WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

System for Detecting Intrusions in the Internet of Vehicles using Deep Learning

Mrs. Vibha MP & Prof. Dilna PM

Department of Computer Science and Engineering
Government Engineering College, Wayanad, Kerala, India

ABSTRACT—

Modern vehicles, including both connected and autonomous types, are progressively integrating with external networks, offering a wide array of services and functionalities. But this increased connectedness also makes the Internet of Vehicles (IoV) more open to cyberattacks and more vulnerable. Systems for intrusion detection (IDSs) are essential for protecting sophisticated automobile systems from network attacks because automotive networks lack authentication and encryption.

This paper introduces two IDS that leverages ensemble learn- ing and transfer learning, incorporating convolutional neural networks (CNNs) and hyper parameter optimization techniques for IoV systems. and a hybrid model consisits of CNN and LSTM. My experiments demonstrate that the proposed ensemble IDS achieves detection rates and F1-scores exceeding 99.94% on the widely recognized public benchmark IoV security dataset, CICIDS2017, and Proposed hybrid model achieves detection rates and F1-scores exceeding 99.97% on the same dataset. This highlights the effectiveness of the proposed IDS in identifying cyber attacks within external vehicular networks.

I. Introduction

As internet of automotive technologies rapidly evolve, no- table advancements have been made in the development of vehicles operated by a network, such as networked and self- driving cars (CVs) [1].External vehicular networks enable interactions between intelligent vehicles and other components of the IoV ecosystem, such as road users, infrastructure, and roadside units [2].

Yet, the heightened connectivity and accessibility of au- tomotive networks have expanded the potential avenues for cyber attacks targeting advanced vehicles. Moreover, because of the constrained length of CAN packets, encryption or au- thentication methods are not employed, leaving these packets susceptible to cyber attacks. The absence of essential security protocols enables cyber attackers to insert harmful messages into Vehicle Networks, enabling them to perpetrate various attacks, such as fuzzing,BOT and other type of web attacks etc. Moreover,the newly formed cellular connections linking linked automobiles with outside networks expose these type of vehicles to traditional internet attacks. [2].Hence, it is imperative to develop system for detecting intrusions (IDSs) to safeguard Internet of Vehicle system and smart automotives by detecting and mitigating internet threats as in Figure 1.

Lately, the progress in Deep and Machine Learning (DL and ML) methods [3] has attracted considerable interest from researchers and automotive manufacturers due to their potential applications in vehicle systems and cybersecurity. Machine Learning (ML) and Deep Learning (DL) approaches are mainly employed to construct IDSs using the classification algorithms adept at discerning between typical internet activity and diverse internet threats by analyzing vehicular data [4] [5]. In this study, I introduce a sophisticated IDS model founded

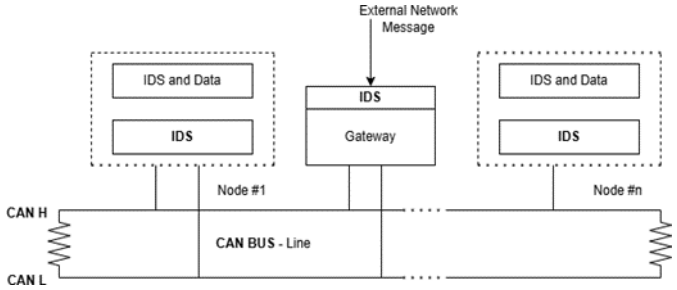

Fig. 1. Architecture of IDs protection in external vehicular network

on fine-tuned CNN(Convolutional Neural Networks), ensem- ble learning method, and transfer learning approaches, aimed at safeguarding Internet of Vehicle systems. I utilize seven sophisticated CNN models—Xception, Inception, VGG16, VGG19, Lenet, EfficientNet, and InceptionResNet—to train foundational learners using vehicle network packets. The hy- perparameters of the Convolutional neural networks moddels undergo optimization via the Particle Swarm Optimization (PSO) method, resulting in refined learning models. Concate- nation is the ensemble strategy that is used to combine the core CNN models to improve the accuracy of intrusion detection. Here suggested IDS framework's efficiency is assessed using the publicly available vehicle network dataset, namely the CICIDS2017 dataset [6]. Secondly a CNN and LSTM hybrid model for Intrusion Detection in the Internet of Vehicles (IoV) leverages the strengths of both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to enhance security. The CNN component efficiently captures spatial features and local patterns in the network traffic data, such as anomalies in packet structures, through convolution and pooling operations. Subsequently, the LSTM component processes these extracted features to learn temporal dependen- cies and sequential patterns, crucial for detecting sophisticated attacks that unfold over time. This hybrid architecture effec- tively combines the local feature extraction power of CNNs with the temporal pattern recognition capabilities of LSTMs, resulting in a robust model capable of identifying a wide range of intrusion types in IoV environments, thereby ensuring more secure and reliable vehicular networks. Here suggested IDS framework's efficiency is assessed using the same vehicle network dataset, namely the CICIDS2017 dataset.

This project primarily contributes in the following ways:

- Framework for Cyber-Attack Detection: It introduces an efficient framework for detecting cyber-attacks in external networks using Convolutional Neural Networks (CNN), ensemble and transfer learning, and approch of Hyper Parameter Optimization (HPO) techniques also.
- Data Conversion Technique: It suggests a cutting-edge technique for converting car network traffic data into pic- tures that makes it easier to distinguish between different types of cyberattacks.
- Benchmark Evaluation: It assesses the suggested ap- proach by contrasting the model's performance with other cutting-edge techniques and using a benchmark cybersecurity dataset that simulates data from external networks.
- The CNN-LSTM hybrid model for Intrusion Detection in the Internet of Vehicles (IoV) Contributes significantly to enhancing security by leveraging the combined strengths of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks.

To the extent of my understanding, no prior research has suggested an intrusion detection system with this level of optimization that combines ensemble learning, CNN, transfer learning, and HPO approaches to efficiently identify differ- ent kinds of attacks on external networks. The document's remaining parts are organized as follows: Relevant research on DL(Deep Learning) and ML(Machine Learning) approches for automotive network intrusion detection is covered in the part 2. In part 3, along with the CNN network, ensemble learning, transformation of data, learning by transfer learning and HPO techniques, the proposed deep learning-based framework is given. The fourth part discusses the experiment's results.

II. Related work

Tasks related to internet of vehicle intrusion detection have made extensive use of DL and ML models. Using Multi-Layer Perceptron (MLP), Rosay et al. [2] propose a deep learning-based IDS for vehicular networks. Using the CICIDS2017 dataset, the MLP model was assessed on an automobile microprocessor. In an Internet of Vehicles (IoV) context, Yang et al. [4] [7] tree-based stacking technique for network traffic analysis. Our suggested stacking approach performs exceptionally well on the CICIDS2017 and Internet of Vehicles datasets.Convolutional neural networks-based IDS development for automotive networks is the subject of several published works. A DCNN (deep CNN) based Intravehic- ular Defense System (IDS) model with reduced Inception Resnet was proposed by Song et al. [5] to identify intrusions in intravehicular networks. Although the Deep CNN model performs more accurately in IoV cyberattack detection tasks, performance still has to be much improved. The intention of The suggested IDS calls for building an ideal IDS framework out of cutting-edge Convolutional Neural Network models that have been hyperparameter tuned and ensemble learning techniques applied. Additionally, transfer learning techniques are employed to increase the efficiency of model training.

III. PROPOSED FRAMEWORK

A. System Overview

The goal of this work is to create an intrusion detection system (IDS) that can identify and stop different kinds of attacks on external vehicular networks. Fig. 2 depicts the architecture of an IDS-protected vehicle as well as a typical threat scenario. Via the On-Board Diagnostics II (OBD II) interface, cybercriminals can initiate internal threats against IVNs and external assaults against external vehicular networks by delivering malicious traffic packets through wireless inter- faces. Deploying the suggested IDS in external networks is therefore necessary.The suggested intrusion detection system (IDS) can be integrated into the external network gateways to detect and stop any malicious packets that try to breach the cars. [3].

This research proposes an IDS based on transfer learning and optimized convolutional neural networks to detect differ- ent kinds of attacks in Internet of Vehicles (IoV) systems. An overview of the suggested IDS framework is shown in Figure

2. First, the quantile transform method is used to convert the time-based chunks of external network data into images. Next, seven cutting-edge CNN models (Efficientnet, VGG16, VGG19, Xception, Inception, Lenet, and InceptionResnet) are trained on the created image set to create basic learners. Particle Swarm Optimization (PSO), an HPO technique that can automatically adjust the hyper-parameters, is used to optimize the CNN models.After

that, the three base CNN models that perform the best are chosen to build the ensemble learning models. Concatenation is the last ensemble technique that is utilized to build the ensemble model for final detection.

The second methodology of the hybrid model involves con- structing a hybrid neural network using Keras' Sequential API in Figure 3. It begins with three convolutional layers (Conv2D) with 32, 64, and 128 filters respectively, each followed by max pooling (MaxPooling2D) and dropout (Dropout) layers to downsample the input and prevent overfitting. The flattened output is reshaped to match the input requirements of an LSTM layer (Reshape((-1, 128))), where -1 denotes the batch size, and 128 is the number of features. This reshaped data is processed by an LSTM layer (LSTM(128)) to capture temporal dependencies. The model includes another dropout layer before the final dense layer (Dense) with a softmax ac- tivation function to output the class probabilities. This design effectively combines CNN for spatial feature extraction and LSTM for temporal sequence learning, making it suitable for tasks like video classification where both spatial and temporal patterns are crucial.

B. Data Transformation and Description

This paper develops the proposed IDS for external vehic- ular networks using a single dataset. Since it is a state-of-

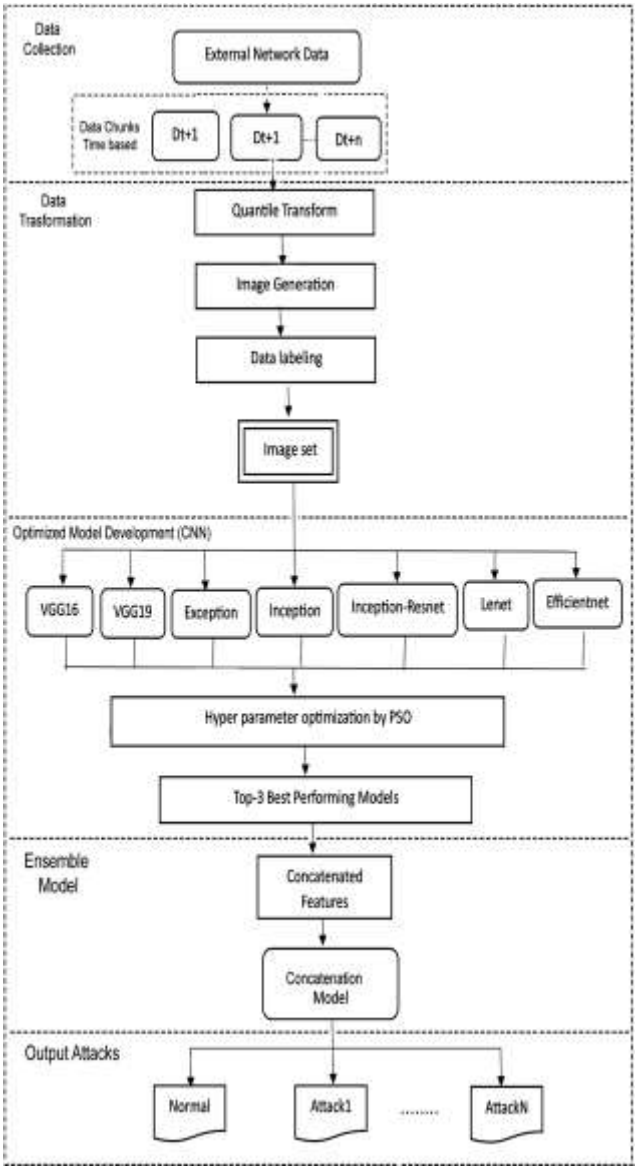

Fig. 2. Ensemble Model IDS Architecture

the-art network security dataset with the most recent attack patterns, the dataset that depicts external network data is the CICIDS2017 dataset [6]. The CICIDS2017 dataset's attack patterns may be categorized into five primary categories of assaults, as per the dataset analysis presented in [8] and [9]: DoS attacks, port-scan attacks, brute-force attacks, web- attacks, and botnets.

To provide an appropriate input for the suggested IDS, the data should be pre-processed after it has been acquired. The original network data should be converted into picture forms since CNN models perform better on image sets and vehicular network traffic datasets are often tabular data sets [10].

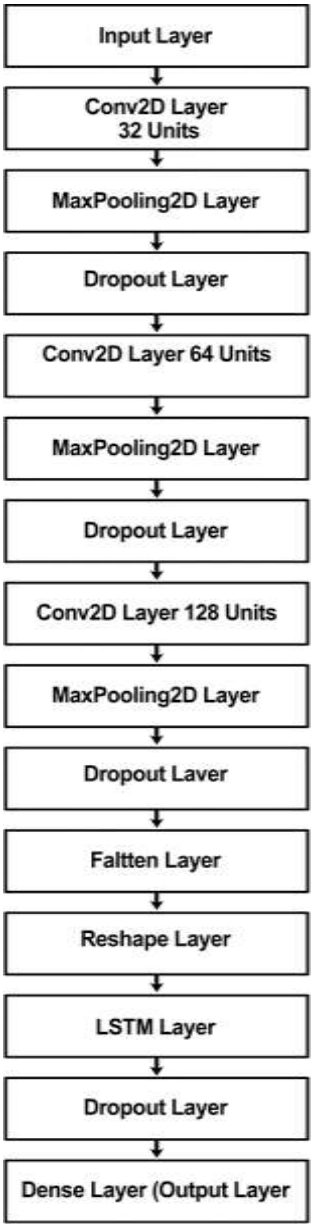

Fig. 3. Hybrid Model IDS Architecture

Data normalization is the first step in the data transfor- mation process. When picture pixel values span from 0 to 255, the network data must likewise be standardized to fall inside this range.Of all the normalization procedures, minmax and quantile normalization are the two most often utilized strategies for transforming data values into the same range. The quantile normalization method converts the feature distribution to a normal distribution and recalculates all the feature values based on the normal distribution. This is why it is used in the proposed framework [11], as min-max normalization does not handle outliers well and may result in most data samples having extremely small values.As a result, managing outliers is successful because most variable values are near to the median values [11].

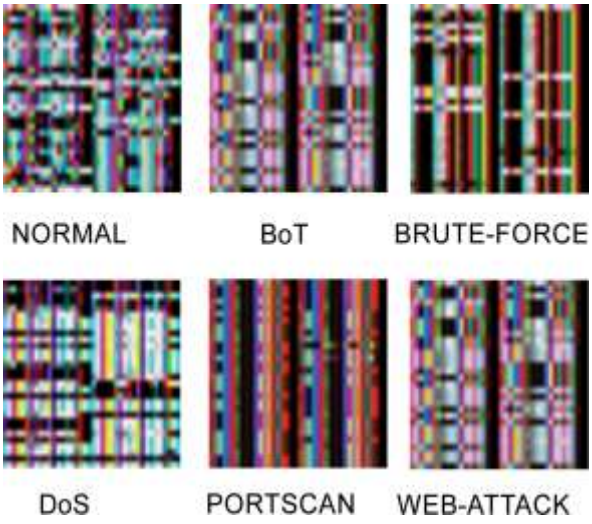

Fig. 4. Sample images of each class in CICIDS datasets

After data normalization, the data samples are divided into chunks according to the timestamps and feature sizes of network traffic datasets.Each chunk of the CICIDS2017 dataset, which was created from [12], has 20 × 20 × 3 color pictures, representing 20 × 3 = 60 consecutive data samples. Since the images are created using the timestamps of the data samples, the time-series correlations of the original network data can be preserved.

In the next stage, the modified images are labeled according to the assault patterns found in the data chunks.If every sample in a chunk or image is a normal sample, the image is termed as ”Normal.” Alternatively, a picture is designated as the most common attack type in a chunk if it contains attack samples. A BOT attack, for instance, will be identified as such in the associated image if it takes place in the chunk with the highest proportion.

Following the aforementioned data pre-processing tech- niques, the final modified image collection is produced and fed into CNN models. Fig. 2 displays the representative samples for each attack type found in the CICIDS2017 dataset.Based on the feature patterns displayed in Fig. 2, it is evidently possible to differentiate the attack patterns of CICIDS2017.

C. Convolutional neural networks and Transfer Learning

A popular deep learning model for image recognition and classification is convolutional neural networks [5].Without the need for supplementary feature extraction and data reconstruc- tion procedures, the images can be immediately entered into CNN models. Three different types of layers are typically seen in a CNN: fully-connected, convolutional, and pooling layers [5].Convolution procedures in convolutional layers allow for the automatic extraction of picture feature patterns. In order to prevent over-fitting in pooling layers, the data complexity can be decreased without sacrificing significant information through local correlations. Fully-connected layers act as a con- duit to connect all features and produce the output.Transferring the weights of a DNN (Deep Neural Network) model learned on one dataset to another is known as TL (Transfer Learning) for DL models [13] . Numerous image processing jobs have seen the successful application of the Transfer learning (TL) approach. This is due to the fact that only features learnt by the top layers of CNN models are specific features for a given dataset, but feature patterns learned by the bottom layers of CNN models are often generic patterns that can be applied to a wide range of tasks. [14] As a result, CNN models' lower layers can be used directly to various applications. Transfer learning can be made more successful by fine-tuning the DL models during the TL process. A few of the top layers of the pre-trained model are unfrozen to retrain it on a different dataset, while the majority of the layers are frozen in fine- tuning (i.e., their weights are retained). By fine-tuning, the learning model can improve the pretrained model's higher- order features to better suit the intended task or dataset. [14]. In the proposed framework, we have selected Xception,Inception,VGG16,VGG19, and InceptionResnet as the base CNN models due to their success in most image clas- sification problems [15]. These CNN models have shown excellent results on a variety of picture classification tasks. They were pre-trained on the ImageNet dataset. Over a mil- lion photos over 1,000 classifications make up the ImageNet dataset, a standard for image processing. [15].

On the ImageNet Challenge, the VGG16 models with 16 layers (VGG16) and 19 layers (VGG19) proposed in [16]] have achieved a reduced error rate of 7.3%.While the VGG16 design consists of three fully connected layers and five blocks of convolutional layers, the VGG19 architecture has three extra convolutional layers.Using convolutional feature extrac- tors that mix several contexts to obtain multiple types of feature patterns, the Inception network, first introduced in [13], lowers computing costs through dimensionality reduction. An expansion of the Inception network, called Xception [17], substitutes depth-wise separable convolutions for the conven- tional network convolutions. Compared to Xception, Inception requires a little more memory. BeginningOne of Inception's extensions, Resnet adds Resnet's leftover connections to the Inception network [15].When it comes to picture classification tasks, InceptionResnet performs better than Inception models, but it uses twice as much memory and compute power.

Following the training of five cutting-edge CNN models on the vehicle network datasets using transfer learning and fine-tuning, The top three CNN models with the highest performance are chosen as the basic learner for the ensemble models that are presented in the following subsection.

D. Ensemble Learning and Proposed Hybrid Model

The aim of ensemble or stacking learning is to build an ensemble or stacking model with higher performance by integrating many base learning models. An ensemble of many learners typically performs more accuracy than single learners, that is why ensemble method of learning is widely employed in data analysis difficulties [12].

For DL models, concatenation [18] is an ensemble ap- proach. Concatenate procedures are used to combine all the attributes into a newly developed concatenated layer which incorporates all the attributes. Main goal of a combined Con- volutional neural network is to get the highest order attributes produced from the better dense layer of basic CNN algorithms. A currepted layer to remove redundant charecteristics and a softmax layer to develop a newly created CNN algorithm come after the concatenated layer. Concatenation has the advantage of combining the highest level features to create a new, comprehensive model. However, it adds to the model training time because the new model needs to be trained again on the whole set of data. The concatenation method's computational cost is expressed as O(NF), Here N is the number of samples of data and the total number of features as F that were taken out of the base CNN models' dense layers.

A CNN-LSTM hybrid model for intrusion detection in the Internet of Vehicles (IoV) leverages the strengths of both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) [19]networks to identify malicious activities. The CNN component extracts spatial features from network traffic data, which is structured as multi-dimensional arrays representing different attributes of the traffic. These features are then passed to the LSTM component, which captures temporal dependencies and patterns that might indicate intru- sions. This hybrid approach allows the model to effectively analyze both the intricate spatial relationships and the temporal sequences within the network traffic, enhancing the detection accuracy and robustness against various types of cyber-attacks in IoV environments.

E. Hyper Parameter Optimization (HPO)

To further enhance the models' performance and better fit the base models to the chosen datasets, the CNN models' hyper-parameters must be adjusted and improved.

There are numerous hyper-parameters in CNN models that require adjustment. like other models in DL. These hyperpa- rameters fall into two categories: those used for model design and model training. [8].The hyper-parameters that ought to be established during the model design phase are known as model design hyper-parameters. The number or percentage of the learning rate, frozen layers, and dropout rate are among the model-design hyper-parameters in the suggested Transfer learning architecture.Model-training hyper-parameters, on the other hand, which include the number of epochs, batch size, and early stop patience, are used to balance the training pace and model performance. The structure, effectiveness, and efficiency of CNN models are directly impacted by the aforementioned hyper-parameters. The automated process of optimizing DL or ML models' hyper-parameters through opti- mization techniques is known as hyper parameter optimization. [8]. PSO is a popular metaheuristic optimization strategy for hyper parameter optimization issues that finds the ideal hyper parameter values by encouraging cooperation and information sharing among swarming particles [8] In order to determine the global optimum, the particles can finally progressively ap- proach the promising places. PSO is selected in the suggested framework because it can handle many hyperparameter types and has an O(NlogN) time complexity. [8]

IV. PERFORMANCE EVALUATION

A. Setup for the Experiment

The Python libraries Keras and Scikit-learn were used to conduct the experiments. The suggested Deeplearning models were evaluated in the trials on an i3 or i5 Google Colab with 4GB of RAM and trained on an i3 or i5 processor with a 500GB or above Hard Disk and 1 GB of RAM, which corresponds to an Internet of vehicle server and a computer at the vehicle level, respectively. As outlined in Section III-B, the suggested methodology is assessed using the CICIDS2017 [6] dataset, a benchmark vehicle network security dataset. Using sevenfold cross-validation, the suggested model is assessed, which can prevent biased and over fitting outcomes. However, as network packets data is typically very unbalanced and includes very few attack samples, performance evaluation is done using four separate metrics:F1-scores,accuracy, recall, and precision. Additionally, model testing time and training time on the server and vehicle level are tracked and do comparison in order to find the accuaracy of the suggested approach.

B. Results and Discussion for the Experiment

To construct optimal models, PSO was utilized for opti- mising the primary parameters of each basic CNN algorithms in the proposed framework. The parameter optimization pro- cesses were applied on the CICIDS2017 set of data. The starting search rate and the ideal hyperparameter measures are shown in the table1 The suggested ensemble models were built using the HPO-optimized CNN models as basis learners.

Table I present the findings from the assessment of the suggested ensemble models and the optimized CNN models using the CICIDS2017 datasets.

CICIDS dataset may contain sensitive information related to cybersecurity incidents, network traffic, and intrusion at- tempts. Protecting this data is essential to prevent unauthorized access, disclosure, or theft, which could potentially expose vulnerabilities or sensitive information. Here the results of the seven different trained model with this CICIDS2017 dataset. By comparing these seven models I chose three top perfoming models. Those are VGG19 (97.62%),inception resnet (99.62%) and Lenet model (97.7%), to concatinate as ensemble model, I got 99.94% accuracy for ensemble model.

99.97% accuracy is attained by using CNN-LSTM hybrid

TABLE I

Hyper-parameters and their optimal values for different CNN models hybrid model on same set of dataset CICIDS2017 as in Table II

Hyper-Parameter	**Model**	**Search Range**	**Optimal Value**
Epoch's number	All CNN models	[5, 50]	20
Batch size		[32, 128]	128
Learning rate		[0.001, 0.1]	0.003
Drop-out rate		[0.2, 0.8]	0.5
Early stop patience		[2, 5]	3
Number of frozen layers	Vgg16	[8, 16]	15
	Vgg19	[10, 19]	19
	Xception	[60, 125]	121
	Inception	[80, 159]	148
	InceptionResnet	[300, 572]	522
	Lenet	[84, 120]	5
	EfficientNet	[32, 256]	4

TABLE II

Performance Evaluation

Method	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train-time (s)**	**Test-time (ms)**
VGG19	97.62	99.17	99.17	99.17	688.1	0.1
Lenet	99.62	99.62	99.62	99.62	688.1	0.3
Inception ResNet	97.70	97.91	97.17	97.70	790.5	0.4
VGG16	96.11	97.12	96.11	96.11	436.3	0.1
Inception	96.11	96.12	96.12	96.11	782.8	0.3
Xception	97.13	97.18	97.13	97.13	655.5	0.2
EfficientNet	40.05	29.26	40.05	28.64	790.4	0.1
Ensemble	99.94	99.94	99.94	99.94	2351.1	1.4
CNN-LSTM	99.97	99.97	99.97	99.97	2400	1.0

Table 1 illustrates that the optimization based CNN models for the CICIDS2017 dataset attain high accuracy and F1-scores of 99.64% to 99.94% following transformation of data as well as PSO. 99.94% is the greatest accuracy and F1-score achieved by the proposed ensemble model, a little higher than the accuaracy and F1-scores of the individual algorithms VGG19 (97.62%), Inception-Resnet (97.7%), and Lenet (99.62%). In the literature, the ensemble model performs better than other contemporary techniques [2]. The small prediction time of the presented models suggests that the suggested IDS can be applied to real-time IoV networks, since vehicle systems that detect anomaly typically require ten milliseconds to carry out each packet's examination [20].

Here training time is much higher for all model than the testing time of those for ensemble model testing and training time is much higher than that of all other model ,because the accuracy and speed is inversely propotional to each other. But the main target of this project work is to achieve a good accuracy by using ensemble model than that of all other model And finally i have got a better accuracy 99.97% by using the CNN-LSTM hybrid model. Hence the CNN-LSTM hybrid model is the better model which gives the better accuracy for the detection of intrusions in the iov with the same CICIDS dataset.

V. CONCLUSION

IoV systems are facing a considerable increase in cyber threats as a result of the increased connectivity of modern automobiles. The work proposed an IDS framework using learning algorithms and ensemble method of learning which is using optimising Convolutional neural networks algorithms to identify various types of assaults in the vehicle network systems, with the goal of preventing cyber attacks on con- nected cars. Additionally, a network packet portions based data conversion technique is put forth to change car network traffic data into image data that CNN models can use as input. Using

data from an external network, the CICIDS2017 dataset is used to assess the suggested IDS. The practical outputs given that the ensemble model with IDS framework is able to successfully detect different forms of threats with a good accuracy and F1-score of 99.94% while comparing to other compared techniques on a single benchmark set of data.And the CNN-LSTM hybrid model IDS framework is able to successfully detect different forms of threats with a better accuracy and F1-score of 99.97% compared to all other models which are trained in this paper.

References

[1] S. T. Mehedi, A. Anwar, Z. Rahman, and K. Ahmed, "Deep transfer learning based intrusion detection system for electric vehicular net- works," *Sensors*, vol. 21, no. 14, p. 4736, 2021.

[2] A. Rosay, F. Carlier, and P. Leroux, "Feed-forward neural network for network intrusion detection," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–6, IEEE, 2020.

[3] L. Yang, D. M. Manias, and A. Shami, "Pwpae: An ensemble framework for concept drift adaptation in iot data streams," in *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 01–06, IEEE, 2021.

[4] L. Yang, A. Moubayed, A. Shami, P. Heidari, A. Boukhtouta, A. Larabi, R. Brunner, S. Preda, and D. Migault, "Multi-perspective content delivery networks security framework using optimized unsupervised anomaly detection," *IEEE Transactions on Network and Service Management*, vol. 19, no. 1, pp. 686–705, 2021.

[5] H. M. Song, J. Woo, and H. K. Kim, "In-vehicle network intrusion detection using deep convolutional neural network," *Vehicular Communications*, vol. 21, p. 100198, 2020.

[6] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization.," *ICISSp*, vol. 1, pp. 108–116, 2018.

[7] L. Yang, A. Moubayed, I. Hamieh, and A. Shami, "Tree-based intelligent intrusion detection system in internet of vehicles," in *2019 IEEE global communications conference (GLOBECOM)*, pp. 1–6, IEEE, 2019.

[8] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020.

[9] R. Panigrahi and S. Borah, "A detailed analysis of cicids2017 dataset for designing intrusion detection systems," *International Journal of Engineering & Technology*, vol. 7, no. 3.24, pp. 479–482, 2018.

[10] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, "Iot dos and ddos attack detection using resnet," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pp. 1–6, IEEE, 2020.

[11] S.-F. Lokman, A. T. Othman, M. H. A. Bakar, and S. Musa, "The impact of different feature scaling methods on intrusion detection for in-vehicle controller area network (can)," in *Advances in Cyber Security: First International Conference, ACeS 2019, Penang, Malaysia, July 30–August 1, 2019, Revised Selected Papers 1*, pp. 195–205, Springer, 2020.

[12] L. Yang, A. Moubayed, and A. Shami, "Mth-ids: A multitiered hybrid intrusion detection system for internet of vehicles," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 616–632, 2021.

[13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818– 2826, 2016.

[14] M. M. Leonardo, T. J. Carvalho, E. Rezende, R. Zucchi, and F. A. Faria, "Deep feature-based classifiers for fruit fly identification (diptera: Tephritidae)," in *2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pp. 41–47, IEEE, 2018.

[15] D. Petrov and T. M. Hospedales, "Measuring the transferability of adversarial examples," *arXiv preprint arXiv:1907.06291*, 2019.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] F. Chollet, "Xception: Deep learning with depthwise separable convolu- tions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258, 2017.

[18] L. Yang and A. Shami, "A lightweight concept drift detection and adaptation framework for iot data streams," *IEEE Internet of Things Magazine*, vol. 4, no. 2, pp. 96–101, 2021.

[19] S. Shende and S. Thorat, "Long short-term memory (lstm) deep learn- ing method for intrusion detection in network security," *International Journal of Engineering Research and*, vol. 9, no. 06, 2020.

[20] A. Moubayed, A. Shami, P. Heidari, A. Larabi, and R. Brunner, "Edge- enabled v2x service placement for intelligent transportation systems," *IEEE Transactions on Mobile Computing*, vol. 20, no. 4, pp. 1380–1392, 2020.