Deep Learning-based Smart Radio Jamming Attacks Detection on 5G V2I/V2N Communications

Badre Bousalem, Vinicius F Silva, Abdelwahab Boualouache, Rami Langar, Sylvain Cherrier

Deep Learning-based Smart Radio Jamming Attacks Detection on 5G V2I/V2N Communications

Badre Bousalem*, Vinicius F. Silva*, Abdelwahab Boualouache†, Rami Langar*‡, Sylvain Cherrier*
*University Gustave Eiffel, LIGM-CNRS UMR 8049, F-77454, Marne-la-Vallée, France
†FSTM, Université du Luxembourg, Luxembourg
‡Software and IT Engineering Department, École de Technologie Supérieure (ÉTS), Montréal, QC H3C1K3, Canada
E-mails: {badre.bousalem, vinicius.fonsecaesilva, rami.langar, sylvain.cherrier}@univ-eiffel.fr ;
abdelwahab.boualouache@uni.lu ; rami.langar@etsmtl.ca

***Abstract*—Vehicular-to-Everything (V2X) communication standards ensure reliable and high-performance data exchange among vehicles, pedestrians, and the roadside infrastructure. 5G New Radio (NR) is a crucial technology that enables Vehicle-to-Network (V2N) and Vehicle-to-Infrastructure (V2I) communications. In the security context, applications and network services that rely on these communication interfaces are subject to external attack sources like radio jamming that target the same control and data frequencies used by them. This causes system and network performance degradation and even Denial of Service (DoS) events, which could lead to traffic accidents involving vehicles and/or Vulnerable Road Users (VRUs). Radio jamming attacks can adopt a smart behavior by changing the targeted center frequency, bandwidth, duration, or time between two consecutive attack bursts over time. Given the context above, we propose in this paper a Deep Learning (DL)-based approach to detect radio jamming attacks on V2I/V2N communication interfaces. Our DL model is trained using a dataset collected from our 5G-V2X testbed. Results show that our DL model outperforms traditional ML algorithms and provides a detection accuracy of up to 96%, a false positive rate of less than 3%, and a detection time decrease of 39% minimum.**

***Index Terms*—5G-V2X; Security; Deep Learning; Attack Detection, Radio Jamming**

I. Introduction

Vehicular systems integrated into 5G and Beyond networks bring a new set of applications and network services to mobile users that aim to enhance their Quality of Experience (QoE) and their safety while on the road or close to it. Effective communication between vehicles and roadside infrastructure, as well as with pedestrians and cyclists, is crucial for successfully implementing automated vehicle assistance and optimal traffic management systems. These communication channels are particularly important for ensuring the safety of Vulnerable Road Users (VRUs) and improving the overall efficiency of our roads. Vehicular-to-Everything (V2X) communication standards come to fulfill these requirements and ensure high robustness, resilience, and stability levels, as well as high throughput and low latency levels, all considering scenarios that may comprise a large number and/or a large density of connected users [1].

5G New Radio (NR) networks is a crucial technology in these standardization efforts since it provides the *Uu* interface that may serve for Vehicle-to-Network (V2N) communications and the *PC5* (*sidelink*) interface that in turn may serve for Vehicle-to-Vehicle (V2V), Vehicle-to-Pedestrian (V2P) and Vehicle-to-Infrastructure (V2I) communications [2, 3]. 5G networks in their pure, independent setup (also called "Stand Alone" – SA) also bring important enablers such as Network Slicing, Software Defined Networking (SDN), and Network Function Virtualization (NFV), which allow to efficiently share and manage the network resources among heterogeneous applications and network services according to their respective Quality of Service (QoS) requirements [4].

In the security context, 5G-V2X communications also bring several concerns that could threaten vehicular users and VRUs safety, which could, in turn, prevent the successful deployment of 5G-V2X standards by the automobile and network industries [5, 6]. For example, applications and network services that depend on remote entities outside the vehicular infrastructure, i.e., that would directly depend on the V2I/V2N communication interfaces, are subject to external attack sources beyond their control domain. An important example is the deployment of radio jamming attacks, which target control and data frequencies used by V2I/V2N communications. System and network performance degradation may happen as a consequence of such attacks, which leads to a series of Denial of Service (DoS) events, preventing not only the exchange of data related to applications from mobile users but also the exchange of key control data for traffic management that could lead to traffic accidents involving vehicles and/or VRUs.

Detecting radio jamming attacks in the V2X context is a challenging task since these can adopt a smart behavior by changing over time either the targeted center frequency, and/or the bandwidth, and/or its duration and time between two consecutive attack bursts. Such behavior prevents network entities from easily detecting and/or predicting radio jamming attacks and taking proper actions to reduce the negative effects while ensuring real-time vehicles' and VRUs' safety. Radio jamming attacks have been widely discussed in the literature. However, their application in the 5G-V2X context, particularly about V2I/V2N communication interfaces, has not been thoroughly explored [7–9]. To ensure efficient detection of these attacks, Machine Learning (ML) and/or Deep Learning (DL) based techniques would enable optimal understanding of the behavior

of external malicious agents over time.

To this end, we propose, in this paper, a DL-based solution to detect radio jamming attacks on V2I/V2N communication interfaces. To do so, we first collect a dataset using our 5G-V2X testbed that is mainly focused on detecting and mitigating cybersecurity attacks, such as Distributed Denial of Service (DDoS) [10, 11]. Then, based on the obtained dataset, we train several traditional ML models and our proposed DL model. Our solution has shown promising results, with a detection accuracy of up to 96%, a false positive rate of less than 3%, and a detection time decrease of 39% minimum, compared to the traditional ML algorithms.

The rest of this paper is organized as follows. Section II discusses the related work. Section III elaborates on the 5G-V2X testbed used to collect data and train the ML/DL models. Section IV describes the methodology adopted to collect our dataset, extract its features, and train the ML/DL models. Finally, Section V shows our performance results, and Section VI concludes this work.

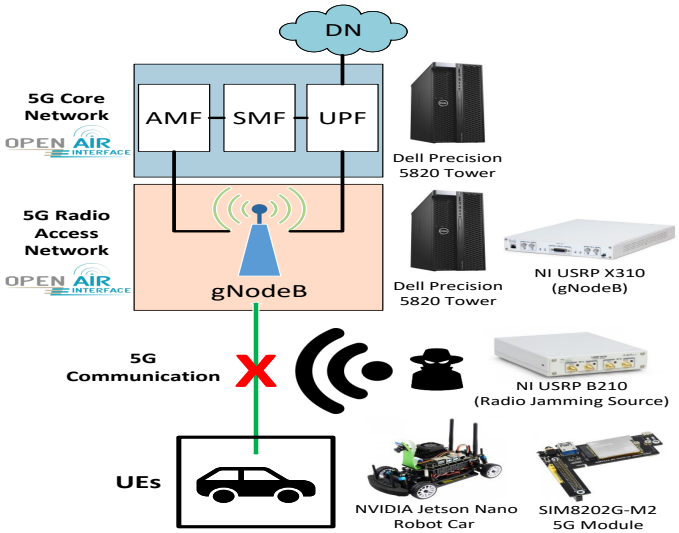


Fig. 1: 5G-V2X testbed's base hardware/software resources.

II. Related work

Several works have been put forth in the literature to maximize security measures while considering V2X communications. However, few studies still consider ML-based approaches to safeguard V2X communications from cyberattacks.

Krayani et al. [12] proposed a V2X-specific joint GPS spoofing and jamming detection method based on learning a generative interactive model and then coding the cross-correlation between RF signals transmitted by multiple vehicles and their trajectories, where their semantic meaning is coupled stochastically at a high abstraction level. In addition, a cognitive Roadside Unit (RSU) outfitted with the obtained Coupled Generalized Dynamic Bayesian Network (C-GDBN) may forecast and estimate vehicle positions based on real-time RF data. This enables RSU to determine whether both RF signals and vehicle trajectories are evolving following the dynamic rules encoded in the C-GDBN and, as a result, to identify the source (i.e., a jammer attacking the V2I or a spoofer attacking the satellite link) of the abnormal behavior observed in the V2X environment. Lyamin et al. [13] considered two jamming strategies. Each transmitted Cooperative Awareness Message (CAM) is jammed separately in random jamming with a probability of p. Additionally, in ON-OFF jamming, a series of k CAMs is destroyed with a probability of 1 only while the ON state is present. The authors used MATLAB simulations to create their dataset while considering radio inference on CAVs. Karagiannis et al. [14] proposed an ML-based scheme to distinguish between unintended and deliberate interference (radio jamming). The authors also created a dataset using the R programming language while considering a situation with interference and various radio jammer kinds. Abhishek et al. [15] proposed an ML-based scheme to discover jamming attacks. Specifically, the authors developed a dataset using the NS3 simulator and selected two features to train their model. Kosmanos et al. [16] suggested an ML-based scheme to detect radio jamming in platoons. The authors used Veins to construct their dataset to identify these attacks, considering data from both the application and physical layers.

However, all these previous works have aimed to detect attacks in V2X communications based on the IEEE 802.11p standard, an old alternative solution for 5G-V2X. In contrast, our approach focuses on detecting radio jamming attacks on 5G-V2X, particularly V2I and V2N communication interfaces. Moreover, our DL model was trained on a dataset obtained from a realistic 5G-V2X testbed involving commercial-grade devices. Interested readers may refer to [7–9] for thorough surveys on different types of cyberattacks and misbehavior detection systems for vehicular networks.

III. 5G-V2X testbed description

Fig. 1 shows the base hardware/software resources deployed in our 5G-V2X testbed. Our work mainly relies on a 5G SA setup consisting of a gNodeB at the Radio Access Network (RAN) and a 5G Core Network (CN), this last one including, among other modules, the Access and Mobility Management Function (AMF), the Session Management Function (SMF), and the User Plane Function (UPF). More specifically, the UPF is responsible to provide users (i.e., the User Equipments – UEs) access to the Internet, i.e., the Data Network (DN). To emulate the RAN and CN elements, we use *OpenAirInterface* (OAI) [17]. OAI is an open-source software developed by Eurecom to support mobile telecommunication systems like 4G Long Term Evolution (LTE) and 5G New Radio (NR).

The 5G CN and 5G RAN elements run in separate machines. Specifically, two Dell Precision 5820 Tower (Intel Xeon W-2265 3.50GHz with 12 cores, 128GB of RAM). The RAN machine is connected to a USRP X310 card which emulates the gNodeB, thus creating a communication interface between the 5G RAN and the UE. Finally, the UE is represented

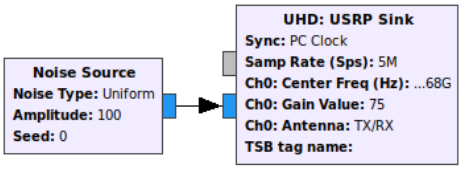


Fig. 2: 5G radio jamming source: Gnuradio block diagram.

by a JetRacer Pro robot car based on the NVIDIA Jetson Nano platform[1]. To connect with the 5G RAN, the robot car integrates a SIMCom SIM8202G-M2 5G module[2].

Our experiments generate benign traffic between the 5G CN and the UE at a fixed throughput. To do that, we use the *iperf3* [18] tool, where the 5G CN is set as the client side, and the UE is set as the server side. We use a USRP B210 card connected to the same machine where the 5G CN runs to deploy radio jamming attacks. The USRP card is controlled by a Gnuradio-based solution, which jams the same frequency band that the 5G network uses to send the data generated by the benign traffic. Fig. 2 shows the Gnuradio block diagram for the radio jamming source.

IV. Methodology

This section presents the methodology adopted to collect and pre-process our dataset and train our models to detect radio jamming attacks. More specifically, we trained multiple ML and DL models while considering each model's accuracy and detection time. We developed the ML and DL models based on multiple algorithms. Since the objective is to compare our DL-based approach with traditional ML algorithms, this section extensively details our DL model's training process.

A. Dataset collection

Several radio jamming scenarios were defined to generate our dataset. Each scenario differs from the others by its specific radio jamming behavior, which in turn is defined by two main parameters: i) the duration of one radio jamming burst and ii) the interval between two consecutive bursts. For each scenario, the UE, the gNodeB, and the radio jamming source are placed at fixed distances between them, and benign traffic generation is triggered simultaneously with the radio jamming source.

Table I shows the values considered for all radio jamming scenarios. We highlight that the values for the radio jamming behavior parameters were defined based on preliminary experiments, where we observed multiple drops in the throughput once radio jamming bursts are deployed.

For each radio jamming scenario, we consider a respective combination of radio jamming burst duration and the interval between bursts, according to the values indicated in Table I. Using the *tshark* tool, we capture network packets at the UE side, thus generating one PCAP file. We also collect *iperf3*'s performance output at the UE (server) side to assess the Rx throughput over time and associate it with the respective network packet capture.

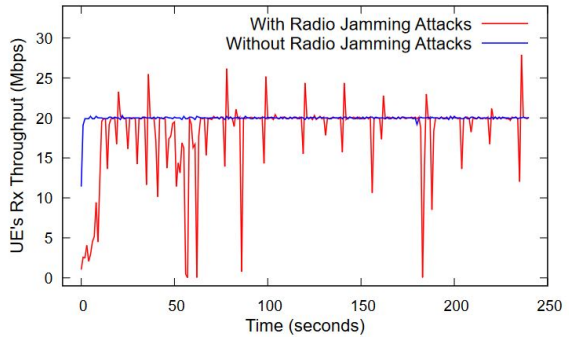
[1]https://www.waveshare.com/jetracer-pro-ai-kit.htm
[2]https://www.waveshare.com/sim8202g-m2-5g-for-jetson-nano.htm

TABLE I: Data collection parameters for the radio jamming scenarios.

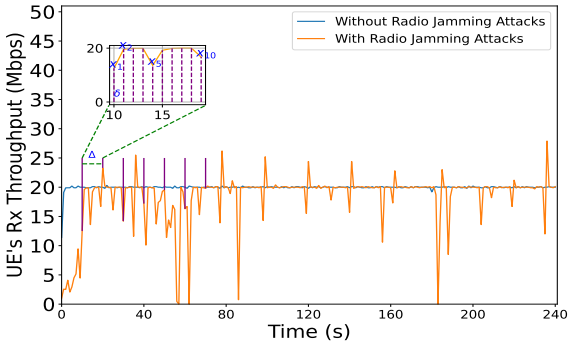
Parameter	Values
Jamming Burst Duration	100, 150, 200, 250, 300 milliseconds
Interval between Jamming Bursts	5, 10 seconds
Jamming Gain	75 dB
Targeted Frequency	3319.18 MHz (5G Band n78)
Benign Traffic Throughput	20 Mbps
Data Collection Time per Scenario	4 minutes

In addition to the radio jamming scenarios, we included in our dataset an extra one where no radio jamming attacks are deployed, i.e., only *iperf3* runs between the 5G CN and the UE, whose data collection duration and throughput is the same as previously done (i.e., 4 minutes and 20 Mbps, respectively).

To further illustrate the radio jamming behavior captured in our dataset, Fig. 3a shows the Rx throughput (in Mbps) measured by the UE in one of the scenarios here considered, where the radio jamming burst duration is of 250 milliseconds and the interval between radio jamming bursts is of 5 seconds. In the same figure, such a scenario is compared to the one where no radio jamming attacks are deployed.



(a) Radio jamming behavior example.



(b) Feature extraction.

Fig. 3: Radio jamming behavior and feature extraction.

As we can observe in Fig. 3a, the Rx throughput suffers multiple drops over time at different magnitudes for the scenario with radio jamming attacks. This can be explained by the radio jamming source being an independent and external entity, which is not timely synchronized with the 5G network elements (i.e., the 5G CN/RAN and UE). Hence, the impacts of each burst on the Rx throughput may vary constantly over time. Additionally, the throughput peaks observed right after most drops may happen due to an immediate adjustment by the *iperf3* tool to recover the fixed throughput of 20 Mbps.

B. Feature extraction

To utilize DL, we must gather features containing information about the throughput changes between two consecutive time instances. Our approach involves using a signal sampling method to extract features that indicate the variation between two sequential throughput values. The procedure entails converting the throughput trace into a series of samples, each one serving as a single feature for the dataset. Fig. 3b illustrates the feature extraction process from the inter-throughput trace. We monitor the sampling process using the sampling length (Δ) and interval (δ). We divide the trace into δ fragments and then compute each fragment's inter-variation (x_i). The resulting consecutive Δ samples represent the feature vector $X = [x_{i+1}, x_{i+2}, ..., x_{i+j}, ..., x_{i+\Delta}]$. As in signal processing, the smaller the sampling interval, the more accurately the features can represent the original signal and, in this scenario, the inter-throughput variation.

C. Model training

Our detector of radio jamming attacks on 5G V2I/V2N communications was built using a DL multi-class model that underwent a training and optimization process involving several key decisions. Firstly, we selected a diverse and substantial dataset that was split into training, validation, and test sub-datasets, 70% allocated for training, of which 10% was used for validation, and 30% for testing. We also applied data augmentation methods in our dataset to improve generalization.

The model's architecture comprises an input layer with 10 neuron nodes, five hidden layers with 1000 hidden nodes for each layer, and an output layer with 3 nodes based on one hot encoding. To speed up learning, we used the *ReLU* activation function for the hidden nodes and the *softmax* function for the output layer. The ADAM optimizer with a learning rate of 0.01 was also used to calculate the weights of the DL model. To prevent overfitting, dropout layers at a rate of 0.3 were added. A fine tuning of hyperparameters was performed, including mini-batches of size 64 and 1500 epochs for training. Please refer to Table II for the DL model's training parameters.

We used Python with the *scikit-learn*, *Tensorflow*, and *Keras* tools to train and test our DL model and compare the performance with traditional ML algorithms.

TABLE II: Training parameters of the DL model.

Parameter	**Value**
Optimizer	ADAM
Learning Rate	0.01
Batch Size	64
Dropout	0.3
Epochs	1500
Ratio of Validation/Test Dataset	10%/30%

V. Performance results

In this section, we describe the performance metrics and the results obtained with the ML and DL models, whose training procedure is described in Section IV, to detect radio jamming attacks using data collected from our 5G-V2X testbed.

A. Performance metrics

This section presents the performance metrics used to evaluate the developed ML and DL models. Our main objective is to compare the test results obtained using our custom dataset, to determine if our developed models detect and classify the presence of radio jamming attacks correctly, and to address any overfitting and underfitting problems. To do so, we use the following metrics:

- **The model's accuracy (ACC):** How accurate a model is in detecting and classifying traffic as normal or jammed:
$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (1)$$
- **Precision (PPV):** The model's positive predictive value:
$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$
- **Recall - True Positive Rate (TPR):** The probability that an actual attack will be detected and classified correctly:
$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$
- **False Positive Rate (FPR):** The probability that normal traffic will be classified as jammed traffic:
$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (4)$$
- **F1 Score (F1):** The harmonic mean between precision (PPV) and recall (TPR) calculated separately for each class and then averaged:
$$\text{F1} = 2 \times \frac{\text{PPV} \times \text{TPR}}{\text{PPV} + \text{TPR}} \quad (5)$$
- **Detection Time (DT):** The time (in seconds) spent by ML/DL models to detect a radio jamming attack.

B. Machine learning models' results

In this section, we present and discuss the results obtained using traditional ML algorithms, in terms of ACC, PPV, TPR, FPR, F1 and DT metrics, by using our custom dataset collected through our 5G-V2X testbed.

With the validation set as input, all ML models achieved an ACC of 99%. Table III shows the performance results obtained for the same ML models with the test set as input.

TABLE III: ML models results with the test set as input.

ML Model	Performance Metric					
	ACC	PPV	TPR	FPR	F1	DT
Logistic Regression	0.4561	0.3852	0.3793	0.6207	0.3356	0.37
KNN	0.7894	0.7867	0.8309	0.1691	0.8027	0.18
SVM	0.6491	0.7222	0.7296	0.2704	0.6701	0.24
Naive Bayes	0.6491	0.7252	0.6896	0.3104	0.6963	0.75
Decision Tree	0.7894	0.7976	0.8327	0.1673	0.8102	0.18
Random Forest	0.7719	0.8069	0.8212	0.1788	0.8033	0.19

Concerning model *Logistic Regression*, we can see that ACC, PPV, TPR, FPR and F1 are quite low for the test set in comparison with the validation set and in comparison with the other ML models. Moreover, DT is not the lowest among the other ML models. Such a result shows that this model is underfitting and does not correctly classify the presence or absence of a radio jamming attack.

Furthermore, models *SVM* and *Naive Bayes* provided similar results in terms of ACC, PPV, TPR, FPR, F1, and DT besides performing better in comparison with *Logistic Regression*. However, these two models still perform poorly with an ACC of around 65% and a DT of 0.24 and 0.75 seconds, respectively, in comparison with *KNN*, *DecisionTree* and *Random Forest*, that provide an ACC of around 78% and a DT of around 0.18 seconds, which shows that model *Naive Bayes* has the slowest DT among all ML models. The same two models also perform poorly when compared to the results with the validation set as input. This indicates that these models are underfitting, preventing us from using them in our 5G-V2X testbed.

It is important to highlight that models *KNN*, *DecisionTree* and *Random Forest* provided the best results in terms of ACC, PPV, TPR, FPR, F1 and DT, in comparison to the previously mentioned algorithms, with the results of *KNN* and *DecisionTree* being the two best ones in terms of ACC (around 79%) and DT (around 0.18 seconds), *DecisionTree* being the best one in terms of TPR (around 83%), FPR (around 16%), F1 (around 81%) and *Random Forest* being the best in terms of PPV (around 80%). However, these results cannot be applied reliably on our 5G-V2X testbed since the accuracy for detecting and correctly classifying each type of radio jamming attack is not very high.

Finally, an overall observation of the results above is that the traditional ML algorithms have a regular performance in detecting radio jamming attacks. However, their performance results can be enhanced since the FPR is still considered high, even in the case of the models that provided the best results. Such a result indicates that the model will likely incorrectly classify radio jamming attacks.

C. Deep learning model's results

In this section, we present and discuss the results obtained using our DL model in terms of ACC, PPV, TPR, FPR, F1 and DT, using our custom dataset obtained through our 5G-V2X testbed. As observed for the aforementioned ML models in Section V-B, our DL model also achieved an ACC, PPV, TPR, FPR and F1 of 99%, with the validation set as input. Table IV shows the performance results obtained for our DL model with the test set as input.

TABLE IV: DL model results with the test set as input.

Performance Metric					
ACC	PPV	TPR	FPR	F1	DT
0.9591	0.9783	0.9794	0.0206	0.9676	0.11

TABLE V: DL model results for each traffic class.

Traffic Class	Performance Metric				
	ACC	PPV	TPR	FPR	F1
Benign	0.9930	1.0000	1.0000	0.0000	0.9997
RJ Class I	0.9146	0.9481	0.9513	0.0487	0.9329
RJ Class II	0.9697	0.9868	0.9871	0.0129	0.9702

By observing the results from Table IV, it is evident that the DL model has better results than the ML models presented in Section V-B, in terms of ACC, PPV, TPR, FPR, F1 and DT. We highlight the results obtained for ACC and FPR, which attained around 96% and less than 3%, respectively. Along with the other metrics above, such results demonstrate the absence of underfitting and overfitting on this model since it performs well on the validation and test sets and shows the capacity of the DL model to identify a radio jamming attack promptly. Indeed, we can see from this table that our DL model reduces considerably the DT by around 39% and 85% compared to the fastest ML models observed in Table III (i.e., *KNN* and *Decision Tree*) and the slowest one (i.e., *Naive Bayes*), respectively.

Table V presents the performance results obtained for each of the three traffic classes in our dataset: (i) the *Benign* class: This class represents the context where we have normal traffic without radio jamming attacks, (ii) the *Radio Jamming (RJ) Class I*: This class represents the scenario where we deploy radio jamming attacks with a time interval of 5 seconds between jamming bursts, and finally (iii) the *RJ Class II*: This class represents the scenario where we deploy radio jamming attacks with a time interval of 10 seconds between radio jamming bursts.

The results observed from Table V show that the DL model accurately classifies and distinguishes between each traffic class, with an ACC of around 99%, 91% and 97% for the *Benign*, *RJ Class I* and *RJ Class II* classes, respectively.

To further illustrate the results obtained with our DL model, we show the model's ACC in Fig. 4 and the loss in Fig. 5 during each epoch, with the test set as input. By observing Fig. 4, we can notice an upward trend, showcasing the model's ability to learn and improve over time. It converges to around 96%, indicating that the DL model has learned enough to make correct predictions with high consistency. On the other hand, Fig. 4 shows a consistent descent in the loss, achieving approximately 3%, which means that the DL model

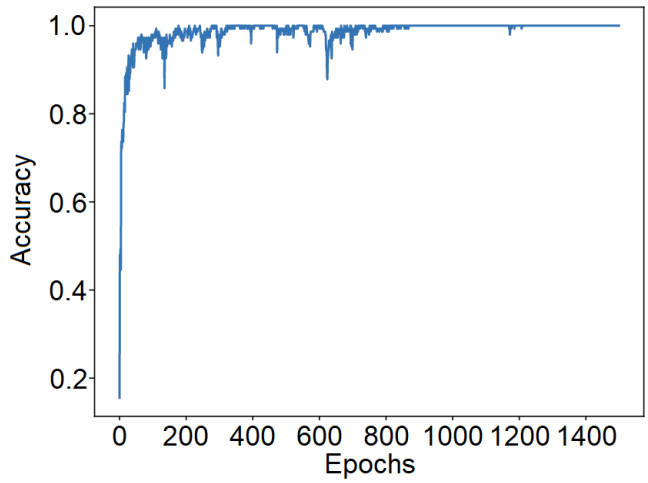


Fig. 4: DL model accuracy results.

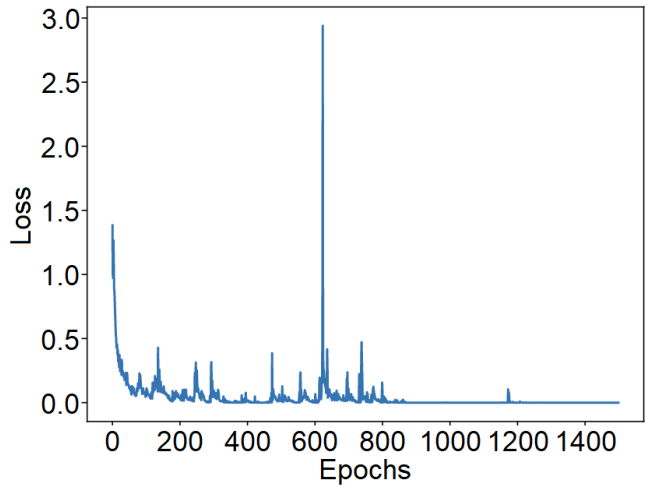


Fig. 5: DL model loss results.

has effectively minimized its error, optimizing its predictive capabilities to near perfection.

Given the behavior observed above, and the fact that the DL model provides the best results compared to all ML models, we can conclude that it can be reliably deployed for detecting and classifying radio jamming attacks in a 5G-V2X environment.

VI. Conclusion

In this work, we presented a new approach based on DL to detect radio jamming attacks on 5G V2I/V2N communication interfaces. To assess the validity of our approaches, we built a 5G-V2X testbed to collect the data needed to train and test our developed models. We compared the performance of traditional ML models with our DL model in terms of prediction accuracy, false positive rate, detection time, and other relevant ML-related performance metrics using our custom dataset generated by the 5G-V2X testbed. For our DL model, we obtained an accuracy of up to 96%, a false positive rate of less than 3%, and a detection time decrease of 39% minimum.

In future work, we intend to mitigate radio jamming attacks by using network slicing to isolate the radio jamming sources. In addition, we intend to explore new techniques to detect radio jamming attacks by using Long Short Term Memory Networks (LSTMs) and Recurrent Neural Networks (RNNs), and compare their performance results with those obtained by our DL model. Finally, we plan to extend our dataset by adding new features and new radio jamming-based scenarios, then make it publicly available to enable the research community to reproduce our experiments.

Acknowledgment

This work was supported by the 5G-INSIGHT bilateral project (ID: 14891397) / (ANR-20-CE25-0015-16), funded by the Luxembourg National Research Fund (FNR), and by the French National Research Agency (ANR). It was also partially supported by funding from the Innovation for Defence Excellence and Security (IDEaS) program from the Department of National Defence (DND) Canada.

References

[1] A. Alalewi *et al.*, "On 5G-V2X use cases and enabling technologies: A comprehensive survey," *IEEE Access*, vol. 9, 2021.
[2] V. Mannoni *et al.*, "A comparison of the V2X communication systems: ITS-G5 and C-V2X," in *Proc. Veh. Technol. Conf. (Spring)*, 2019.
[3] D. Garcia-Roger *et al.*, "V2X support in 3GPP specifications: From 4G to 5G and beyond," *IEEE Access*, vol. 8, pp. 190 946–190 963, 2020.
[4] I. Afolabi *et al.*, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Commun. Surv. Tuts.*, vol. 20, no. 3, pp. 2429–2453, 2018.
[5] 3GPP, "Study on security aspects of enhanced network slicing," TR 33.813, V0.8.0, Tech. Rep., Nov 2019.
[6] ETSI, "Intelligent transport systems security: Threat, vulnerability and risk analysis," TR 102 893 V1.2.1, Tech. Rep., Mar. 2017.
[7] M. Gayathri and C. Gomathy, "A deep survey on types of cyber attacks in VANET," *J Crit Rev*, vol. 8, no. 01, pp. 1029–1039, 2021.
[8] T. Sapala *et al.*, "A survey on VANET attacks and its security mechanisms," in *Proc. Intl. Conf. on Par., Dist. and Grid Comp.*, 2022.
[9] A. Boualouache and T. Engel, "A survey on machine learning-based misbehavior detection systems for 5G and beyond vehicular networks," *IEEE Commun. Surv. Tuts.*, 2023.
[10] B. Bousalem *et al.*, "Deep learning-based approach for DDoS attacks detection and mitigation in 5G and beyond mobile networks," in *Proc. IEEE 8th Int. Conf. on Netw. Softwar. (NetSoft)*, 2022, pp. 228–230.
[11] B. Bousalem *et al.*, "DDoS attacks detection and mitigation in 5G and beyond networks: A deep learning-based approach," in *Proc. IEEE Glob. Commun. Conf.*, 2022, pp. 1259–1264.
[12] A. Krayani *et al.*, "Integrated sensing and communication for joint GPS spoofing and jamming detection in vehicular V2X networks," 2023.
[13] N. Lyamin *et al.*, "AI-based malicious network traffic detection in VANETs," *IEEE Netw.*, vol. 32, no. 6, pp. 15–21, 2018.
[14] D. Karagiannis and A. Argyriou, "Jamming attack detection in a pair of RF communicating vehicles using unsupervised machine learning," *Veh. Commun.*, vol. 13, pp. 56–63, 2018.
[15] N. V. Abhishek and M. Gurusamy, "JaDe: Low power jamming detection using machine learning in vehicular networks," *IEEE Wirel. Commun. Lett.*, 2021.
[16] D. Kosmanos *et al.*, "Intrusion detection system for platooning connected autonomous vehicles," in *Proc. South East Eur. Des. Autom., Comp. Eng., Comp. Netw. and Soc. Med. Conf.* IEEE, 2019.
[17] Eurecom, "OpenAirInterface." [Online]. Available: https://openairinterface.org/
[18] ESnet, "iperf3." [Online]. Available: https://downloads.es.net/pub/iperf/