Multibranch Reconstruction Error (MbRE) Intrusion Detection Architecture for Intelligent Edge-Based Policing in Vehicular Ad-Hoc Networks

Amit Chougule, Varun Kohli, Vinay Chamola, *Senior Member, IEEE*, and Fei Richard Yu, *Fellow, IEEE*

***Abstract*— There has been a notable increase in the research and development of Vehicular Ad-hoc Networks (VANETs) to efficiently and safely manage large amounts of traffic. Such networks are, however, also prone to various cyber threats to data integrity, privacy, authentication, and network availability, and given the potential risk to life under the event of a malfunction and misinformation, it is important to provide security measures against such threats. This paper presents the Multi-branch Reconstruction Error (MbRE) Intrusion Detection System (IDS) for edge-based anomaly detection in VANETs for data integrity, network availability and user authentication-based misbehaviors without the need to train on them. Vehicular data is first sequenced and separated into three data branches - frequency (F) derived from the message timestamps, pseudo-identities (I), and the motion data (M) i.e. position and velocity. The proposed model comprises of three Convolutional Neural Networks (CNN)-based reconstruction models trained to reconstruct normal F-I-M vehicular behavior. The IDS classifies each branch of a sequence as 0/1 based on the reconstruction error threshold for the respective branch and, therefore, has the ability to detect 8 possible binary encoded behaviors for each sequence of vehicular data. These results are then used to find the overall behavior of each vehicle using carefully selected detection thresholds. MbRE is able to classify frequency, identity and motion-based behavior samples with an accuracy of 100%, 98.5-100%, and 95.4-100%, respectively, without the need to train on such behaviors. The study also emulates the IDS on Google Colaboratory and Jetson Nano to show its practicality in cloud and edge environments.**

***Index Terms*— Vehicular ad-hoc networks (VANETs), deep neural networks, intelligent transportation system (ITS), anomaly detection.**

I. Introduction

THE advancements in the Internet of Things (IoT), edge and cloud computing, and 5G technology have let to a surge in the development of Intelligent Transportation Systems (ITS) such as Vehicular Ad-hoc Networks (VANETs) [1], [2]. Vehicular nodes in a VANET exchange information via Vehicle to X (V2X) communication to make the network safer, greener, and more efficient [3], [4]. V2X communication may be Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I), Vehicle to Sensor (V2S), and Vehicle to Pedestrian (V2P), and the increase in V2X links is expected to increase the attack surface for cyber-threats which may include attacks on data integrity, verification, user privacy, and network availability to name a few [5], [6]. Studies have shown that nearly 1.3 million people lose their lives and 8 million are injured due to road accidents every year [3], [7], [8]. Therefore, any misinformation and network unavailability in VANETs [9] can prove to be fatal for users given the high risks, and it is imperative to propose robust solutions to identify, analyze, eradicate and prevent such cyber-threats.

Due to the large amount of data generated and shared in VANETs, effective management and analysis of big data is challenging. Deep learning has proven to be a promising solution to this problem based on the success of recent deep learning-based intrusion detection systems in detecting known vehicular misbehaviors [10], [11]. Furthermore, recent studies on efficient resource management solutions for edge servers [12] have made edge-deployed security solutions a lucrative choice, alleviating the limitations of high latency and cost associated with cloud-based solutions.

However, most intelligent security measures are trained on a limited number of known cyber-threats, and the introduction of new threats in the future would require an in-depth analysis, data collection, and a re-training of the Deep Learning-based solutions to provide updated intrusion detection capabilities. The upkeep of such solutions is, thus, very expensive and time-consuming in the long term, making it one of the biggest shortcomings of state-of-the-art solutions. So it is necessary to develop systems that can identify new misbehavior efficiently.

This study proposes an intelligent and statistical intrusion detection system (IDS) called the Multi-branch Reconstruction Error (MbRE) to address the discussed shortcoming of the state-of-the-art intelligent intrusion detection systems. The contributions of this study are summarised as follows:

i We categorize data integrity, network availability, and sender identity-based cyber threats in VANETs into eight generalized categories to create "themes" of vehicular behavior for simplified detection.

Manuscript received 23 February 2022; revised 9 June 2022 and 27 July 2022; accepted 13 August 2022. The work of Vinay Chamola and Fei Richard Yu was supported by the Shastri Indo-Canadian Institute (SICI) Shastri Institutional Collaborative Research Grant (SICRG) Grant. The Associate Editor for this article was H. Lv. *(Corresponding author: Vinay Chamola.)*

Amit Chougule and Vinay Chamola are with the APPCAIR, Department of Electrical and Electronics Engineering, BITS-Pilani, Pilani Campus, Pilani 333031, India (e-mail: amitchougule121@gmail.com; vinay.chamola@pilani.bits-pilani.ac.in).

Varun Kohli is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077 (e-mail: varun.kohli@u.nus.edu).

Fei Richard Yu is with the Department of Information Technology, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: richard.yu@carleton.ca).

Digital Object Identifier 10.1109/TITS.2022.3201548

ii We propose a novel, edge-deployed, deep learning and statistics-based IDS for the cost-effective and long-term security of VANETs against a wide range of cyber-threats without the need to train on them.
iii We demonstrate the detection capability of the IDS on nineteen data integrity, frequency and identity-based misbehavior, all of which are previously unknown to the IDS.
iv We also emulate the proposed IDS on Google Colaboratory (CPU) and Jetson Nano to demonstrate its practicality in cloud and edge environments respectively.

The remainder of the paper is structured as follows: Section II provides a survey of related works and current solutions. Section III then gives a taxonomy of vehicular misbehavior in the scope of this study. Section IV explains the underlying concepts of the MbRE IDS, followed by a discussion of the simulation environment in Section V. The results and analysis of this study are presented in Section VI, and the study is concluded in Section VII.

II. Literature Survey

Due to the large surface of vulnerabilities in VANETs [13], [14], [15], several cyber threats are likely to the security of such networks [7], [8]. Misbehaving vehicular nodes may share incorrect position or velocity data with other nodes in the network, or the network may be flooded with messages to deny legitimate communication. Intruding nodes may also collect data from other nodes and broadcast it as their own. Such intrusions in the network may prove to be fatal. Therefore, various studies have been conducted in the past decade for intrusion detection [16], with tool varying from statistics [17], blockchain [18], machine learning [19] and deep learning [11].

Authors of [20] proposed a Support Vector Machine (SVM)-based framework to evaluate the trustworthiness of nodes, wherein vehicles plan upcoming data hops based on the evaluated trustworthiness of the next node. The proposed scheme detected Blackhole and Jellyfish intrusions, and prevented data deletion and delay-based intrusions in the network. The authors of [17] proposed a statistical approach for rogue node detection. The proposed IDS was deployed on the vehicular OBUs and analysed statistical differences of received vehicular data from normal behavior. If identified, data from the intruding nodes was rejected and other vehicles in the network were notified of the same. Another study used an entropy-based approach to detect deviation from normal behavior [21]. Authors of [19] proposed an IDS based on random forest classifier to detect Distributed-Denial of Service (DDoS) attacks. An IDS using ensemble learning was proposed by the authors of [22] to identify fraud attacks. Another machine learning-based framework was proposed in [23] which used tree-based machine models such as decision tree, random forest, and extra tree for intrusion detection in Control Area Networks (CAN).

As discussed previously, given the time-bound nature of security requirements in vehicular networks, cloud platforms are not a viable solution due to their high latency. Further, classical machine learning is limited by data volume, and therefore, deep learning proves to be the better solution to handle the big data scenario in VANETs [24], [25]. Studies on optimizing resource management and allocation in edge and fog computing [26], [27] have encouraged the development of various edge and fog-deployed deep learning solutions. The authors of [10] propose a Convolutional Neural Network (CNN)-based IDS wherein vehicular data was sequenced, converted into images and analysed by the proposed CNN model. The proposed framework can detect normal, DoS, disruptive, sybil, traffic congestion and their combinations at high recall and precision. Another study [11] proposed a scheme to detect twenty different frequency, identity and data-based anomalies through binary or ternary coarse-grained classification. A blockchain and edge-based learning framework was proposed in [18], wherein the pre-trained IDS models are broadcasted by the RSUs for use by OBUs in the network through a federated blockchain. Authors of [28] proposed a multi-tiered hybrid IDS which could detect a wide array of attacks included in the CAN-intrusion dataset. Authors of [29] proposed an IDS based on Generative Adversarial Networks for fog environments. Although this scheme was made for Local Area Networks (LAN), it is a good example of long term security for networks by using normal behavior as a standard for evaluating other behavior. Along similar lines, another study [30] proposed a reconstruction and thresholding-based IDS using Long Short Term Memory (LSTM) to detect DoS, Fuzzy, RPM spoofing and Gear spoofing attacks.

While the security solutions discussed above perform well in their scope of cyber-threats, it is important to note that they cover limited number of malicious behaviors. Furthermore they are also limited by the evolution of vehicular misbehaviors. For example, learning models trained to detect ten behaviors will not work when introduced to the eleventh threat. It is costly and time-consuming to retrain such models to incorporate new attack variants.

III. Preliminary Background

Figure 1 shows the structure of a typical VANET. As seen in the figure, a VANET comprises of vehicles of different types (cars, trucks, buses, and so on) communicating with each other, and with the RSUs via wireless V2X communication following the 5.9GHz band of the IEEE 802.11p wireless communication standard for Intelligent Transportation Systems (ITS). To ensure safety of other vehicles, support good decision making and maintain smooth operation, each vehicle must broadcast correct information regarding their positions and velocities at acceptable transmission frequencies, and under authorised pseudo-identities. Each broadcast message contains the position and velocity (along with other relevant information depending on the context of the application) of the vehicle at that time. It can be expected that some vehicles may violate the rules in the network and based on the type of misbehavior exhibited by them, vehicles may be classified into one or more of the following basic behaviors.

1) **Normal:** Normal vehicles broadcast true position and velocity data at standard frequencies, and under authorized pseudo-identities.

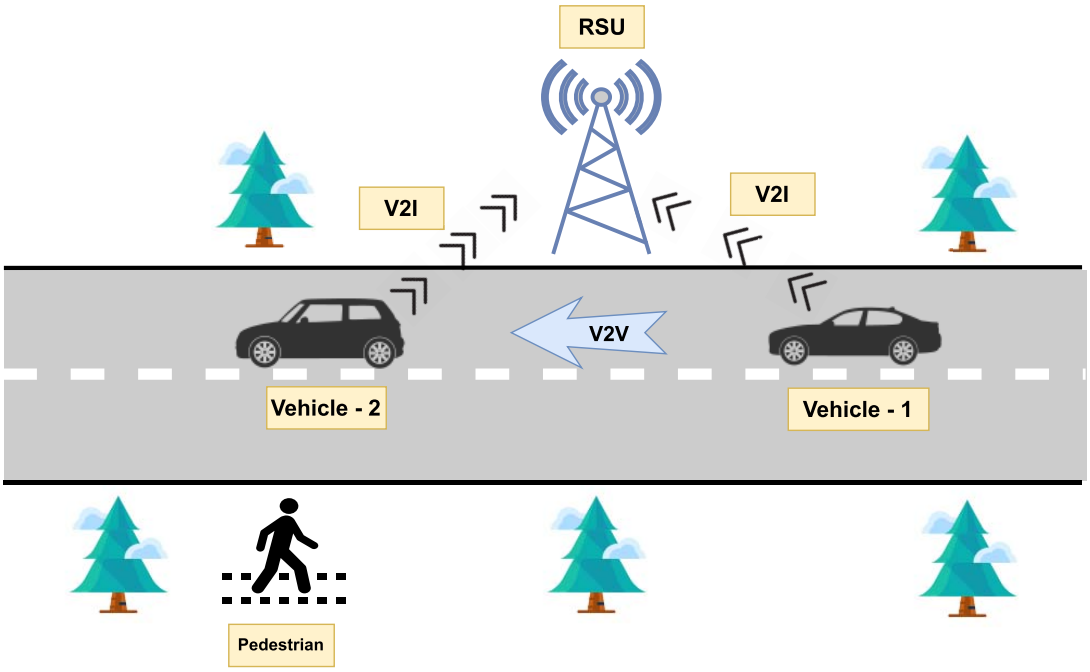


Fig. 1. A typical VANET.

2) **Position and Velocity Faults:** Vehicles broadcast incorrect position or velocity data. This may include constant, offset or random data.
3) **Eventual Stop:** This behavior is characterised by the broadcast of constant position data after a vehicle spends some time in the network.
4) **Delayed Messages:** Vehicles broadcast zero valued position and velocity data for some time, after which they broadcast their correct motion data.
5) **Disruptive:** Vehicles rebroadcast a few instances of positions and velocities received from random nearby vehicles as their own.
6) **Data replay:** Vehicles rebroadcast the real-data of a target vehicle in the network as their own.
7) **Denial of Service:** Vehicles flood the network with messages at high frequencies.
8) **Sybil:** Vehicles assume multiple valid pseudo-identities to broadcast their data, thereby gaining an unfair advantage over the network resources.
9) **Traffic Congestion:** Vehicles broadcast their information as well as falsified information under different pseudo-identities emulating vehicles in close proximity.

IV. Proposed Intrusion Detection Architecture

Figure 2 gives an overview of the proposed MbRE IDS. MbRE is a statistical and intelligent IDS developed to detect known and unknown vehicular misbehaviors based on message frequency, identity and data without the need to train on them.

A. Multi-Branch Sequences

The data broadcast from a vehicle comprises of three main pieces of information - first, the position and velocity data (M), the frequency of messages (F), and the pseudo-identities assumed for the messages (I). These three branches, F-I-M, form the core of the proposed IDS. The time-series information of every vehicle participating in the network is sequenced, and split into F-I-M branches for independent analysis.

B. F-I-M Encoding

Each of the behaviors discussed in Section III exhibit misbehavior in one or more of the F-I-M branches. For instance, DoS attacks consist of an F-fault, while Sybil and Random Position attacks consist of I and M-faults respectively. Similarly, based on the type of faults present in the broadcast, a vehicle's behavior can be encoded into 3-bit strings of F-I-M i.e. 8 unique F-I-M categories of vehicular behavior. This creates "themes" of vehicular behaviors. Table I shows the eight general F-I-M encodings and the corresponding twenty sample vehicular behaviors considered in this study. As seen in the table, DoS attack is encoded as 1-0-0, while a more complicated attack such as DoS Disruptive Sybil is encoded as 1-1-1 since it comprises of misbehavior in each of the three data branches. There were no behvaiors for 1-1-0 behavior available in the dataset (discussed in a later section). It is important to note that all behaviors shown in the table, with the exception of normal behavior, are new to the IDS.

C. Reconstruction Error

Reconstruction is a machine learning technique used to recreate the input given to a model. In this study, three Convolutional Neural Network (CNN) models were trained to reconstruct the individual F-I-M branches of normal vehicular behavior. The Reconstruction Error (RE) of each reconstructed branch can be evaluated as its Mean Square Error

TABLE I
SAMPLE VEHICULAR BEHAVIORS FOR THE EIGHT POSSIBLE F-I-M ENCODINGS

S.No.	F-I-M Encoding	Sample Behaviors
1	0-0-0	Normal
2	0-0-1	Constant/random/offset position and speed, eventual stop, disruptive, data replay, delayed messages
3	0-1-0	Sybil
4	0-1-1	Traffic congestion sybil
5	1-0-0	DoS
6	1-0-1	DoS random, DoS disruptive
7	1-1-0	Not available
8	1-1-1	Data replay, DoS random sybil, DoS disruptive sybil

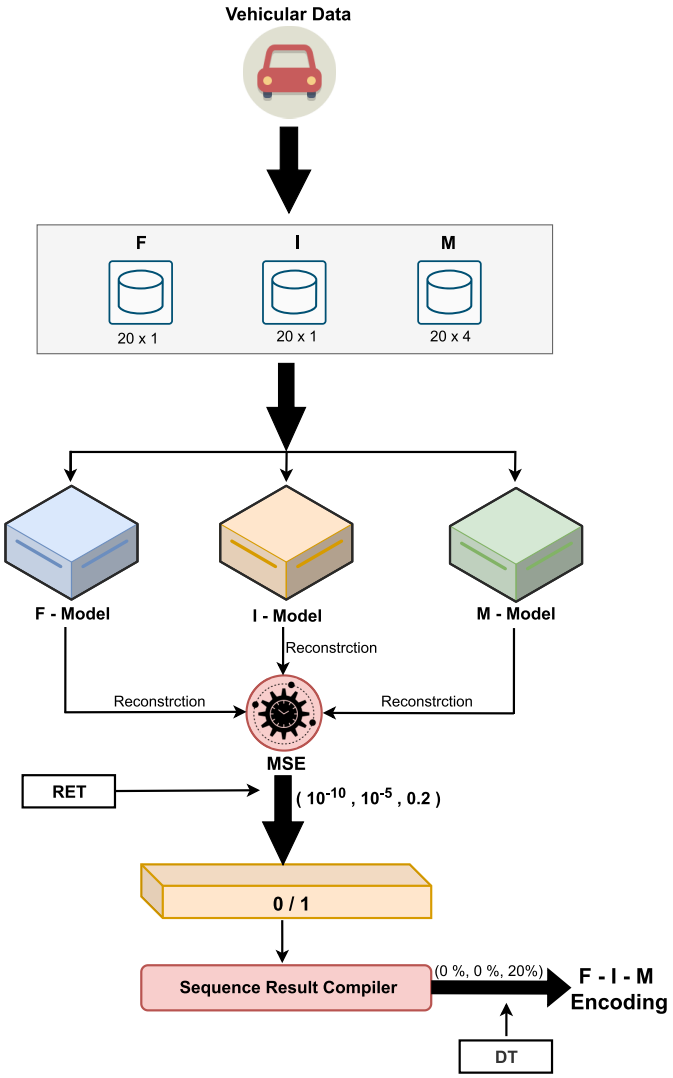


Fig. 2. Structure of the MbRE IDS.

relative to the original input sequence. Therefore, for a given i^{th} input sequence S_i and the reconstructed output sequence S'_i, the RE_i is calculated as,

$$RE_i = 1/m \sum_{j=1}^{m} (S_{ij} - S'_{ij})^2 \quad (1)$$

where m is the dimension of the input sequence vector, S_{ij} is the j^{th} data point in sequence S_i. RE, therefore, provides a measure of the closeness of a reconstructed input sequence to the original one. It should be noted that, the three CNN models reconstruct the F-I-M data branches separately, and therefore a single complete reconstructed sequence of vehicular data comprises of 3 separate RE values, one each for F,I and M.

D. Sequence-Level Encoding and Reconstruction Error Threshold

Sequence-level encoding refers to the predicted F-I-M encoding of an input sequence. This is achieved with the help of a Reconstruction Error Threshold (RET). RET is defined as the minimum error above which the behavior of a vehicle in a data branch is considered to be malicious. The RET of each branch is selected slightly larger than the maximum RE of normal behavior in that branch after filtering outliers. The i^{th} reconstructed sequence branch b (b is one of F, I or M) is encoded as 0 or 1 based on its value relative to the respective RET as per the following equation,

$$SE_{bi} = \begin{cases} 0, & \text{for } RE_{bi} \leq RET_{bi} \\ 1, & \text{for } RE_{bi} > RET_{bi} \end{cases} \quad (2)$$

where SE_{bi} is the sequence level encoding of branch b of the ith sequence. On compiling the three branches together the overall vector SE_i can be written as,

$$SE_i = [SE_{Fi}, SE_{Ii}, SE_{Mi}] \quad (3)$$

The selected RET will be discussed in Section V.

E. Sequence Result Compiler

The sequence result compiler stores the historical encoding of vehicular behavior to evaluate its overall behavior type in the network using the detection threshold discussed in the next subsection.

F. Vehicular Encoding and Detection Threshold

While the RET performs sequence level encoding of vehicular behavior, a Detection Threshold (DT) is used to assess the overall behavior of a vehicle in the network. DT is defined as the minimum proportion (%) of sequences identified to

have errors (using RET) above which the overall behavior of a vehicle is labeled malicious. More information about the selected DT can be found in section V. For a vehicle with n sequences, the Branch Ratio (BR), i.e. the ratio of sequence level misbehavior predictions relative to the total number of sequences, is calculated as,

$$BR_b = 1/n \sum_{i=1}^{n} (SE_{bi}) \tag{4}$$

Accordingly, the Vehicular Encoding (VE) for data branch b can be calculated as,

$$VE_b = \begin{cases} 0, & \text{for } BR_b \leq DT_b \\ 1, & \text{for } BR_b > DT_b \end{cases} \tag{5}$$

where DT_b is the DT of branch b. Similar to the compiled RET, the overall VE can be written in a concise form as,

$$VE = [VE_{Fi}, VE_{Ii}, VE_{Mi}] \tag{6}$$

V. Simulation Environment

This section provides details of the dataset, data preprocessing, model hyperparameters, and threshold filters used in this study.

A. Dataset and Pre-Processing

This study uses the VeReMi Extension dataset for intrusion detection [31], due to its robustness and notable variety of vehicular behaviors. The dataset comprises of twenty-four V2V and V2I communication folders, one each for 24 hours of simulation. Each simulation folder contains one traceGroundTruth.json file of V2I communication data and multiple traceJSON.json files of V2V data for each vehicle present in the network in the simulation hour. Since the current study focuses on edge-deployed intrusion detection, only the V2I communication data is used for training and testing the proposed IDS. The traceGroundTruth.json files contain multiple data points of V2I communication, which include the vehicles' (X, Y) coordinates of position, velocity, acceleration and heading, the pseudo-identity used by the vehicle to broadcast the message, as well as the timestamp at which the message is received by the infrastructure. Given the lack of acceleration and heading-based misbehaviors, only six data fields were used for the purpose of intrusion detection - time stamps for the F-branch, transformed pseudo-identities for the I-branch, and (X,Y) position and speed coordinates for the M-branch. The data was compiled into a dictionary of vehicles labeled with their respective behaviors (obtained from the traceJson.json file names), and converted into (20,6) dimension sequences, of which the F, I and M branches had dimensions (20,1), (20,1) and (20,4) respectively. The sequences were generated at a window length of 5, i.e. a vehicle with 50 data points would have 6 total sequences of data: first sequence as data point 1-20, second as data point 5-25, and so on. The goal of windowed sequence generation was to ensure completeness of information within a sequence while minimizing redundancy in data.

TABLE II

F-I-M Encoding and Branch Recall Results of MbRE for the 20 Vehicular Behaviors Included in the VeReMi Extension Dataset

S.No.	Behavior	Encoding			Recall		
		F	I	M	F	I	M
0	Normal	0	0	0	100	100	98
1	Constant position	0	0	1	100	100	42.5
2	Constant position offset	0	0	1	100	100	5.5
3	Random position	0	0	1	100	100	100
4	Random position offset	0	0	1	100	100	8
5	Constant speed	0	0	1	100	100	94.5
6	Constant speed offset	0	0	1	100	100	42
7	Random speed	0	0	1	100	100	100
8	Random speed offset	0	0	1	100	100	100
9	Eventual stop	0	0	1	100	100	9
10	Disruptive	0	0	1	100	100	98.5
11	Data replay	0	0	1	100	100	98.5
12	Delayed messages	0	0	1	100	100	8.5
13	DoS	1	0	0	100	100	98
14	DoS random	1	0	1	100	100	100
15	DoS disrptive	1	0	1	100	100	99.5
16	Data replay sybil	1	1	1	100	99.5	98.5
17	Traffic congestion sybil	0	1	1	100	98.5	97.5
18	DoS random sybil	1	1	1	100	100	100
19	DoS disruptive sybil	1	1	1	100	99	99

The data was then scaled. The first timestamp of each F-branch sequence was subtracted from all timestamps in that sequence to remove the possibility of bias to exact timestamp values. Similarly, the pseudo-identities were also transformed as a function of the vehicle's actual identity. The positions and velocities in the M-branch were scaled down by a factor of 100 and 10 respectively.

B. Train and Test Data

10,000 normal behavior sequences were used to train the proposed reconstruction models. Once trained, sequences from 300 vehicles from each of the 20 behavior types listed in Table II were reconstructed.

C. Model Hyperparameters

Three CNN reconstruction models were used for the F-I-M data branches, all of which were trained to reconstruct normal data. The F and I models comprised of one Conv1D layer with 32 filters and a kernel size of 2, and a MaxPool1D layer with a pooling size of 2. This was followed by a Flatten layer, and two Dense layers of 50 neurons and 20 output neurons. The M-model comprised of two Conv2D layers of 64 (2×1) kernels, and 32 (2×1) kernels, followed by a MaxPool2D layer of size (2×1), a Flatten layer, and two Dense layers of sizes 50 neurons and 80 output neurons. The F and I-models had 20 output neurons to reconstruct the (20,1) dimension F and I-branches, while the M-model had 80 output neurons to reconstruct the (20,4) M-branch sequences in (80,1) format. All models use the ReLU activation function and were optimized with Adam at a learning rate of 0.001. Alternatively, experiments using Stochastic Gradient Descent (SGD), which

is another popular optimizer choice, showed worse results. The F and I-models were trained for 10 epochs, while the M-model was trained for 200 epochs due to the inherent complexity of the M-branch data.

D. Reconstruction Error and Detection Thresholds

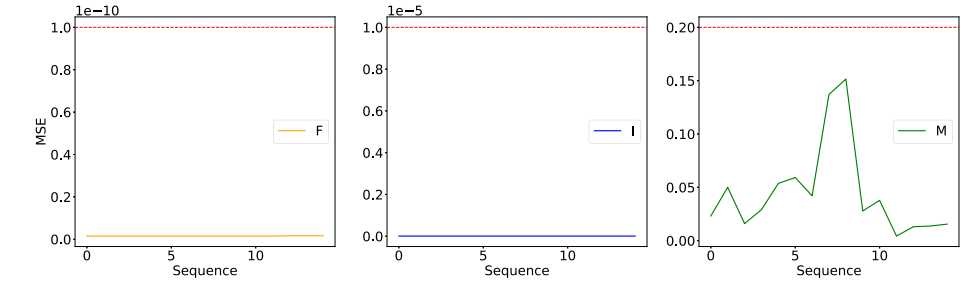
An RET of $[10^{-10}, 10^{-5}, 0.2]$ and DT of $[0\%, 0\%, 0.2\%]$ was selected after various experiments to obtain the highest possible final recall scores. F and I reconstruction showed no outliers for normal behavior, therefore thresholds slightly higher than the maximum reconstruction errors were selected for these data branches. In contrast, it was observed that the M-model reconstruction had 5% data points as outliers, and an error of 0.2 was selected for the M-branch. Based on the severity of misbehavior of the data branches, the DT for F and I-branches were fixed at 0% so as to classify vehicles showing any misbehavior in these categories as malicious, while the M-branch DT was set to 20% to accomodate the expected high variance and possible values of position and velocity.

VI. Results and Analysis

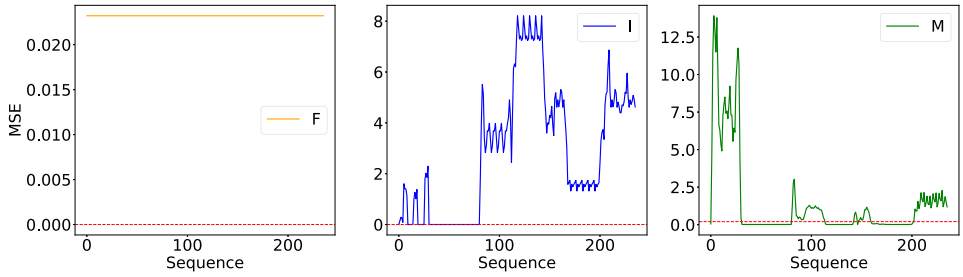
This section presents the numeric and graphical results of the proposed MbRE IDS for a wide range of vehicular misbehaviors. Since the IDS classifies each branch into two classes (0 or 1), the precision, recall and accuracy are interchangeable and this study uses recall for the same. Table II shows the recall scores for test samples of 300 vehicles from each of the twenty vehicular behaviors included in the VeReMi Extension dataset. It can be seen that the F and I models perform exceptionally well in identifying all types of vehicular behavior with a 100% recall for the F-model and 98.5-100% recall for the I-model. Further, the M-model had recall scores in the range 5.5%-100% wherein a majority of the behaviors were classified at recall scores higher than 94.5%. Figure 3 shows the visual representation of the IDS' error generation and RET (dashed red line) for all sequences of one sample vehicle each of normal (type 0) and complex behavior types - 16 (data replay sybil), 17 (traffic congestion sybil), 18 (DoS random sybil) and 19 (DoS disruptive sybil). As the figure shows, the RE of the F branch stays nearly the same across all sequences for a given vehicles while the RE for I and M is variable. This is due to the simplicity of F-based misbehavior. However, if there were varying frequencies of messages for a single vehicle across different sequences, the relative change in the RE would be observable. Examples of this can be seen in the variance in RE results for different I and M branch sequences of the same vehicle which is due to the high variance in I and M data present in the dataset. These results show the ability of the IDS to detect different variations in data for a specific behavior type. Figure 4 shows the corresponding classification of the vehicular sequence reconstruction errors seen in the previous figure into the respective sequence-wise encoding using the RET. All sequence errors above the RET are classified as 1 (i.e. misbehavior) and otherwise 0 (normal). An effort was made to select a vehicle for which the IDS wrongly classified some sequences, but ultimately correctly identified the vehicular behavior type using the DT. For instance, as the figure shows, the M branch of all sample vehicles except data replay sybil were predicted reliably, however, data replay sybil has spikes in 0 and 1 prediction. This is due to the high variance in motion data which is observable even in some samples of normal behavior. On carefully examining the sequences predicted as 0, it was seen that the M behavior seemed nearly normal and therefore the model was unable to predict those specific sequences as 1. However, due to the history sensitive nature of the IDS to consider all available sequences of a vehicle to predict its overall behavior, the vehicle was still labeled to have an an M-fault, therefore making it reliable in detecting vehicles that may be intermittently exhibiting misbehavior. Furthermore, the results show that the IDS is able to detect combinations of misbehaviors without the need to train on them, which gives insight into its ability to detect unknown behaviors that deviate from the norm. Additionally, Figure 5 shows the error and prediction performance of MbRE for sample vehicles from behavior types 9 (eventual stop) and 12 (delayed messages). These behaviors are identified as special cases in the context of this study despite the poor identification performance after the DT as seen from the table, and will be discussed in more detail shortly.

Various inferences can be made regarding the performance of MbRE. First, the IDS is able to detect any type of frequency-based fault that may occur in a real-world VANET. Second, among the vehicular behaviors involving Sybil attacks (types 16, 18 and 19) it was observed that the incorrectly labeled sample vehicles had only one sequence of data within which the identity misbehavior was not as dominant as seen in vehicles that spent more time in the simulation network. Therefore, the IDS is adept in identifying unnatural identity behavior as well. However, the low scores observed for behavior types 1 (constant position, 42.5%), 2 (constant position offset, 5.5%), 4 (random position offset, 8%), 6 (constant speed offset, 42%), 9 (eventual stop, 9%) and 12 (delayed messages, 8.5%) show some limitations of MbRE in identifying such misbehaviors. Types 1, 2, 4 and 6 do not generate high enough reconstruction errors to be identified as misbehavior by the RET due to their high similarity with normal vehicular behavior. Types 9 and 12 however, are special cases due to the nature of these misbehaviors. Eventual stop exhibits the behavior of a vehicle that reaches an abrupt halt after initially broadcasting normal vehicular data. Similarly delayed messages comprise of 0 motion values until the real data broadcast begins. As shown in Figure 5, these behaviors can be observed from spikes in the M-error. Since this at a small number of points, such vehicles will go undetected through the DT. However, they may spotted through manual examination of vehicular behavior plots after deeper study of exact error thresholds for each behavior.

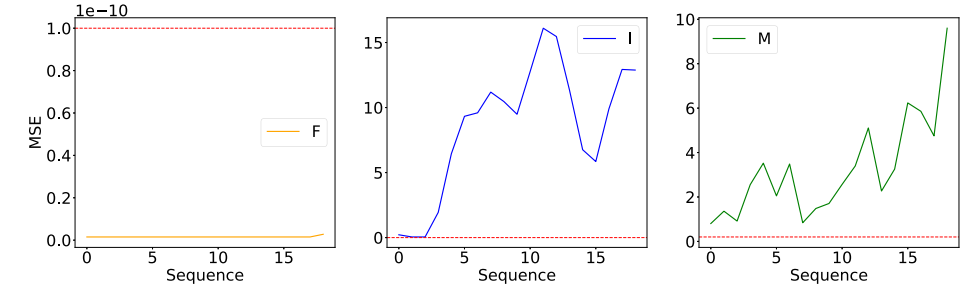
Table-III shows the run-time performance (in milliseconds) averaged over 1000 runs of different parts of the proposed IDS in two different environment - the cloud-based Google Colaboratory (CPU), and the Jetson Nano (for a more real-world example). As shown, the IDS performs the sequence generation, prediction and encoding of each vehicle within 511.89 milliseconds on Jetson Nano with series prediction. The time can be reduced to nearly 180 milliseconds per



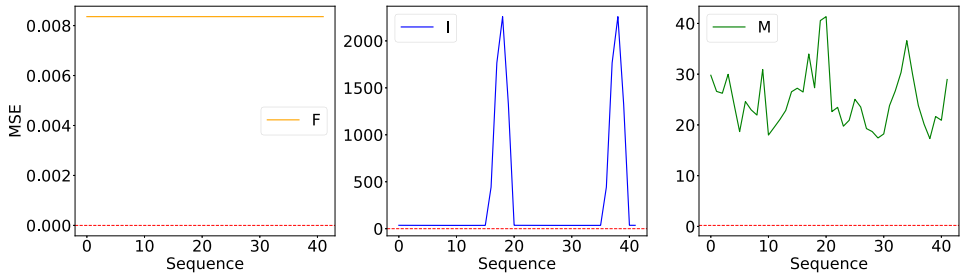
(a) Normal



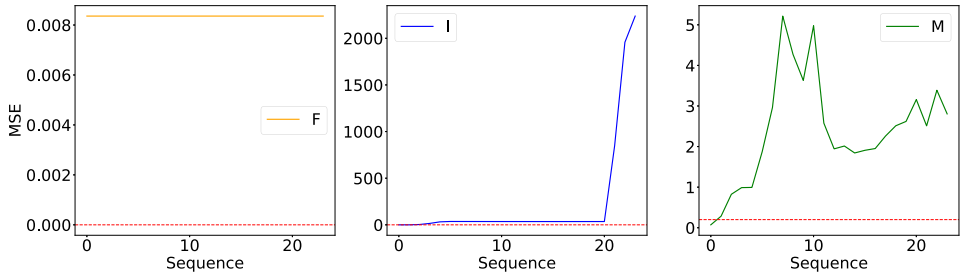
(b) Data replay sybil



(c) Traffic congestion sybil

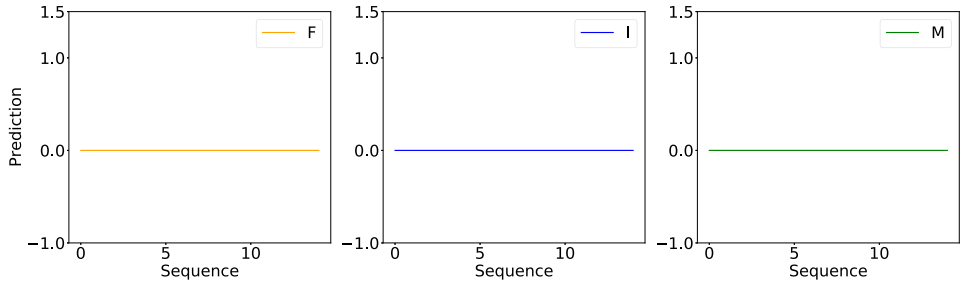


(d) DoS random sybil

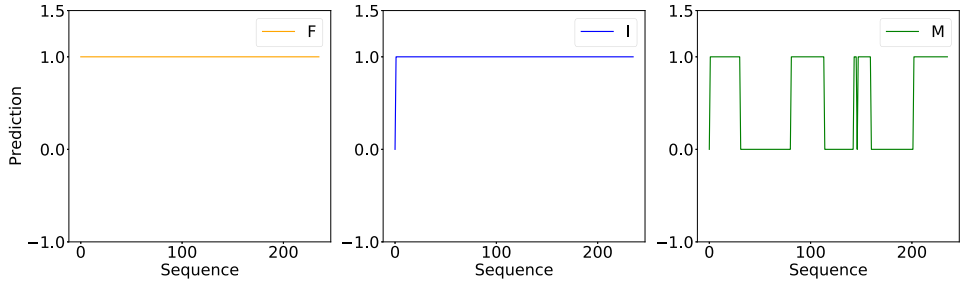


(e) DoS disruptive sybil

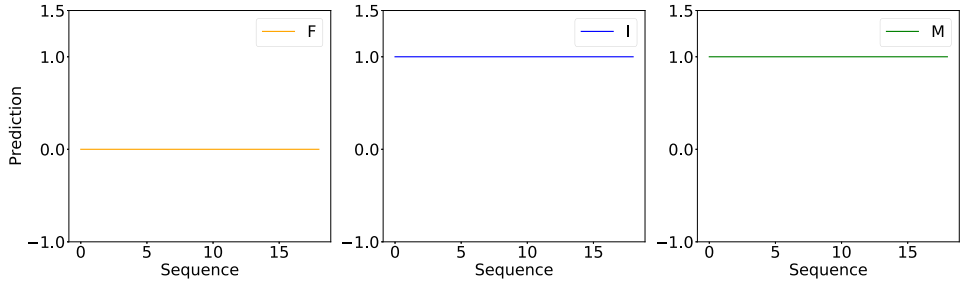
Fig. 3. F-I-M reconstruction error plots for normal behavior and four complex misbehaviors for sample vehicles.



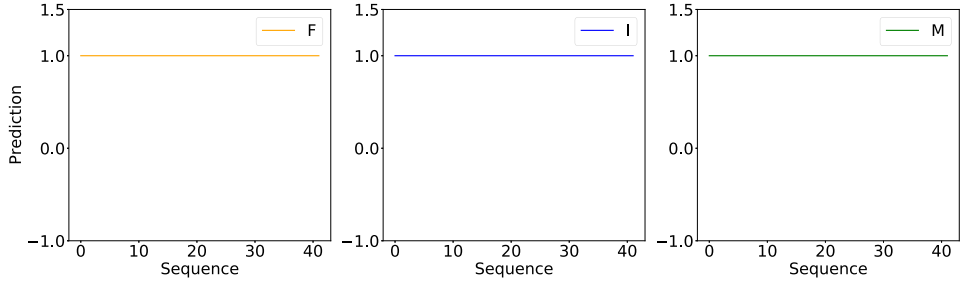
(a) Normal



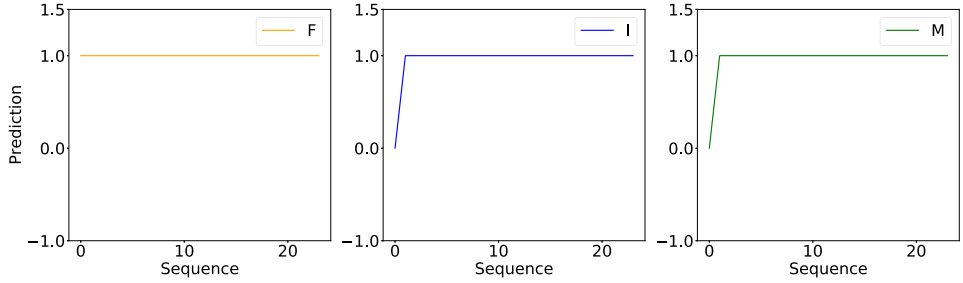
(b) Data replay sybil



(c) Traffic congestion sybil

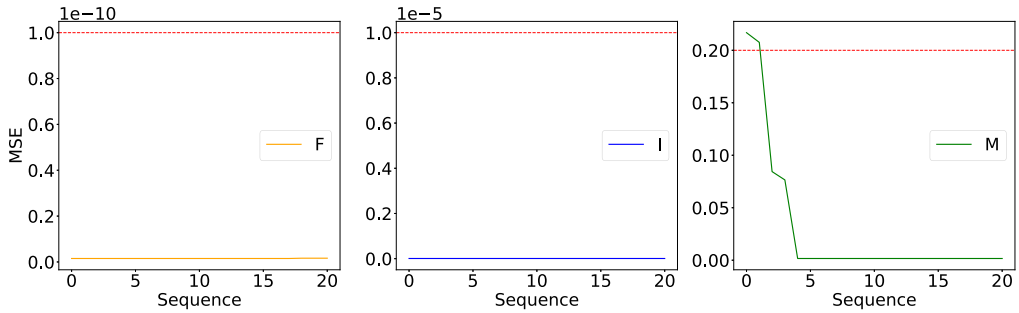


(d) DoS random sybil

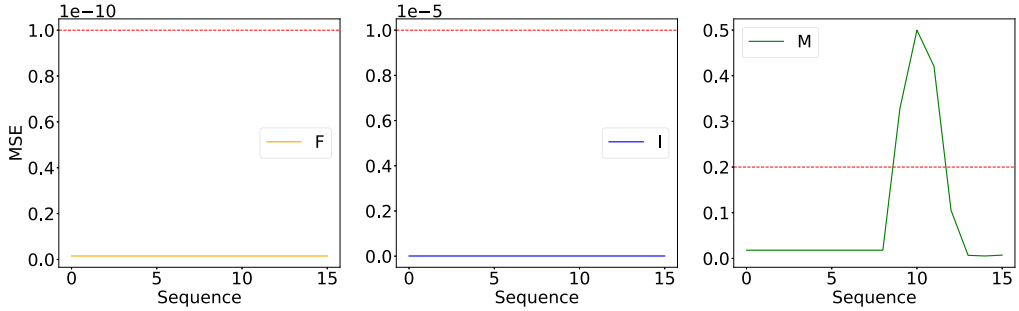


(e) DoS disruptive sybil

Fig. 4. F-I-M encoding plots for the previously shown reconstruction errors.



(a) Eventual stop



(b) Delayed messages

Fig. 5. F-I-M reconstruction error plots for special case behaviors.

TABLE III
RUN-TIME RESULTS OF MBRE ON GOOGLE COLAB (CPU) AND JETSON NANO

Device	Setup	Prediction			Encoding	Total
		F	I	M		
Google Colab	0.15	56.55	50.91	50.77	0.21	158.59
Jetson Nano	0.07	171.91	171.68	167.68	0.55	511.89

vehicle if the predictions are made in parallel. Further, the IDS performs faster on Google Colaboratory as expected with a total series average run-time of 158.59 milliseconds (which can also be reduced to 60 milliseconds with parallel processing). These results shows the ability of the MbRE IDS to perform its operation within reasonable time spans, making it suitable for real-world VANET edge deployment.

VII. CONCLUSION

This study proposed a lightweight and intelligent deep learning-based intrusion detection architecture called the MbRE IDS. MbRE was trained only on normal vehicular behavior. The IDS reconstructed the F-I-M vehicular data sequences and classified each branch as 0 or 1 by comparing it to the carefully defined RET. Finally a DT was used to classify every vehicle in the test-sample based on its overall performance in the network. The results showed MbRE's ability to detect 19 different vehicular behaviors included in the VeReMi Extension dataset, which it was not previously exposed to. Numerical results included a 100% recall for the F-branch, 98.5-100% recall for the I-branch and a 94.5-100% recall for the M-branch with a few limitations. The run-time results show the practicality of MbRE for real-world edge deployment in VANETs. The proposed work can be taken further by a deeper analysis of reconstruction errors to create more exhaustive RETs which will help perform fine-grained intrusion detection without the need to train on misbehavior.

REFERENCES

[1] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, "Sensor technologies for intelligent transportation systems," *Sensors*, vol. 18, no. 4, p. 1212, 2018.

[2] Y. Lin, P. Wang, and M. Ma, "Intelligent transportation system(ITS): Concept, challenge and opportunity," in *Proc. IEEE IEEE 3rd Int. Conf. Big Data Secur. Cloud (BigDataSecurity) Int. Conf. High Perform. Smart Comput., (HPSC) IEEE Int. Conf. Intell. Data Secur. (IDS)*, May 2017, pp. 167–172.

[3] J. Contreras-Castillo, S. Zeadally, and J. A. Guerrero-Ibañez, "Internet of vehicles: Architecture, protocols, and security," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3701–3709, Oct. 2018.

[4] B. Ji *et al.*, "Survey on the internet of vehicles: Network architectures and applications," *IEEE Commun. Standards Mag.*, vol. 4, no. 1, pp. 34–41, Mar. 2020.

[5] Z. Lu, G. Qu, and Z. Liu, "A survey on recent advances in vehicular network security, trust, and privacy," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 760–776, Feb. 2018.

[6] S. Sharma and B. Kaushik, "A survey on internet of vehicles: Applications, security issues & solutions," *Veh. Commun.*, vol. 20, Dec. 2019, Art. no. 100182.

[7] S. G. Philipsen, B. Andersen, and B. Singh, "Threats and attacks to modern vehicles," in *Proc. IEEE Int. Conf. Internet Things Intell. Syst. (IoTaIS)*, Nov. 2021, pp. 22–27.

[8] S. K. Khan, N. Shiwakoti, P. Stasinopoulos, and Y. Chen, "Cyber-attacks in the next-generation cars, mitigation techniques, anticipated readiness and future directions," *Accident Anal. Prevention*, vol. 148, Dec. 2020, Art. no. 105837.

[9] D. B. Rawat, M. Garuba, L. Chen, and Q. Yang, "On the security of information dissemination in the internet-of-vehicles," *Tsinghua Sci. Technol.*, vol. 22, no. 4, pp. 437–445, 2017.

[10] T. Alladi, V. Kohli, V. Chamola, F. R. Yu, and M. Guizani, "Artificial intelligence (AI)-empowered intrusion detection architecture for the internet of vehicles," *IEEE Wireless Commun.*, vol. 28, no. 3, pp. 144–149, Jun. 2021.

[11] T. Alladi, V. Kohli, V. Chamola, and F. R. Yu, "Securing the internet of vehicles: A deep learning-based classification framework," *IEEE Netw. Lett.*, vol. 3, no. 2, pp. 94–97, Jun. 2021.

[12] X. Xu, H. Li, W. Xu, Z. Liu, L. Yao, and F. Dai, "Artificial intelligence for edge service optimization in internet of vehicles: A survey," *Tsinghua Sci. Technol.*, vol. 27, no. 2, pp. 270–287, Apr. 2022.

[13] N. Sharma, N. Chauhan, and N. Chand, "Security challenges in internet of vehicles (IoV) environment," in *Proc. 1st Int. Conf. Secure Cyber Comput. Commun. (ICSCCC)*, Dec. 2018, pp. 203–207.

[14] A. Samad, S. Alam, S. Mohammed, and M. Bhukhari, "Internet of vehicles (IoV) requirements, attacks and countermeasures," in *Proc. 12th INDIACom; INDIACom 5th Int. Conf. Comput. Sustain. Global Develop. IEEE Conf.* New Delhi, India, 2018, pp. 1–4.

[15] Y. Sun *et al.*, "Security and privacy in the internet of vehicles," in *Proc. Int. Conf. Identificat., Inf., Knowl. Internet Things (IIKI)*, Oct. 2015, pp. 116–121.

[16] O. Y. Al-Jarrah, C. Maple, M. Dianati, D. Oxtoby, and A. Mouzakitis, "Intrusion detection systems for intra-vehicle networks: A review," *IEEE Access*, vol. 7, pp. 21266–21289, 2019.

[17] K. Zaidi, M. B. Milojevic, V. Rakocevic, A. Nallanathan, and M. Rajarajan, "Host-based intrusion detection for VANETs: A statistical approach to rogue node detection," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6703–6714, Aug. 2016.

[18] H. Liu *et al.*, "Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 6073–6084, Jun. 2021.

[19] Y. Gao, H. Wu, B. Song, Y. Jin, X. Luo, and X. Zeng, "A distributed network intrusion detection system for distributed denial of service attacks in vehicular ad hoc network," *IEEE Access*, vol. 7, pp. 154560–154571, 2019.

[20] E. A. Shams, A. Rizaner, and A. H. Ulusoy, "Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks," *Comput. Secur.*, vol. 78, pp. 245–254, Jul. 2018.

[21] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J. P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1557–1568, Oct. 2007.

[22] P. Sharma and H. Liu, "A machine-learning-based data-centric misbehavior detection model for internet of vehicles," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4991–4999, Mar. 2021.

[23] L. Yang, A. Moubayed, I. Hamieh, and A. Shami, "Tree-based intelligent intrusion detection system in internet of vehicles," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[24] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Inf. Fusion*, vol. 42, pp. 146–157, Jul. 2018.

[25] W. Xu *et al.*, "Internet of vehicles in big data era," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[26] Y. Dai, D. Xu, S. Maharjan, G. Qiao, and Y. Zhang, "Artificial intelligence empowered edge computing and caching for internet of vehicles," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 12–18, Jun. 2019.

[27] Z. Ning *et al.*, "Deep reinforcement learning for intelligent internet of vehicles: An energy-efficient computational offloading scheme," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 4, pp. 1060–1072, Dec. 2019.

[28] L. Yang, A. Moubayed, and A. Shami, "MTH-IDS: A multitiered hybrid intrusion detection system for internet of vehicles," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 616–632, Jan. 2022.

[29] P. F. de Araujo-Filho, G. Kaddoum, D. R. Campelo, A. G. Santos, D. Macedo, and C. Zanchettin, "Intrusion detection for cyber–physical systems using generative adversarial networks in fog environment," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6247–6256, Apr. 2021.

[30] K. Agrawal, T. Alladi, A. Agrawal, V. Chamola, and A. Benslimane, "NovelADS: A novel anomaly detection system for intra-vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 7, 2022, doi: 10.1109/TITS.2022.3146024.

[31] J. Kamel, M. Wolf, R. W. van der Hei, A. Kaiser, P. Urien, and F. Kargl, "VeReMi extension: A dataset for comparable evaluation of misbehavior detection in VANETs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

Amit Chougule received the M.Tech. degree from PES University, Bengaluru, India, in 2020. He is currently pursuing the Ph.D. degree with the Department of Electrical and Electronics Engineering, BITS-Pilani, Pilani Campus, India. He is also working as a Research Scholar with the Department of Electrical and Electronics Engineering, BITS-Pilani, Pilani Campus. He also has industrial experience working on artificial intelligence for healthcare in MNCs, such as Philips Healthcare and AIvolved Technologies. His research interests include developing artificial intelligence-based solutions for the autonomous driving as well as for healthcare using computer vision and deep learning technologies.

Varun Kohli received the Bachelor of Engineering (B.E.) degree in electrical and electronics engineering from the Birla Institute of Technology and Science (BITS), Pilani, in July 2021. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, National University of Singapore (NUS), and specializes in communications and networks. Prior to this, he worked at Pricewaterhouse Coopers India as a Technology Consultant from July 2021 to December 2021. His research interests include the Internet of Things, security, artificial intelligence, brain–computer interfaces, sustainability, and blockchain.

Vinay Chamola (Senior Member, IEEE) received the B.E. degree in electrical and electronics engineering and the master's degree in communication engineering from the Birla Institute of Technology and Science, Pilani, India, in 2010 and 2013, respectively, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2016. In 2015, he was a Visiting Researcher with the Autonomous Networks Research Group (ANRG), University of Southern California, Los Angeles, CA, USA. He is currently with the Department of Electrical and Electronics Engineering, BITS-Pilani, where he heads the Internet of Things Research Group/Laboratory. His research interests include the IoT security, blockchain, UAVs, VANETs, 5G, and healthcare. He is a fellow of the IET. He serves as an Area Editor for the *Ad Hoc Networks* journal (Elsevier) and the *IEEE Internet of Things Magazine*. He also serves as an Associate Editor for the *IEEE Consumer Electronics Magazine*, IEEE NETWORKING LETTERS, *IET Quantum Communications*, *IET Networks*, and several other journals.

Fei Richard Yu (Fellow, IEEE) received the Ph.D. degree in electrical engineering from The University of British Columbia (UBC) in 2003. He is currently a Professor at Carleton University, Canada. His research interests include blockchain, security, and green ICT. He is a fellow of the IET. He has served as a technical program committee (TPC) co-chair of numerous conferences. He serves on the editorial boards of several journals, and is the Co-Editor-in-Chief of *Ad Hoc & Sensor Wireless Networks*, the Editor-in-Chief for IEEE VTS Mobile World Newsletter, and is a Lead Series Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and IEEE COMMUNICATIONS SURVEYS AND TUTORIALS.