Artificial Intelligence (AI)-Empowered Intrusion Detection Architecture for the Internet of Vehicles

Tejasvi Alladi, Varun Kohli, Vinay Chamola, F. Richard Yu, and Mohsen Guizani

Abstract

Recent advances in the Internet of Things (IoT) and the adoption of IoT in vehicular networks have led to a new and promising paradigm called the Internet of Vehicles (IoV). However, the mode of communication in IoV being wireless in nature poses serious cybersecurity challenges. With many vehicles being connected in the IoV network, the vehicular data is set to explode. Traditional intrusion detection techniques may not be suitable in these scenarios with an extremely large amount of vehicular data being generated at an unprecedented rate and with various types of cybersecurity attacks being launched. Thus, there is a need for the development of advanced intrusion detection techniques capable of handling possible cyberattacks in these networks. Toward this end, we present an artificial intelligence (AI)-based intrusion detection architecture comprising Deep Learning Engines (DLEs) for identification and classification of the vehicular traffic in the IoV networks into potential cyberattack types. Also, taking into consideration the mobility of the vehicles and the real-time requirements of the IoV networks, these DLEs will be deployed on Multi-access Edge Computing (MEC) servers instead of running on the remote cloud. Extensive experimental results using popular evaluation metrics and average prediction time on a MEC testbed demonstrate the effectiveness of the proposed scheme.

Introduction

In recent times, the adoption of IoT has been seen as an enabler for many futuristic wireless networks including vehicular networks such as the Internet of Vehicles (IoV). IoV, as a promising paradigm for next-generation Intelligent Transportation Systems (ITS), is expected to bring in significant changes in the underlying communication and transportation infrastructure [1]. It is expected to support a wide range of applications for the next generation ITS using wireless communications technologies.

As part of the IoV architecture, the participating vehicles would send and receive a huge amount of data. The IEEE standard 802.11p based DSRC and 5G based cellular Vehicle to Everything (V2X) are widely being adopted as the communication protocols for Vehicle to Infrastructure (V2I) communication, that is, between the vehicle On-board Units (OBUs) and the Road Side Units (RSUs), and also for Vehicle to Vehicle (V2V) communication. The communicated data may include various parameters such as the position and velocity coordinates of the vehicle, the time stamp at which the data is sent, and the real/pseudo identities of the vehicles, and so on. This data received at the infrastructure nodes can be used for better vehicle mobility management, road traffic management, driver assistance, road safety applications, and many such applications.

Along with the various benefits offered, this transformation also brings forth various challenges to address. With the explosion of vehicular traffic on the roads, and with the number of V2V and V2I communication links expected to grow at an unprecedented rate, the possibility of the number and type of cybersecurity attacks also increases manifold [2, 3]. This calls for the development of state-of-the-art security schemes that can handle this unprecedented attack scenario. Blockchain [4] and cryptography-based authentication schemes [5] have been proposed in the past; however, there is also a need for intrusion detection schemes that can serve as additional layers of security to address the greater security requirements of IoV networks compared to traditional vehicular networks.

Several intrusion detection techniques addressing vehicular networks that use traditional detection and classification approaches such as secure access authentication [6], big data collection with mutual authentication [7], and machine learning approaches have been presented in the literature. However, with the extremely large amount of vehicular data being generated at an unprecedented rate and with the various types of security attacks being launched, traditional intrusion detection schemes may not be suitable in these scenarios. Thus, there is a need to develop more advanced Artificial Intelligence (AI)-based intrusion detection techniques for detecting malicious entities in vehicular networks. Deep learning algorithms in particular are better suited in the IoV scenario, which unlike machine learning algorithms do not require extensive feature extraction techniques to be carried out on the dataset.

Some deep learning-based techniques have been studied in this regard by researchers for securing vehicular networks in recent works. A Con-

Digital Object Identifier:
10.1109/MWC.001.2000428

Tejasvi Alladi and F. Richard Yu are with Carleton University; Varun Kohli and Vinay Chamola (corresponding author) are with BITS Pilani; Mohsen Guizani (corresponding author) is with Qatar University.

1536-1284/21/$25.00 © 2021 IEEE

volutional Neural Network (CNN) based sensor anomaly detection technique has been proposed by researchers [8] for preventing crashes in autonomous vehicles. Another work by the authors of [9] used CNNs for intrusion detection. However, the number of attacks considered in their work is limited. CNNs nevertheless have been found to be quite useful for intrusion detection in vehicular networks.

Further, vehicular networking applications are real-time, thus there is also a need to consider the time-sensitive nature of these applications. In addition to meeting the security requirements, the proposed security solution should also be a low latency time-sensitive one. Loukas *et al*. [10] present a deep learning-enabled technique for intrusion detection in vehicular networks in which they propose offloading the detection tasks onto the cloud servers. However, offloading to the cloud leads to high latency issues.

Multi-access Edge Computing (MEC) has emerged as a promising initiative for mitigating the latency issues observed in cloud-based offloading applications. MEC brings the computing resources to the network edge, that is, closer to the nodes which require the resources. Several research works have shown the feasibility of deploying edge servers in vehicular networks also. Ning *et al*. proposed a deep reinforcement learning algorithm running on MEC servers to show the benefits of offloading to the edge servers [11]. Another work [12] proposed an intelligent path planning scheme based on vehicular edge computing architecture. Other works combining deep learning techniques such as CNNs with edge computing have also been discussed in the literature [13].

The major contributions of this article are as follows:

- We propose an AI-empowered MEC-based intrusion detection architecture for identifying and classifying various types of cybersecurity attacks in the IoV network.
- Two different classification techniques are presented as part of this architecture. One classification technique works on time sequences generated from the vehicular broadcast data received at the RSUs, and the other technique works on the image representations of these time sequences.
- We demonstrate the feasibility of the proposed architecture by simulating four different Deep Learning Engines (DLEs) on a real-environment edge device such as a Raspberry Pi 3B.

The rest of this article is organized as follows. The following section discusses the taxonomy of security attacks in IoV. The proposed intrusion detection architecture is then presented. We then discuss the dataset and pre-processing steps involved. Following that we discuss the simulation environment used and the numerical results. The article is concluded in the final section.

Taxonomy of Security Attacks in IoV

In this section, we first discuss the most important security attack types possible in IoV networks before presenting the proposed intrusion detection architecture in the next section. A combination of one or more of these attack types can also be carried out.

Denial of Service Attack: A vehicle transmitting data at a frequency higher than the standard limit set for the network leads to a denial of service (DoS) attack. Through a DoS attack in the current IoV scenario, the legitimate vehicles in the network are denied service from the RSUs. A possible variant of DoS is a DoS random attack in which the data fields of the messages sent by an attacker vehicle are all random.

Sybil Attack: A vehicle may assume multiple valid pseudonymous identities and thus gain a disproportionate advantage in comparison to the other vehicles in the network. Since the pseudonyms are all valid identities, the detection of this type of attack is a challenge.

Data Replay Attack: A vehicle in the network may re-transmit or replay the data previously received from some other vehicle. However, it uses its own identity to replay the data. Thus, identifying this attack type is also a challenge since the replayed data appears as if it is coming from a normal non-attacker vehicle transmitting data, without changing the transmission frequency or without the use of multiple pseudonymous identities.

Disruptive Attack: This is a type of data replay attack where a vehicle replays the previously received messages from multiple vehicles selected at random. The attacker vehicle could launch this attack to prevent genuine information from being broadcast by flooding the network with such redundant data.

AI Empowered MEC-Based Intrusion Detection Architecture

In this section, we discuss the proposed AI-empowered MEC-based intrusion detection architecture. As part of the proposed architecture, we discuss the network model considered and the AI classification techniques used.

IoV Network Model

The IoV network model considered in this article consists of vehicle OBUs, RSUs, and MEC servers as shown in Fig. 1. The messages sent by the vehicle OBUs are received by the RSUs, which forward the messages to the MEC servers connected to the RSUs. Since RSUs themselves are generally not deployed with heavy computing resources, in this network model the MEC servers provide the necessary computational power to run the pre-trained DLEs. When a vehicle comes in the communication range of an RSU, these DLEs are run to predict whether the vehicle is of normal type or one of the attacker types and classify it accordingly.

Intrusion Detection Techniques

Here we discuss the two intrusion detection techniques proposed for detecting intrusion and classifying the vehicular network traffic into their respective attack types. In both techniques, the first step is to pass the vehicle data through a sequence generating pre-processing engine to create sequences of time series data. These sequences are used further in either of the techniques discussed below for classifying into a possible attack type.

Sequence Classification: In the first technique, time sequences are sent directly into a time-sequence classification DLE without further pre-pro-

Since RSUs themselves are generally not deployed with heavy computing resources, in this network model the MEC servers provide the necessary computational power to run the pre-trained DLEs. When a vehicle comes in the communication range of an RSU, these DLEs are run to predict whether the vehicle is of normal type or one of the attacker types and classify it accordingly.

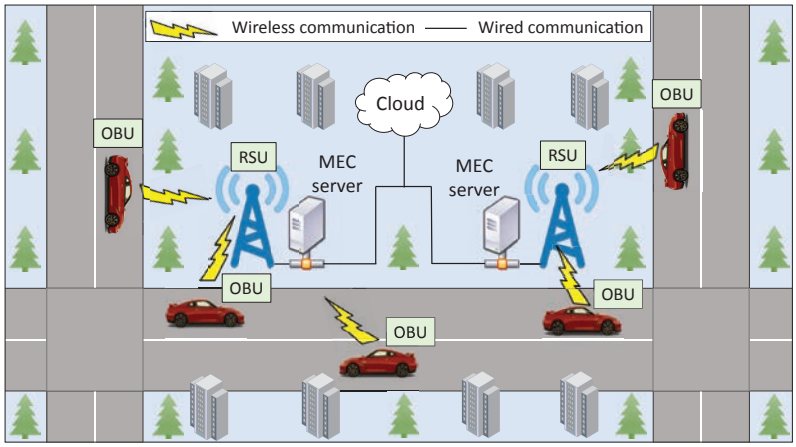

FIGURE 1. IoV network model.

cessing. This classification engine can be based on Long Short Term Memory (LSTM), which is a type of deep neural network/deep learning model.

Sequence-Image Classification: In the second technique, time sequences are first converted into images by passing through an image generation engine. These images are then fed into an image-classification DLE. This classification engine can be based on CNN, another type of deep neural network/deep learning model.

DLEs for Sequence Classification

LSTM is a type of artificial recurrent neural network (RNN) deep learning model with feedback connections, unlike feed-forward neural networks. It finds applications in areas involving time-series data such as speech recognition, handwriting recognition, and intrusion detection systems for network traffic. An LSTM unit consists of a cell and three gates, namely the input, output, and forget gate, each of which performs specific functions to regulate the flow of information into the cell, out of the cell, and retention of information in the cell, respectively. The ability of LSTMs to remember information while being insensitive to the time gaps between them makes them well suited for time-series classification and processing. LSTMs overcome the problem of vanishing gradients that have plagued other RNN architectures and hidden Markov models. In this study, we use a stacked LSTM model of four LSTM layers and an output dense layer activated by Rectified Linear Unit (ReLU) activation.

A variant of LSTMs called the CNN-LSTM model is an LSTM model designed for sequence prediction for time series data. In this study, we use a CNN-LSTM model comprised of two CNN layers and one LSTM layer. The hyper-parameters of the LSTM and CNN-LSTM models will be discussed in more detail later.

DLEs for Sequence-Image Classification

Multilayer perceptrons (MLPs) are a class of feed-forward artificial neural networks (ANNs) that consist of at least three layers, namely an input layer, a hidden layer, and an output layer. Each neuron in every layer is connected to all the neurons of the next layer, thus making it a fully connected network. All neurons apart from the input nodes are activated by a non-linear activation function which may be sigmoid, hyperbolic tangent (tanh), softmax, or ReLU. MLPs employ back-propagation, a supervised deep learning technique. The multiple layer network combined with non-linear activation enables an MLP to learn non-linear functions and thus distinguish data that is linearly inseparable. MLPs are function approximators that use regression analysis to create mathematical models, and since classification is a regression problem, MLPs can be used to make good classifier algorithms. We use a four-layer MLP in our study.

CNNs are a type of deep neural network primarily employed to analyze visual imagery. They are also referred to as space-invariant or shift-invariant ANNs. CNNs take advantage of hierarchical patterns in data to identify complex patterns using simpler patterns. Their connectivity within the network is less extensive than MLPs, which are fully connected neural networks, and thus CNNs are less complex. In comparison to other image classification algorithms, CNNs require little data pre-processing. The network learns the filters to identify features in the data independently, thus removing the requirement of feature engineering which is a major advantage compared to other neural network models. The CNN model used in this study is comprised of two CNN layers. The hyper-parameters of the CNN and MLP models used will also be discussed in detail later.

Deployment on MEC Servers

The proposed DLEs are deployed on MEC servers in this IoV network. The training of various deep learning models on the training dataset is carried out on the resource-rich cloud servers, while the time-critical prediction tasks are carried out on the DLE corresponding to the trained deep learning models on locally deployed MEC servers. As discussed earlier, the MEC servers are connected to the RSUs. An overview of the proposed intrusion detection architecture is graphically depicted in Fig. 2.

Dataset and Pre-Processing

In this section, we discuss the details of the dataset used and the pre-processing steps carried out on it. The above-discussed classification techniques require the generation of time-series data as the first step before they can be passed through either of the classifiers. We demonstrate the working of these classification DLEs by using a dataset termed VeReMi Extension [14], which was generated by the authors of [15] using a popular open-source simulation tool called VEINS. This dataset consists of log files containing traces collected from each vehicle that has traversed the network in a 24-hour time duration. It also has one ground truth file each for every hour of simulation consisting of data points for all the vehicles that have traversed the network in that hour. Each data point is a message transmitted by a traveling vehicle in the network and consists of different types of fields such as timestamp, pseudo-identity of the vehicle, and X, Y coordinates of position, velocity, acceleration, and heading.

We identified and considered a total of 10 class types, including normal vehicle type (labeled as class 0) and nine attack types (labeled as classes 1-9) from the original VeReMi Extension dataset for the creation of our customized dataset. Class-

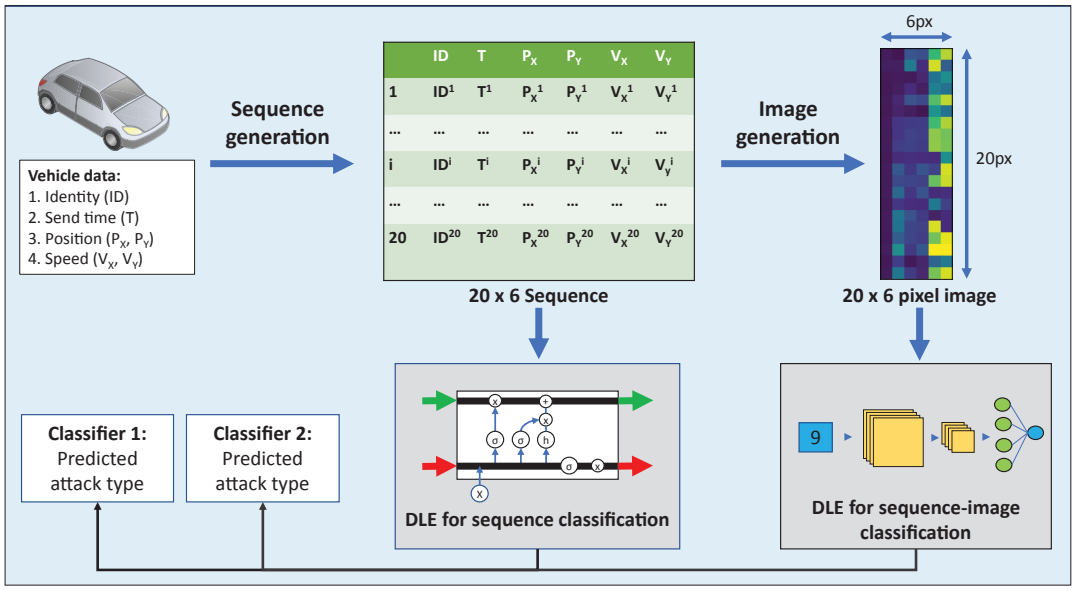

FIGURE 2. Artificial Intelligence-empowered intrusion detection architecture.

es 1-3 refer to disruptive, data replay, and DoS attacks, while the rest of the classes are different possible combinations of the four primary attack types we discussed earlier. To pre-process the dataset, we chose six fields, namely time stamp, pseudo-identity, X and Y position coordinates, and X and Y velocity coordinates, from the fields discussed earlier. We also labeled each data point (which is originally unlabelled in the ground truth file) through a preliminary match of the messages from the ground truth file with the individual vehicle log trace file. We created multiple sequences of dimension 20x6, which consists of 20 data points comprising of the above mentioned six fields. For sequence-image classification, the sequences were normalized and converted to 20x6 pixel grayscale images, that are fed as (20x6x1) dimensional arrays to the CNN model, and as a 120 dimension input to the MLP. A pictorial example of the sequence-images for each of the 10 class types, from class 0 to class 9, is shown in Fig. 3. Although these are originally grayscale images, for a better visual contrast we are showing them using shades of yellow and blue colors.

Simulation Environment

In this section, we present the simulation environment used to simulate the MEC servers in the IoV network considered. We also present the hyper-parameters of the DLEs discussed earlier. A Raspberry Pi 3B (Rpi) device was used as a real-environment MEC server to test the performance of the proposed DLEs on the vehicular traffic data. This IoV scenario data is readily available from the VeReMi Extension dataset as discussed above. The four DLEs were developed using Keras, a Python library for neural networks. They were trained on Google Colab cloud services which run on GPU backend engines and tested on Rpi.

Each of the four DLEs discussed was trained using Adam optimizer at an initial learning rate of 10^{-3}, which was then changed to 3×10^{-4} for a smoother convergence to a local minimum. For the direct sequence classification technique, the sequences of dimension 20×6 were given to the

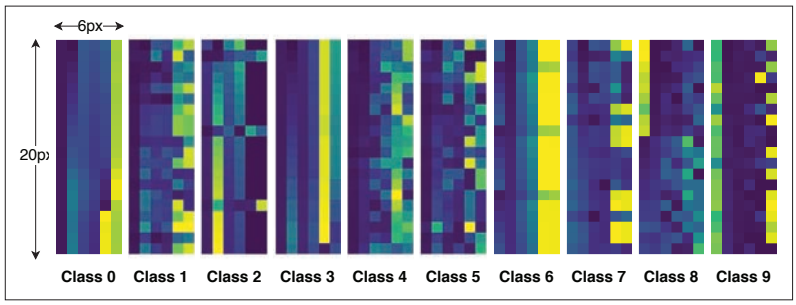

FIGURE 3. Pictorial representation of sequence-images for all the class types considered.

LSTM and CNN-LSTM models, and the loss was calculated as the mean absolute error for a single output node prediction. The LSTM model is comprised of four stacked LSTM layers of 512 units each, followed by an output dense layer of one unit, activated by ReLU. The CNN-LSTM model, on the other hand, consists of ReLU activated 1D CNN layers of 1024 and 512 filters, each with kernel sizes of 8 and stride lengths of 1. These are followed by a 1D MaxPooling layer, an LSTM layer of 512 units, and a dense layer with one unit which is activated by ReLU.

In contrast to the sequence classification DLEs, the CNN and MLP models were trained to make sequence-image classification DLEs which predict one-hot encoded label vectors of dimension 10 for the corresponding input, with categorical cross-entropy as the loss function. The CNN model used in our study consists of two 2D convolutional layers with 100 and 50 filters each, and both with filter sizes and strides respectively of (3, 3) and (1, 1). These are followed by a 0.25 dropout and the data is flattened, to be sent to two fully connected dense layers of 50 and 10 neurons, respectively. The MLP model, on the other hand, is comprised of an input layer (120 neurons), two hidden layers (1024 neurons and 256 neurons), and an output layer (10 neurons). All layers are activated by the ReLU activation function, except the output layer, which is activated by the softmax activation function.

Method	Sequence classification								Sequence-image classification							
Model	LSTM				CNN LSTM				MLP				CNN			
Class	$\mathcal{A}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}1$	$\mathcal{A}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}1$	$\mathcal{A}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}1$	$\mathcal{A}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}1$
0	100	100	100	100	99.07	92.06	99.43	95.6	98.48	90.54	95.71	93.06	100	100	100	100
1	98.13	89.47	92.29	90.86	98.55	94.91	90.57	92.69	95.58	78.35	78.57	78.46	98.36	92.96	90.57	91.75
2	98.59	96.59	89.14	92.72	97.81	91.27	86.57	88.86	94.31	73.33	69.14	71.18	98.37	90.81	93.14	91.96
3	99.53	95.63	100	97.77	99.82	98.31	100	99.15	99.47	97.71	97.43	97.57	100	100	100	100
4	99.77	99.71	98	98.85	99.91	99.71	99.43	99.57	99.91	99.71	99.43	99.57	99.97	99.72	100	99.86
5	99.77	97.77	100	98.87	99.91	99.71	99.43	99.57	98.96	95.14	95.14	95.14	99.97	100	99.71	99.86
6	99.94	100	99.43	99.71	99.94	99.43	100	99.72	99.82	99.43	98.86	99.14	100	100	100	100
7	99.77	97.77	100	98.87	99.74	97.49	100	98.73	99.24	95.01	98	96.48	99.91	99.15	100	99.57
8	100	100	100	100	99.85	98.59	100	99.29	99.88	98.87	100	99.43	100	100	100	100
9	99.77	100	97.71	98.84	99.59	100	96	97.96	99.51	99.41	96	97.67	99.91	100	99.14	99.57
Average	99.53	97.69	97.66	97.65	99.42	97.15	97.14	97.11	98.52	92.75	92.83	92.77	**99.65**	**98.26**	**98.26**	**98.26**

TABLE 1. Comparison of the evaluation metrics for the four DLEs considered.

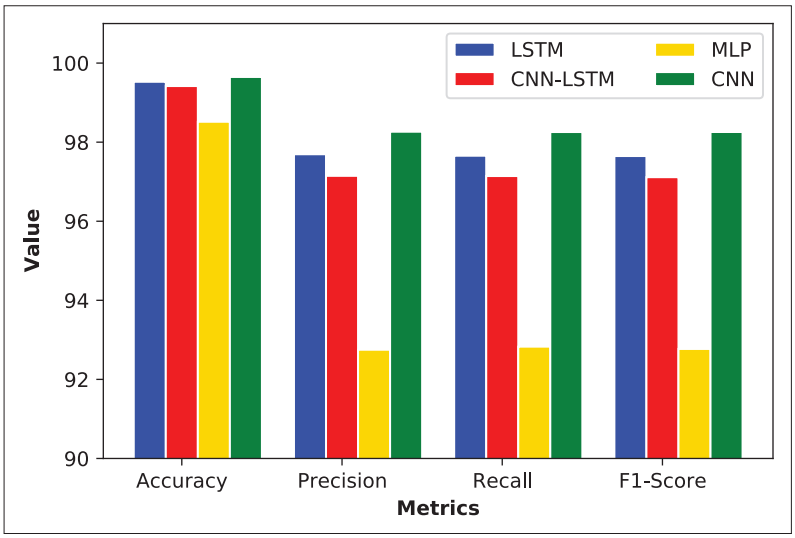

FIGURE 4. Averages of the evaluation metrics for all the models considered.

Time/model	LSTM	CNN-LSTM	MLP	CNN
Prediction	570.12	336.84	14.78	12.93
Data setup	0.36	0.32	1.31	1.36
Total	570.48	337.16	16.09	**14.29**

TABLE 2. Average run time (in milliseconds) per time sequence for each deep learning engine on Rpi.

The input dataset consists of a total of 35,000 sequences with 3,500 sequences of each class type (0-9) considered. It was split into 90 percent, 5 percent, and 5 percent to derive train, validation, and test datasets, respectively, with each of the datasets containing equal proportions of all the classes considered to ensure unbiased classification.

Numerical Results

Next, we present the results of the training experiments in terms of four popular evaluation metrics used in deep learning, namely accuracy ($\mathcal{A}$), precision ($\mathcal{P}$), recall ($\mathcal{R}$), and F1-score ($\mathcal{F}1$). These results, as shown in Table 1, present the four metrics for each of the class types considered in each of the four DLEs. Along with the metrics for each class type, we also show the average of each metric across all the classes considered. From the averages of the metrics, for sequence classification, we can see that the performance of the LSTM model is slightly better among LSTM and CNN-LSTM models; whereas for sequence-image classification, the CNN model performs significantly better compared to the MLP model. Across the two classification techniques and all four DLEs considered, the CNN model has the best performance. It can be seen that all four metrics score 100 percent for class 0 (normal vehicle data) in both LSTM and CNN models. In the CNN model, most of the attack classes fare well in the range of 99-100 percent, except for class 1 and class 2. In LSTMs, the performance is slightly worse, with most of the metrics in the range of 98-100 percent. It can also be observed that class 1 (disruptive attack) and class 2 (data replay attack) perform the worst across all the metrics for all the models considered including the CNN model. This is expected since the disruptive attack is just another variant of the data replay attack. For classification techniques without a requirement for individually identifying disruptive and data replay attacks, the two classes can be merged into one, which would further improve the prediction performance of the CNN model. It can also be noted that the MLP model being a basic deep learning model performs the worst among all the models considered, as expected. The average of the performance metrics for each model is also plotted in the bar graph shown in Fig. 4.

Taking the real-time requirements of the IoV network, and the need for low latency intrusion detection and classification into consideration, we calculate the time taken for intrusion detection on Rpi. The average run time per sequence is split into data-setup time (for converting the vehicle data into sequences for sequence classification and for converting the vehicle data into sequences followed by conversion into images for image classification) and actual prediction time (prediction of the normal/attack class type). These results are presented in Table 2. In the table, the total time taken is the sum of the data setup and prediction times in milliseconds, with the data setup time being negligible in comparison to the prediction time. While the data setup time for the sequence classification DLEs, namely LSTM and CNN-LSTM models, is less than that for sequence-image classification DLEs, namely, MLP and CNN models, the prediction time is significantly lower for the latter. As is evident from the results, the proposed CNN model is faster than all other DLEs on Rpi, with a total time of 14.29 ms only. Thus, it can be observed that the sequence-image classification

DLE based on CNNs is a more viable option for deployment on the MEC servers in the proposed architecture. Also, following the given experimentation of the discussed DLEs on Rpi, these DLEs are cost-effective since they can be deployed on the existing MEC servers without the need for investment in more infrastructure.

Conclusion

In this article, we proposed an AI-empowered intrusion detection architecture for securing the Internet of Vehicles. Deep learning-based classifier engines (DLEs) were deployed on Multi-access Edge Computing servers in this network for identifying and classifying the vehicles into potential attack types using the messages received from the vehicles. Two classification techniques, namely direct time-sequence-based classification and sequence-image-based classification, were considered. Sequence-image-based classification using Convolutional Neural Networks was shown to perform the best among all the models considered. This work considered major possible cybersecurity attack types for prediction in an IoV scenario. In the future, we plan to extend this work to include all possible misbehavior types, which can range from vehicular sensor malfunctioning to faulty data transmission scenarios in the next-generation ITS.

References

[1] B. Ji *et al.*, "Survey on the Internet of Vehicles: Network Architectures and Applications," *IEEE Commun. Standards Mag.*, vol. 4, no. 1, 2020, pp. 34–41.

[2] T. Alladi *et al.*, "Consumer IoT: Security Vulnerability Case Studies and Solutions," *IEEE Consumer Electronics Mag.*, vol. 9, no. 2, 2020, pp. 17–25.

[3] Z. Lu, G. Qu, and Z. Liu, "A Survey on Recent Advances in Vehicular Network Security, Trust, and Privacy," *IEEE Trans. Intelligent Transportation Systems*, vol. 20, no. 2, 2018, pp. 760–76.

[4] T. Alladi *et al.*, "Blockchain Applications for Industry 4.0 and Industrial Iot: A Review," *IEEE Access*, vol. 7, 2019, pp. 176 935–51.

[5] T. Alladi *et al.*, "A Lightweight Authentication and Attestation Scheme for In-Transit Vehicles in IoV Scenario," *IEEE Trans. Vehicular Technology*, vol. 69, no. 12, 2020, pp. 14 188–97.

[6] M. Tao *et al.*, "Accessauth: Capacity-Aware Security Access Authentication in Federated-IoT-Enabled V2G Networks," *J. Parallel and Distributed Computing*, vol. 118, 2018, pp. 107–17.

[7] L. Guo *et al.*, "A Secure Mechanism for Big Data Collection in Large Scale Internet of Vehicle," *IEEE Internet of Things J.*, vol. 4, no. 2, 2017, pp. 601–10.

[8] F. van Wyk *et al.*, "Real-Time Sensor Anomaly Detection and Identification in Automated Vehicles," *IEEE Trans. Intelligent Transportation Systems,* vol. 21, no. 3, 2019, pp. 1264–76.

[9] L. Nie *et al.*, "Data-Driven Intrusion Detection for Intelligent Internet of Vehicles: A Deep Convolutional Neural Network-Based Method," *IEEE Trans. Network Science and Engineering*, 2020.

[10] G. Loukas *et al.*, "Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning," *IEEE Access*, vol. 6, 2017, pp. 3491–3508.

[11] Z. Ning *et al.*, "Deep Reinforcement Learning for Intelligent Internet of Vehicles: An Energy-Efficient Computational Offloading Scheme," *IEEE Trans. Cognitive Commun. Networking*, vol. 5, no. 4, 2019, pp. 1060–72.

[12] C. Chen *et al.*, "An Intelligent Path Planning Scheme of Autonomous Vehicles Platoon Using Deep Reinforcement Learning on Network Edge," *IEEE Access*, vol. 8, 2020, pp. 99 059–69.

[13] A. Dalgkitsis *et al.*, "Data Driven Service Orchestration for Vehicular Networks," *IEEE Trans. Intelligent Transportation Systems*, 2020, pp. 1–10.

[14] "VeReMi Extension," https://github.com/josephkamel/VeReMi-Dataset, 2020; accessed 20-September-2020.

[15] J. Kamel *et al.*, "Veremi Extension: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs," *Proc. IEEE Int'l. Conf. Commun. (ICC)*, June 2020.

Biographies

Tejasvi Alladi is currently a postdoctoral researcher at Carleton University, Canada (under Prof. F. Richard Yu). He obtained his B.E. from Birla Institute of Technology and Science, Pilani, India, and an M.S. from North Carolina State University, USA in 2010 and 2015, respectively. He completed his Ph.D. degree from BITS, Pilani, India in 2020. He has approximately six years of industrial experience working on embedded systems in MNCs such as Qualcomm and Samsung. His research interests include Internet-of-Things (IoT) security, blockchain and unmanned aerial vehicles(UAVs).

Varun Kohli is currently pursuing his B.E. in electrical and electronics from Birla Institute of technology and Science (BITS), Pilani. He completed his thesis on deep learning-based security of IoT systems from the National University of Singapore in 2020 under the supervision of Prof. Biplab Sikdar. His research interests include artificial intelligence, machine learning, deep learning, IoT, and brain-computer interfaces.

Vinay Chamola is an assistant professor in the EEE Department & APPCAIR, BITS-Pilani, India. He received his B.E. (2010) and M.E. (2013) degrees from BITS, Pilani, and the Ph.D. (2016) from National University of Singapore (NUS), Singapore. His research interests include IoT, 5G network management, blockchain and security. He is an area editor of *Ad Hoc Networks*, Elsevier, and also an associate editor of *IEEE Internet of Things Magazine*, *IET Quantum Communications* and *IET Networks*.

Fei Richard Yu is a professor at Carleton University, Canada. His research interests include blockchain, security, and green ICT. He serves on the editorial boards of several journals, and is a Co-Editor-in-Chief for *Ad Hoc & Sensor Wireless Networks*, and a Lead Series Editor for *IEEE Transactions on Vehicular Technology* and *IEEE Communications Surveys & Tutorials*. He has served as a Technical Program Committee (TPC) Co-Chair of numerous conferences.

Mohsen Guizani received his B.S. (with distinction) and M.S. degrees in electrical engineering and M.S. and Ph.D. degrees in computer engineering from Syracuse University, New York, in 1984, 1986, 1987, and 1990, respectively. He is currently a professor in the Department of Computer Science and Engineering, Qatar University. He is the author/coauthor of nine books and more than 450 publications in refereed journals and conferences. His research interests include wireless communications and mobile computing, smart grid, cloud computing, and security.

This work considered major possible cybersecurity attack types for prediction in an IoV scenario. In the future, we wish to extend this work to include all possible misbehavior types, which can range from vehicular sensor malfunctioning to faulty data transmission scenarios in the next-generation ITS.