[1]Nujud Al-Aql

Abdulaziz Al-Shammari

Hybrid RNN-LSTM Networks for Enhanced Intrusion Detection in Vehicle CAN Systems

Journal of Electrical Systems

Abstract: - Electric vehicles (EVs) use electric motors for propulsion, relying on electric energy stored in batteries or other energy storage devices. The standard communication protocol used in EVs is the Control Area Network (CAN), a communication protocol widely used in the automotive industry for networking and communication between various components within a vehicle. CAN protocol, designed without any care about protection, as automotive systems become more connected, the vulnerability to cyber threats, including intrusion attacks. The most common intrusion attacks on EVs are Denial of Service (DoS), Fuzzy, and Impersonation Attacks. These become a significant challenge due to the imperative need for robust Intrusion Detection Systems (IDS) in CAN networks. This paper explores the application of advanced deep learning techniques, specifically Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks, to enhance the effectiveness of intrusion detection in the EV domain. We will use a hybrid Deep-learning model to improve the analysis. First, we will apply the RNN model, and the output will come as input for the second model, LSTM. Our proposed hybrid model achieved an accuracy of 93%. The outcomes of this research contribute to advancing cybersecurity measures in vehicular networks, ensuring the integrity and safety of connected vehicles. The applicability of RNN and LSTM techniques in the context of CAN networks demonstrates their potential to evolve as integral components of next-generation intrusion detection systems, fostering a secure and resilient automotive ecosystem.

Keywords: Intrusion detection system (IDS); controller area network (CAN) bus; deep learning (DL); Recurrent Neural Networks (RNN); Long Short-Term Memory (LSTM) networks; Denial of Service (DoS); Electronic Control Units (ECUs).

I.INTRODUCTION

They say electric vehicle (EV) technology is fast advancing in the modern car [1]. For this cause, the modern car will be more intelligent with many more valuable applications, contemporary, and ranging across numerous functions. The various features are controlled by many Electronic Control Units (ECUs) interconnected by the CAN bus. One ECU can control, monitor, and allow the subsystems to work at low vehicle noise, vibration, and energy [2]. After ECUs started being used in the automotive system. This furthered the functionality. These advances have undoubtedly increased the quality of our living standards and the susceptibility of cars as victims of cyber-attacks [3]. CAN lacks security functionality; for example, it has no authentication and encryption to protect communication from web attacks [4]. Researchers have shown that there are significant security flaws in in-car networks [5]. An attacker might get physical control over a car by injecting bogus signals into the car's security system and tampering with and accessing an ECU through weak interfaces. Some examples of susceptible interfaces are Compact Disc (CD) players, On-Board Diagnostics, and flash drives (USB). Furthermore, automobiles are becoming highly intelligent machines that can now communicate with their environment thanks to the development of wireless technologies like Bluetooth, Wi-Fi, mobile communication, and 5G [6]. For instance, a live system has been successfully hacked using car key fobs. Furthermore, ECUs cannot identify who provided the transmitted signals and can receive any ECU-to-ECU transmission on the same bus. Malicious assaults, such as packet injection and data manipulation, can create fictitious packets that conquer crucial components that ensure drivers' safety [7]. Other vehicle assault methods include radio, GPS, Electronic Windows, and steering and brake hacking [8], [9]. Because they put the driver in danger for life, vehicular assaults are therefore detrimental to the car as well as the driver [10]. Consequently, it is critical to identify car intrusions to prevent vehicle damage and save lives [11].

Cars can face various kinds of attacks. These could include things like Denial-of-Service (DoS) attacks, flooding, fuzzy attacks, spoofing, malfunctions, vulnerabilities when nearby, impersonation, replaying data, rerouting, remote sensor tampering, and pretending to be someone else [12]. Numerous research works have examined safety concerns related to intra- and inter-vehicular communications [13]. For instance, due to their

[1] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU) ,

Riyadh 11432, Saudi Arabia , e-mail:435030595@sm.imamu.edu.sa

increasing effectiveness and ease of use in detecting intrusions, intrusion detection sensors are becoming increasingly popular [14] Noura et al. [15] devised a method that uses the Advanced Encryption Standard (AES) encryption to offer data secrecy while using the least resources. It uses less memory, power, and calculations. With limited hardware and software resources, Castiglione et al. [16] suggested a method for securing in-car communication that uses lightweight block ciphers. Mundhenk et al. [17] introduced a security method called Lightweight Authentication for Secure Automotive Networks (LASAN) to protect car communication while using little in the way of computing resources, such as electricity and network bandwidth. Only specific threat models previously considered throughout the design phases may make these IDS effective [18]. Most current research on the security of the CAN protocol has concentrated on physical aspects, such as encrypting CAN transmission and restricting access restrictions [19]. Still, the development of a more effective IDS is required. Physical access restrictions will reduce the effectiveness of CAN bus communications. Using cryptography with a system this light is only sometimes successful. Machine learning (ML) based intrusion detection systems address the issue with traditional communication networks. The objective is to document the essential statistical characteristics of data and use them to identify any assault. To categorize different forms of attacks, intrusion detection techniques such as Random Forest (RF), Multilayer Perceptron (MLP), Decision Tree (DT), and Support Vector Machine (SVM) are created [20]. ML techniques are used for a vehicular network as the typical ECU's processing capacity is restricted to handling such a complicated operation. Throughout the past ten years, ML tools have produced significant and influential results for complex challenges, such as fault detection [21] and detection of cyber threats [22]. ML techniques can be helpful for intrusion detection. However, there still needs to be a widely agreed-upon framework or model for consistently identifying and categorizing cyberattacks [7]. They may be enhanced by extensively using the dataset and other ML models. To address the present security issues with in-vehicle CAN buses, this motivates us to investigate the potential of practical Deep Learning (DL) algorithms, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) and Generative Adversarial Network (GAN) or use hybrid algorithms to get good results with analysis.

II.BACKGROUND

- *CAN Protocol*

The Controller Area Network (CAN) bus protocol is a crucial communication standard in automotive and industrial systems. Developed in the 1980s by Robert Bosch GmbH,[33] originally for vehicle-to-vehicle communication within the car, but now finding wide acceptance in all areas of the industry due to its strength and reliability. Basically, CAN bus protocol mediates communications of electronic control units (ECUs) on the network but not through a central computer host. This, therefore, creates an invisible relationship between interconnected devices to exchange critical real-time data in the modern-day way of operating vehicles and industrial machinery. CAN works on a message-based protocol, wherein data is sent as frames. The frames carry an identifier that indicates both the priority and content of the message, coupled with the actual data payload. CAN uses a differential signaling system and is, therefore, well adapted to transmit without problems in noisy surroundings, subsequently enabling this bus to apply effectively in automotive applications, where intensities of electromagnetic interference are very high. One of the most critical features of the CAN protocol is that it is designed to be fault tolerant. When a transmission error, CAN uses a bitwise mechanism of arbitration since hose messages with lower identifiers have priority over those with higher identifiers. By itself, it attempts to resolve the conflicting transmissions smoothly allows for the integrity of the communication network. Besides, it allows for the variation of the communication speed settings. It is, therefore, flexible in the setup of any system, providing data rates of a few kilobits per second up to several megabits per second, which can be used and are supported by CAN across all industries. Over the years, the CAN bus protocol has had revisions with increments meant for improved data throughput, better fault detection, and security. This has further propped up its place in strengthened domains such as automotive networking, industrial automation, and many more. The CAN bus protocol, therefore, forms the foundation for all modern communication systems, promising reliable interchange even in hostile data environments. That, with the fault-tolerant design, decentralized architecture, and flexibility it adheres to, makes it preferred for applications requiring robust and efficient communications.

- *Electronic Control Units*

Electronic Control Units (ECUs) form the heart of today's modern vehicle architecture, playing the role of a maestro in the sophisticated interplay of the various subsystems required for optimized performance, efficiency, and safety. The high-end embedded systems include functionalities from the vehicle's engine management to Advanced Driver Assistance Systems (ADAS)[34], which have the most significant role in vehicle adaptation and reactive behavior. Until now, the domain of automotive control systems has relied entirely on mechanical and hydraulic components, which facilitate basic operations, among them fuel injection and ignition timing. On this continent comes a technological innovation toward super-specialized computational units capable of managing complex algorithms and protocols in real time. In principle, ECUs are composed of microcontrollers or microprocessors furnished with dedicated memory, input/output interfaces, and communication protocols. The ECUs sense the inputs required by using the different types of sensors located in the vehicle, process them using sophisticated algorithms, and, in return, command the actuators to make a series of required movements. The trend of innovations and optimizations in this area really mirrors the continued growth in the number of ECUs in modern-day vehicles. Today, a typical car might include tens of ECUs, which include ECUs for managing the engine, controlling the transmission, modulating the brakes, and many others for infotainment. Such an architecture would support modularization and specialization, hence allowing easy diagnosis, maintenance, and upgrade of the system. It has also enabled the integration of ECUs and high-end functionalities developed, such as adaptive cruise control and lane-keeping assistance. The ECUs apply sophisticated machine learning algorithms to onboard sensors in a way that they can analyze the driving pattern of the driver in a bid to predict danger and proactively take measures to prevent it, hence enhancing safety and comfort. As a result, the number of ECUs is projected to grow further, bringing challenges, especially in cybersecurity, and increased system complexity. Thus, interconnected systems and communication interfaces make ECUs vulnerable to hacking and unauthorized access. So, enforcing high-security measures and strict validation procedures to ensure vehicle integrity and user privacy are sacrosanct. All told, electronic control units make up the technological backbone of modern automotive systems when digital innovation and mechanical exactitude come together. As vehicles continue to roll more and more towards electrification and autonomy, ECUs will surely be at the forefront of inciting changes and aiding in shaping a way forward for mobility.

- *Attack on EVs*

The development of more ECUs in the systems of such "smart" cars creates a high level at which the automobile will be connected with the digital world, hence exposing the automobile to cyber-attacks. CAN protocol is extensively accepted, but it makes no room for such security mechanisms as authentication and encryption; it thereby really exposes the cars to many threats. Electronic vehicle attacks can happen remotely or through direct physical contact. For this reason, the communication channels within the car, including Bluetooth, Wi-Fi, or cellular networks, and other channels over the CAN bus from remote places, are open for hackers to exploit to transmit their malicious messages. Physical attacks can also be placed in effect either at the time of manufacturing or while tampering with on-board diagnostic (OBD) ports in a car, from message spoofing, such as attackers sending fake commands to the vehicle to control some part of its behavior, to much more insidious attacks. These attacks may result in the loss of life, risk to passenger safety, and compromise of operations integrity. Threat actors might potentially take over the control of functions with major criticality levels like braking or steering. Besides, there is this risk of hijacking the car's infotainment system or stealing private data, which might, in fact, seriously compromise the public trust in the safety of today's vehicles. The danger is not only theoretical but some high-profile cases are proven facts that demonstrate these vulnerabilities. For example, researchers have shown that it is practically feasible for attackers to take control of a vehicle's engine and brakes over the Internet. Such incidents do, in reality, put forward the necessity for very strong security within the in-vehicle communication network against such kinds of threats. Therefore, the full understanding of these risks and methods of their possible attacks is necessary for the in- depth involvement of the automotive industry to develop more secure systems against these future threats. This implies thorough, not only benevolent but security measures that breed confidence in the reliability of the connected vehicle.

- *Deep Learning for Detection*

Deep Learning in Cybersecurity: With the exceptional power of deep learning to identify anomalies and possible threats in the vehicle network, much has transpired in this regard for cybersecurity. Its capabilities in learning from extensive data and knowing complex patterns find perfect use in Supervising the complex exchanges in the CAN systems— often, the old ways of security do not suffice. Deep learning models are optimized for normal message separation from those with potentially harmful impacts in a CAN context. Other models, instead, analyze the timing, frequency, and sequence of CAN messages to spot the anomalies that refer to an attack, e.g., from irregular message patterns to an increase in deviant requests. Here, it is the bedrock for some deep-learning architectures, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), which have been made operational. In this light, therefore, RNNs will be highly appropriate for time- pattern observation within flows of CAN messages and quite capable of detecting state-of-the-art attacks that develop over time. The challenges and opportunities arise in the huge benefits that come with the high detection capabilities of deep learning. If all else is held constant, it is not free of many challenges. These include the fact that it requires massive training data and has other risks like adversarial attacks, where attackers manipulate data to beat the defenses. Nevertheless, there's a strong push in research to make these models more robust and less susceptible to such tactics. Deep learning is a powerful tool for bolstering the security of vehicle CAN networks, providing an adaptable and forward-looking approach to counteracting the diverse cyber risks.

III.RELATED WORK

Significant research has contributed to developing a methodology for detecting attacks on CAN bus protocol. In this context, many studies have focused on solving related problems using Machine learning and deep learning. This section reviews recent state-of-the-art research and methods that address this problem.

- *ML For Detection on CAN Protocol*

Yang et al. [23] proposed a lightweight framework using machine learning models for IDS in autonomous vehicles. The authors focused on the tree-based decision classifier on the standard dataset, and the tree-based decision classifier showed significant results in detecting cyberattacks in CAN. The proposed work needs to validate the model on the real-time dataset further to evaluate the models' effectiveness and robustness. Alfardus and Rawat [24] do excellent research. They tried four different algorithms using HCRL Car-Hacking dataset [35]. Three of them are ML, and the last one is DL; for ML SVM, RF, and K-Nearest Neighbor (KNN). They got almost similar Results, except KNN, which got on ACC 98.82% and Recalled 98.04%; otherwise, it's identical. Alshammari et al. [25] proposed IDS on CAN Bus Protocol using two different ML. The analysis using HCRL CAN Intrusion Detection dataset [32], KNN gave great results more than the SVM. D'Angelo, Castiglione, and Palmieri [26] present two algorithms designed to execute a data-centric anomaly detection framework. The Cluster-based Learning Algorithm's initial algorithm is employed to grasp the typical patterns of messages transmitted across the CAN bus, serving as a baseline reference. Conversely, the second algorithm, the Data-driven Anomaly Detection Algorithm, is utilized for instantaneous classification of these messages, distinguishing between legitimate and malicious ones, thus facilitating early detection in cases of misuse. They used the HCRL CAN Intrusion Detection dataset [32], and they got on ACC 99.98%, and for Precision, they got 99.86%. Refat et al.[27] used the HCRL Car-Hacking dataset [35] to analyze it, and they tried a new approach to get a bitter result. First, they transferred the CAN message to Graph Feather, and then they started the analysis. They tried two different ML algorithms, SVM and KNN. SVM gave a higher result than KNN.

Table 1ML INTRUSION DETECTION

Reference number	Type	Accuracy	Recall	F1 score	Precision	DATASET
23	DT	99.99	99.99	0.999	-	HCRL Car-Hacking dataset [35]
23	DT	99.72	99.3	0.998	--	CICIDS2017 [36]
24	SVM	99.99	100	100	--	HCRL Car-Hacking dataset [35]
24	RF	99.99	100	100	--	HCRL Car-Hacking

Reference number	Type	Accuracy	Recall	F1 score	Precision	DATASET
						dataset [35]
24	KNN	98.82	98.04	100	--	HCRL Car-Hacking dataset [35]
25	SVM	96.4	97.7	93.3	98.4	HCRL CAN Intrusion Detection dataset [32]
25	KNN	96.9	98.5	93.5	99.9	HCRL CAN Intrusion Detection dataset [32]
26	Clustring	99.98	--	--	99.86	HCRL CAN Intrusion Detection dataset [32]
27	SVM	99.67	97.23	98.04	99.03	HCRL Car-Hacking dataset [35]
27	KNN	98.59	97.06	97.98	99.11	HCRL Car-Hacking dataset [35]

- *DL For Detection on CAN Protocol*

To achieve more secure communication, hardware and software ciphers might be combined. Hossain et al. [28] proposed the LSTM for attack detection in the CAN system. For the LSTM's learning, the authors developed a custom dataset using the experimental vehicle. Furthermore, they injected different types of attacks and collected malicious samples. The proposed LSTM model showed a 99.99% accuracy score for training and testing instances.Similarly, kan et al. [29] also proposed a Bi-LSTM model for anomaly detection in CAN that enabled the system to classify the anomalies into DoS, reply, and fuzzy attack. Khatri et al. [30] proposed the transfer learning-based model for intrusion detection in CAN. The author extracted the quality features of CAN using the DL model and fine-tuned the model for robust classification of attacks. Javed et al[31] they try a new hybrid approach and compare it with previous studies. Their model was designed with Two DL CNN and Grated attention Recurrent Neural Networks (GRU), they used CAN Dataset for intrusion detection (OTIDS) [32] data set.

Table 2 DL INSTRUCTION DETECTION

Reference number	Type	Accuracy	Recall	F1 score	Precision	DATASET
28	LSTM	99.98	99.97	99.90	--	Simulation
29	Bi-LSTM	95.5	-	-	--	Simulation
30	Transfer learning Hybrid (CNN+ LSTM)	100	100	100	100	HCRL Car-Hacking dataset [35]
30	Transfer learning Hybrid (CNN+	99.91	99.91	99.91	99.91	Car hacking: attack & defense challenge 2020 dataset [37]

Reference number	Type	Accuracy	Recall	F1 score	Precision	DATASET
	LSTM)					
31	Hybrid (CNN+ GRU)	94.23	93.91	93.79	93.69	HCRL CAN Intrusion Detection dataset [32]

IV.METHODOLOGY

In previous studies, DL showed a greater detection result than ML, so we designed a new approach that used two DL algorithms. First, we take CAN massage, which comes as input for the RNN model. Then, the result goes as input for the second method, LSTM. This showed a great detection result. This section describes the proposed methodology with the dataset description. The overview of the proposed method is also shown in Fig 1

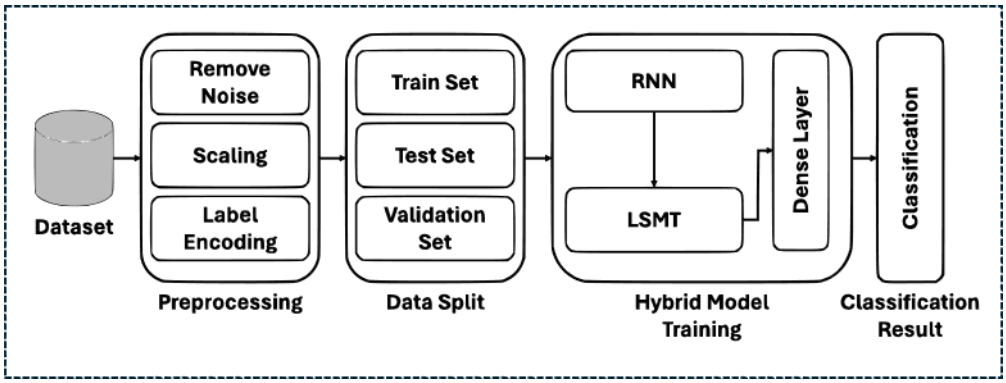

Figure 1 Overview of the proposed methodology.

- *Dataset Description*

We used one of the most published datasets, which is the CAN Dataset for Intrusion Detection (OTIDS)[32]; it contains three kinds of attacks (DoS, Fuzzy, and Impersonation) and contains Free attack data. The distribution of these classes inside the dataset is shown in Fig 2. In the dataset, about 2.3 million records are normal or attacks free, along with Dos, fuzzy and Impersonation have 656.5K, 591.9K, and 995.4K, respectively. The dataset contains a total of about 4.6 million records. Originally, the dataset was in the form of text files.

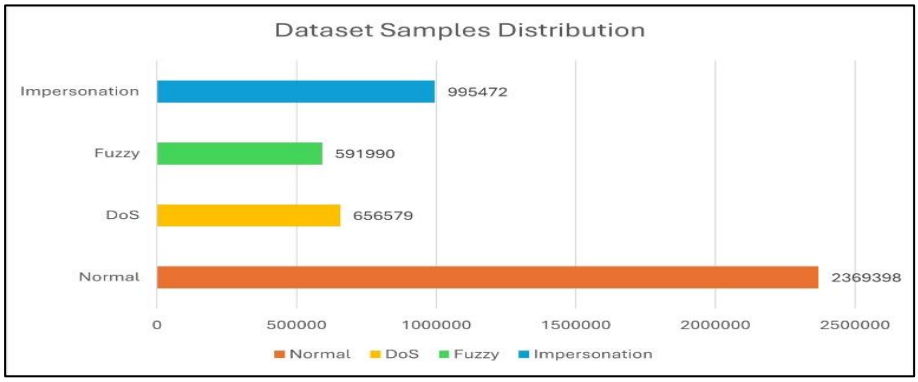

Figure 2 Dataset Class Distribution

- *Dataset Pre-Processing*

This section discusses the preprocessing of the dataset before splitting it into training and testing and applying the model to it. Initially, the data in 4 text files for Dos, Fuzzy, Impersonation, and attacks free, referred to as Normal. We have read the text files using Python and merged all four text. files into a single data frame, and set the proper column names as 'Timestamp,' 'ID,' 'Protocol,' and 'DLC,' etc. After creating the data frame, we removed all those instances that have no CAN message. Some columns are also converted to appropriate data types that were initially in string or Object type to get the best results for the model. The labels of the training and testing set were encoded into 0, 1, 2, and 3 for Dos, Fuzzy, Impersonation, and Normal classes.

After preprocessing all the data, we split the data into train, validation, and test sets with 60% for training, 20% for validation, and 20% for testing. We have also used the argument of the splitting function to split the data equally based on their classes—the count of the records after the split is shown in Table III. We have also converted the data into the appropriate format that the RNN model expects, a 3d format.

Table 3 Samples Distribution in each Subset.

Split	Fuzzy	Dos	Impersonation	Normal	Total
Train	371142	413302	634081	1451852	2870377
Validation	92785	103326	158521	362963	717595
Test	115982	129157	198150	45370	896993

- *Proposed Architecture*

Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), and fully connected layers are all integral parts of deep learning in their respective ways of managing and processing serial data. RNNs are well-designed to process sequential data. They retain the data from the previous input to scale up their competence in tasks like time series prediction, voice recognition, and processing language. LSTMs help provide a way to solve the vanishing gradient problem in conventional RNNs, thus enabling long-term data to be handled more effectively. Dense or fully connected layers are central architectural elements of a neural network, enabling a network of interlinked neurons to perform intricate feature detection and categorizing.

We have proposed a hybrid deep learning model that combines RNN with the efficient LSTM model to get optimized results of classification for the effective intrusion detection task. Initially the preprocessed data passed to the RNN architecture that will do initial feature extraction and get the embeddings of our data, then these embeddings passed to the LSTM architecture that have the capability for extracting long term dependent features to find the exact representation of our data. Once the feature extraction phase has been completed, these extracted features then go through the fully connected layers stacked one after the other. In this phase the extracted features undergo through the interconnected neuron layers which further refine the features. Finally, the output of this layer passed through the classification layer which classify the input record into one of the four predefined classes.

V.RESULTS AND DISCUSSION

The evaluation measures that were employed to ascertain the effectiveness of the recommended strategy are thoroughly examined in this section. It also looks at the hardware and software requirements needed for training and evaluating models. A thorough description of the various hyper-parameters and the values that correspond with them is provided. This part also meticulously presents a thorough examination of the results obtained using the proposed methodology.

- Evaluation Measures

Some of the quantitative measures, also referred to as assessment metrics to measure a deep learning model, are effective. It assesses the performance at which different models or algorithms perform for some given tasks, estimates the performance of a model or algorithm to solve a given problem, and provides areas of potential improvement. This research employed Recall, ROC curve, accuracy, f1-score, precision, and confusion matrix

in the assessment measures. All taken together, these measurements provide an all-rounded verdict of the model's efficacy and thereby detail both the benefits and possible areas for its development.

Accuracy: Accuracy measures how well the model classifies correctly, taking everything into account, including all the instances and samples. Yet, if different types of lapses differ in their degree of importance or if datasets are imbalanced, one might quickly conclude that strict reliance on the attribute of accuracy is insufficient for a complete evaluation.

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

Precision: Precision is related to how well the model can correctly select positive samples from the set of actual positives. The number of true positives against the summation of the number of true positives and the number of false positives computes precision. In short, precision indicates how well the model performs when it generates a positive forecast.

$$Precision = \frac{TP}{TP+FP} \qquad (2)$$

Recall: Recall measures how successfully the model separates positive samples from the actual positive pool. It is also frequently referred to as sensitivity or the true positive rate. This statistic is computed as the ratio of true positives to the sum of true positives and false negatives. In essence, recall offers an assessment of the extent to which the model's favorable predictions hold true.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1_Score: A total statistic that strikes a balance between recall and precision is the f1-score. The harmonic mean of these two measurements is used to calculate it. This is particularly helpful in situations where there is an unequal distribution of lapses among the classes or where the significance of the two error categories is the same. Using a range of 0 to 1, the f1-score is a comprehensive assessment of the model's precision and recall skills. It functions optimally at 1.

$$F1\_score = \frac{2*Precision*Recall}{Precision+Recall} \qquad (4)$$

- *Environmental Setup*

Several experiments were conducted within the Colab environment to test the models. Throughout this study, the model was trained and evaluated using the Python programming language TensorFlow and Keras. An NVIDIA Tesla T4 GPU with 15 GB of RAM was used in the trials, which ran on Google Colab's free edition.

- *Hyper-Parameter Settings*

Extensive empirical testing was conducted to optimize the model training performance for Intrusion detection in the CAN system by fine-tuning several hyperparameters. Batch size, selection of optimizer, learning rate, epochs, and loss function selection are some of these critical factors. The goal was to determine which combination of hyperparameter values produced the most significant outcomes in identifying intrusion in CAN by methodical testing and iteration. The model was optimized iteratively to get the appropriate degree of resilience and accuracy in identifying various forms of CAN intrusion. The selected parameters are shown in Table 4.

Table 4 Optimal Values of Hyperparameters.

Parameter	Value
Optimizer	Adam
Learning rate	0.001
Epochs	10
Batch Size	64

Activation (RNN & LSTM)	ReLU
Dense layer Neuron (RNN & LSTM)	64

- *Evaluation Measures*

In this study, we have suggested a hybrid RNN-LSTM architecture for categorizing various CAN system assaults on automobiles. Following preprocessing and the removal of records without any messages, the dataset's initial contents were 4.61 million, which were reduced to 4.48 million records. Firstly, 20% of the dataset was put aside for testing, leaving 80% for training. From the training set, 20% of that was used for validation when our model was in the training stage. As seen in Figure 3, the model functioned incredibly well and displayed accuracy rates of 93.2% and 93.2% for both training and validation. The performance of the model can be seen from the losses which are 0.1831 and 0.1833 for training and validation respectively, as presented in Fig 3.

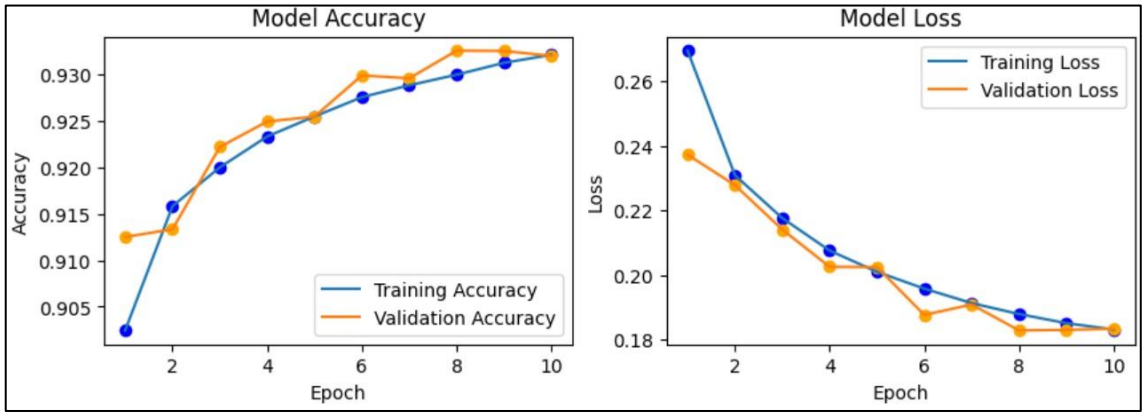

Figure 3 The loss and accuracy of the model during training.

ROC curve and confusion matrix technique is used for the evaluation of the model on the unseen data that was separated out initially. An assessment metric called a confusion matrix shows the actual label of the samples on the y-axis and the anticipated label of the model on the x-axis. Additionally, it determines the number of correct matches in cases where the projected label and the actual label match exactly. Fig 4 displays the confusion matrix of the trained hybrid model.

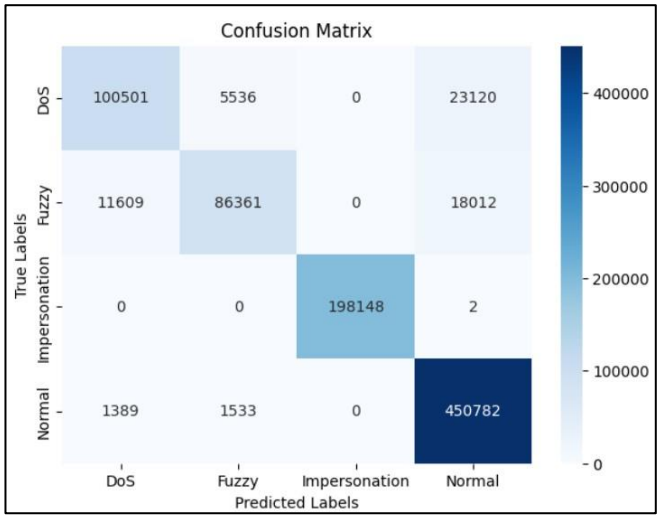

Figure 4 Confusion matrix of our proposed hybrid model.

The confusion matrix was used to construct the remaining assessment metrics, such as accuracy, precision, recall, and f1-score, as stated in equations 1-4. The table comprehensively summarizes the model for various attack categorizations on unseen samples. The unweighted average of measurements determined individually for each class is referred to as "Macro AVG" in Table 5. Put another way, it computes the average of each class's unique performance metrics and considers all classes equally, regardless of how frequently or how important they are. This ensures that every class receives a fair evaluation. Calculating the average of the metrics weighted by the number of samples in each class, also known as "Weighted AVG," which considers the class imbalance. This indicates that, in comparison to classes with fewer instances, classes with more instances have a more significant impact on the total average. "Weighted AVG" is especially helpful when working with unbalanced datasets since it assigns greater weight to class performances that are more typical of the distribution.

Table 5 Classification Report of the Proposed Model.

	Precision	Recall	F1Score	Support
DoS	0.89	0.78	0.83	129157
Fuzzy	0.92	0.74	0.82	115982
Impersonation	1	1	1	198150
Normal	0.92	0.99	0.95	453704
Accuracy			0.93	896993
Macro Avg	0.93	0.88	0.9	896993
Weighted Avg	0.93	0.93	0.93	896993

The ROC curve is an extremely useful visual tool in which the plot of the performance is made at different levels of decision-making thresholds to assess how well the classification system works. It's basically a plot of the True Positive Rate (TPR) against the False Positive Rate (FPR) with a changing threshold. True Positive Rate is also called sensitivity or recall; it specifies the number of actual positives that are correctly identified through a test. The False Positive Rate tells how many negatives are wrongly assigned to positives. On the other hand, the area under the ROC curve gives overall information about the accuracy of the test in discriminating effectively between the two groups. A ROC curve plots sensitivity on the y-axis against 1-specificity on the x-axis. An AUC of 0.5 suggests the test does no better than making random guesses, and an AUC of 1.0 suggests the test perfectly separates the two groups. The proposed hybrid model gives the AUC score of 0.97, 0.98, 1.00, and 0.99 for Dos, Fuzzy, Impersonation, and Normal Classes, respectively, as shown in Figure 5.

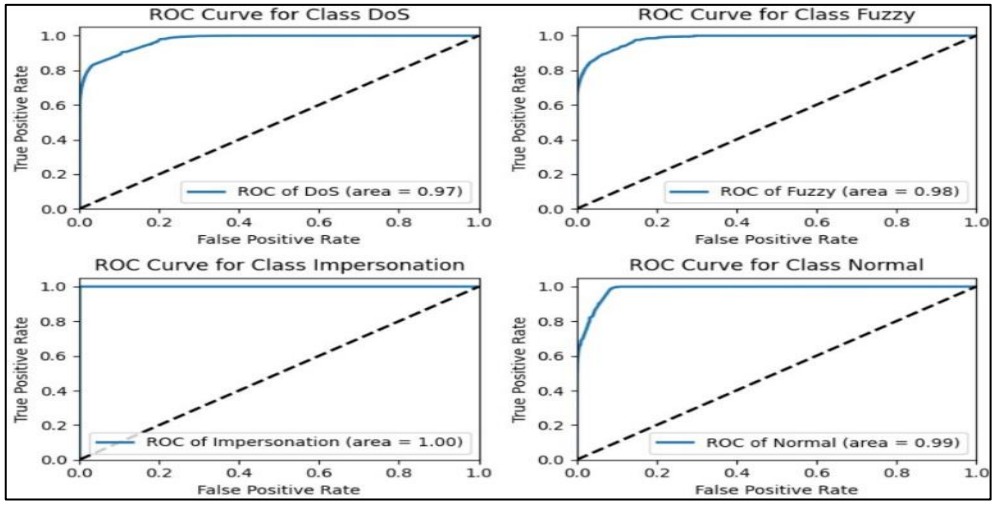

Figure 5 ROC Curve of all Classes.

VI.CONCLUSTION

This study proposed a hybrid deep learning model by combining Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) networks for an improved electric vehicle cybersecurity setup. Our groundbreaking hybrid deep-learning model goes beyond off-the-shelf solutions, not just in dealing with the CAN protocol's known susceptibilities but by setting new standards for both accuracy and efficiency for intrusion detection by giving a 93% success rate. This work shows the potential for enhancement in the security of the vehicular network through the fusing of RNN and LSTM architectures. As the trend of connecting electric vehicles to an ecosystem increases, there is an urgent need to develop a robust intrusion detection system. These advancements have made it necessary to move to an advanced level of cybersecurity protection against sophisticated threats, including Denial of Service (DoS), fuzzy, and impersonation attacks, which are increasing daily in the changed automotive technology environment. The effectiveness of the proposed model would, in essence, lead to the efficacy of the current security protocols while opening doors for further research and development. The study promotes the continued development and exploration of deep-learning strategies to enhance the accuracy and efficiency of intrusion detection systems.

This research, therefore, provides a critical step in the security of electric vehicles from cyber threats, thus leading to improved intrusion detection systems using RNN and LSTM. These, in turn, would undergird continuous innovation in vehicle cybersecurity and form a solid base for future actions to secure connected and autonomous vehicles to provide a safe, reliable, and resilient automotive future.

ACKNOLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University for funding and supporting this work through Graduate Students Research Support Program.

REFERENCES

[1] Elkhail, R. Refat, R. Habre, … A. H.-I., and undefined 2021, "Vehicle security: A survey of security issues and vulnerabilities, malware attacks, and defenses," ieeexplore.ieee.orgAA Elkhail, RUD Refat, R Habre, A Hafeez, A Bacha, H MalikIEEE Access, 2021•ieeexplore.ieee.org, Accessed: Nov. 06, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9625934/

[2] T. Park, C. Han, S. L.- Mechatronics, and undefined 2005, "Development of the electronic control unit for the rack-actuating steer-by-wire using the hardware-in-the-loop simulation system," Elsevier, Accessed: Nov. 06, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957415805000681?casa_token=BUf4Trnbv3EAAAAA:DXqC2C4moxECULaJiR_LtupnDufsvlqoK35fZcYqeHJKDwwkX6CJGCKcAzYfyIiXO6s8aKTGF-bA

[3] M. Ring, D. Frkat, M. S.-A. C. S. In, and undefined 2018, "Cybersecurity evaluation of automotive e/e architectures," wp.mpi-inf.mpg.deM Ring, D Frkat, M SchmiedeckerACM Computer Science In Cars Symposium (CSCS 2018), 2018•wp.mpi-inf.mpg.de, pp. 123–4567, 2018, Accessed: Nov. 06, 2023. [Online]. Available: https://wp.mpi-inf.mpg.de/cscs/files/2018/09/02-Cybersecurity-Evaluation-of-Automotive-E_E-Architectures.pdf

[4] T. H. H. Aldhyani and H. Alkahtani, "Attacks to Automatous Vehicles: A Deep Learning Algorithm for Cybersecurity," Sensors, vol. 22, no. 1, Jan. 2022, doi: 10.3390/S22010360.

[5] M. Dibaei et al., "Attacks and defences on intelligent connected vehicles: a survey," Digital Communications and Networks, vol. 6, no. 4, pp. 399–421, Nov. 2020, doi: 10.1016/J.DCAN.2020.04.007.

[6] R. C. Shit, S. Sharma, K. Yelamarthi, and D. Puthal, "AI-Enabled Fingerprinting and Crowdsource-Based Vehicle Localization for Resilient and Safe Transportation Systems," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 7, pp. 4660–4669, Jul. 2021, doi: 10.1109/TITS.2021.3053942.

[7] M. J. Kang and J. W. Kang, “Intrusion detection system using deep neural network for in-vehicle network security,” PLoS One, vol. 11, no. 6, Jun. 2016, doi: 10.1371/JOURNAL.PONE.0155781.

[8] W. Xu, C. Yan, W. Jia, X. Ji, and J. Liu, “Analyzing and Enhancing the Security of Ultrasonic Sensors for Autonomous Vehicles,” IEEE Internet Things J, vol. 5, no. 6, pp. 5015–5029, Dec. 2018, doi: 10.1109/JIOT.2018.2867917.

[9] M. Kamal and D. A. Talbert, “Toward Never-Ending Learner for Malware Analysis (NELMA),” Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020, pp. 2291–2298, Dec. 2020, doi: 10.1109/BIGDATA50022.2020.9378357.

[10] Z. Cai, A. Wang, W. Zhang, M. Gruffke, H. S.-B. H. USA, and undefined 2019, “0-days & mitigations: roadways to exploit and secure connected BMW cars,” i.blackhat.comZ Cai, A Wang, W Zhang, M Gruffke, H SchweppeBlack Hat USA, 2019•i.blackhat.com, Accessed: Nov. 06, 2023. [Online]. Available: https://i.blackhat.com/USA-19/Thursday/us-19-Cai-0-Days-And-Mitigations-Roadways-To-Exploit-And-Secure-Connected-BMW-Cars-wp.pdf

[11] H. Lee, S. H. Jeong, and H. K. Kim, “OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame,” Proceedings - 2017 15th Annual Conference on Privacy, Security and Trust, PST 2017, pp. 57–66, Sep. 2018, doi: 10.1109/PST.2017.00017.

[12] A. Appathurai, G. Manogaran, N. C.-I. Networks, and undefined 2019, “Trusted FPGA-based transport traffic inject, impersonate (I2) attacks beaconing in the Internet of Vehicles,” Wiley Online LibraryA Appathurai, G Manogaran, N ChilamkurtiIET Networks, 2019•Wiley Online Library, vol. 8, no. 3, pp. 169–178, May 2018, doi: 10.1049/iet-net.2018.5171.

[13] W. Wu et al., “Sliding Window Optimized Information Entropy Analysis Method for Intrusion Detection on In-Vehicle Networks,” IEEE Access, vol. 6, pp. 45233–45245, Aug. 2018, doi: 10.1109/ACCESS.2018.2865169.

[14] B. Groza and P. S. Murvay, “Efficient Intrusion Detection with Bloom Filtering in Controller Area Networks,” IEEE Transactions on Information Forensics and Security, vol. 14, no. 4, pp. 1037–1051, Apr. 2019, doi: 10.1109/TIFS.2018.2869351.

[15] H. N. Noura, O. Salman, R. Couturier, and A. Chehab, “LoRCA: Lightweight round block and stream cipher algorithms for IoV systems,” Vehicular Communications, vol. 34, Apr. 2022, doi: 10.1016/J.VEHCOM.2021.100416.

[16] A. Castiglione, F. Palmieri, F. Colace, M. Lombardi, D. Santaniello, and G. D’Aniello, “Securing the internet of vehicles through lightweight block ciphers,” Pattern Recognit Lett, vol. 135, pp. 264–270, Jul. 2020, doi: 10.1016/J.PATREC.2020.04.038.

[17] P. Mundhenk et al., “Security in automotive networks: Lightweight authentication and authorization,” dl.acm.orgP Mundhenk, A Paverd, A Mrowca, S Steinhorst, M Lukasiewycz, SA Fahmy, S ChakrabortyACM Transactions on Design Automation of Electronic Systems (TODAES), 2017•dl.acm.org, vol. 22, no. 2, Mar. 2017, doi: 10.1145/2960407.

[18] X. Sun, B. Yan, X. Zhang, and C. Rong, “An integrated intrusion detection model of cluster-based wireless sensor network,” PLoS One, vol. 10, no. 10, Oct. 2015, doi: 10.1371/JOURNAL.PONE.0139513.

[19] S. Woo, H. Jo, D. L.-I. T. on intelligent, and undefined 2014, “A practical wireless attack on the connected car and security protocol for in-vehicle CAN,” ieeexplore.ieee.orgS Woo, HJ Jo, DH LeeIEEE Transactions on intelligent transportation systems, 2014•ieeexplore.ieee.org, vol. 16, no. 2, p. 993, Apr. 2015, doi: 10.1109/TITS.2014.2351612.

[20] T. Moulahi, S. Zidi, A. Alabdulatif, and M. Atiquzzaman, “Comparative Performance Evaluation of Intrusion Detection Based on Machine Learning in In-Vehicle Controller Area Network Bus,” IEEE Access, vol. 9, pp. 99595–99605, 2021, doi: 10.1109/ACCESS.2021.3095962.

[21] M. Emperuman, S. C.- Sensors, and undefined 2020, “Hybrid continuous density hmm-based ensemble neural networks for sensor fault detection and classification in wireless sensor network,” mdpi.comM Emperuman, S ChandrasekaranSensors, 2020•mdpi.com, vol. 20, no. 3, Feb. 2020, doi: 10.3390/s20030745.

[22] D. Perakovi et al., “Swarm Optimization and Machine Learning Applied to PE Malware Detection towards Cyber Threat Intelligence,” mdpi.comSJ Kattamuri, RKV Penmatsa, S Chakravarty, VSP MadabathulaElectronics, 2023•mdpi.com, vol. 12, no. 2, Jan. 2023, doi: 10.3390/electronics12020342.

[23] L. Yang, A. Moubayed, I. Hamieh, and A. Shami, “Tree-based intelligent intrusion detection system in internet of vehicles,” 2019 IEEE Global Communications Conference, GLOBECOM 2019 - Proceedings, Dec. 2019, doi: 10.1109/GLOBECOM38437.2019.9013892

[24] A . Alfardus and D. B. Rawat, “Intrusion detection system for can bus in-vehicle network based on machine learning algorithms.” 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2021. [Online]. Available:

[25] A. Alshammari, M. A. Zohdy, D. Debnath, and G. Corser, “Classification approach for intrusion detection in vehicle systems.” Wireless Engineering and Technology, vol. 9, pp. 79 – 94, 2018. [Online]. Available: https://www.scirp.org/journal/PaperInformation. aspx?PaperID=88247

[26] G. D’Angelo, A. Castiglione, and F. Palmieri, “A cluster-based multidimensional approach for detecting attacks on connected vehicles.” IEEE Internet of Things Journal, vol. 8, pp. 12518 – 12 527, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/ 9235336

[27] Refat, R.U.D., Elkhail, A.A., Hafeez, A., Malik, H. (2022). Detecting CAN Bus Intrusion by Applying Machine Learning Method to Graph Based Features. In: Arai, K. (eds) Intelligent Systems and Applications. IntelliSys 2021. Lecture Notes in Networks and Systems, vol 296. Springer, Cham. https://doi.org/10.1007/978-3-030-82199-9_49

[28] M. D. Hossain, H. Inoue, H. Ochiai, D. Fall, and Y. Kadobayashi, “LSTM-based intrusion detection system for in-vehicle can bus communications,” IEEE Access, vol. 8, pp. 185489–185502, 2020, doi: 10.1109/ACCESS.2020.3029307.

[29] X. Kan, Z. Zhou, L. Yao, and Y. Zuo, “Research on Anomaly Detection in Vehicular CAN Based on Bi-LSTM,” Journal of Cyber Security and Mobility, vol. 12, no. 5, pp. 629-652–629–652, Aug. 2023, doi: 10.13052/JCSM2245-1439.1251.

[30] N. Khatri, S. Lee, and S. Y. Nam, “Transfer Learning-Based Intrusion Detection System for a Controller Area Network,” IEEE Access, vol. 11, pp. 120963–120982, 2023, doi: 10.1109/ACCESS.2023.3328182.

[31] A. R. Javed, S. u. Rehman, M. U. Khan, M. Alazab and T. R. G, "CANintelliIDS: Detecting In-Vehicle Intrusion Attacks on a Controller Area Network Using CNN and Attention-Based GRU," in IEEE Transactions on Network Science and Engineering, vol. 8, no. 2, pp. 1456-1466, 1 April-June 2021, doi: 10.1109/TNSE.2021.3059881.

[32] https://ocslab.hksecurity.net/Dataset/CAN-intrusion-dataset

[33] https://www.can-cia.org/can-knowledge/can/can-history/

[34] https://iotmktg.com/understanding-the-role-of-electronic-control-units-ecus-in-modern-cars/

[35] https://ocslab.hksecurity.net/Datasets/car-hacking-dataset

[36] https://www.unb.ca/cic/datasets/ids-2017.html

[37] https://ocslab.hksecurity.net/Datasets/carchallenge2020