6th International Conference on Smart Cities, Internet of Things and Applications (SCIoT2022) September 14-15- Ferdowsi University of Mashhad.

Ensemble Learning-based Intrusion Detection System for Autonomous Vehicle

Jay Thaker
Department of Comp. Sci. & Engg.
Institute of Technology
Nirma University, Ahmedabad, India
20mcei13@nirmauni.ac.in

Nilesh Kumar Jadav
Department of Comp. Sci. & Engg.
Institute of Technology
Nirma University, Ahmedabad, India
21ftphde53@nirmauni.ac.in

Sudeep Tanwar
Department of Comp. Sci. & Engg.
Institute of Technology
Nirma University, Ahmedabad, India
sudeep.tanwar@nirmauni.ac.in

Pronaya Bhattacharya
Department of Comp. Sci. & Engg.
Institute of Technology
Nirma University, Ahmedabad, India
pronaya.bhattacharya@nirmauni.ac.in

Hossein Shahinzadeh
Department of Electrical Engg.
Amirkabir University of Technology
Tehran Polytechnic, Tehran, Iran
h.s.shahinzadeh@ieee.org

2022 Sixth International Conference on Smart Cities, Internet of Things and Applications (SCIoT) | 978-1-6654-8891-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/SCIoT56583.2022.9953697

***Abstract*—Autonomous vehicles (AVs) are a potential technology for improving safety and driving efficiency in intelligent transportation systems (ITSs). However, AVs are subject to various cyber-attacks, comprising denial-of-service, spoofing, sniffing, and cross-site scripting. To overcome the security issues in AV, this paper proposed an intelligent framework that impersonates the intrusion detection system (IDS) that intelligently classifies malicious and non-malicious data requests of AVs. For that, we utilized ensemble-based machine learning classifiers, such as decision tree, random forest, extra tree, XGboost, K-nearest neighbor, and support vector machine (SVM), to train them on different attacks and simultaneously use their learning for classification. The proposed model is bifurcated into different phases of machine learning, like data collection, pre-processing, and prediction. Finally, we evaluate the ensemble models using different evaluation metrics, such as accuracy, precision, recall, and f1-score. XGBoost outperforms other classifiers in terms of accuracy, i.e., 98.57%, which benefits from attaining a high detection rate and low computational cost at the same time for the AV systems.**

***Index Terms*—Tree based learning, Autonomous Vehicle, Internet of Vehicle, IDS,**

I. Introduction

Modern technologies are updating and revolutionizing different sectors, such as agriculture, health, industry, and many more, to offer satisfactory services to the end-users and raise the nation's economy. An autonomous vehicle (AV) is one such technology that is constantly overseen to upgrade and enhance its performance to reduce the accident severity and offer an automated driving experience by utilizing the essential benefits of artificial intelligence (AI) [9]. The deployed sensors in AV continually monitor and supervise the surrounding environment of the AV to detect any dire experience, like road blockage, pedestrians walking on the road, traffic jams, etc., and alert the AV driver at the same time. However, the sensor communication utilizes the public Internet, which can arise security issues, such as distributed-denial-of-service (DDoS), man-in-the-middle (MiTM), session hijacking, sniffing, cross-site scripting, and many more. Attackers can exploit the AV by discovering the software/hardware vulnerability or misconfiguring AVs' communication links. Not only that, it can perform an active attack where it modifies the information of enroute data packets, which directly leads to the risk of AV. Further, controller area network as a communication protocol is used between different electronic controller units (ECU) of AV. It facilitates an error monitoring and detection mechanism for secure transmission of information and can help to reduce electric wiring cost and massive weight resulting into a low complexity system. However, communication between ECU through the CAN bus protocol can make them vulnerable to various security attacks.

To overcome the aforementioned security loopholes in AV, the research community has provided different security solutions that eradicate the security and privacy constraints and enhance the performance of the AV systems [7] [4]. The scientific community has incorporated blockchain as a promising solution to confront security attacks in AV systems. For instance, the authors of [11] proposed a secure architecture that ensures the security of sensing and tracking in AVs. They employed blockchain technology to securely store the sensing and tracking object into the immutable ledger. Further, the authors of [6] present a secure smart contract-based blockchain network to securely store the detected object by the AV. The results show that their proposed method outperforms the baseline work in terms of efficiency and reliability. However, blockchain technology leverages higher computational complexity that is not feasible for a critical application like AV. Therefore, the researchers have adopted AI-based solutions, for example Cui *et al.* [3] proposed a proactive mechanism against security threats; their main objective is to analyze the security implication on sensors and actuators. The authors have used low computational resources and a lightweight algorithm that requires periodic checks

978-1-6654-8891-4/22/$31.00 ©2022 IEEE

because it collects information from different sensor nodes. Then, Taylor *et al.* in [14] presented a detection method by inspecting and observing the data packets and message frequency from one sensor node to another. However, they have not specified data integrity or data manipulation attacks that jeopardize the performance of AV systems.

Seo *et al.* in [12], employed a generative adversarial network (GAN)-based IDS to confront the security attacks in the AV. This replaces the conventional IDS, which utilizes the inefficient signature-based method to classify the attacks. The GAN-based IDS outperforms the traditional IDS in terms of accuracy and shows a better detection rate to detect unknown attacks. However, GANs are computationally expensive, which is not a suitable option to use with critical systems, i.e., AV systems. Further, the authors of [8] proposed an adaptive network-based fuzzy inference system that acts as a feature database. The authors have deployed the inference system in the real-time environment of AV, where attacks and their detection systems are deployed on the gateway. Their proposed approach shows a significant improvement in detecting the anomalous behavior of the CAN network.

Additionally, a few of the researchers have presented an algorithmic aspect based on payloads to detect the attack in AV. For instance, the authors in [10] investigated an entropy-based learning mechanism to analyze and identify the attack. They show the importance of their proposed approach in a system where there is a requirement for automation. They found essential parameters that the attackers can easily leverage and try to protect it from the attackers by their proposed solution. Despite their significant results, they have not explored the proposed solution's data manipulation attack and overall computational complexity. Later, Stabili *et al.* in [13] presented a detection algorithm that particularly observes the manipulated CAN message in the AV systems. The algorithm has low computational complexity by operating on small processing capabilities. The results show that the proposed approach outperforms in detecting the fuzzing attack by 100%. However, the authors have not explored other attacks, such as DDoS, MiTM, or sniffing, due to which the attackers can still exploit the AV. Therefore, this is a stringent requirement for an AI-based model that encourages low computation and high detection rates to efficiently classify different types of security attacks that deteriorate the AV systems' performance [16]. Further, the authors of [15] proposed predictive and encoder-based deep learning algorithms to secure the AV cruise control from different security attacks. Then, Ref. [2] studied the CAN bus protocol and its security implications on AV. The authors used long short term memory LSTM model to identify intrusions in the AV communication. The result shows that the proposed system outperforms in terms of accuracy, i.e., 97.30%. However, the proposed algorithms are computationally expensive and therefore, there is a need for a simplistic approach that can detect and secure the AV from different security vulnerabilities.

To overcome the aforementioned security issues, we proposed an AI-based IDS that employs ensemble methods to adroitly classifies the malicious and non-malicious CAN traffic of AV. Toward this goal, first, we have simulated the CAN traffic in the ICSim simulator, where different attacks are performed, such as resource starvation, injection, and MiTM attack, then the network traffic is captured inside the Wireshark tool. Next, the generated traffic is converted into a comma-separated value (CSV) file with both malicious and non-malicious CAN traffic of AV that is fed as an input to the AI models. Then, the dataset is pre-processed using standard Pandas, Numpy, and Python libraries, and data is normalized using Min-Max normalization. After pre-processing, the dataset is assessed for imbalance classification, which is resolved using the synthetic minority oversampling technique (SMOTE). Then based on the output of SMOTE, we proceed with feature selection and data splitting to reduce an input variable and collect only relevant data for the efficient classification. Next, we train the ensemble-based classifiers, i.e., decision tree, random forest, extra tree, XGboost, K-nearest neighbor, and SVM on a standard dataset, i.e., CICIDS2017. Then, these pre-trained classifiers use their learning to classify the unknown or new attack from the newly generated CAN traffic by the ICSIm simulator. Finally, the proposed framework is evaluated with different performance metrics, such as accuracy, precision, recall, and f1-score. The result shows that the XGboost outperforms other classifiers in terms of accuracy, i.e., 98.57%.

A. Research contributions

- We proposed an IDS-based framework that efficiently classifies the malicious and non-malicious CAN traffic of the AV. For that, a standard dataset, i.e., CICIDS2017, is utilized to train different ensemble-based AI classifiers, such as decision tree, extra tree, random forest, XGBoost, SVM, and KNN.
- Additionally, a new dataset is generated using the ICSim simulator that impersonates the CAN traffic of AV. This helps in enhancing the performance of the proposed framework by learning new features and security attacks from the generated dataset.
- Finally, the proposed framework is evaluated against different performance parameters, such as accuracy, precision, recall, and f1-score. The XGBoost classifier outperforms other classifiers in terms of accuracy, i.e., 98.57%.

B. Organization

The rest of the paper is organized as follows. Section 2 introduces the system model and problem formulation. Section 3 presents the proposed framework. Section 4 presents the results and discussion. Finally, Section 5 gives the concluding remarks.

II. System model and problem formulation

This section describes the system model for our proposed framework, i.e., to efficiently detect the malicious CAN traffic

from the AV system. Fig. 1 shows the AI-based proposed framework, which comprises the AV and different sensors, such as $\{s_1, s_2, \ldots, s_n\} \in S$ associated with each other. The s_1 communicates with s_2 to share real-time monitored data (D) of AV by using the CAN network.

$$s_1 \xrightarrow[\text{monitored data (D)}]{\text{via CAN traffic}} s_2 \tag{1}$$

An attacker can use (s_k) vulnerable sensor, $s_k \notin S$ to exploit the CAN traffic and jeopardize the performance of the AV systems.

$$s_k \xrightarrow[\text{forged data}]{\text{via manipulated CAN traffic}} s_2 \tag{2}$$

To overcome the aforementioned security issues, there is a requirement to employ AI intelligence that efficiently bifurcates malicious (τ), and non-malicious (τ') CAN traffic and maximizes the security of real-time monitored data (D) of the AV systems. In that direction, we formulated an objective functions described as follows.

$$\mathbb{C} \longrightarrow \tau, \tau' \tag{3}$$

$$O = \max \sum_{i=1}^{n} \text{Secure}(D) \tag{4}$$

where is the AI classifier, τ, and τ' are the malicious and non-malicious CAN traffic of AV. To achieve this objective function, we utilize CICIDS2017 data to train different AI classifiers. Subsequently, a dataset is generated from the ICSim simulator that gives both τ and τ' data of the AV systems. Further, the data is pre-processed by observing and rectifying the missing values, normalization, outliers, and class imbalance problems from both datasets. Then, the ensemble-based AI classifiers (C) attempt to correctly classify the τ and τ' of the AV's CAN traffic. Lastly, the proposed ensemble-based AI intelligence framework is assessed using various performance metrics, such as precision, recall, accuracy, and f1-score.

III. The proposed framework

Fig. 1 displays the proposed framework, which is divided into three layers, i.e., data collection, data pre-processing, and data classification layer. A detailed description of each layer is as follows.

A. Data collection layer

This layer comprises the AV and its deployed sensors (S), such as light detection and ranging (LiDAR), global positioning systems (GPS), radio detection and ranging (RADAR), and many more to collect the surrounding environment data, like pedestrians and vehicles on the road, traffic jams, etc., additionally, it accumulates the current working conditions of AV, such as brakes, wheel alignment, steering movement, flattens tires, and engine revolution per minute (RPM). The sensor shares this real-time monitored data (D) with each other to make a soothing driverless experience in the AV. The sensors use the CAN protocol to transmit the D to other sensors to accomplish the shared objective of the intelligent transportation system. However, the CAN traffic is manipulated with numerous attack types, like DDoS, MiTM, sniffing, data integrity attacks, etc., that hinder the performance of AV. Therefore, there is a requirement for an intelligent IDS that constantly monitors the CAN traffic and subsequently classifies the malicious (τ) and non-malicious (τ') CAN traffic.

B. Data pre-processing layer

This layer is used to clean and prepare the dataset for the classification layer. The proposed framework utilizes two datasets, i.e., a standard CICIDS2017 [1] and the manually generated dataset. Both datasets have to pre-process in order to remove the inconsistencies that occlude the performance of classification results. First, the dataset is analyzed for missing and NA values which are resolved by employing Python, Pandas, and NumPy libraries that have essential functions, such as fillna(), isnull().sum(), dropna(), and filling with regression value. Next, the dataset is normalized using Min-Max normalization, wherein a specific column value is larger or smaller than others; it severely affects the classification results. Incorporating Min-Max normalization standardizes the dataset and removes the outliers.

$$N_d = \frac{X - Min}{Max - Min} \tag{5}$$

where, N_d is the normalized data and X represents the specific column value of the dataset. Further, the dataset is examined for the class imbalance problem, where if the count of one class label (y=1) is larger than the class label (y=0), then it directly affects the performance of classification. Therefore, the synthetic minority oversampling technique (SMOTE) is utilized to oversample the data in the dataset. Both datasets are gone through pre-processing steps to improvise the prediction of classifiers.

C. Data classification layer

Once both datasets are pre-processed, the standard dataset, i.e., CICIDS2017, is grouped into two datasets, i.e., the training and testing dataset using *train_test_split*(). The training dataset is operated by different ensemble-based classifiers, i.e., decision tree, random forest, extra tree, XGBoost, KNN, and SVM, to correctly classify τ and τ' of AV's CAN traffic. Algorithm 1 shows the step-by-step approach of the proposed framework. The reason behind the inclusion of KNN and SVM is to analyze and compare the behavior of the proposed framework with the ensemble and conventional AI classifiers. Once the AI classifiers train on the CICIDS2017 training data, their learning helps the testing dataset, which is the generated dataset from the ICSim simulator. This benefits the proposed framework because the test datasets have new attacks and are not pre-trained. In addition, it validates the results of training datasets, i.e., if the result of the testing dataset is nearly equal to the result of the training dataset, then it shows that the classifiers appropriately do the classification [5]. Moreover, stacking is used to strengthen the accuracy of the classifiers. It is

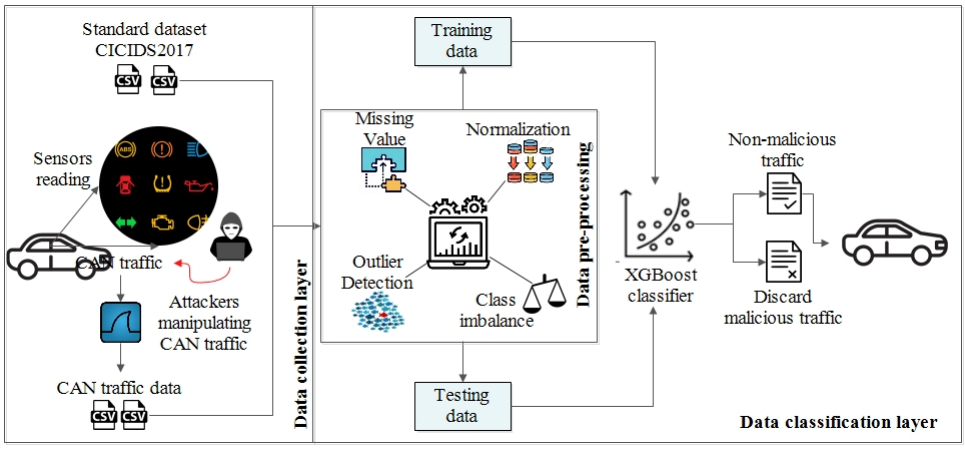

Fig. 1. Phases of machine learning model for AV

a typical ensemble approach that consists of two layers in which the output of the first layer, i.e., trained base predictors (input) is applied as a input to a meta-learner of the second layer to yield a powerful classifier [12].

First layer of the stacking ensemble mechanism consider four trained tree structure as a base model algorithm in which the algorithm with the highest accuracy can be adopted as meta-classifier in the second layer. To work on the certainty of the chosen prediction, a prediction determination feature selection procedure is used by calculating the normal element significance records produced by the four tree-based ML selected models. They are picked for prediction determination since tree-based calculations compute the significance of each element in light of every tree and afterward normalize the result of the trees to make the outcome more dependable. Moreover, unique conventional component determination strategies, for example, entropy, information gain, and Gini index are used in the framework by assigning various boundaries in tree-based techniques to create the persuading prediction significance. The amount of the complete component significance is 1.0. Finally, the proposed framework is evaluated with different performance metrics, such as accuracy, recall, precision, and f1-score. The result shows that the XGBoost shows significant improvement in terms of accuracy, i.e., 98.57%.

Algorithm 1 Working of the proposed framework

Input: $\{d\}$ dataset $\in D$
Output: Clasification of normal and malicious data

procedure EDGE_CLASSIFIER(D,RF_c)
  $dataset \leftarrow$ accumulates randomly sample instances from majority classes $df_s = dfsort_index()$
  if dataset has empty value Normalize using Min-Max normalization $x : (x - x.min())/(x.max() - x.min())$ **then**
    $dataset \leftarrow df = df.fillna(0)$
  else
    no change in dataset
  end if
  if $C_0 << C_1$ **then**
    processed_dataset $\leftarrow SMOTE(K,C)$
  else
    no change in _dataset

    Feature selection Average feature
  end if
  if $(fs = all\_tree\_features/4)$ **then**
    for $(Imp_{Feat} = 0; Imp_{Feat} <= 0.9; Imp_{Feat}++)$ **do**
      $\mathbb{C}$ = $xgb.XGBClassifier().fit(x_train, y_train)$
      $XGBoost_c(\delta_{tr1}', \delta_{tri}') \xrightarrow[\text{Precision}]{\text{Accuracy}} \mathbb{C}_o$
    end for
  end if
end procedure

IV. Result and Performance Evaluation

This section describes the result analysis of the proposed framework, which comprises the experimental setup, dataset description, and the proposed ensemble-based IDS performance analysis in terms of accuracy, precision, recall, and f1-score. A summarized description of each component is explained as follows.

A. Simulation System

In this experimental evaluation, we used the Anaconda distribution (version 2.1.1.), which is a prominent tool for formulating AI models. In that, we utilize Jupyter Notebook and other standard libraries, such as Pandas, Numpy, Matplotlib, and sklearn, that provides an environment to load the dataset, write the AI code and visualize the data for better understanding. Additionally, we employ the ICSim simulator to impersonate the CAN traffic of AV, and later the Wireshark (stable version 3.6.5) is used to accumulate the CAN traffic from the simulator. All the tools used in this work are open-source, which are downloaded from the Internet and installed

on a desktop having specifications, such as Intel core i3, 8GB RAM, and Intel iRIS graphic card that improves the performance of our experiment.

B. Dataset description

In this proposed IDS to evaluate this work, we used a dataset for inter and external communication in AV. Initially, we collected a CAN-intrusion dataset, i.e., CICIDS2017 [1] used for car hacking, and proposed new IDS development on the CAN bus. It has class labels, such as DoS, BENIGN, Port-Scan, Brute-Force, Web-Attack, Botnet, and Infiltration. In this dataset, there are 78 columns and 56661 rows that include information on forwarding Packet, total backward packet length and flow bytes, flow packets, and other attributes of the dataset that help identify some security attacks and generate alerts with IDS. Table I describes a class label of attacks that we can use in our experiments.

TABLE I
DESCRIPTION OF CLASS LABEL IN CICICDS2017 DATASET

Class label	Number of instances
DoS	19035
BENIGN	22731
Port-Scan	7946
Brute-Force	2767
Web-Attack	2180
Botnet	1966
Infiltration	36

C. IDS performance analysis

The result of the proposed IDS-based framework is evaluated in terms of accuracy, precision, recall, and f1-score. In addition, Table II shows the comparison of performance metrics of each classifier used in the proposed framework. The proposed IDS-based framework shows significant improvement in detecting the traditional security and modern-day security attacks from both datasets. Ensemble methods outperform other conventional AI classifiers, i.e., SVM and KNN. Additionally, it consumes less time and provides better results using tree-based learning. It can be seen from Table II that XGBoost received the highest accuracy for the detection of attacks in the proposed framework. This happens because XGBoost always gives more importance to functional space and reduces the cost of the model.

TABLE II
PERFORMANCE ANALYSIS RESULT OF IDS

Method Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-score
Decision Tree	98.25	98.19	98.25	0.9818
Random Forest	98.41	98.38	98.41	0.9837
Extra trees	98.09	98.04	98.09	0.9803
XGboost	98.57	98.52	98.57	0.9852
KNN [8]	97.40	96.26	96.30	0.9675
SVM [8]	96.50	95.30	95.35	0.9780
Stacking	98.25	98.19	98.25	0.9818

TABLE III
FEATURES AND PACKET WEIGHT OF ATTACK METHOD

Attack Method Name	Attack Features	Packet Weight
Port Scan	Length of forward packet	0.302
	Average packet size	0.1034
	Client server flag count	0.1019
Brute force	Receiver destination port	0.3725
	Min forward packet length	0.102
	Packet length variance	0.0859
Web attack	Init win bytes backward	0.2463
	Average packet size	0.165
	Destination port	0.061
Botnet	Source port	0.234
	Packet length mean	0.123
	Average segment size	0.114
Infiltration	Total length of forward pacekt	0.2295
	Subflow forward bytes	0.134
	Destination port	0.115
DoS	Packet length standard	0.175
	Average packet size	0.123
	Destination port	0.0783

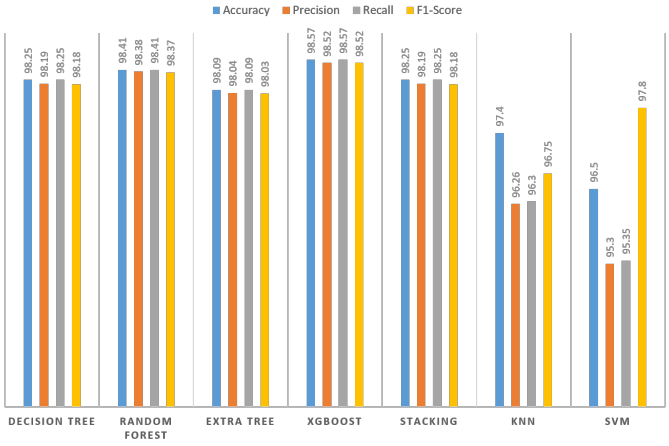

Fig. 2. Accuracy, precision, recall and f1-score of different classifier

Fig. 2 shows the different AI classifiers along with the parameters such as accuracy, precision, recall, and f1-score. From the graph, it is clear that XGBoost outperforms other AI-based classifiers in all performance metrics. This is because the XGBoost classifier supports regularization, thereby reducing its cost function and weight matrix. Moreover, the precision specifies the states that are correctly classified. Formally, it is defined as the ratio of the True Positive (TP) class to the total number of positive classes. It shows the correctness of the proposed framework. Further, it helps to identify and improve reliability of the ML model in classifying the positive classes. The XGBoost shows the precision value of 98.52% which is higher than other classifiers. Furthermore, the proposed framework also calculates the recall value, which is defined as the number of positive classes rightly predicted as positive from the total classes. The graph shows that the XGBoost again shows higher recall values, i.e., 98.57% compared to other classifiers. Lastly, the proposed

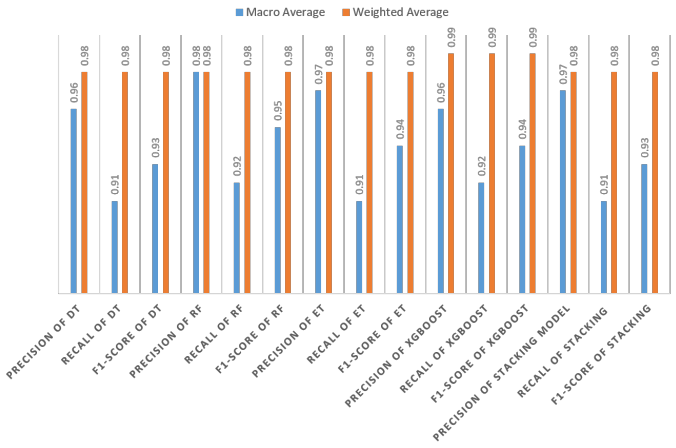

Fig. 3. Comparison of macro average and weighted average

framework is assessed using the f1-score, described as the harmonic mean of the parameters, i.e., precision and recall. It helps when the classifier has the same value of precision and recall. It depends on the precision and recall values, i.e., if both precision and recall values tends to be high, the model brings a high f1-score. Similarly, if both precision and recall values tends to be low and medium, the model brings a low and medium f1-score. Fig. 2 it is clear that once again, the XGBoost shows substantiate improvement in f1-score, i.e., 98.52% compared to other classifiers.

In addition, Fig. 3 shows the comparison of a macro and weighted average for precision, recall, and f1-score of all the ensemble-based classifiers. It helps when the dataset is imbalanced and the performance metrics are devalued. Macro average scores specify the mean value of precision, recall, and f1-score. Contrary, the weighted average scores are the sum of all classes multiplied by their respective class proportion. From the Fig. 3, it is clear that the XGBoost shows a better macro and weighted average compared to other ensemble and conventional AI classifiers. Further, Table III, describes the attack name and its features along with the packet length that is fetched from the Wireshark tool. It shows the change in the packet length once the attack happened on the CAN traffic of the AV.

V. Conclusion

In this paper, we proposed an IDS framework for intelligently classifying malicious and non-malicious CAN traffic of the AV. Toward this goal, we employ the indispensable benefits of ensemble-based AI models to correctly classify CAN traffic. First, a standard dataset, i.e., CICIDS2017, is utilized to train different ensemble-based AI classifiers, such as decision tree, extra tree, random forest, XGBoost, SVM, and KNN. Further, a new dataset is generated using the ICSim simulator, which mimics the CAN traffic to analyze new and unknown attacks of AV. Then, learning of the aforementioned pre-trained classifiers is utilized on the generated dataset to improve the performance of the proposed framework. Prior to training, the dataset is preprocessed with missing values, normalization, outlier detection, stacking, and balancing the class labels. Finally, the proposed framework is evaluated against various performance parameters, such as accuracy, precision, recall, and f1-score. The XGBoost classifier outperforms the other classifiers in terms of accuracy, i.e., 98.57%. In future work, we will improvise the security aspects of the proposed framework by analyzing modern-day attacks, such as malware attacks, replay, and Sybil attacks.

References

[1] MS Windows NT kernel description. https://www.unb.ca/cic/datasets/ids-2017.html. Accessed: 2010-09-30.

[2] Theyazn H. H. Aldhyani and Hasan Alkahtani. Attacks to automatous vehicles: A deep learning algorithm for cybersecurity. *Sensors*, 22(1), 2022.

[3] Jin Cui, Giedre Sabaliauskaite, Lin Shen Liew, Fengjun Zhou, and Biao Zhang. Collaborative analysis framework of safety and security for autonomous vehicles. *IEEE Access*, 7:148672–148683, 2019.

[4] Rajesh Gupta, Aparna Kumari, and Sudeep Tanwar. A taxonomy of blockchain envisioned edge-as-a-connected autonomous vehicles. *Transactions on Emerging Telecommunications Technologies*, 32(6):e4009, 2021.

[5] Mariam Ibrahim, Ahmad Alsheikh, Feras Awaysheh, and Mohammad Alshehri. Machine learning schemes for anomaly detection in solar power plants. *Energies*, 15:1082, 02 2022.

[6] Xiantao Jiang, F. Richard Yu, Tian Song, Zhaowei Ma, Yanxing Song, and Daqi Zhu. Blockchain-enabled cross-domain object detection for autonomous driving: A model sharing approach. *IEEE Internet of Things Journal*, 7(5):3681–3692, 2020.

[7] Aparna Kumari, Rajesh Gupta, Sudeep Tanwar, and Neeraj Kumar. A taxonomy of blockchain-enabled softwarization for secure uav network. *Computer Communications*, 161:304–323, 2020.

[8] Fang Li, Lifang Wang, and Yan Wu. Research on can network security aspects and intrusion detection design. Technical report, SAE Technical Paper, 2017.

[9] Harsh Mankodiya, Mohammad S. Obaidat, Rajesh Gupta, and Sudeep Tanwar. Xai-av: Explainable artificial intelligence for trust management in autonomous vehicles. In *2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*, pages 1–5, 2021.

[10] Michael Müter and Naim Asaj. Entropy-based anomaly detection for in-vehicle networks. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 1110–1115. IEEE, 2011.

[11] Dakshita Reebadiya, Tejal Rathod, Rajesh Gupta, Sudeep Tanwar, and Neeraj Kumar. Blockchain-based secure and intelligent sensing scheme for autonomous vehicles activity tracking beyond 5g networks. *Peer-to-Peer Networking and Applications*, 14(5):2757–2774, 2021.

[12] Eunbi Seo, Hyun Min Song, and Huy Kang Kim. Gids: Gan based intrusion detection system for in-vehicle network. In *2018 16th Annual Conference on Privacy, Security and Trust (PST)*, pages 1–6. IEEE, 2018.

[13] Dario Stabili, Mirco Marchetti, and Michele Colajanni. Detecting attacks to internal vehicle networks through hamming distance. In *2017 AEIT International Annual Conference*, pages 1–6. IEEE, 2017.

[14] Adrian Taylor, Nathalie Japkowicz, and Sylvain Leblanc. Frequency-based anomaly detection for the automotive can bus. In *2015 World Congress on Industrial Control Systems Security (WCICSS)*, pages 45–49. IEEE, 2015.

[15] Sheng-Li Wang, Sing-Yao Wu, Ching-Chu Lin, Srivalli Boddupalli, Po-Jui Chang, Chung-Wei Lin, Chi-Sheng Shih, and Sandip Ray. Deep-learning-based intrusion detection for autonomous vehicle-following systems. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 865–872, 2021.

[16] Zhang Yong, Zhang Xiaoming, and Mohammad Alshehri. A machine learning-enabled intelligent application for public health and safety. *Neural Computing and Applications*, 08 2021.