INTERNET OF VEHICLES SECURITY IMPROVEMENT BASED CONTROLLER AREA NETWORK AND ARTIFICIAL INTELLIGENCE

Ibraheem Hatem Mohammed

Department of Computer Communications Engineering, Al-Rafidain University College, Baghdad, Iraq

*E-mail of corresponding author: ibraheemdoser77@gmail.com

Ibraheem Hatem Mohamed 0000-0002-7362-7870

Resume

The Internet of Vehicle (IoV) is revolutionizing the automobile sector by allowing vehicles to interact between them and with roadside infrastructure. The Controller Area Network (CAN) is a vital component of such Autonomous vehicles (AVs), allowing communication between various Electronic Control Units (ECUs). However, the CAN protocol's intrinsic lack of security renders it opens to a variety of cyber-attacks, posing substantial hazards to both safety and privacy. This research proposes a defence mechanism for the real-time threat detection. It investigates the use of deep learning with multi-layer perceptron to improve the security of CAN networks inside the IoV framework. The suggested method is highly effective in identifying and mitigating potential risks, as evidenced by extensive testing on real-world CAN datasets.

Article info

Received 23 August 2024
Accepted 13 January 2025
Online 29 January 2025

Keywords:

autonomous vehicle
Internet of Vehicles (IoV)
Controller Area Network (CAN)
cyber security
deep learning
vehicular communication systems

Available online: https://doi.org/10.26552/com.C.2025.017

ISSN 1335-4205 (print version)
ISSN 2585-7878 (online version)

1 Introduction

With the latest technological developments, autonomous vehicles (AVs) that were formerly deemed science fiction have become a reality. Despite the fact that it is still in its early stages of development, the concept of autonomous vehicles is gaining global acceptance [1]. Autonomous cars are capable of sensing their environment and operating independently of people. A passenger is not required to drive the automobile at any time, nor is their presence within the vehicle required. An autonomous vehicle can go anywhere a conventional car can go and accomplish all the functions carried out by a competent human driver.

The applicability of a proposed solution to CVs or AVs depends on the specific nature of the proposed solution in terms of driving function, which can be classified hereunder:

- Conventional vehicles (non-AVs) participating in IoV rely on their human drivers for decision-making. The IoV solutions for these vehicles typically focus on" information sharing", warnings, or driver assistance, e.g., alerting drivers to nearby accidents or real-time traffic updates.
- Autonomous vehicles (AVs), on the other hand, can process the IoV data to directly execute actions such as rerouting or autonomous braking without human intervention.

According to the Society of Automotive Engineers (SAE), automation in autonomous cars can be categorized into six separate levels, ranging from SAE Level 0 (fully manual) to SAE Level 5 (entirely autonomous) [2-3]. Table 1 gives a description of these levels.

Although customers are not yet able to acquire fully autonomous vehicles, we are already in the phase of partially automated automobiles [3]. The Internet of Vehicles (IoV), an interconnected network of autonomous vehicles, roadside infrastructure, and components that communicate and interact with one another using wireless technology, is derived from the Internet of Things (IoT), with the objective of improving the effectiveness, efficiency, and safety of autonomous vehicles [4-5].

The electrical and electronic system of autonomous vehicles is a scattered and complicated network of Electronic Control Units (ECUs), sensors, and actuators. ECUs, which are computing units, are required to operate a specific subsystem and make critical autonomous driving decisions. They must interact with one another and exchange sensitive data using a set of standard protocols. The CAN bus is regarded as the de facto standard for the in-vehicle communication

Table 1 *SAE levels of driving automation*

SAE Level	Description
0	The driver is the one who controls the entire vehicle. In the form of alerts, such as lane departure or blind spot warnings, driver aid is offered.
1	With one autonomous function to help, the driver has complete control over the car. Adaptive cruise control, for example, uses automated acceleration and braking to maintain a safe distance from oncoming traffic. Alternatively, automated steering can be used, which involves the assistance of lane centering and other features to keep the car moving at a consistently high speed.
2	The driver has complete control over how the vehicle performs, with assistance from two automated operations such as steering, braking, and acceleration.
3	The car may function autonomously under a set of predetermined configurations, and the driver can take control of the vehicle at any time.
4	The vehicle may operate autonomously under specified settings, eliminating the need for the driver to oversee it. The car is extremely close to being totally autonomous.
5	At this level, the car is supposed to be completely autonomous and capable of operating without restrictions. There is no need for the driver to supervise it.

network, and it is ubiquitously used in almost all the automobiles [1, 6-7].

The remaining sections of this article are organized as follows. In Section 2 the relevant background knowledge is introduced. In Section 3 is discussed related work and their limitations. In Section 4, the specific design details of the intrusion detection model are described, while the performance evaluation is shown in Section 5, followed by the conclusion in Section 6.

2 Background knowledge

The power train, chassis and safety, body and comfort, and telematics and infotainment domains are the four main segments of an autonomous vehicle's internal communication system [8]. The airbag control, anti-lock braking, suspension, and Advanced Driver Assistance System, which perform real-time, safety-critical operations, are included in the Chassis and Safety domain. The Body and Comfort domain includes operations that do not frequently need real-time processing, such as in-vehicle climate control, seat control, door, window, or light control. The remote communication, information, and entertainment services are managed by the Telematics and infotainment domain. Each domain's performance and reaction time requirements vary depending on the function performed. Figure 1 depicts how these domains are integrated via various standards, like CAN, Media Oriented Systems Transport (MOST), and Local Interconnect Network (LIN). The CAN Bus protocol is most commonly used in the internal communication network of vehicles to support the aforementioned operations [9-14].

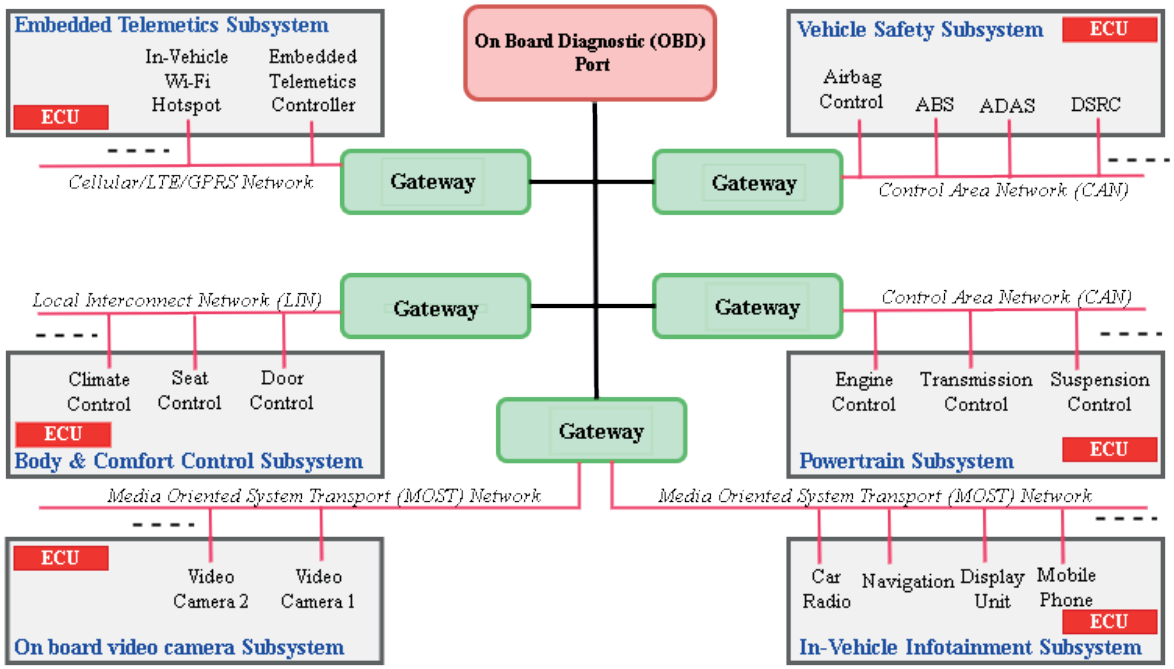

Figure 1 *In-Vehicle sub-systems adapted from [9]*

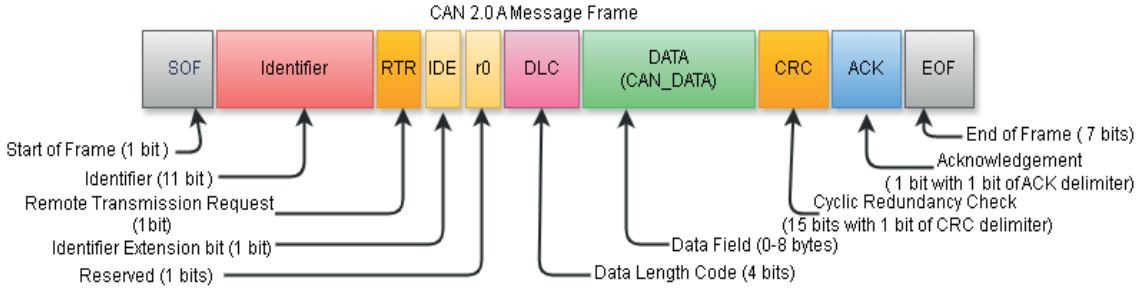

Figure 2 *The CAN frame structure*

Table 2 *Attacks executed and analysed on IoV subsystem*

Ref.	Attack Surface	Impact
Eisenberth et al. [15]	Keyless entry system	Control the door lock, unlock, and the engine
Koscher et al. [16]	Interfaces Infotainment using OBD-II/ USB port	CAN Bus injection, full access to the vehicle
Miller et al. [17]	OBD-II port	Control brakes, wheels, and get access to the CAN Bus of a real vehicle
Petit et al. [18]	LiDAR, Cameras Sensors	Signal jamming
Zorz et al. [19]	OBD-II Cellular Dongle	CAN Bus injection in Real vehicle
Palanca et al. [20]	OBD-II interface	DoS attack on CAN Bus
Woo et al. [21]	OBD-II interface	Replay, impersonation attack using the smartphone application
Nie et al. [22]	Wi-Fi, GSM	Replay, impersonation using access to CAN network using browser exploit
Mukherjee et al. [23]	OBD-II port	DoS attack by compromise ECU using the data link layer exploit

Figure 2 depicts the CAN data frame, which begins with an 1-bit Start of Frame (SOF) field, followed by an Arbitration field containing an 11- or 29-bit Identifier (ID) (CAN 2.0A has an 11-bit ID, whereas CAN 2.0B is the extended format with a 29-bit ID), and an 1-bit Remote Transmission Request (RTR).

Cybercriminals can get access to the internal communication network of the targeted vehicle using the aforementioned interfaces and carry out a range of attacks, including "replay", "DoS", or "spoofing" attacks [11]. Table 2 shows a list of effective attacks undertaken and analysed by various researchers on the IoV subsystem.

3 Literature review

Encryption, authentication, protocol stack redesign, and intrusion detection systems are among the suggested security options for the CAN Bus protection [8]. Some studies in recent years have focused on encryption techniques to secure the CAN system. However, adopting similar algorithms may need extra hardware or modifications to current ECUs. Intrusion detection methods that do not need changes to the network protocol or hardware are a better alternative for security inside AVs [7]. Researchers have utilised a range of ways to identify the CAN bus intrusions, including rule-based, machine learning-based, and other technologies. The authors of the article [24] present a deep learning-based Convolutional Neural Network (CNN) model for protecting the CAN bus in autonomous vehicles. The findings are also compared to various traditional methods; among them, the deep learning system achieves excellent accuracy. The study conducted in [25], describes another deep learning-based intrusion detection model that utilizes Long Short-Term Memory (LSTM) and CNNs network models. LSTM is a type of recurrent neural network (RNN) architecture used in deep learning, particularly for analysing sequential or time-series data, whereas identical research has been done by authors in [12, 26-29]. These studies have shown that standard CAN network data is growing increasingly sophisticated, and neural network-based models, particularly deep learning models, are the most effective way to handle the identified weaknesses in IoV security [30].

The majority of DNN-based solutions were built using the Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), which are extensively utilised to solve complicated problems in computer vision, text processing, audio recognition and classification, and so on. Due to their complexity, the DNNs often take a long time to train on input data.

They also require powerful computers with specialized processing units like Tensor, and Neural Processing Units. In this study, the deep learning-based IDS for CAN bus networks is presented, which outperforms previous work due to its simpler and more optimized network model.

4 Defence mechanism

To detect and categorize various assaults on autonomous vehicles by recognizing abnormal CAN network traffic patterns, the Multi-Layer Perceptron (MLP) based deep learning model is presented that may be deployed as an extra CAN Bus node, such as an OBD-2 dongle. It is more affordable and practical, and there is no need to modify the CAN Bus. It can detect and identify several types of attacks on the IoV CAN network. In Section 4.1 are described the methods to mitigate identified threats in the proposed IoV defence solution, while in Section 4.2 are described the realistic and most recent dataset used to train the model, whereas in section 4.3 is described the structure of the suggested solution.

4.1 Methods to mitigate identified threats in the proposed IoV defence solution

The proposed deep learning-based defence mechanism (e.g., using CNNs and LSTMs) provides an effective approach to intrusion detection. However, as IoV networks are highly dynamic and interconnected, they are susceptible to a wide range of cyber threats, such as DoS attacks, spoofing, man-in-the-middle attacks, data manipulation, and more.

This section outlines potential attack mitigation techniques that could complement or enhance the proposed solution to improve security within IoV systems.

4.2 Description of the dataset

In this study, the Canadian Institute for Cyber security (CIC) IoV2024 dataset is used [31-32]. It is a benchmark dataset; generated using a testbed of the real vehicle, to encourage the development of innovative security solutions for IoV processes, and it is published on the CIC dataset homepage. It contains traces for normal as well as five attack scenarios: DoS, "spoofing-steering wheel", "spoofing-RPM", "spoofing-GAS", and "spoofing-SPEED" attack, carried out by leveraging the unique characteristics of the CAN protocol in a real testbed of a Ford automobile equipped with all ECUs [31]. The original dataset was cleaned to eliminate noise (irrelevant data) and extract a specific amount of samples for each class to maintain the dataset balanced and in a format suitable for Deep Neural Networks [33-37]. Table 3 shows the class and sample's information, while Table 4 shows the retrieved features. The dataset is subsequently divided into training and testing datasets. The data is divided into 60:40 ratios, which means that 60% is utilised for model training and 40% is used for model validation.

4.3 Proposed deep learning model and experimental setup

Multi-Layer Perceptron (MLP), which serves as the foundation for this deep learning approach, is a neural network with multiple hidden layers. It is best suited for regression or classification problems in which inputs are allocated to a class. The neurons (or nodes) are arranged in different layers, as illustrated in Figure 3, and are connected to every neuron in the next layer, so the output of one neuron becomes the input of the next. Each connection between neurons has a weight, which is one of the variables that change throughout training. The weight of the link influences how much information is sent between neurons. Once a neuron gets inputs from all the other neurons linked to it, the output (y) is determined using the formula provided in Equation (1).

$$y = \sum_{i=1}^{N} (x_i * w_i) + b\,, \tag{1}$$

where x_i is an input of the neuron, w_i is the associated weight, and b is the bias. The output value (y) is then given to the activation function $g(y)$, which introduces nonlinearity into the neuron's output. Finally, the model employs the backpropagation algorithm to update the weights of the input layer based on the error at the output layer.

The proposed MLP-based deep learning model consists of an input layer, an output layer, and two dense hidden layers. The model receives input that is extracted from the data packet transmitted over the internal communication channel of the vehicle. Due to 153 features being extracted from the in-vehicle network traffic, the same number of neurons is inserted in the first layer. It is followed by the two dense hidden layers of two and eight neurons, respectively. Since the CAN network traffic is to be classified into six classes (TARGET LABEL in Table 4), the output layer is made up of completely linked six neurons. The model has a total of 386 trainable parameters. Figure 4 depicts the layer relationships, while Table 5 provides the model's summary.

The activation functions in MLP are critical for generating complex decisions and predictions. This article uses the ReLU activation function in the intermediate layers, which operates by performing a basic mathematical operation on the input value. If the input value is higher than or equal to zero, the output is the same as the input. If the input value is negative, the

Table 3 *Number of samples collected for each class*

S. No.	Class	# of Samples
1	BENIGN	80000
2	DoS	74660
3	SPOOFING_GAS	9991
4	SPOOFING_RPM	54899
5	SPOOFING_SPEED	24950
6	SPOOFING_STEERING_WHEEL	19976
	TOTAL	264476

Table 4 *Features extracted from the dataset*

S. No.	Features	Description
1	ID	Arbitration ID
2 to 9	"DATA_0" to "DATA_7"	1st to 8th byte of data transmitted through CAN data frame
10	LABEL	Type of traffic (Benign/Malicious)
11	TARGET LABEL	Six Specific Class of the traffic ("Benign", "DoS", "Spoofing_GAS", "Spoofing_RPM", "Spoofing_SPEED", and "Spoofing_STEERING_WHEEL")

Table 5 *Proposed MLP Model Summary*

Layer	Shape of the Output	Number of Parameters	Activation function
dense_136 (Dense)	(None, 2)	308	Rectified Linear Units (ReLU)
dense_137 (Dense)	(None, 8)	24	Rectified Linear Units (ReLU)
dense_138 (Dense)	(None, 6)	54	Softmax
Total parameters: 386 (1.51 KB) Trainable parameters: 386 (1.51 KB) Non-trainable parameters: 0 (0.00 Byte) Optimizer: "Adam", Loss function: "categorical_crossentropy", Performance Metrics: "Accuracy"			

result is zero. The mathematical representation of the ReLU function is as follows:

$$f(x) = \max(0,x). \tag{2}$$

The output layer employs the softmax activation function, which transforms the raw output scores of the model into probabilities, facilitating the distribution of these probabilities across various classes. This transformation is mathematically represented by:

$$Soft\max(Z_i) = \frac{e^{Z_i}}{\sum_{j=1}^{K} e^{Z_j}}. \tag{3}$$

Here, Z_i represents the input value for the Softmax function and is the output value of the node for i-th class at the output layer. K is the total number of nodes at the output layer.

Cross entropy measures the difference between the predicted probability and the true probability. Multiclass Cross-Entropy Loss, also known as categorical cross-entropy, is used as a loss function in the proposed deep-learning model. The loss function is mathematically represented by:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}(y_{i,j}\log(p_{i,j})). \tag{4}$$

Here, N is the number of instances in the dataset, K is the number of classes, $y_{i,j}$ is the true output for the i-th sample and j-th class, and $p_{i,j}$ is the predicted probability for i-th sample and j-th class. The Adam optimizer, which stands for "Adaptive Moment Estimation", is employed to iteratively minimize the loss function during training.

5 Results and comparative analysis

In the domain of deep learning, a perfect fit model is desired since it assures strong generalization and consistent performance on new data. Overfitting and underfitting, on the other hand, provide unreliable results and poor generalization. A well-performing deep learning model should have training and validation loss curves that converge to a comparable, low data point.

Table 6 *Training and Validation efficiency of the proposed model*

Epoch	Tr_loss	Tr_accuracy	Val_loss	Val_accuracy
	1.4293	0.4142	1.1413	0.4878
	1.0016	0.4883	0.8921	0.4884
	0.8277	0.4973	0.7646	0.5072
	0.6848	0.6311	0.5988	0.7578
	0.5273	0.7775	0.4573	0.8402
	0.3966	0.8415	0.3480	0.8423
	0.3133	0.8789	0.2821	0.9425
	0.2539	0.9430	0.2272	0.9428
	0.2022	0.9431	0.1805	0.9428
	0.1628	0.9594	0.1486	0.9805
	0.1366	0.9809	0.1274	0.9805
	0.1187	0.9809	0.1123	0.9805
	0.1056	0.9809	0.1010	0.9805
	0.0956	0.9809	0.0922	0.9805
	0.0877	0.9809	0.0850	0.9805
	0.0811	0.9809	0.0790	0.9805
	0.0757	0.9809	0.0739	0.9805
	0.0710	0.9809	0.0696	0.9805
	0.0670	0.9809	0.0658	0.9806
	0.0635	0.9809	0.0625	0.9806
	0.0604	0.9809	0.0595	0.9806
	0.0577	0.9809	0.0570	0.9806
	0.0553	0.9809	0.0547	0.9806
	0.0532	0.9809	0.0527	0.9806
	0.0514	0.9811	0.0509	0.9808
	0.0497	0.9811	0.0494	0.9809
	0.0483	0.9811	0.0480	0.9809
	0.0470	0.9811	0.0468	0.9809
	0.0459	0.9811	0.0457	0.9809
	0.0449	0.9812	0.0447	0.9809
	0.0439	0.9812	0.0435	0.9809
	0.0423	0.9812	0.0415	0.9809
	0.0399	0.9812	0.0387	0.9809
	0.0367	0.9812	0.0353	0.9809
	0.0333	0.9973	0.0318	0.9999
	0.0298	0.9999	0.0283	0.9999
	0.0265	0.9999	0.0251	0.9999
	0.0234	0.9999	0.0222	0.9999
	0.0207	0.9999	0.0196	0.9999
	0.0184	0.9999	0.0174	0.9999

Tr_loss: Training Loss; Tr_accuracy: Training Accuracy; Val_loss: Validation Loss; Val_accuracy: Validation Accuracy

This shows that the model is generalizing properly and not overfitting or underfitting. Analysing the behaviour of these curves during the training gives vital insights into the model's learning process and aids in making the required changes to increase performance. Table 6 shows the training and validation losses reported for each epoch throughout the simulation, which are also represented in Figure 5. The convergence of the training and validation loss curves demonstrates that the suggested model is learning the core trends in the data and generalizing successfully to the validation set. It indicates that the model is neither overfitting nor underfitting, as well.

Evaluating the performance of a deep learning model incorporates a series of procedures and metrics that offer a full picture of how well the model is doing.

Figure 6 depicts the confusion matrix obtained as a consequence of the simulation and used to calculate accuracy, precision, recall, and F1-score. Precision and recall are measures used to assess the effectiveness of a classification model, particularly in the cases with unbalanced classes or where different types of classification errors have varying costs.

The performance evaluation metrics, shown in Table 7, can be produced using Equations (5) to (8), where α, β, γ, and μ denote True Positive, True Negative, False Positive, and False Negative, respectively.

Precision and recall should both be high, however, they should be used in conjunction with other assessment measures like accuracy and F1-score to have a thorough view of a classifier's performance. The F1-score, which is the harmonic mean of the Precision and Recall values, provides a more balanced assessment of the model's performance.

$$Accuracy = \frac{\alpha + \beta}{\alpha + \beta + \gamma + \mu}, \tag{5}$$

$$Precision\,(P) = \frac{\alpha}{\alpha + \gamma}, \tag{6}$$

$$Recall\,(R) = \frac{\alpha}{\alpha + \mu}, \tag{7}$$

$$F1_{score}(F1) = 2 * \frac{Precesion * Recall}{Precesion + Recall}. \tag{8}$$

The results from the Confusion matrix (Figure 6) and Tables 7, 8, and 9 show that the proposed deep learning model can detect and classify an attack on an autonomous vehicle's CAN network with an average Recall of 0.999927477, Precision of 0.999930671, and F1-Score of 0.999929069. The model performed better than the benchmark research [31] in terms of accuracy, recall, precision, and F1-Score. It also outperformed previous studies [24] with the highest accuracy of 99.99%. Achieving an average accuracy of 99.99% is an impressive feat, but it is important to provide transparency regarding the factors that contributed to this result and to discuss any potential limitations or biases. To clarify achieved results some succeeded points can be addressed such as: dataset quality and size, model architecture, training process, evaluation metrics.

On the other hand, there are some potential limitations and biases that can be summarized due to: dataset bias concerning both causes (class imbalance or synthetic data), overfitting risk, limited attack types, real-world conditions, model complexity and deployment feasibility.

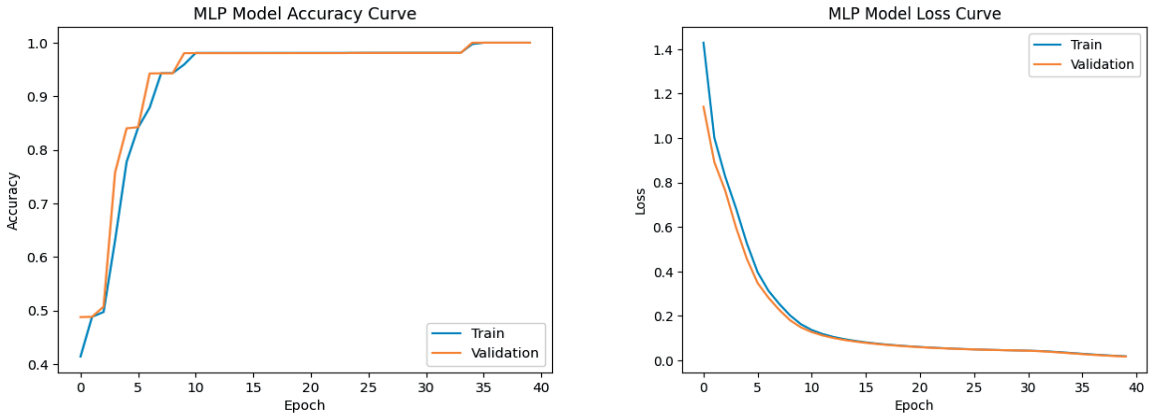

Figure 5 *(a) Accuracy curve, (b) Loss curve*

Actual	BENIGIN	31997	1	0	2	0	0
	DOS	0	29865	0	0	0	0
	SPOOFING_GAS	0	0	3996	0	0	0
	SPOOFING_RPM	0	0	0	21958	2	0
	SPOOFING_SPEED	0	0	0	0	9980	0
	SPOOFING_STEERING_WHEEL	0	0	0	2	0	7988
		BENIGIN	DOS	SPOOFING_GAS	SPOOFING_RPM	SPOOFING_SPEED	SPOOFING_STEERING_WHEEL
		Predicted					

Figure 6 *Confusion Matrix*

Table 7 *Recall, Precision, and F1-Score values for the proposed model*

Class	Recall	Precision	F1-Score
BENIGN	0.99990625	1	0.999953123
DoS	1	0.999966517	0.999983258
SPOOFING_GAS	1	1	1
SPOOFING_RPM	0.999908925	0.999817867	0.999863394
SPOOFING_SPEED	1	0.999799639	0.99989981
SPOOFING_STEERING_WHEEL	0.999749687	1	0.999874828
Macro Average	0.999927477	0.999930671	0.999929069

Table 8 *Proposed solution vs. the benchmark study*

Ref.	Accuracy	Recall	Precision	F1-Score
Neto et al. [31]	95%	0.68	0.74	0.63
Proposed Model	99.99%	0.999927477	0.999930671	0.999929069

Table 9 *Proposed solution vs. related work*

Ref.	Solution Type	Attack type	(Avg. Accuracy)
Sudhakar et al. [38]	Deep learning (CNN)	Malware	98.63%
Ahmed et al. [24]	Deep learning (CNN)	DoS, Fuzzy	96%
Neto et al. [31]	Deep learning (MLP)	DoS, Spoofing	95%
Proposed Model	Deep learning (MLP)	DoS, Spoofing	99.99%

6 Conclusion and scope for future work

The CAN protocol is a key component of the internal communication network of autonomous. However, the protocol's inherent lack of security makes it susceptible to a wide range of cyber threats, posing significant risks to both the safety and privacy of the driver, passenger or the vehicle itself.

This paper has explored the effectiveness of the DL-based approach to enhance the security of internal communication networks of autonomous vehicles in the IoV framework. A novel deep-learning based defence mechanism is proposed that provides the real-time threat detection. The findings highlight the potential of deep learning to significantly enhance the security of CAN networks in IoV, contributing to safer and more reliable vehicular communication systems.

The simulation results indicate that the proposed DL-based model is capable of successfully detecting and classifying attacks (DoS and spoofing) on the CAN network of an autonomous vehicle, with an average Recall of 0.999927477, Precision of 0.999930671, and F1-Score of 0.999929069. The proposed model outperformed benchmark studies and other related work in terms of accuracy, recall, precision, and F1-Score, achieving the highest accuracy of 99.99%. Future work should focus on improving the scalability of the proposed system and integrating it with broader IoV security frameworks to provide a holistic defence strategy.

Acknowledgment

The authors received no financial support for the research, authorship and/or publication of this article.

Conflicts of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] ISLAM, R., REFAT, R. U. D. Improving CAN bus security by assigning dynamic arbitration IDs. *Journal of Transportation Security* [online]. 2020, **13**(1-2), p. 19-31. ISSN 1938-7741, eISSN 1938-775X. Available from: https://doi.org/10.1007/s12198-020-00208-0

[2] O.-R. A. D. (ORAD) Committee. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. SAE International, 2021.

[3] GERSHON, P., SEAMAN, S., MEHLER, B. REIMER, B., COUGHLIN, J. Driver behavior and the use of automation in real-world driving. *Accident Analysis and Prevention* [online]. 2021, **158**, 106217. ISSN 0001-4575, eISSN 1879-2057. Available from: https://doi.org/10.1016/j.aap.2021.106217

[4] SHARMA, N. CHAUHAN, N., CHAND, N. Security challenges in Internet of Vehicles (IoV) environment. In: 2018 First International Conference on Secure Cyber Computing and Communication ICSCCC: proceedings [online]. IEEE. 2018. eISBN 978-1-5386-6373-8, p. 203-207. Available from: https://doi.org/10.1109/ICSCCC.2018.8703272

[5] ASWAL, K., DOBHAL, D. C., PATHAK, H. Comparative analysis of machine learning algorithms for identification of BOT attack on the Internet of Vehicles (IoV). In: 2020 International Conference on Inventive Computation Technologies ICICT: proceedings [online]. IEEE. 2020. eISBN 978-1-7281-4685-0, p. 312-317. Available from: https://doi.org/10.1109/ICICT48043.2020.9112422

[6] ZHAO, Q., CHEN, M., GU, Z., LUAN, S., CAN bus intrusion detection based on auxiliary classifier GAN and out-of-distribution detection. *ACM Transactions on Embedded Computing Systems* [online]. 2022, **21**(4), p. 1-30. ISSN 1539-9087, eISSN 1558-3465. Available from: https://doi.org/10.1145/3540198

[7] SUN, H., HUANG, W., WENG, J., LIU, Z., TAN, W., HE, Z., CHEN, M., WU, B., LI, L., PENG, X. CCID-CAN: cross-chain intrusion detection on CAN bus for autonomous vehicles. *IEEE Internet of Things Journal* [online]. 2024, p. 26146 - 26159. eISSN 2327-4662. Available from: https://doi.org/10.1109/JIOT.2024.3393122

[8] ALIWA, E., RANA, O., PERERA, C., Cyberattacks and countermeasures for in-vehicle networks. *ACM Computing Surveys* [online]. 2022, **5**(1), p. 1-37. ISSN 0360-0300, eISSN 1557-7341. Available from: https://doi.org/10.1145/3431233

[9] ZHANG, T., ANTUNES, H., AGGARWAL, S. Defending connected vehicles against malware: challenges and a solution framework. *IEEE Internet of Things Journal* [online]. 2014, **1**(1), p. 10-21. eISSN 2327-4662. Available from: https://doi.org/10.1109/JIOT.2014.2302386

[10] CHEN, S.-H., LIN, C.-H. R. Evaluation of DoS attacks on vehicle CAN bus system. In: *Recent advances in intelligent information hiding and multimedia signal processing. Vol. 110* [online]. PAN, J.-S., ITO, A., TSAI, P.-W., JAIN, L. C. (Eds.). Cham: Springer International Publishing, 2019. ISBN 978-3-030-03747-5, eISBN 978-3-030-03748-2, p. 308-314. Available from: https://doi.org/10.1007/978-3-030-03748-2_38

[11] JO, H. J., CHOI, W. A Survey of attacks on controller area networks and corresponding countermeasures. *IEEE Transactions on Intelligent Transportation Systems* [online]. 2022, **23**(7), p. 6123-6141. ISSN 1524-9050, eISSN 1558-0016. Available from: https://doi.org/10.1109/TITS.2021.3078740

[12] HANSELMANN, M., STRAUSS, T., DORMANN, K., ULMER, H. CANet: An unsupervised intrusion detection system for high dimensional CAN bus data. *IEEE Access* [online]. 2020, **8**, p. 58194-58205. eISSN 2169-3536. Available from: https://doi.org/10.1109/ACCESS.2020.2982544

[13] LIU, J., ZHANG, S., In-vehicle network attacks and countermeasures: challenges and future directions. *IEEE Network* [online]. 2017, **31**(5), p. 50-58. ISSN 0890-8044, eISSN 1558-156X. Available from: https://doi.org/10.1109/MNET.2017.1600257

[14] BOZDAL, M. SAMIE, M., ASLAM, S., JENNIONS, I. Evaluation of CAN bus security challenges. *Sensors* [online]. 2020, **20**(8), 2364. eISSN 1424-8220. Available from: https://doi.org/10.3390/s20082364

[15] EISENBARTH, T. KASPER, T., MORADI, A., PAAR, C., SALMASIZADEH, M., SHALMANI, M. T. M. On the power of power analysis in the real world: a complete break of the KeeLoq code hopping scheme. In: 28th Annual International Cryptology Conference Advances in Cryptology-CRYPTO 2008: proceedings [online]. Springer. 2008. ISBN 978-3-540-85173-8, eISBN 978-3-540-85174-5, p. 203-220. Available from: https://doi.org/10.1007/978-3-540-85174-5_12

[16] KOSCHER, K., CZESKIS, A., ROESNER, F., PATEL, S., KOHNO, T., CHECKOWAY, S., MCCOY, D., KANTOR, B., ANDERSON, D., SHACHAM, H., SAVAGE, S. Experimental security analysis of a modern automobile. In: 2010 IEEE Symposium on Security and Privacy: proceedings [online]. IEEE. 2010. ISSN 1081-6011, eISSN 2375-1207, ISBN 978-1-4244-6894-2, eISBN 978-1-4244-6895-9, p. 447-462. Available from: https://doi.org/10.1109/SP.2010.34

[17] VALASEK, C., MILLER, C. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA*. 2015, **2015**(S91), p. 1-91.

[18] PETIT, J., STOTTELAAR, B., FEIRI, M., KARGL, F. Remote attacks on automated vehicles sensors: experiments on camera and lidar. *Black Hat Eur*. 2015, **11**(2015), 995.

[19] ZORZ, Z. Backdooring connected cars for covert remote control - help net security. 2018. Retrieved August, 2020.

[20] PALANCA, A., EVENCHICK, E., MAGGI, F., ZANERO, S. A stealth, selective, link-layer denial-of-service attack against automotive networks. In: 14th International Conference, Detection of Intrusions and Malware, and Vulnerability Assessment DIMVA 2017: proceedings [online]. 2017. Springer. 2017. ISBN 978-3-319-60875-4, eISBN 978-3-319-60876-1, p. 185-206. Available from: https://doi.org/10.1007/978-3-319-60876-1_9

[21] WOO, S., JO, H. J., LEE, D. H. A practical wireless attack on the connected car and security protocol for in-vehicle CAN. *IEEE Transactions on Intelligent Transportation Systems* [online]. 2014, **16**(2), p. 993-1006. ISSN 1524-9050, eISSN 1558-0016. Available from: https://doi.org/10.1109/TITS.2014.2351612

[22] NIE, S., LIU, L., DU, Y. Free-fall: hacking tesla from wireless to can bus. *Defcon*. 2017, p. 1-16.

[23] MUKHERJEE, S., SHIRAZI, H., RAY, I., DAILY, J., GAMBLE, R. Practical DoS attacks on embedded networks in commercial vehicles. In: 12th International Conference Information Systems Security ICISS 2016: proceedings [online]. 2016. Springer. 2016. ISBN 978-3-319-49805-8, eISBN 978-3-319-49806-5, p. 23-42. Available from: https://doi.org/10.1007/978-3-319-49806-5_2

[24] AHMED, I., JEON, G., AHMAD, A. Deep learning-based intrusion detection system for internet of vehicles. *IEEE Consumer Electronics Magazine* [online]. 2023, **12**(1), p. 117-123. ISSN 2162-2248, eISSN 2162-2256. Available from: https://doi.org/10.1109/MCE.2021.3139170

[25] ALLADI, T., KOHLI, V. CHAMOLA, V., YU, F. R. A deep learning based misbehavior classification scheme for intrusion detection in cooperative intelligent transportation systems. *Digital Communications and Networks* [online]. 2023, **9**(5), p. 1113-1122. ISSN 2468-5925, eISSN 2352-8648. Available from: https://doi.org/10.1016/j.dcan.2022.06.018

[26] HOSSAIN, M. D., INOUE, H. OCHIAI, H., FALL, D., KADOBAYASHI, Y. LSTM-Based intrusion detection system for in-vehicle Can bus communications. *IEEE Access* [online]. 2020, **8**, p. 185489-185502. eISSN 2169-3536. Available from: https://doi.org/10.1109/ACCESS.2020.3029307

[27] SONG, H. M., WOO, J., KIM, H. K. In-vehicle network intrusion detection using deep convolutional neural network. *Vehicular Communications* [online]. 2020, **21**, 100198. eISSN 2214-210X. Available from: https://doi.org/10.1016/j.vehcom.2019.100198

[28] ZHANG, J., LI, F., ZHANG, H., LI, R., LI, Y. Intrusion detection system using deep learning for in-vehicle security. *Ad Hoc Networks* [online]. 2019, **95**, 101974. ISSN 1570-8705, eISSN 1570-8713. Available from: https://doi.org/10.1016/j.adhoc.2019.101974

[29] ZHOU, A., LI, Z., SHEN, Y. Anomaly detection of CAN bus messages using a deep neural network for autonomous vehicles. *Applied Sciences* [online]. 2019, **9**(15), 3174. eISSN 2076-3417. Available from: https://doi.org/10.3390/app9153174

[30] MANSOURIAN, P., ZHANG, N., JAEKEL, A., KNEPPERS, M. Deep learning-based anomaly detection for connected autonomous vehicles using spatiotemporal information. *IEEE Transactions on Intelligent Transportation Systems* [online]. 2023, **24**(12), p. 16006-16017. ISSN 1524-9050, eISSN 1558-0016. Available from: https://doi.org/10.1109/TITS.2023.3286611

[31] NETO, E. C. P., TASLIMASA, H., DADKHAH, S., IQBAL, S., XIONG, P., RAHMAN, T., GHORBANI, A. A. CICIoV2024: advancing realistic IDS approaches against DoS and spoofing attack in IoV CAN bus. *Internet of Things* [online]. 2024, **26**, 101209. ISSN 2543-1536, eISSN 2542-6605. Available from: https://doi.org/10.1016/j.iot.2024.101209

[32] NETO, E. C. P., TASLIMASA, H., DADKHAH, S. IQBAL, S., XIONG, P., RAHMANB, T., GHORBANI, A. A. CIC IoV dataset 2024 [online]. Canadian Institute for Cybersecurity, 2024. Available from: https://www.unb.ca/cic/datasets/iov-dataset-2024.html

[33] SPITALOVA, Z. Vehicle-to-everything communication. *Communications - Scientific Letters of the University of Zilina* [online]. 2023, **25**(1), p. 24-35. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2023.017

[34] SABIHA, A. D., KAMEL, M. A., SAID, E., HUSSEIN, W. M. Path planning algorithm based on teaching-learning-based-optimization for an autonomous vehicle. *Communications - Scientific Letters of the University of Zilina* [online]. 2022, **24**(2), p. C33-C42. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2022.2.C33-C42

[35] MORGOS, J., KLCO, P., HRUDKAY, K. Artificial neural network based MPPT algorithm for modern household with electric vehicle. *Communications - Scientific Letters of the University of Zilina* [online]. 2022, **24**(1), p. C18-C26. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2022.1.C18-C26

[36] DANKO, M., HANKO, B., DRGONA, P. Optimized control of energy flow in an electric vehicle based on GPS. *Communications - Scientific Letters of the University of Zilina* [online]. 2021, **23**(1), p. C7-C14. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2021.1.C7-C14

[37] KAPOOR, J., PATHAK, A., RAI, M., MISHRA G. R. Vehicle cabin noise cancellation model using pre-filter for improved convergence rate and better stability. *Communications - Scientific Letters of the University of Zilina* [online]. 2023, **25**(1), p. C1-C12. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2023.003

[38] SUDHAKAR, KUMAR, S. An emerging threat fileless malware: a survey and research challenges. *Cybersecurity* [online]. 2020, **3**(1), 1. eISSN 2523-3246. Available from: https://doi.org/10.1186/s42400-019-0043-x