An Advanced Intrusion Detection System for SDVNs Using Deep Learning Techniques

Kamlesh Maurya
Department of Computer Science and Engineering
National Institute of Technology, Rourkela
Rourkela, India
224cs2015@nitrkl.ac.in

Arun Kumar
Department of Computer Science and Engineering
National Institute of Technology, Rourkela
Rourkela, India
kumararun@nitrkl.ac.in

***Abstract*—The shift from Vehicular Ad-hoc Networks (VANETs) to Software-Defined Vehicular Networks (SDVNs) entails a centralized management of the network that increases the control but at the same time creates new spots for security breaches, most likely at the controller. A majority of existing Intrusion Detection Systems (IDS) including the ones powered by machine learning, often miss the time aspect of the advanced network attacks, viewing them instead as static classification problems. The main aim of the research is to fill the gap by creating a deep learning-powered sequence-aware intrusion detection system for SDVNs. Recurrent Neural Networks (RNNs) are employed in this technique to assess the network data as a time series, thus attack detection is done based on the detection of behavioral patterns. The scheme goes through a thorough training and testing process with a standard dataset which enables the model to accurately tell apart the benign (good) and malicious (bad) network traffic flows by spotting complex temporal patterns. The research claims that already trained and tested the system achieves high accuracy in the identification of different simulated attack types. Additionally, the results attribute a crucial role to the temporal sequence modeling in maintaining solid security in the rapidly changing SDVN environments and even laying the groundwork for smarter, more secure transportation systems.**

***Index Terms*—SDVN, Intrusion Detection, Deep Learning, LSTM, RNN, Time-Series Analysis, Vehicular Networks, Network Security**

I. Introduction

Intelligent Transportation Systems (ITS) are the modern-day conveniences that rely on the connectivity offered by the Vehicular Ad-hoc Networks (VANETs) which facilitate the vehicles and the roadside infrastructure communication (V2X) [1], [2]. The aspect of safety on the roadway is amplified by this connectivity along with the aspect of traffic efficiency. But at the same time, the decentralized and dynamic nature of the traditional VANETs leads to major difficulties in terms of management, scalability, and security [1].

To begin with, the Software-Defined Vehicular Network (SDVN) is the novel network paradigm that solves the aforementioned problems. Essentially, the SDVN clears the way for controlling the network and the management of all the data through the use of an SDN controller while still being termed a network due to its dependable nature [1], [3]. The routing is thus optimized and policy enforcement becomes flexible. Even though SDVN resolves the majority of the VANET problems, it does so at the price of a new significant vulnerability—the centralized controller which now becomes the single point of failure and the most attractive target for the Distributed Denial of Services (DDoS) kind of cyber-attacks [1], [4]. Therefore, the need for a sophisticated, intelligent Intrusion Detection System (IDS) at this critical component's defense is recognized.

A review of the currently available Intrusion Detection Systems (IDS) reveals a significant gap in research: most of the systems, among which a large number are machine learning based ones, consider intrusion detection as a static classification problem [1], [5]. By means of traffic data aggregation over a specific time frame, they miss what is the most critical temporal sequence of events. This, in turn, renders them unable to detect sophisticated, time-dependent attacks such as the low-and-slow intrusions.

The present paper fills the research void by putting forth a sequence-aware deep learning framework. The method employs Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) units, to interpret network traffic as a time series [1], [6]. The model thereby not only learns the sequential patterns of normal communication but also detects anomalies based on their behavior over time. The primary objective of this research is to create, realize, and evaluate a newly developed IDS for Software Defined Wireless Networks (SDWNs) that is not constrained by the deficiencies of static classification models. In particular, the project aims at proposing an IDS architecture that treats network traffic as a temporal sequence and uses a sequence-aware deep learning model (LSTM) that can discover long-range patterns in SDWN traffic data.

II. Background and Related Work

A. SDVN Architecture and Threat Landscape

The SDVN architecture takes advantage of the principles of SDN to enhance the control and efficiency of vehicular networks. It is primarily the segregation of the network's control plane from the data plane that is responsible for this [1], [7]. The Control Plane represented by the centralized SDN controller, is termed the ”brain” of the network, as it makes intelligent routing and policy decisions based on a global view provided by the Data Plane.

On the other hand, the centralized design in question puts the whole network at risk as it has a multi-plane attack surface. The Data Plane attackers may cause traffic forwarding disruptions by using methods like Man-in-the-Middle (MitM) and Sybil attacks while the Control Plane, in turn, becomes a major target for Distributed Denial of Service (DDoS) attacks, which can exhaust the controller's resources and thus, take down the entire network [1], [4], [8]. Moreover, the Application Plane is not safe either, where the attacks may originate from malevolent apps that would tamper with the network behavior [1], [9]. An effective IDS is a crucial requirement as it should span events from all three planes to be able to recognize such intricate attacks.

B. Literature Review

Research on applying ML and DL to intrusion detection in SDN is still ongoing and active. An early study by Ye and colleagues proposed the use of a Support Vector Machine (SVM) for detecting DDoS attacks, however, the method did not provide a clear direction for feature selection [10]. Tang et al. explored the use of a Deep Neural Network (DNN) in their research and later on even a Gated Recurrent Unit (GRU), however, the limitation was that they could only work with six basic statistical features which are not sufficient for identifying time based complex attacks [11]. K-means and Self-Organizing Maps were among the models used in other studies that also applied static features, thus ignoring the sequence of the traffic flow [12], [13].

Recently, GRU-LSTM and other RNNs, have been used, but they still tend to pre-process data in a way that feature aggregation is applied, which limits the model's capability of analyzing the raw sequence. The difficulty of modeling the complex time and space dynamics of vehicular networks is a common issue in the literature [14], [15]. The works presented in Table I indicate reliance on static features and also highlight the gap that this research is filling, as they summarize the literature.

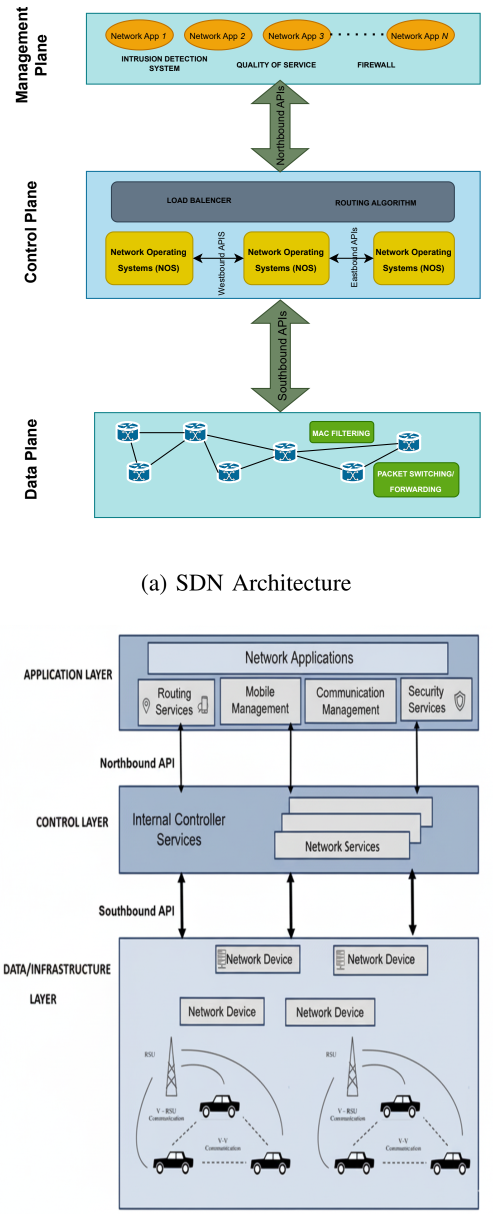
III. Methodology

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.

A. Data Conditioning Pipeline

Firstly, raw network traffic data is very large and sometimes dirty, and therefore, it cannot be directly used for ML models. Secondly, data is gradually cleaned and prepared for modeling purposes through this pipeline.

- Data Cleaning and Pre-processing: Initially, it replaces missing values, changes data types to the correct ones, and drops unnecessary features. As an example, it removed columns that contained the Flow ID, Src IP, Dst IP, and Timestamp.
- Feature Engineering: It labels text-based features such as 'Protocol' and 'Label' with numbers by applying label encoding, thus making them readable for the model.



(a) SDN Architecture

(b) SDVN Architecture

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

- Normalization: It applies Standard Scaler to all numerical features to scale them to a similar range. Thus, features with large values cannot dominate the model.
- Reshaping for Sequential Analysis: The most important step is preparing data for the temporal model. It changes the 2D data (samples × features) into 3D (samples, timesteps, features). For this initial research, each sample corresponds to a single timestep.

TABLE I: Comparative Analysis of Existing ML/DL-Based IDS for SDVNs

Reference	**Methodology**	**Dataset(s)**	**Detected Attacks**	**Limitations / Gap Addressed**
Ye et al. [10]	Support Vector Machine (SVM)	Simulated (Mininet)	DDoS (Acc: 95.24%)	No feature selection; relies on static features; controller bottleneck unaddressed.
Tang et al. [11]	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. [11]	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. [16]	Advanced SVM (ASVM)	Simulated	DDoS (Acc: 97%)	Tested with a small, unrealistic number of packets; lacks temporal context; performance degradation likely in real attacks.
Silva et al. [12]	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. [13]	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 [17]	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. [18]	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.
Dey & Rahman [19]	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.

B. 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, timesteps, 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.

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

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

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

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

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

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

3) Output Layer (Dense): A standard fully connected layer with a softmax activation function. The number of neurons corresponds to the number of classes (e.g., Non-Tor, Non-VPN, Tor, and VPN).

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

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.

IV. Results and Discussion

The method involves implementing and training the proposed LSTM model using the CIC-Darknet2020 dataset. The model's performance was evaluated on a separate portion of the data (20%) that it had not encountered during training.

The model achieved a final test accuracy of 96.59%, which shows it is very effective at classifying different types of network traffic. The detailed performance metrics are shown in the classification report in Table II.

TABLE II: Classification Report Summary for LSTM Model

Class	**Precision**	**Recall**	**F1-score**	**Support**
Non-Tor	1.00	1.00	1.00	18655
NonVPN	0.91	0.90	0.90	4752
Tor	0.98	0.86	0.91	282
VPN	0.90	0.91	0.91	4608
Accuracy				**0.9659**
Macro Avg	0.95	0.92	0.93	28297
Weighted Avg	0.97	0.97	0.97	28297

The performance of the model was exceptional for the Non-Tor class and also quite strong for the NonVPN and VPN classes. In the case of the Tor class, which was represented by the least number of samples, the precision was great (0.98), but the recall was less (0.86). This indicates that when the model labels a traffic as Tor, it is nearly always right, but it may sometimes overlook a few cases.

The confusion matrix shown in Fig. 2 provides a more granular view of the outcomes. The numbers on the diagonal are the instances where the predictions were correct. To illustrate, 419 samples from the NonVPN class were wrongly predicted to be from the VPN class, and 383 samples from the VPN class were incorrectly predicted to be from the NonVPN class. This is quite reasonable because their traffic patterns might be closely related.

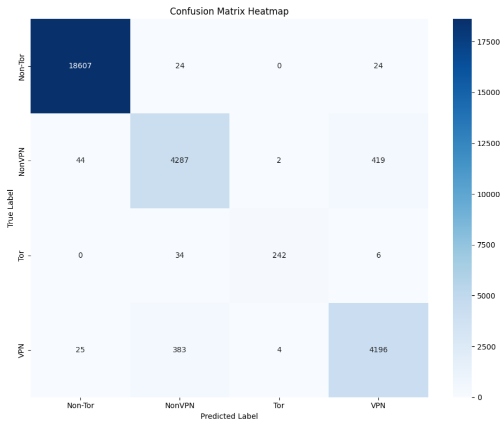


Fig. 2: Confusion Matrix Heatmap

The training and validation graphs (Fig. 3) show that the model learned effectively without significant overfitting. The training was stopped early after 43 epochs when the performance on the validation set stopped improving.

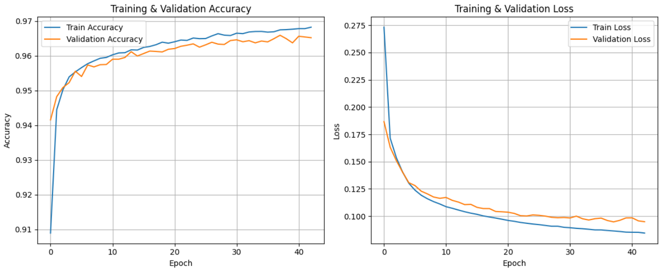


Fig. 3: Training & Validation Accuracy/Loss

The outcomes have already established that the sequence-aware LSTM has the capability to learn the features of various types of network traffic, which is surprisingly good considering its simplicity. This gives a solid basis for the application of this technique in more sophisticated intrusion detection cases for SDVNs in the future.

V. Conclusion and Future Work

This paper proposes a basic proof-of-concept for a sequence-aware deep learning framework for intrusion detection. By regarding network traffic as a time series and applying an LSTM model, the suggested method circumvents the drawbacks of traditional static classification methods. The experimental results, which achieved a total accuracy of 96.59%, confirm the idea that modeling the temporal dynamics of network traffic is a strong approach for intrusion detection.

The success of LSTM-based method validation was a significant milestone in our future comprehensive research. The next steps are going to be through the following main areas:

- Transfer to a Domain-Specific Dataset: Next, we will take and implement the methodology on a vehicular network dataset, such as VeReMi, in order to test the system for specific vehicular attacks.
- Model Improvement: The current model will be enhanced by incorporating the LSTM attention mechanism [20], which will enable the model to focus on the most crucial aspects within a traffic sequence.
- Thorough Benchmarking: A complete side-by-side evaluation will be performed not only with the traditional ML methods (e.g., Random Forest) but also with non-sequential DL models (e.g., CNNs) [15].
- Real-Time Performance Emphasis: A crucial part of our future work will be to quantify and optimize the inference latency in order to confirm the IDS is up to par with the low-latency requirements of automotive safety applications [21].

This extended research ultimately seeks to create a strong and smart IDS, one that would be able to fit right into an SDVN controller, thus paving the way to a truly resilient and secure intelligent transportation system.

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.

[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] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] S. Sharma and A. Kaul, “A comprehensive survey on vanets security: Attacks, challenges, and solutions,” *Wireless Personal Communications*, vol. 117, pp. 1645–1686, 2021.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] H. Bekele, "Deep learning-based intrusion detection systems in VANETs: A systematic literature review," Master's thesis, University of Turku, 2025.

[20] 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.

[21] 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.