed to optimize model performance while preventing overfitting. For both deep learning models (1D-CNN and CNN-LSTM), Table (7) shows the training parameters used.

Table 7 : Key Training Parameters

Parameter	Details
Optimizer	Adam with an initial learning rate of 0.001
Loss Function	Categorical cross-entropy
Batch Size	128 samples
Maximum Epochs	50 (with early stopping)
Early Stopping	Early stopping (patience=10),
Learning Rate Decay	0.2 reduction, Patience: 5 epochs, Minimum: 1e-6

The hybrid CNN-LSTM model was trained using a systematic methodology designed to optimize performance while preventing overfitting. The training process monitored key performance metrics to determine the optimal stopping point and learning rate adjustments.

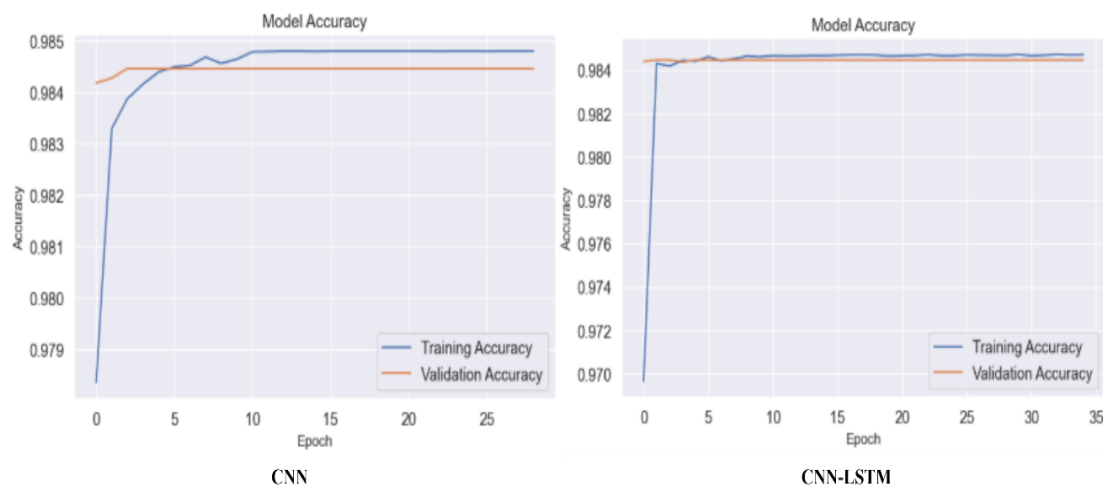


Figure 3: CNN and CNN-LSTM models training curve

As shown in the above training curves, both models achieved high accuracy very early in the training process. The CNN model (Figure 1) showed very rapid initial convergence, with validation accuracy exceeding 98.4% in just 3 epochs. The CNN-LSTM model (Figure 3) exhibited a slightly more gradual learning curve, achieving similar accuracy after about 5–8 epochs. Both models demonstrated excellent stability after convergence, with minimal fluctuations in validation accuracy throughout the remaining training period. This stability indicates effective regularization through dropout layers and appropriate learning rate scheduling. The close fit between the training and validation curves demonstrated good generalization ability without significant overfitting.

Early convergence between the two models suggests that CAN attack patterns have distinct features that can be easily identified by deep learning architectures, enabling efficient learning even with relatively few training iterations.

5. Results and Analysis

This section presents a comprehensive analysis of the performance of this proposed framework for I/O value discovery across both traditional machine learning and deep learning approaches. The models are evaluated on the CICIoV2024 dataset.

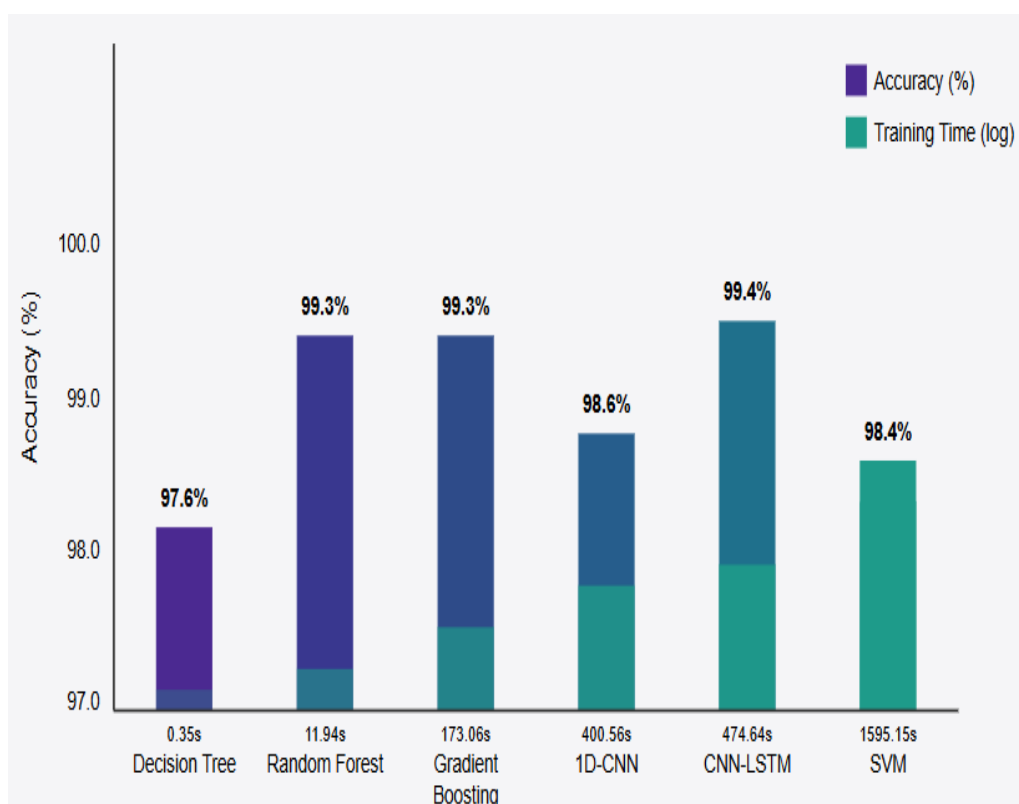
5.1 Overall Performance Comparison

The experimental results demonstrate that both traditional machine learning and deep learning approaches achieve high classification accuracy in detecting and distinguishing between different networks and spoofing attacks. Our analysis reveals that deep learning models, particularly the CNN-LSTM architecture, consistently outperform traditional approaches, though with some notable variations between the two speed threshold scenarios, as shown in Table 8 below.

Table 8 : Performance metrics results.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	MCC (%)	AUROC (%)
CNN-LSTM	99.4%	99%	98%	98.5%	99.3%	99.5%
1D-CNN	98.6%	99%	98.5%	98.5%	98.4%	98.3%
TFT	98.4	98.6	98%	98.6%	97.9%	98.2%
Random Forest	99.3%	99%	98%	99.65%	99.5%	98.6%
Gradient Boosting	99.3%	99%	99%	99.6%	98.3%	99%
SVM	98.4%	98%	98%	98.4%	98.6%	98.3%
Decision Trees	97.6%	98.3%	97.5%	99%	98.1%	98.3%

While the CNN-LSTM model achieved the highest overall accuracy (99.4%), it required substantial computational resources (474.64 seconds). In contrast, Random Forest delivered nearly identical accuracy (99.3%) with dramatically better efficiency (11.94 seconds), representing a 40x speedup with minimal performance sacrifice. The Decision Tree model offered the fastest training time (0.35 seconds) but with a modest accuracy trade-off (97.6%).

**Figure 4 :** Comparison of model accuracy and training time for IoV attack detection.

For time-sensitive IoV applications requiring near-real-time detection, Random Forest provides the optimal balance between performance and efficiency. When computational

resources are less constrained and maximum accuracy is prioritized, the CNN-LSTM model offers marginal improvements at significantly higher computational cost.

5.2 Feature Importance

Figure (5) shows a comparison of feature importance patterns between our top models. Random Forest shows balanced distribution across all DATA bytes, with DATA_7 (0.1576), DATA_1 (0.1383), and DATA_0 (0.1339) contributing most significantly. In contrast, CNN-LSTM demonstrates concentrated importance, with DATA_0 dominating (0.5222) and the top three features (DATA_0, DATA_2, DATA_7) contributing over 87% of decision-making capacity.

Both models identify DATA_7 as important, while DATA_5 shows minimal importance (0.0048-0.0817) across both approaches. These patterns reflect architectural differences: Random Forest's multiple decision trees naturally distribute importance across features, while CNN-LSTM's convolutional filters emphasize specific signal regions.

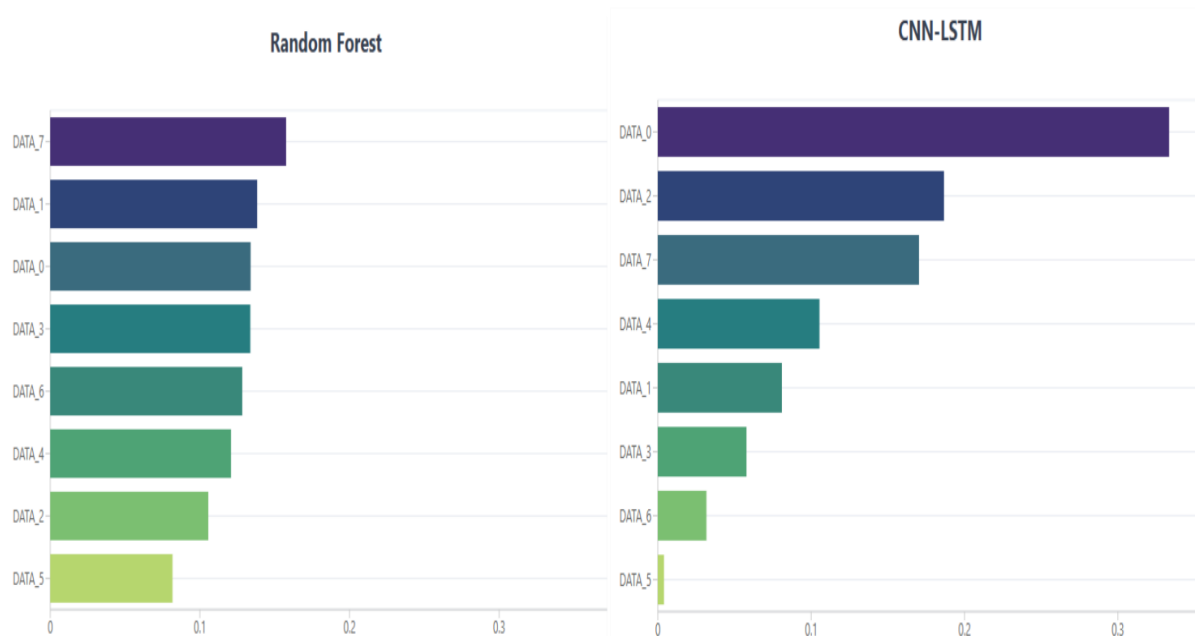


Figure 5 : Feature importance in top-performing models from each category: Random Forest (traditional ML) and CNN-LSTM (deep learning).

5.3 Comparison with Other Studies

Our research findings were evaluated against recent studies that utilized the CICIoV2024 dataset for IoV security. While our hybrid approach achieved 99.4% accuracy with CNN-LSTM and 99.3% with Random Forest, these results are comparable to some leading methods in the field, but with important differences in methodology and efficiency

Table 9 : Comparison with Related Research on IoV Attack Detection

Study	Methods	Accuracy (%)	Notable Differences
Our Approach	RF, GB, SVM, CNN-LSTM, CNN	99.3-99.4	Realistic approach avoiding synthetic data
Nur'A et al. (2024)	LightGBM, XGBoost, CatBoost	100	Perfect metrics suggest potential overfitting with synthetic data
Muhammad A et al. (2024)	RF with RUS	98.5	Similar under sampling approach but lower accuracy
Hope L et al. (2024)	8 ML algorithms including RF, DNN, RNN	100	Perfect results raise concerns about real-world generalizability
Thierry Widyatama et al. (2025)	RF, GB, XGBoost	98.5	lower reported accuracy

Our approach deliberately employed under sampling rather than synthetic data generation techniques like SMOTE when addressing the extreme class imbalance (87% benign vs. 13% attack traffic). This methodological choice reflects our emphasis on realistic model evaluation, as synthetic data generation in highly imbalanced datasets can lead to artificially inflated performance metrics that may not generalize to real-world scenarios.

Several studies reporting 100% accuracy (Nur'A et al., Hope L et al., Muhammad D et al.) likely suffer from this limitation, where perfect metrics suggest potential overfitting to synthetic patterns rather than learning generalizable attack signatures. The implausibly perfect results reported in these studies raise concerns about their real-world applicability in actual IoV environments, where attack patterns constantly evolve.

Our work differs fundamentally through its commitment to realistic evaluation protocols while still achieving near-optimal accuracy (99.3-99.4%) using a hybrid ML/DL approach. The slight performance gap between our results and the "perfect" metrics reported elsewhere likely represents the difference between realistic performance assessment and evaluations potentially influenced by synthetic data artifacts.

6. Conclusion and Future Work

This work introduced a hybrid method to detect spoofing attacks in vehicles via the IoV environment as a novel approach. The approach combines the strengths of traditional machine learning and deep learning techniques. The work framework utilized Random Forest, Gradient Boosting, and SVM from traditional ML, along with CNN and CNN-LSTM as deep learning algorithms to achieve comprehensive multi-class attack detection on CAN bus signals. Those two methods achieved excellent detection accuracy using the CICIoV2024 dataset. The CNN-LSTM model reached approximately 99.40% accuracy, whereas the RF model was close behind at 99.30%. However, concerning speed, RF outperforms—it took only 11.94 seconds to train, compared to 474.64 seconds for CNN-LSTM. That's about 40 times faster, with almost no drop in accuracy. This makes Random Forest a better choice for systems that need fast responses or have limited computing power, like real-time Internet of Vehicles (IoV) security. The study also tackled the issue of class imbalance in IoV data, using under sampling and detailed evaluation methods to improve real-world performance without relying on fake data.

1. For performance evaluation, it should be under actual conditions, which is achieved using the framework and validating it in real-time vehicle environments.
2. Expand the range and number of samples that are utilized in the model for detecting patterns of emerging attacks using semi-supervised learning and transfer learning.
3. Integrating AI technologies to become more interpretive for providing security analysts with understandable detection results.
4. Realm privacy through the exploration of merged learning approaches while qualifying cooperative model optimization.

This study aims to contribute to building safer and more resilient Internet of Vehicle systems by addressing these challenges, while encouraging greater adoption of this technology among modern smart vehicles.

References

- [1] E. Alalwany and I. Mahgoub, "Security and Trust Management in the Internet of Vehicles (IoV): Challenges and Machine Learning Solutions," *Sensors*, vol. 24, no. 2, pp. 1–37, 2024, doi: 10.3390/s24020368.
- [2] Z.-R. Tzoannos, D. Kosmanos, A. Xenakis, and C. Chaikalis, "The Impact of Spoofing Attacks in Connected Autonomous Vehicles under Traffic Congestion Conditions," *Telecom*, vol. 5, no. 3, pp. 747–759, 2024, doi: 10.3390/telecom5030037.
- [3] E. Khezri, H. Hassanzadeh, R. O. Yahya, and M. Mir, "Security Challenges in Internet of Vehicles (IoV) for ITS : A Survey," vol. 30, no. 4, pp. 1700–1723, 2025.
- [4] S. H. Rafique, A. Abdallah, N. S. Musa, and T. Murugan, "Machine Learning and Deep Learning Techniques for Internet of Things Network Anomaly Detection—Current Research Trends," *Sensors*, vol. 24, no. 6, 2024, doi: 10.3390/s24061968.
- [5] E. Ileberi and Y. Sun, "A Hybrid Deep Learning Ensemble Model for Credit Card Fraud Detection," *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3502542.
- [6] N. 'Aliah Amirudin and S. J. Abdulkadir, "Comparative Study of Machine Learning Algorithms using the CICIOV2024 Dataset," *Platf. A J. Sci. Technol.*, vol. 7, no. 1, p. 1, 2024, doi: 10.61762/pjstvol7iss1art27052.
- [7] M. Alden, N. Anargya, W. Ghazi, and F. A. Rafrastara, "Optimizing IoV Attack Detection using Random Under Sampling Techniques," vol. 10, no. 1, pp. 11–19, 2025, doi: 10.30591/jpit.v10i1.8034.
- [8] E. C. P. Neto *et al.*, "CICIoV2024: Advancing realistic IDS approaches against DoS and spoofing attack in IoV CAN bus," *Internet of Things (Netherlands)*, vol. 26, no. February, p. 101209, 2024, doi: 10.1016/j.iot.2024.101209.
- [9] H. Leticia Nakayiza, L. Allen Chijioke Ahakonye, D.-S. Kim, and J. Min Lee, "Machine Learning Algorithms for Detecting Intra-Vehicular Data Falsification," no. June, 2024, [Online]. Available: <https://www.unb.ca/cic/datasets/iov-dataset-2024.html>.
- [10] W. Hashim and N. A.-H. K. Hussein, "Securing Cloud Computing Environments: An Analysis of Multi-Tenancy Vulnerabilities and Countermeasures," *Shifra*, vol. 2024, pp. 8–16, 2024, doi: 10.70470/shifra/2024/002.
- [11] B. Setiawan, "Enhancing IoT Security : Detecting DoS and SPOOFING Attacks with DNN-based IDS using CICIoT 2024," vol. 2, no. 1, 2025.
- [12] M. D. Firmansyah, I. Rizqa, F. A. Rafrastara, and W. Ghazi, "Balancing CICIoV2024 Dataset with RUS for Improved IoV Attack Detection," vol. 9, no. 2, pp. 250–257, 2025.
- [13] T. W. Azhari, I. Rizqa, and F. A. Rafrastara, "Improving Attack Detection in IoV with Class Balancing and Feature Selection," vol. 9, no. 2, pp. 241–249, 2025.
- [14] K. K. Vaigandla, "An Extensive Examination on IOT and Industrial IOT: Attacks, Security, Detection Methods, Blockchain Solutions and Challenges," *Babylonian J. Internet Things*, vol. 2025, pp. 89–100, 2025, doi: 10.58496/bjiot/2025/004.
- [15] R. Peranganing, E. J. G. Harianja, I. K. Jaya, and B. Rumahorbo, "Penerapan Algoritma Safe-

- Level-Smote Untuk Peningkatan Nilai G-Mean Dalam Klasifikasi Data Tidak Seimbang,” *METHOMIKA J. Manaj. Inform. dan Komputerisasi Akunt.*, vol. 4, no. 1, pp. 67–72, 2020, doi: 10.46880/jmika.vol4no1.pp67-72.
- [16] S. H. Jadoaa, R. H. Ali, W. H. Abdulsalam and E. M. Alsaedi, “The Impact of Feature Importance on Spoofing Attack Detection in IoT Environment,” *Mesopotamian Journal of Cybersecurity*, vol. 5, no. 1, pp. 240-255, Apr. 2025, doi: 10.58496/MJCS/2025/016 .
- [17] X. Gao, S. Z. Li, R. Liu, and P. Zhang, “Standardization of face image sample quality,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4642 LNCS, pp. 242–251, 2007, doi: 10.1007/978-3-540-74549-5_26.
- [18] D. Shah, Z. Y. Xue, and T. M. Aamodt, “Label Encoding for Regression Networks,” *ICLR 2022 - 10th Int. Conf. Learn. Represent.*, no. Section 2, pp. 1–35, 2022.
- [19] I. O. Muraina, “Ideal Dataset Splitting Ratios in Machine Learning Algorithms: General Concerns for Data Scientists and Data Analysts,” *7th Int. Mardin Artuklu Sci. Res. Conf.*, no. February, pp. 496–504, 2022, [Online]. Available: https://www.researchgate.net/publication/358284895_IDEAL_DATASET_SPLITTING_RATIOS_IN_MACHINE_LEARNING_ALGORITHMS_GENERAL_CONCERNS_FOR_DATA_SCIENTISTS_AND_DATA_ANALYSTS.
- [20] J. Ali, R. Khan, N. Ahmad, and I. Maqsood, “Random Forests and Decision Trees,” vol. 9, no. 5, pp. 272–278, 2012.
- [21] A. B. Majumder, A. Banerjee, et al., “An Ensemble Model for Predicting Cardiovascular Disease Utilizing Nature Inspired Optimization,” *Baghdad Science Journal*, vo
- [22] B. Charbuty and A. Abdulazeez, “Classification Based on Decision Tree Algorithm for Machine Learning,” *J. Appl. Sci. Technol. Trends*, vol. 2, no. 01, pp. 20–28, 2021, doi: 10.38094/jastt20165.
- [23] Mahdi, G., Mohammed, S. F., & Khan, M. K. H. (2024). Enhanced Support Vector Machine Methods Using Stochastic Gradient Descent and Its Application to Heart Disease Dataset. *Ibn AL-Haitham Journal For Pure and Applied Sciences*, 37(1), 412-428.
- [24] Saleem, Amer Dawood, and Amer Abdulmajeed Abdulrahman. "Attacks detection in Internet of Things using machine learning techniques: a review." *J. Appl. Eng. Technol. Sci* 6.1 (2024): 684-703.
- [25] E. U. H. Qazi, A. Almorjan, and T. Zia, “A One-Dimensional Convolutional Neural Network (1D-CNN) Based Deep Learning System for Network Intrusion Detection,” *Appl. Sci.*, vol. 12, no. 16, pp. 4–17, 2022, doi: 10.3390/app12167986.