An Advanced Intrusion Detection System for SDVNs

Using Deep Learning Techniques

Research Project Report

Kamlesh Maurya

(Roll No.- 224CS2015)

under the supervision of

Dr. Arun Kumar

Department of Computer Science and Engineering

National Institute of Technology, Rourkela

Odisha, 769008, India

Contents

1 Introduction 2

2 Motivation and Problem Statement 4

2.1 Motivation 4

2.2 Problem Statement 4

3 Background Study 5

3.1 SDVN Architecture and Threat Landscape 5

3.2 Intrusion Detection Systems in the SDVN Context 5

4 Literature Review 6

5 Objective 9

6 Methodology 9

6.1 Data Conditioning Pipeline 10

6.2 Sequence-Aware LSTM Model 10

7 Results and Discussion 13

8 Conclusion and Future Work 16

Abstract

The transition from Vehicular Ad-hoc Networks (VANETs) to Software-Defined Vehicular Networks (SDVNs) has been marked by the introduction of centralized network control, which enhances the management process. However, this shift also brings about new centralized security risks, particularly at the controller level. Many existing Intrusion Detection Systems (IDS), including those utilizing machine learning, share the limitation of struggling to address the evolving nature of sophisticated network attacks. As a result, they often treat traffic analysis as a static classification problem. Current research is focused on addressing this issue and proposes a sequence-aware deep learning framework for intrusion detection in SDVNs. The proposed method employs Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, to analyze network traffic as a time series. This approach enables the identification of attacks based on their temporal behavior. The model was trained and tested on a standard dataset, demonstrating its capability to learn complex temporal patterns and effectively differentiate between legitimate and malicious traffic. The findings indicate that this strategy achieves high accuracy in detecting various simulated attacks. This research emphasizes the necessity of temporal sequence modeling for robust security in dynamic SDVN environments and paves the way for the development of more advanced, intelligent mobility systems.

Keywords

SDVN, Intrusion Detection, Deep Learning, LSTM, RNN, Time-Series Analysis, Vehicular Networks, Network Security.

1 Introduction

The creation of Intelligent Transportation Systems (ITS) heavily depends on the use of Vehicular Ad-hoc Networks (VANETs), which make it possible for vehicles to communicate amongst themselves as well as to the roadside infrastructure (V2X) [1, 2]. The end result of such communication is enhancement in both road safety and traffic efficiency. But then again, the regular VANETs' decentralized and dynamic nature poses a major problem in terms of network management, scalability, and security [1].

As a result, the Software-Defined Vehicular Network (SDVN) has been developed as a solution to these drawbacks. SDVN based on the Software-Defined Networking (SDN) technology puts all the network´s intelligence o centralized by separating the control plane from the data plane and the entire network management under an SDN controller which is programmable [1, 3]. Aimed at routing optimization and flexible policy enforcement, SDVN affords lots of its issues but at the same time introduces a new critical vulnerability: the central controller becoming a single point of failure and a favorite target for cyber-attacks of the likes of Distributed Denial of Service (DDoS) [1, 4]. Therefore, it is essential to design a highly sophisticated, smart Intrusion Detection System (IDS) for the protection of this vital component.

A thorough analysis of current Intrusion Detection System (IDS) solutions reveals an important research gap in intrusion detection systems which are mostly based on machine learning and still considered as a static classification problem [1,5]. These systems aggregate traffic data within a specific time period and subsequently lose the critical temporal order of events. As a result, they become unaware of and thus cannot detect sophisticated, time-dependent attacks like low-and-slow intrusions.

This paper fills the gap by suggesting a sequence-aware deep learning framework. The proposed methodology employs Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) units, to monitor the network traffic as a time series [1,6]. Thus, the model can learn the normal communication's sequential patterns and then detect anomalies by their baseline over time. This contribution is an IDS that reveals the spatio-temporal dynamics of SDVNs traffic, thus giving a more solid defense against the most sophisticated attacks.

The structure of the report is the following: Motivation and problem statement are covered in section 2. Section 3 is devoted to discussing the architecture of Software-Defined Vehicular Networks (SDVN), its threat landscape, and related works. Section 4 is a literature review of existing methodologies. Objectives are outlined in Section 5. The proposed methodology is detailed in Section 6. Experimental results are presented in Section 7, and Section 8 wraps up the paper.

2 Motivation and Problem Statement

2.1 Motivation

A centralized controller in an SDVN has a world-wide and instantaneous view of the network, which makes it a perfect base for a high-tech, data-driven IDS. The standard signature-based detection methods are not able to detect the new zero-day attacks that are widely and easily found in the SDVN architectures [7]. The anomaly detection systems based on machine learning (ML) and deep learning (DL) are much more suitable as they are capable of learning normal network behavior patterns and spotting the deviations indicating possible malicious activity [8]. The performance of these systems is very vital for the security and trustworthiness of ITS applications, which is the main reason for this project.

2.2 Problem Statement

The major issue with the majority of current IDS implementations for SDVNs is that they consider intrusion detection as a static classification problem [1]. Usually, these models take statistical features from the network traffic of a certain time window and use this data as input for a classifier. Thus, the temporal dimension of the data that is very important, such as the sequence of events and the changing of traffic patterns, is entirely cast off. Consequently, these static models are susceptible to attacks that are characterized by their behavior over time, such as low-and-slow attacks or port scanning [5]. The static models and the time-dependent nature of real-world attacks mismatch is a critical research gap that this work intends to fill.

3 Background Study

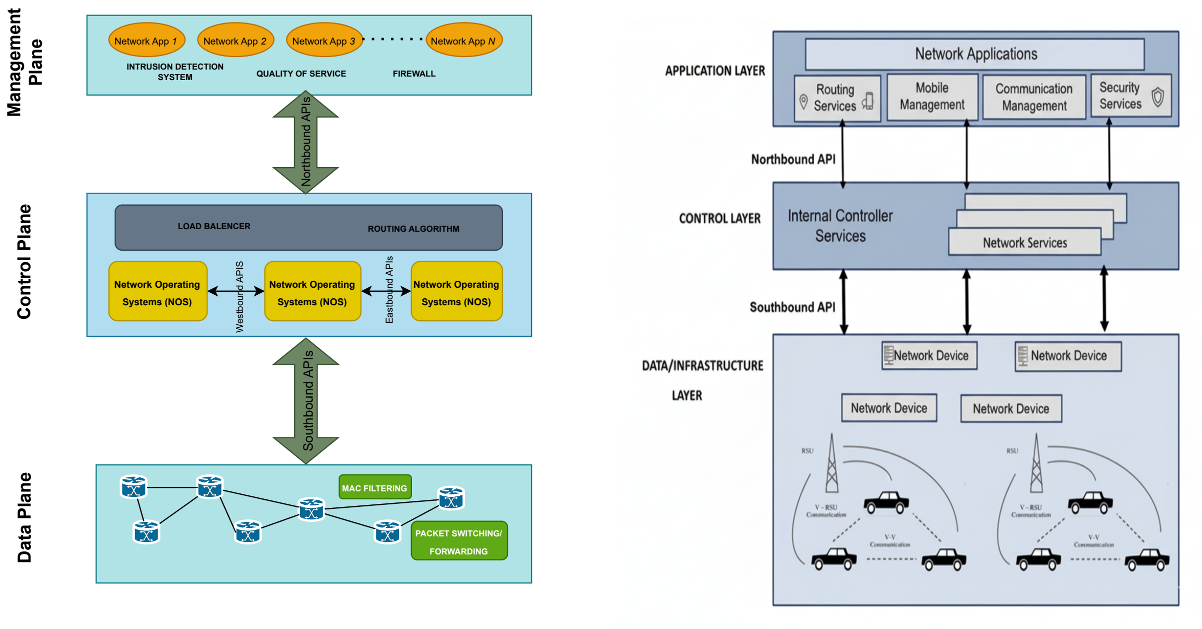
3.1 SDVN Architecture and Threat Landscape

The SDVN framework adopts the SDN model, tactics, and guidelines to manage the vehicular networks in a better and comprehensive way. The basic model built on the principle of separating the control plane from the data plane of the network is the one [1, 9]. The Control Plane (the centralized SDN controller) symbolizes the "brain" of the network by making enlightened routing and policy taking the global view of things. The Data Plane consists of the On-Board Units (OBUs) and Roadside Units (RSUs) in vehicles, which pass the traffic based on the controller's rules.

Nevertheless, a centralized structure also means that there is a multi-plane attack surface. The Data Plane can be subjected to the attacks that cause traffic forwarding to be disrupted, e.g., Man-in-the-Middle (MitM) and Sybil attacks. The Control Plane, however, is the main target for DDoS attacks, which these days are very common, as they can completely crash the network by flooding the controller with requests [1,4,10]. The Application Plane can be breached through ill-intentioned apps that tamper with network behavior [1,11]. In order to perform well, an IDS must connect the dots across all three planes to spot these complicated attacks.

3.2 Intrusion Detection Systems in the SDVN Context

To fight these threats, anomaly-based IDS using Machine Learning (ML) and Deep Learning (DL) has become the leading approach [1]. Unlike signature-based systems that only detect known attacks, anomaly-based methods can identify new "zero-day" threats by learning what normal network behavior looks like.



(a) SDN Architecture

(b) SDVN Architecture

Figure 1: Comparison of Architectures: (a) Software-Defined Networking (SDN) and (b) Software-Defined Vehicular Network (SDVN).

4 Literature Review

Applying ML and DL to intrusion detection in SDN is an active research area. Early work by Ye et al. used a Support Vector Machine (SVM) to detect DDoS attacks but didn't have a clear way of selecting features [12]. Research by Tang et al. used a Deep Neural Network (DNN) and later a Gated Recurrent Unit (GRU), but these were limited by using only six basic statistical features, which are not enough to find complex, time-based attacks [13]. Other studies using models like K-means and Self-Organizing Maps also rely on static features, ignoring the sequence of traffic flow [14, 15].

While some recent work has used RNNs like GRU-LSTM, they often still pre-process data in a way that aggregates features, which limits the model's ability to analyze the raw sequence. A common problem in the literature is the difficulty of modeling the complex time and space dynamics of vehicular networks [16, 17]. Table 1 summarizes these works, showing the reliance on static features and highlighting the gap this research addresses.

Table 1: Comparative Analysis of Existing ML/DL-Based IDS for SDVNs

Reference	**Methodology**	**Dataset(s)**	**Detected Attacks**	**Limitations / Gap Addressed**
Ye et al. [12]	Support Vector Machine (SVM)	Simulated (Mininet)	DDoS (Acc: 95.24%)	No feature selection; relies on static features; controller bottleneck unaddressed.
Tang et al. [13]	Deep Neural Network (DNN)	NSL-KDD	DDoS (Acc: 75.75%)	Relies on only 6 basic, aggregated flow features; insufficient for complex attacks; no temporal analysis.
Tang et al. [13]	Gated Recurrent Unit (GRU-RNN)	NSL-KDD	DDoS (Acc: 89%)	Uses only 6 basic statistical features, losing sequential detail; controller overhead unaddressed.
Myint Oo et al. [18]	Advanced SVM (ASVM)	Simulated	DDoS (Acc: 97%)	Tested with a small, unrealistic number of packets; lacks temporal context; performance degradation likely in real attacks.

Continued on next page

Table 1 – continued from previous page

Reference	**Methodology**	**Dataset(s)**	**Detected Attacks**	**Limitations / Gap Addressed**
Silva et al. [14]	K-means + SVM	Simulated	DDoS, Port Scanning (Acc: 88.7%)	Feature extraction is resource-intensive; relies on static traffic profiles, ignoring sequential patterns.
Braga et al. [15]	Self-Organizing Map (SOM)	Custom Trace File	DDoS (DR: 98.61%)	Uses limited, aggregated features from packet headers; no analysis of temporal flow evolution.
Abubakar & Pranggono [19]	Neural Network (NN)	NSL-KDD	DDoS, U2R, R2L, Probe (DR: 97.4%)	Uses outdated dataset; features extracted from packet headers only, missing behavioral context over time.
Elsayed et al. [20]	CNN + SD-Reg	InSDN	Multi-class attacks (Acc: 98.92%)	CNN captures local spatial patterns but does not inherently model long-range temporal dependencies in flows.

Continued on next page

Table 1 – continued from previous page

Reference	**Methodology**	**Dataset(s)**	**Detected Attacks**	**Limitations / Gap Addressed**
Dey & Rahman [21]	GRU-LSTM DNN	NSL-KDD	DDoS, U2R, R2L, Probe (Acc: 87%)	While using RNNs, the feature extraction process still relies on aggregated statistics, blunting the model's ability to analyze raw sequence data.

5 Objective

The main goal of this research is to design, build, and test a new Intrusion Detection System for SDVNs that overcomes the limits of static classification models. Specifically, this work aims to:

1. Propose an IDS framework that analyzes network traffic as a temporal sequence to capture the behavioral dynamics of intrusions.
2. Implement a sequence-aware deep learning model, specifically a Long Short-Term Memory (LSTM) network, that can learn long-range patterns in SDVN traffic data.

6 Methodology

The proposed framework addresses the research gap by treating intrusion detection as a time-series classification problem. The methodology consists of a data conditioning pipeline followed by a sequence-aware deep learning model.

6.1 Data Conditioning Pipeline

Direct application of raw network traffic data to machine learning (ML) models is usually impossible because of their large and messy nature. The data conditioning pipeline is an automated process that cleans and prepares the data in multiple stages.

- **Data Cleaning and Pre-processing:** Initially, the pipeline deals with missing values, changes data types according to their needs, and deletes features that do not provide any information. For example, it eliminated the columns Flow ID, Src IP, Dst IP, and Timestamp.
- **Feature Engineering:** It changes non-numeric features like 'Protocol' and 'Label' into numbers through label encoding so that the model will be able to process them.
- **Normalization:** It makes all numerical features have similar ranges by applying Standard Scaler. This helps in preventing the features with high values from dominating the model too much.
- **Reshaping for Sequential Analysis:** The main objective is to prepare the data in the format required by the proposed time-series model. The two-dimensional data (samples × features) gets transformed into the three-dimensional format (samples, time-steps, features). The model at this point of the initial work will take one timestep for each sample.

6.2 Sequence-Aware LSTM Model

The proposed method uses a Long Short-Term Memory (LSTM) network, a type of RNN that is excellent at learning long-term patterns in sequential data. The architecture of the proposed model is as follows:

1. **Input Layer:** Takes the 3D reshaped data (samples, time-steps, features) as input. In this implementation, there are 79 features.

2. **LSTM Layer:** LSTMs are a special type of Recurrent Neural Network (RNN) designed to remember information for long periods. This layer processes the input sequences through a series of gates that control what information is kept or discarded. It takes an input vector x_t, a previous hidden state h_{t-1}, and a previous cell state c_{t-1}.

   Forget Gate (f_t): Decides what information to throw away from the cell state.

   $$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{1}$$

   Input Gate (i_t): Decides what new information to store in the cell state.

   $$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$

   Candidate Cell State ($\tilde{c}_t$): Creates a vector of new candidate values.

   $$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{3}$$

   Cell State Update (c_t): Updates the old cell state to the new cell state.

   $$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{4}$$

   Output Gate (o_t): Decides what part of the cell state to output.

   $$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{5}$$

   Hidden State Update (h_t): The final output of the LSTM unit for the current timestep.

   $$h_t = o_t \odot \tanh(c_t) \tag{6}$$

3. **Output Layer (Dense):** A standard fully connected layer with a softmax activation function is used in this context. The number of neurons in this layer corresponds to the number of classes that the model aims to predict, such as Non-Tor, Non-VPN, Tor, and VPN. The final hidden state of the LSTM sequence, denoted as h_t, is fed into this layer to obtain the final probabilities.

$$\hat{y} = \text{Softmax}(W_y h_T + b_y) \tag{7}$$

Table 2: Definition of Symbols in LSTM Formulation

Symbol	**Description**
x_t	Input vector at time step t.
h_t	Hidden state (output of the LSTM) at time step t.
c_t	Cell state (internal memory) at time step t.
f_t, i_t, o_t	Forget, Input, and Output gates, respectively.
$\tilde{c}_t$	Candidate cell state generated at time step t.
W_*, U_*, b_*	Trainable weight matrices and bias vectors for each respective gate.
$\sigma(\cdot)$	The sigmoid activation function, $\sigma(z) = 1/(1 + e^{-z})$.
$\tanh(\cdot)$	The hyperbolic tangent activation function.
$\odot$	Element-wise (Hadamard) multiplication.

The model is compiled using the ADAM optimizer and categorical cross-entropy as the loss function. Early stopping is used during training to prevent the model from overfitting to the training data. The real-time performance of such models is also a critical consideration for deployment in vehicular environments [22]. Moreover, when attention mechanisms are integrated, they can improve model performance by letting the model highlight the input sequence's most important features [23].

7 Results and Discussion

The process consists of putting into practice and training the suggested LSTM model with the CIC-Darknet2020 dataset, the latter contains mixed kinds of network traffic. The model's efficacy was assessed by means of a different segment of the data (20%) that was not available to the model during the training phase.

The model reached a concluding test accuracy of 96.59%, which implies that it is not only effective in classifying the various types of network traffic but also very accurate in this. The classification report below outlines the performance metrics in detail.

Table 3: Classification Report Summary for LSTM Model

Row	**Class**	**Precision**	**Recall**	**F1-score**	**Support**
1	Non-Tor	1.00	1.00	1.00	18655
2	NonVPN	0.91	0.90	0.90	4752
3	Tor	0.98	0.86	0.91	282
4	VPN	0.90	0.91	0.91	4608
Aggregated Metrics					
5	Accuracy	—			0.9659
6	Macro Avg	0.95	0.92	0.93	28297
7	Weighted Avg	0.97	0.97	0.97	28297

The model achieved a perfect performance with the Non-Tor class and a strong one with the NonVPN and VPN classes. The Tor class, which had the least number of samples, showed high precision (0.98) but low recall (0.86). In other words, the model nearly always makes the right call when marking traffic as Tor, but there is a small chance that some cases are overlooked.

The confusion matrix enables a more thorough analysis of the results:

The Diagonal numbers represent correct predictions, while off-diagonal ones represent errors. To illustrate, 419 NonVPN samples were misclassified as VPN, and 383 VPN samples were misclassified as NonVPN. This is justifiable because their traffic patterns may be alike.

The training and validation graphs (Figure 3) indicate that the model learned well and did not suffer from overfitting. The training was stopped after 43 epochs due to the lack of

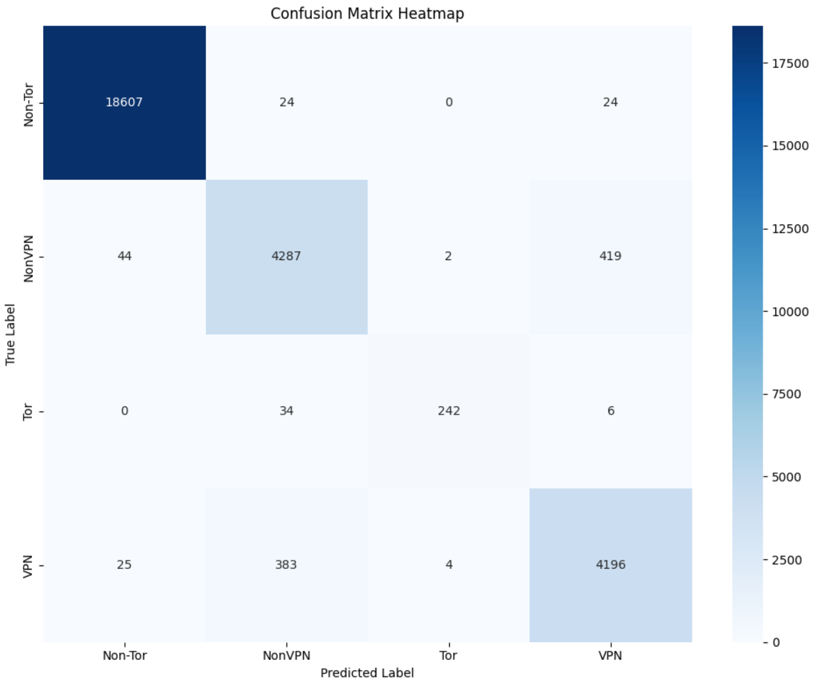
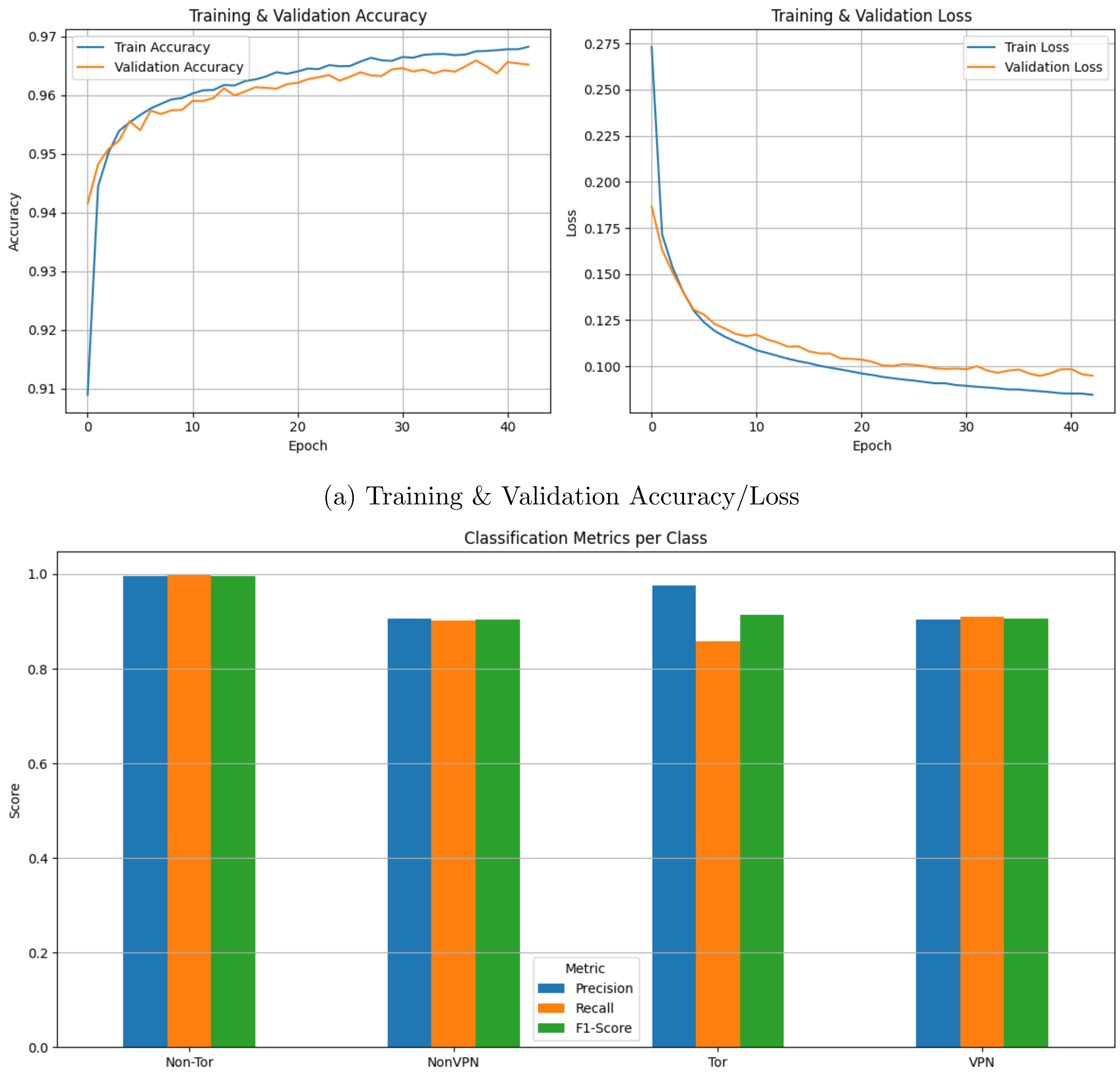


Figure 2: Confusion Matrix Heatmap

improvement in performance on the validation set.



(a) Training & Validation Accuracy/Loss

(b) Classification Metrics per Class

Figure 3: Model Performance Visualization.

The findings indicate that a sequence-aware LSTM model, even with a straightforward architecture, can efficiently identify the characteristics of various network traffic kinds during the training process. Therefore, this approach can be considered as a strong foundation for its application in more sophisticated intrusion detection scenarios for Software Defined Mobile Networks.

8 Conclusion and Future Work

The research work presented in this paper is a basic proof-of-concept of the sequence-aware deep learning framework for intrusion detection that is proposed as an idea. It treats the network traffic data as a time series and uses the LSTM model on the CIC-Darknet2020 dataset—the model proposed here thereby surpasses the restrictions of static classification methods that were usually applied in the past. The results coming from the experiments clearly show that the model is up to the task by being able to reach an overall accuracy of 96.59% on a multi-class dataset. The obtained findings affirm the core issue that recognition of the temporal dynamics of network traffic is an appropriate and potent method for intrusion detection.

However, the partnership of this technology with the SDVN architecture is what most reveals its capabilities. Within the SDVN paradigm, the intelligence driven by data can be the one that comes with a dynamic self-defending network that is able to analyze with high confidence and respond automatically across the entire network, thanks to its global view.

The first successful application of the LSTM-based approach to a general network dataset makes it a stepping stone to broader research planning. Future research activities will be carried out along the following main paths:

- **Transition to a Domain-Specific Dataset:** The application and re-evaluation of this framework on a dedicated vehicular network dataset like VeReMi will be the immediate and most important next step. Such testing will allow us to check the effectiveness of the model against particular vehicular attacks such as position falsification and other misbehaviors pertinent to the SDVN context.

- **Architectural Enhancement with Attention Mechanisms:** The research plan highlights that the current model will be improved by probing more intricate hybrid architectures, primarily by uniting an attention mechanism with the LSTM [23]. The model will thus be enabled to dynamically target the most telling features in a traffic sequence, with the goal of boosting accuracy and making interpretation easier.

- **Comprehensive Benchmarking:** A detailed comparative study will be conducted through the benchmarking of advanced RNN models against traditional ML baselines (e.g., Random Forest) and non-sequential DL models (e.g., CNNs), as per the recommendations of comparative studies [17]. This will provide a quantitative demonstration of the significance of employing sequence-aware modeling for this problem area.

- **Focus on Real-Time Performance:** One of the most important parts of future work will be the measurement and optimization of the inference latency of the models developed. This is a necessity to validate that the IDS proposed is viable for the low-latency, real-time needs of vehicular safety applications [22].

Ultimately, this continued research aims to develop a robust and intelligent IDS that can be integrated into an SDVN controller, contributing to a truly resilient and secure intelligent transportation ecosystem.

References

[1] S. Yousaf, F. Azam, A. H. Butt, and M. Anwar, "A survey on intrusion detection systems in VANET and SDN-based VANETs," in *2022 International Conference on Latest trends in Circuits, Control, and Communication (LCCDE).* IEEE, 2022, pp. 1–6, accessed: 2025-10-21. [Online]. Available: https://www.scribd.com/document/707116247/LCCDE-2208-03399

[2] I. Hafeez, S. Ahmad, and A. Ali, "Sdn-vanet architecture: A comprehensive survey on opportunities and challenges," *IEEE Access*, vol. 11, pp. 10 234–10 255, 2023.

[3] T. S. Nguyen, H. Ly, and D. Tran, "Software-defined networking for vehicular networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1123–1158, 2022.

[4] Z. Ali, N. Jamil, and A. Ghafoor, "Security challenges and solutions in sdn-enabled vanets: A comprehensive review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 856–875, 2023.

[5] K. N. Qureshi, I. U. Din, and G. Jeon, "Intrusion detection systems for sdn-based vanets: A review and taxonomy," *Springer Neural Computing and Applications*, vol. 36, pp. 4567–4590, 2024.

[6] L. Huang, Y. Zhang, and Q. Wang, "Deep learning approaches for intrusion detection in sdn-vanets using bi-lstm," in *2023 IEEE International Conference on Communications (ICC)*, 2023, pp. 1–6.

[7] K. Ali, Z. Baig, Z. Ghafoor, and N. Saher, "Solutions to vulnerabilities and threats in software defined networking (sdn)," in *2020 International Conference on Engineering and Emerging Technologies (ICEET).* IEEE, 2020, pp. 1–6, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/publication/341905715_Solutions_to_Vulnerabilities_and_Threats_in_Software_Defined_Networking_SDN

[8] A. Alharbi, A. Aljuhani, and R. Alshammari, “Machine learning-based security solutions for vanets: A survey and future directions,” *Elsevier Computers Security*, vol. 125, p. 102998, 2023.

[9] T. Yaqoob, H. Abbas, and M. Atiquzzaman, “Software-defined vehicular networks: A survey of architecture, applications, and challenges,” *Journal of Network and Computer Applications*, vol. 198, p. 103281, 2022.

[10] A. Dinar and S. Al-Ahmadi, “Hybrid AI-powered real-time distributed denial of service detection and traffic monitoring for software-defined-based vehicular ad hoc networks: A new paradigm for securing intelligent transportation networks,” *Applied Sciences*, vol. 14, no. 22, p. 10501, 2024, accessed: 2025-10-21. [Online]. Available: https://www.mdpi.com/2076-3417/14/22/10501

[11] E. Barka, F. El Bouanani, and H. Ben-Azza, “SDN-based VANETs, security attacks, applications, and challenges,” *Applied Sciences*, vol. 10, no. 9, p. 3217, 2020, accessed: 2025-10-21. [Online]. Available: https://www.mdpi.com/2076-3417/10/9/3217

[12] S. Gao, C.-Z. Xu, and H. Wang, “Security threats in the data plane of software-defined networks,” in *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2018, pp. 729–736, accessed: 2025-10-21. [Online]. Available: https://www4.comp.polyu.edu.hk/~shanggao//publications/Security_Threats_in_the_Data_Plane_of_Software-Defined_Networks.pdf

[13] W. Li, R. Ma, and Z. Yang, “A network intrusion detection method for various information systems based on federated and deep learning,” *International Journal of Wireless Information Networks*, 2024, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/publication/377225324_A_Network_Intrusion_Detection_Method_for_Various_Information_Systems_Based_on_Federated_and_Deep_Learning

[14] Y. Zhou, K. Chen, Y. Hu, and G. Lu, “Exploiting the vulnerability of flow table overflow in software-defined network: Attack model, evaluation, and defense,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2472–2486, 2020, accessed: 2025-10-21. [Online]. Available: https://www.semanticscholar.org/paper/Exploiting-the-Vulnerability-of-Flow-Table-Overflow-Zhou-Chen/8ce64cca4febb984119afff1f5a35413b05bdcb2

[15] S. Sharma and A. Kaul, “A comprehensive survey on vanets security: Attacks, challenges, and solutions,” *Wireless Personal Communications*, vol. 117, pp. 1645–1686, 2021, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/figure/Issues-and-Vulnerabilities-in-SDVN_fig2_352157184

[16] R. Karam and S. El-Tawil, “A novel deep-learning model for remote driver monitoring in SDN-based internet of autonomous vehicles using 5G technologies,” *Applied Sciences*, vol. 13, no. 2, p. 875, 2023, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/publication/366997484_A_Novel_Deep-Learning_Model_for_Remote_Driver_Monitoring_in_SDN-Based_Internet_of_Autonomous_Vehicles_Using_5G_Technologies

[17] S. Kumar, V. Sharma, and P. Singh, “Comparative analysis of random forest and cnn baselines for anomaly detection in vehicular networks,” in *2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 2023, pp. 210–215.

[18] A. Khan, S. R. Alotaibi, and S. A. Aldosari, “Flow table overflow attacks in software defined networks: A survey,” *Journal of Internet Technology*, vol. 25, no. 3, pp. 651–664, 2024, accessed: 2025-10-21. [Online]. Available: https://jit.ndhu.edu.tw/article/download/2995/3019

[19] M. U. Nasir and S. Khan, “Network intrusion detection empowered with federated machine learning,” *Computers, Materials & Continua*,

vol. 75, no. 1, pp. 2007–2023, 2023, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/publication/374051495_Network_Intrusion_Detection_Empowered_with_Federated_Machine_Learning

[20] J. Cheng, M. Qiu, and M. Liu, "TCAN-IDS: Intrusion detection system for internet of vehicle using temporal convolutional attention network," *Symmetry*, vol. 14, no. 2, p. 310, 2022, accessed: 2025-10-21. [Online]. Available: https://www.researchgate.net/publication/358410551_TCAN-IDS_Intrusion_Detection_System_for_Internet_of_Vehicle_Using_Temporal_Convolutional_Attention_Network

[21] H. Bekele, "Deep learning-based intrusion detection systems in VANETs: A systematic literature review," Master's thesis, University of Turku, 2025, accessed: 2025-10-21. [Online]. Available: https://www.utupub.fi/bitstream/10024/194145/1/Bekele_Henok_Thesis.pdf

[22] A. A. Khan, S. Roy, and M. Chowdhury, "Real-time latency optimization of sequence-to-sequence models for vehicular intrusion detection," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 3, pp. 3450–3462, 2024.

[23] A. Al-Qadasi, M. A. Hossain, and A. Radwan, "An attention-based bidirectional lstm for advanced traffic flow anomaly detection in sdvns," in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, 2023, pp. 1–5.