IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 9, No. 5, October, 2022, pp. 1298-1304
Journal homepage: http://iieta.org/journals/mmep

Congestion Intrusion Detection-Based Method for Controller Area Network Bus: A Case for KIA SOUL Vehicle

Kennedy Okokpujie[1,2,3*], Daniella Mughole[1,2,4], Joke A. Badejo[1,2], Emmanuel Adetiba[1,2,5]

[1] Department of Electrical and Information Engineering, Covenant University, Ota 11212, Ogun State, Nigeria
[2] Covenant Applied Informatics and Communication Africa Center of Excellence, Covenant University, Ota 11212, Ogun State, Nigeria
[3] Africa Centre of Excellence for Innovative & Transformative STEM Education, Lagos State University, Ojo 120101, Lagos State, Nigeria
[4] Génie Electrique et Informatique, Université Libre des Pays des Grands Lacs, BP 360 Goma, Goma 6110104, Democratic Republic of the Congo
[5] HRA, Institute for Systems Science, Durban University of Technology, Durban 4001, South Africa

Corresponding Author Email: kennedy.okokpujie@covenantuniversity.edu.ng

https://doi.org/10.18280/mmep.090518

Received: 6 July 2022
Accepted: 23 August 2022

Keywords:
attacks, Controller Area Network (CAN) bus, deep feedforward neural network, long short-term memory, intrusion detection, in-vehicle network

ABSTRACT

In the vehicle industry, connectivity and autonomy are becoming increasingly important features. One of the most used protocols for in-vehicle communication is the Controller Area Network (CAN) bus which manages the communication between networked components. However, the CAN bus, despite its critical importance, lacks sufficient security features to protect its network as well as the overall car system. Thus, vehicle network security is becoming increasingly crucial. Methods of intrusion detection help to improve the security of the in-vehicle network. This work aims to provide a model that enables effective detection of attacks such as fuzzy, DoS, and impersonation using the Deep Feedforward Neural Network (DeepFNN) model as well as the Long Short-Term Memory model. Moreover, the LSTM model presents the most satisfying outcome in terms of precision and recall metrics.

1. INTRODUCTION

The functions and the complexity of modern vehicles have been revolutionized with the introduction of various technologies including advanced features such as automation and interconnectivity with the external world to improve safety and enable communication between vehicles [1]. Modern vehicles integrate new hardware, software, and protocols for communication. The automotive industry is improving with the use of the Controller Area Network (CAN) bus system as a central system for managing the communication between the set of networked components such as sensors, actuators, and Electronic Control Units (ECUs), and communication devices [2]. The modern car consists of about 50 to 100 ECUs, which communicate using CAN bus protocol. The intra-vehicle network allows sensors, ECUs, and actuators to share data, allowing the vehicle to function [3]. ECU controls and monitors the vehicle subsystem for the improvement of energy efficiency as well as vibration and noise reduction. Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) are automotive services that require computer-based inter-vehicle and intra-vehicle communications [4].

The Controller Area Network (CAN) bus protocol is one of the most important transformations introduced to the car industry [5]. Initially, the CAN bus protocol was engineered for an industrial machine, however, it has been adopted in the automotive industry for vehicle network systems because of its effectiveness, low cost, and centralized systems. It allows the coordination of movements between the engine, the brakes, the steering wheel, etc., which makes the modern car connected [6]. In CAN bus, a message or a frame is essentially composed of an ID (identifier), which indicates the communication's priority, Data Length Code (DLC), Data part, and Cyclic Redundancy Check (CRC) [7].

CAN bus is the communication protocol of in-vehicle networks that allows the message to broadcast to all nodes without authentication or encryption which makes it vulnerable and increases the probability of attacks [8]. An ECU can communicate with a vehicle's external element via a network system, which increases the CAN bus protocol's attack surface [9, 10]. Therefore, the CAN bus protocol security is a crucial concern that needs to be considered. Several ways compromise the security of the CAN bus system such as Denial of Service (DoS) attacks, Fuzzy attacks, and Impersonation attacks which are considered in this work.

1.1 Types of attacks

(a) Fuzzy attack

The attacker injects packets containing CAN ID and data at random. This exposes strange functionalities to all nodes, causing unexpected automobile behavior. A fuzzy attack occurs When an attacker notices and selects an identifier and data to create abnormal behavior [11]. These behaviors are dangerous and risk to life. For example, irregular lighting of the turn signal lamps, a huge steering wheel shake, spontaneous gear shift, etc. In contrast to the DoS assault,

which occupies the bus and so maintains crucial messages, the fuzzy attack paralyzes the functionalities of the vehicle [12]. To successfully attack the car, the attacker simply needs to send a malicious message in the same format as a legitimate CAN message all the time.

(b) Denial of Service (DoS) attack

In a DoS attack, the attacker interrupts the service briefly or indefinitely to make the network or system unavailable to authorized users [13]. Using a theoretical identity, high-priority messages are injected into the bus over a short period. These messages occupy space on the bus, obstructing the delivery of important frames. The attacker repeatedly injects messages which makes the bus communication busy while treating first the injected messages due to their high priority level, and thus degrades the performance of the system by delaying normal messages between nodes [14]. BUS DoS and ECU DoS are two different types of DoS attacks. BUS DoS is a simple attack that blocks ECUs from accessing the CAN bus, whereas an ECU DoS attack is aimed at a single ECU and influences safety [15].

(c) Impersonation attack

It entails assuming the identity of an authentic ECU, either physically or logically. The attacker can mimic an ECU by providing frames on its behalf [15]. As an ECU is designed to respond instantly by transmitting a data frame when it receives a remote frame, a late or no response will be considered an assault or dysfunction of the node [16]. However, in an impersonation attack, the attacker gives the response to the remote frame on behalf of the ECU.

Illicit access to a computer or network system is detected using an intrusion detection system. Precise and appropriate recognition of an attack is required to respond to a network intrusion using various techniques [17]. An intrusion detection system (IDS) records and monitors malicious activity on a computer or network. The detection approach, technology kind, and detection time are essential descriptive features of an IDS [18]. Unauthorized users who access network assets to cause damages are known as intruders.

New technology, software, and communication protocols increase the complexity of vehicles. From a security perspective, this results in an expansion of the attack area. The CAN standard definition does not include an internal security system. This increases the cybersecurity issues for modern vehicles.

Numerous publications on the subject are the result of these facts, which give rise to valid security issues. It reveals, particularly, that deep learning is an excellent technique for detecting CAN bus threats.

The security mechanism to protect the CAN bus against these attacks is a crucial need. Considering models are not always suitable for all applications, the technique used in this work differs from that of the authors in Lee et al. [19], who did not adopt a machine learning approach. On the other hand, in Okokpujie et al. [20], DeepFNN and SVM models were used for the classification. However, all features were used as one. This work focuses on the security of in-vehicle networks by providing a model that enables an efficient prediction and classification of the attacks. According to the results, for Normal, DoS, Fuzzy, and Impersonation respectively, the F-measures of each class obtained in the LSTM model are greater than the F-measures in the DeepFNN model in the proportion of 88.66%, 78.88%, 80.10%, and 73.78%. The LSTM model has consequently offered excellent performance in terms of the metrics, in comparison to the DeepFNN model performance.

The remainder of this paper is organized as follows: Section 2 reviews other researchers' works; the methodology of this work is presented in Section 3, Section 4 presents the discussion and the results of the model, and finally, Section 5 concludes the work.

2. REVIEW OF RELATED WORKS

The authors [19] implemented an intrusion-detection-based method that analyses the time interval and the offset ratio to determine if an event is an attack or not. The method is based on transmitting remote frames or messages to and from nodes. Furthermore, to identify various attacks, the authors analyzed metrics such as the lost reply ratio, the instant reply ratio, the time intervals, the offsets correlation coefficient, and the average time of responding. Therefore, the model is limited to a certain amount of data through additional nodes deployed. However, a machine learning approach could be used to detect intrusions effectively.

Okokpujie et al. [20] developed an anomaly-based detection technique of a CAN using two different models, mainly the Support Vector Machine (SVM) model and the Deep Feedforward Neural Network (DeepFNN) model. The authors presented a comparison between the results of each model. According to that, they showed that the SVM model provided a satisfactory classification to the DeepFNN model. In addition to that, the result obtained revealed that the performance of a model is not evaluated by its accuracy itself. However, the authors considered all the features of the dataset as one input that affected the performance evaluation of the models used.

Gmiden et al. [21] proposed a method for detecting intrusions in a CAN bus based on the analysis and monitoring of the time between the transmission of a message and the response. This method does not need to be implemented in each ECU and modify the CAN protocol. Moreover, an overview and classification of attacks were provided with the mechanisms to protect a system against them. However, this method does not perform on other types of attacks, such as the DoS attack which is one of the attacks presented in this work.

In Hossain et al. [6], an Intrusion Detection System (IDS) based on Long Short-Term Memory (LSTM) was developed to detect attacks and defend the CAN bus network against them. The authors developed a dataset by injecting three kinds of attacks; examples are fuzzing, spoofing, and DoS. The proposed LSTM model classified the attacks with an accuracy of 99.995% and provided an efficient detection rate. However, their dataset was unbalanced according to the proportion of data associated with each class, leading to the up-sampling process of data. It is important to note that the up-sampling process helps to adjust the proportion of data but can affect the performance of a model.

Dönmez [22] studied an intrusion detection system by analyzing the sequence of message identifiers with k as sequence length. The system aimed to detect anomalies in a CAN bus network. The authors considered that an attacker might prevent any device connected to the CAN system from delivering a message without affecting the order of messages in the queue. Therefore, their method yields small false-positive rates. In addition to that, they considered the CAN bus messages as input and processed them one at a time. During the training phase, a data structure internal to the program

records and saves every k-sequence encountered and its contents corresponds to the model being learned. However, the learning is limited to memorization and the intrusion detection system lacks the ability of generalization which is not efficient.

Jin [23] proposed an intrusion detection system based on the signature lightweight applied directly to the ECU to detect anomalies on a CAN bus generated by multiple attacks. The ID, correlation, time, value range, and amplitude of the context change are variables considered as signatures for the drop, replay, and temper attacks. To detect anomalies, different mechanisms were used for each signature. Furthermore, some parameters must be predefined to enable the ECU to calculate the signature when a new message is sent and evaluate the bus state. However, the detection of temper attacks is not effective.

Hossain et al [24] suggested a Long Short-Term Memory (LSTM)-based intrusion detection system that detects attacks against the CAN bus network without decoding the CAN bus's raw packets. The dataset used was composed of CAN's raw messages collected using Vehicle Spy 3 and the attacks, especially Fuzzing, Spoofing, and DoS, that were injected into a real car Toyota Hybrid. In addition to that, the authors demonstrated that the detection accuracy of the system can be significantly affected by the values of the hyperparameter. Moreover, the model was trained with eleven features including CAN ID, DLC, Data [D0-D7], as well as the label. However, the proportion of data in the dataset is low to ensure the scalability of the model.

Machine learning methods are effective at detecting attacks, but they were not used in some of the works reviewed. Furthermore, the low amount of data, as well as the processing of data used, had a significant impact on the performance of the developed models.

3. METHODOLOGY

To achieve this work, a CAN dataset for intrusion detection (OTIDS) was used. It contains a free attack state representing the normal events and different attacks namely fuzzy attack, DoS attack, and impersonation attack. The data was acquired from the KIA SOUL vehicle and recorded from CAN traffic through its On-Board Diagnostics (OBD-II) port. It is important to note that the attack messages were injected into the dataset. The dataset comprises different features that determine the state of an event whether it is normal or an attack. The different features are described as follows:

- ID: It is a hexadecimal number that identifies a payload within the CAN traffic.
- DLC: It is an integer number ranging from 0 to 8 that gives information about the number of bytes carried by a certain event (payload).
- Data: It is the payload conveyed within the CAN traffic. This feature is split into eight samples including Data [0], Data [1], Data [2], Data [3], Data [4], Data [5], Data [6], and Data [7].

The dataset contains 4,613,435 tuples divided into 2,369,397 for normal, 591,989 for Fuzzy attack, 656,578 for DoS attack, and 995,471 for impersonation attack CAN-Intrusion-Dataset (OTIDS) - Hacking and Countermeasure Research Lab, 2020 [25].

In addition to that, the above features define four different classes found in the dataset. These classes are given as follows:

- Normal: It represents the CAN attack-free messages.
- Fuzzy: It is the attack that represents random injected messages that can cause unexpected behaviour of the vehicle.
- Dos: It is the attack that represents high-priority information injected that can generate latencies and provoke a delay in getting a response to the driver's commands.
- Impersonation: It is the attack where the messages from a specific node are stopped by the attacker who takes control by sending wrong information to the CAN bus.

The data was analyzed using Python on Google Collaboratory. The data classification approaches used in this work is the Deep learning approach which comprises several phases including the data acquisition, data pre-processing, training and validation phases and testing phase as shown in Figure 1. The flowchart in Figure 1 depicts the steps taken for data classification according to the various phases of deep learning, from data acquisition to results.

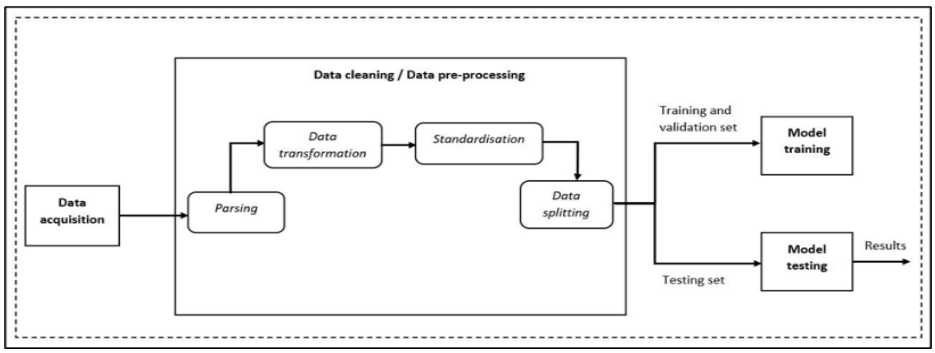

Figure 1. Data classification flow chart

- In the data acquisition phase, the data was available in terms of four distinct text files where each file is associated with a particular class.
- In the data pre-processing phase: The first step in this phase is the parsing process where the data from each text file was converted into Comma Separated Values (CSV) file before being merged and shuffled into a single file. Figure 2 illustrates the parsing process from the txt file to the CSV file. The second step is the data transformation where the data in hexadecimal type was converted into a decimal type. Figure 3 shows the transformation of the data from hexadecimal to decimal type. The third step consists of data standardization which is followed by the data splitting process in the proportion of 20% for the testing set and under the 80% dedicated to the training set, 25% is allocated to the validation. Note that the target classes were all one hot encoded.
- In the training phase, two deep learning models are considered, the Deep Feedforward Neural Network

(DeepFNN) and the Long Short-Term Memory (LSTM).

- The testing phase consists of the performance evaluation of the models.

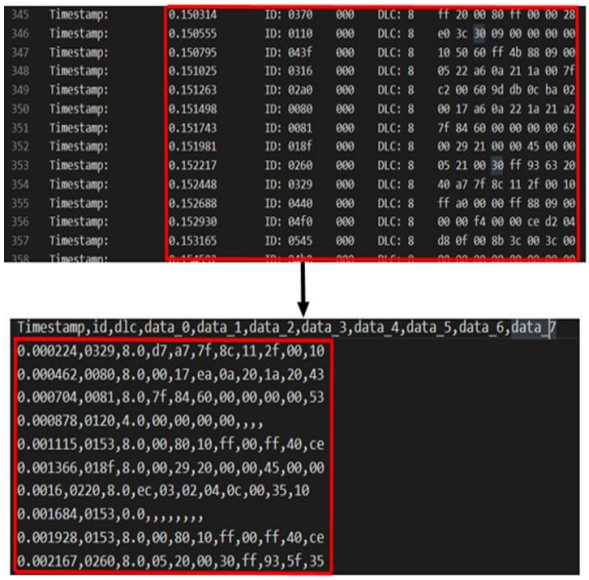

Figure 2. Parsing process

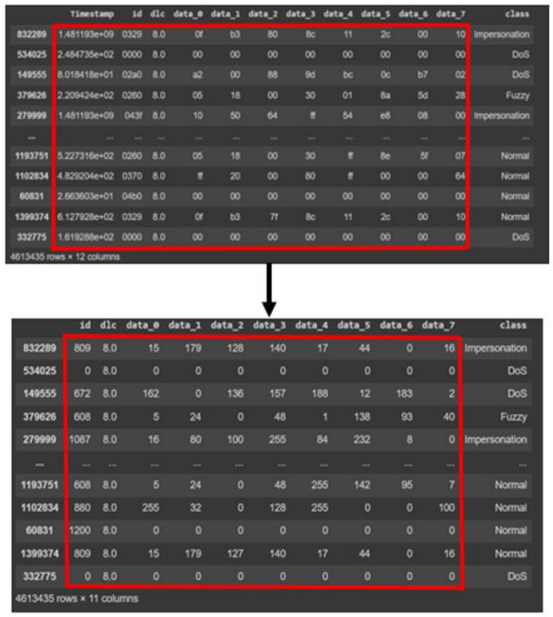

Figure 3. Data transformation process

The following presents the architecture of the models:

(a) Deep Feedforward Neural Network

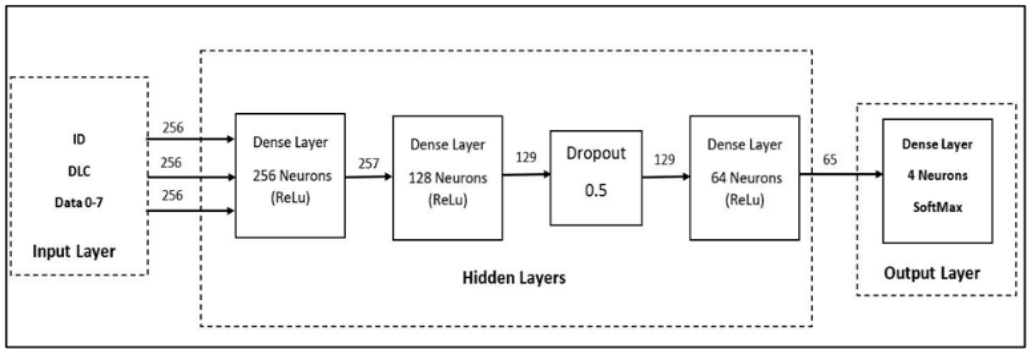

Figure 4. DeepFNN model architecture

Figure 4 shows the architecture of the DeepFNN model trained using the CAN dataset for intrusion detection (OTIDS). The model design has ten inputs including ID, DLC, and Data 0-7, three dense hidden layers where the first dense layer comprises 256 neurons, the second 128 neurons, and the third 64 neurons. All the layers have the Rectified Linear Unit (ReLu) as the activation function and between the second and the third hidden layer, a Dropout of 0.5 is set to avoid overfitting. Moreover, the output layer is composed of four neurons representing each class.

(b) Long Short-Term Memory

Figure 5 illustrates the architecture of the LSTM model which comprises ten inputs including ID, DLC, and Data 0-7. The hidden layers consist of an LSTM layer that counts 128 units with sigmoid as activation function and a Recurrent Dropout of 0.5 (R_D : 0.5). In addition to that, a dropout of 0.5 is set between the LSTM layer and the output layer to avoid overfitting. As for the DeepFNN, the output layer, in this case, contains also four neurons representing each class.

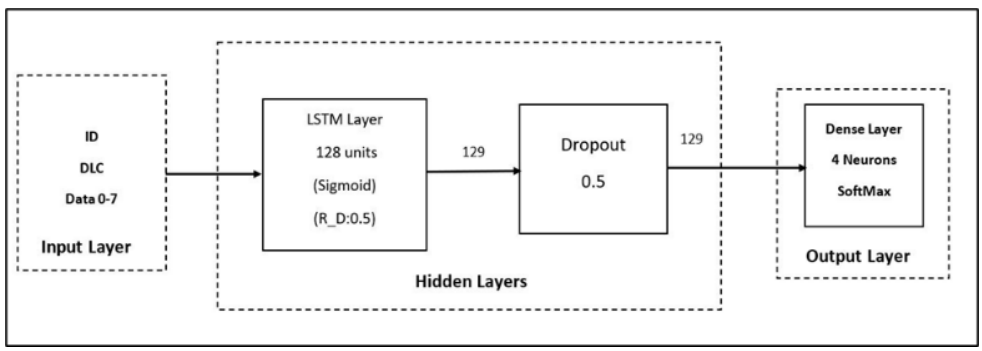

Figure 5. LSTM model architecture

4. RESULTS AND DISCUSSION

4.1 Results

Performance evaluation is a critical phase in anomaly detection. This section presents the performance metrics of the trained models in terms of precision, recall, accuracy, and F-measures. These metrics are obtained with the help of the confusion matrix describing the way that the models were able to classify the different classes. Eqns. (1) – (4) determine how each metric is calculated:

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

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

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

$$F - measures = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

where, TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.

The following confusion matrix describes the classification report of the DeepFNN model.

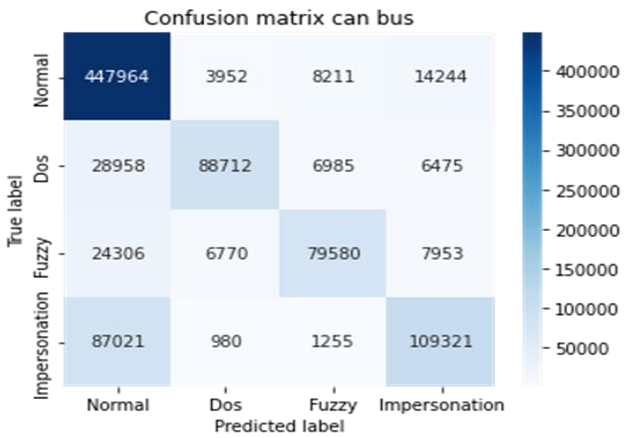

Figure 6. Confusion matrix of DeepFNN

According to the confusion matrix above in Figure 6, the DeepFNN model was able to classify all the classes with an accuracy of 78.637%. However, several misclassifications are observed and lead to the performance detailed in Table 1.

Table 1. Classification report of DeepFNN

Classes	Precision	Recall	F-measures
Normal	76.152%	94.433%	84.312%
DoS	88.346%	67.651%	76.625%
Fuzzy	82.869%	67.094%	74.151%
Impersonation	79.222%	55.052%	64.961%

The following confusion matrix describes the classification report of the LSTM model.

As for the DeepFNN model, Figure 7 presents the LSTM confusion matrix that shows its classification report in terms of TP, FP, TN, and FN. The model reports an accuracy of 83.74%. However, Table 2 presents the performance of the model as follows:

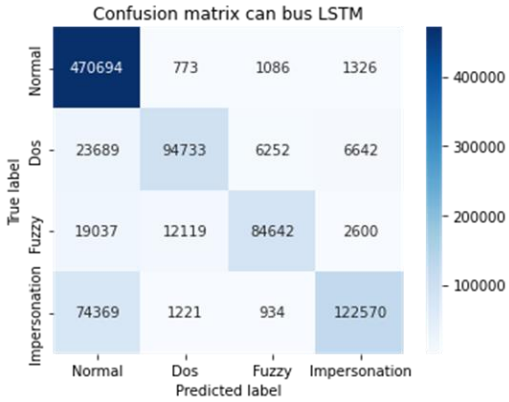

Figure 7. Confusion matrix of LSTM

Table 2. Classification report of LSTM

Classes	Precision	Recall	F-measures
Normal	80.07%	99.32%	88.66%
DoS	87.03%	72.14%	78.88%
Fuzzy	91.09%	71.48%	80.10%
Impersonation	92.06%	61.56%	73.78%

4.2 Discussion

Based on the results obtained, as the F-measures metric is proportionally dependent on the precision and the recall metrics, the F-measures determines better how the model can classify each class properly. The F-measures of each class obtained in the LSTM model is greater than the F-measures in the DeepFNN model in the proportion of 88.66%, 78.88%, 80.10%, and 73.78% for Normal, DoS, Fuzzy, and Impersonation respectively. Therefore, unlike the DeepFNN model performance, the LSTM model has provided a high performance in terms of metrics.

According to the result provided in Okokpujie et al. [20], the authors used the Support Vector Machine (SVM) model and the DeepFNN. After the performance evaluation of both models, the classification report revealed that the Radial basis kernel of SVM provided satisfactory results in terms of F-measures in the proportion of 64% and 80% for fuzzing and flooding attacks respectively. However, the result shows that the FNN model in this work outperforms the one proposed previously by Okokpujie et al. [26] and the LSTM model used in this work provided better performance than all other models. Moreover, this work considered eight independent data features compared to the authors in Okokpujie et al. [27] who used the eight data features as one feature.

5. CONCLUSIONS

Modern automobiles are subject to a variety of security vulnerabilities that attackers can use to obtain access to, and eventually, control them. Since attacks may have fatal consequences, effective attack detection is crucial. In-vehicle networks are typically not suitable for the use of standard security measures, even though they can defend targeted systems from external attacks. The study of intrusion detection is an attractive field that enables both academics and industry

to understand and improve the security of computer and network systems. There is no perfect way to evaluate a model's performance for a classification task, however, various metrics provide useful information about how a classification model performs. This work provided a general overview of attacks including DoS, Fuzzy, and Impersonation as well as their classification using deep learning techniques like DeepFNN and LSTM models. The dataset was pre-processed, and the models were able to classify with normal, DoS, Fuzzy, and Impersonation with an accuracy of 78.637% for the DeepFNN model and 83.74% for the LSTM model. In comparison to the methods employed in the literature, machine(deep) learning techniques are better suited for identifying attacks. According to the results obtained the deep learning models used in this work presented an excellent performance unlike the approach used in the literature for the same data. However, in terms of the F-measures metric, the LSTM model provided a satisfactory outcome, unlike the DeepFNN model.

Further work can be done by performing the LSTM model to detect other types of attacks such as flooding, spoofing, and replay attacks.

ACKNOWLEDGMENT

The authors acknowledge the sponsorship of the Covenant University Centre for Research, Innovation, and Discovery (CUCRID), Ota, Ogun State, Nigeria. The master's degree program of the second author is supported by the World Bank-funded Covenant Applied Informatics and Communication Africa Center of Excellence (CApIC-ACE) at Covenant University.

REFERENCES

[1] Al-Jarrah, O.Y., Maple, C., Dianati, M., Oxtoby, D., Mouzakitis, A. (2019). Intrusion detection systems for intra-vehicle networks: A review. IEEE Access, 7: 21266-21289. https://doi.org/10.1109/ACCESS.2019.2894183

[2] Han, S., Xie, M., Chen, H.H., Ling, Y. (2014). Intrusion detection in cyber-physical systems: Techniques and challenges. IEEE Systems Journal, 8(4): 1052-1062. https://doi.org/10.1109/JSYST.2013.2257594

[3] Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., Savage, S. (2010). Experimental security analysis of a modern automobile. In 2010 IEEE Symposium on Security and Privacy, pp. 447-462. https://doi.org/10.1109/SP.2010.34

[4] Biswas, S., Tatchikou, R., Dion, F. (2006). Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. IEEE Communications Magazine, 44(1): 74-82. https://doi.org/10.1109/MCOM.2006.1580935

[5] Alfardus, A., Rawat, D.B. (2021). Intrusion detection system for CAN bus in-vehicle network based on machine learning algorithms. In 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), pp. 0944-0949. https://doi.org/10.1109/UEMCON53757.2021.9666745

[6] Hossain, M.D., Inoue, H., Ochiai, H., Fall, D., Kadobayashi, Y. (2020). LSTM-based intrusion detection system for in-vehicle can bus communications. IEEE Access, 8: 185489-185502. https://doi.org/10.1109/ACCESS.2020.3029307

[7] Iehira, K., Inoue, H., Ishida, K. (2018). Spoofing attack using bus-off attacks against a specific ECU of the CAN bus. In 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 1-4. https://doi.org/10.1109/CCNC.2018.8319180

[8] Lin, C.W., Sangiovanni-Vincentelli, A. (2012). Cyber-security for the controller area network (CAN) communication protocol. In 2012 International Conference on Cyber Security, pp. 1-7. https://doi.org/10.1109/CyberSecurity.2012.7

[9] Han, M.L., Kwak, B.I., Kim, H.K. (2018). Anomaly intrusion detection method for vehicular networks based on survival analysis. Vehicular Communications, 14: 52-63. https://doi.org/10.1016/j.vehcom.2018.09.004

[10] Seo, E., Song, H.M., Kim, H.K. (2018). GIDS: GAN based intrusion detection system for in-vehicle network. In 2018 16th Annual Conference on Privacy, Security and Trust (PST), pp. 1-6. https://doi.org/10.1109/PST.2018.8514157

[11] Jeong, S., Jeon, B., Chung, B., Kim, H.K. (2021). Convolutional neural network-based intrusion detection system for AVTP streams in automotive Ethernet-based networks. Vehicular Communications, 29: 100338. https://doi.org/10.1016/j.vehcom.2021.100338

[12] Bi, Z., Xu, G., Xu, G., Tian, M., Jiang, R., Zhang, S. (2022). Intrusion detection method for in-vehicle can bus based on message and time transfer matrix. Security and Communication Networks, 2022: Article ID 2554280. https://doi.org/10.1155/2022/2554280

[13] Borkar, A., Donode, A., Kumari, A. (2017). A survey on Intrusion Detection System (IDS) and Internal Intrusion Detection and protection system (IIDPS). In 2017 International Conference on Inventive Computing and Informatics (ICICI), pp. 949-953. https://doi.org/10.1109/ICICI.2017.8365277

[14] Studnia, I., Nicomette, V., Alata, E., Deswarte, Y., Kaâniche, M., Laarouchi, Y. (2013). Survey on security threats and protection mechanisms in embedded automotive networks. In 2013 43rd Annual IEEE/IFIP Conference on Dependable Systems and Networks Workshop (DSN-W), pp. 1-12. https://doi.org/10.1109/DSNW.2013.6615528

[15] Boudguiga, A., Klaudel, W., Boulanger, A., Chiron, P. (2016). A simple intrusion detection method for controller area network. In 2016 IEEE International Conference on Communications (ICC), pp. 1-7. https://doi.org/10.1109/ICC.2016.7511098

[16] Cho, K.T., Shin, K.G. (2016). Fingerprinting electronic control units for vehicle intrusion detection. In 25th USENIX Security Symposium (USENIX Security 16), pp. 911-927.

[17] Ahmad, I., Abdullah, A.B., Alghamdi, A.S. (2009). Application of artificial neural network in detection of DOS attacks. In Proceedings of the 2nd International Conference on Security of Information and Networks, pp. 229-234. https://doi.org/10.1145/1626195.1626252

[18] Liao, H.J., Lin, C.H.R., Lin, Y.C., Tung, K.Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1): 16-24. https://doi.org/10.1016/J.JNCA.2012.09.004

[19] Lee, H., Jeong, S.H., Kim, H.K. (2017). OTIDS: A novel

intrusion detection system for in-vehicle network by using remote frame. In 2017 15th Annual Conference on Privacy, Security and Trust (PST), pp. 57-5709. https://doi.org/10.1109/PST.2017.00017

[20] Okokpujie, K., Kennedy, G.C., Nzanzu, V.P., Molo, M.J., Adetiba, E., Badejo, J. (2021). Anomaly-based intrusion detection for a vehicle can bus: A case for hyundai avante cn7. Journal of Southwest Jiaotong University, 56(5): 144-156. https://doi.org/10.35741/issn.0258-2724.56.5.14

[21] Gmiden, M., Gmiden, M.H., Trabelsi, H. (2016,). An intrusion detection method for securing in-vehicle CAN bus. In 2016 17th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), pp. 176-180. https://doi.org/10.1109/STA.2016.7952095

[22] Dönmez, T.C. (2021). Anomaly detection in vehicular CAN bus using message identifier sequences. IEEE Access, 9: 136243-136252. https://doi.org/10.1109/ACCESS.2021.3117038

[23] Jin, S., Chung, J.G., Xu, Y. (2021). Signature-based intrusion detection system (IDS) for in-vehicle can bus network. In 2021 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1-5. https://doi.org/10.1109/ISCAS51556.2021.9401087

[24] Hossain, M.D., Inoue, H., Ochiai, H., Fall, D., Kadobayashi, Y. (2020). Long short-term memory-based intrusion detection system for in-vehicle controller area network bus. In 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), pp. 10-17. https://doi.org/10.1109/COMPSAC48688.2020.00011

[25] "CAN-intrusion-dataset (OTIDS) - Hacking and Countermeasure Research Lab." (2020). https://sites.google.com/a/hksecurity.net/ocslab/Dataset/CAN-intrusion-dataset, accessed on Apr. 26, 2022.

[26] Okokpujie, K., Noma-Osaghae, E., John, S.N., Ndujiuba, C., Okokpujie I.P. (2021). Comparative analysis of augmented datasets performances of age invariant face recognition models. Bulletin of Electrical Engineering and Informatics, 10(3): 1356-1367. https://doi.org/10.11591/eei.v10i3.3020

[27] Okokpujie, K., John, S., Ndujiuba, C., Noma-Osaghae, E. (2020). Development of an adaptive trait-aging invariant face recognition system using convolutional neural networks. In: Kim, K., Kim, HY. (eds) Information Science and Applications. Lecture Notes in Electrical Engineering, vol 621. Springer, Singapore. https://doi.org/10.1007/978-981-15-1465-4_41