DeepIDPS: An Adaptive DRL-based Intrusion Detection and Prevention System for SDN

Nadia Niknami and Jie Wu
Center for Networked Computing, Temple University, USA
Emails: {nadia.niknami, jiewu}@temple.edu

***Abstract*—Most intrusion detection systems (IDS) are vulnerable to novel attacks and struggle to maintain a balance between high accuracy and a low false positive rate. Furthermore, the relevant features of Distributed Denial of Service (DDoS) attacks in conventional networks may not necessarily apply to the Software-defined network (SDN) environment. Additionally, weak feature selection algorithms can omit critical parameters and result in significant data loss. Although earlier works on network flow analysis using Long Short-Term Memory (LSTM) show excellent ability, they fall short in obtaining deep features from network flow, resulting in lower accuracy. The emergence of Attention Mechanism(AM) and deep reinforcement learning (DRL) present a promising solution for intrusion detection and enhancing security in SDN. AM has the capability to assign varying weights to different network traffic features, enabling IDS to extract and emphasize more crucial information. This paper introduces DeepIDPS, a novel DRL-based network intrusion detection system utilizing a CNN-LSTM approach and Attention Mechanism specifically designed for SDN environments. DeepIDPS demonstrates an exceptional ability for continuous auto-learning within the network context, effectively identifying diverse forms of network intrusions while significantly augmenting both prevention and detection capabilities.**

***Index Terms*—Attention Mechanism (AM), Deep reinforcement learning (DRL), Distributed Denial of Service (DDoS), Intrusion Detection System (IDS), Long Short Term Memory (LSTM), Software Defined Network (SDN).**

I. Introduction

Software-defined networking (SDN) represents a modern network architecture and design paradigm that uses software to facilitate communication between the control and data planes. By decoupling these planes, SDN streamlines network device complexity and provides adaptable network management capabilities. However, SDN networks face a substantial threat in the form of Distributed Denial of Service (DDoS) attacks. These attacks can target all layers of the SDN framework, including the data, control, and application planes, as well as the communication channels that connect the devices of the data plane and the control plane [1]. To mitigate this risk, SDN controllers often deploy Machine Learning-based Intrusion Detection Systems (IDSs) to identify and response to network attacks. The IDS functions by analyzing network traffic from SDN switches via the control layer. It scrutinizes incoming and outgoing traffic to detect patterns indicative of suspicious behavior. When such activity is identified, the IDS triggers an alarm to alert the SDN controller about the ongoing attack. This proactive approach empowers the SDN controller to take immediate action to mitigate the attack and ensure network security. Optimizing feature selection in Machine Learning models can significantly enhance classifier accuracy and detection rates while reducing execution time. The effectiveness of an IDS heavily relies on the quality of the feature construction and selection algorithms employed. However, despite the existence of various feature selection methods coupled with machine learning models for DDoS attack detection [2], these mechanisms have often proven ineffective in SDNs. A comprehensive feature extraction approach should consider both *Temporal* and *Global* [3]–[5] features to provide a robust defense against DDoS attacks.

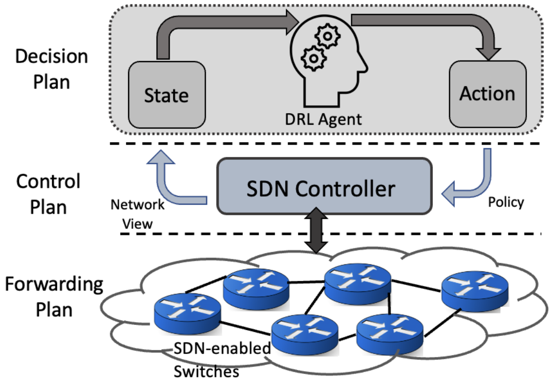


Fig. 1: RL-based SDN.

In response to these challenges, Deep Learning (DL) methods, including Artificial Neural Networks, have emerged as a promising solution. Convolutional Neural Networks(CNNs) are a common DL architecture used for various applications [6]. Its strength lies in learning intricate patterns and extracting essential features, which suits them for detecting anomalies in network traffic. While CNNs have been employed for anomaly detection, they may fall short when detecting subtle differences between normal and malicious traffic, as these differences are often minimal.

To strengthen CNN anomaly detection capabilities, we propose combining them with Long Short-Term Memory (LSTM) [7]. LSTMs networks are exceptionally adept at modeling sequential data and mitigating the vanishing gradient problem. This fusion enables the extraction of both spatial and temporal features from input data, and our approach

This research was supported in part by NSF grants CNS 2214940, CPS 2128378, CNS 2107014, CNS 2150152, CNS 1824440, and CNS 1828363.

leverages an Attention Mechanism to assign varying weights to input data, enhancing the model's capacity to extract critical information without significantly increasing computational and storage overhead. Thus, our paper introduces an Attention-CNN-LSTM method applied to intrusion detection, which has demonstrated favorable results.

Deep reinforcement learning (DRL) has shown great promise in tackling complex real-world challenges. In DRL, an agent is trained to acquire an optimal policy by actively interacting with the environment and mapping input states to corresponding actions. For this task, the agent must effectively perceive the current input state and make informed decisions. However, the proliferation of social networking platforms has raised significant concerns about privacy and data storage, making the transfer of substantial data volumes to the cloud a challenging endeavor. One potential solution to address this issue involves integrating DRL with deep neural networks (DNNs), enhancing the capabilities of DRL algorithms, and ultimately improving efficiency and effectiveness.

This paper presents a novel approach called DeepIDPS, which is an adaptive Intrusion Detection and Prevention System (IDPS) designed specifically for SDNs. DeepIDPS leverages the power of DRL to enhance its capabilities. To capture complex observation features in RL, a DNN is employed. In real-time, an RL-based agent explores the SDN environment, dynamically analyzes its changing properties, and formulates appropriate security policies. These policies are then implemented by the SDN controller, which, in turn, applies them to the switches. This iterative process continues to adapt to the evolving network conditions.

Fig. 1 depicts the architecture of SDN integrated with RL, showcasing the addition of an intelligent decision-making layer based on deep learning to the SDN controller. By utilizing network measurement techniques and gaining a holistic view of the entire network, the *intelligent decision-making* layer can generate efficient policies for intelligent network control. These policies are capable of addressing global, real-time, and personalized network control requirements. The main contributions of this paper can be summarized as follows.

- We introduce DeepIDPS, a DRL-based IDS by deploying a hybrid deep neural network to a RL component. It includes CNN and LSTM that are deployed in an intelligent decision-making layer. The RL agent has different actions based on the current situation of the network. Our proposed approach can detect and prevent attacks.
- With the help of combining CNN and LSTM, we extract important global and temporal features. Selected promising features can reduce the complexity of the model and improve the performance of the intrusion model.
- We have integrated the Attention Mechanism into the CNN-LSTM model to direct the model's attention towards high-impact features.
- We assess the effectiveness of our proposed model based on different measurements in the detection of different types of attacks and for diverse datasets.

The rest of this paper is organized as follows: Section II gives an overview of the existing machine learning (ML) and deep learning (DL) techniques that are currently used to monitor and detect threats in SDNs. The proposed model including the evaluation dataset and the experimental setup are provided in Section III. The obtained experimental results are discussed in Section IV. Finally, Section V discusses and concludes the paper.

II. Background and Related Works

A. Intrusion in SDN

SDN technology has become increasingly popular for developing network management and cybersecurity applications in networked systems. IDSs are commonly used to defend against attacks by monitoring network traffic and generating alerts upon detecting malicious activity. However, the processing capacity of the basic IDS is limited, making it impractical to inspect vast amounts of network traffic. When traffic flows exceed the IDS's processing capacity, packets are discarded without inspection. In recent years, numerous defense and mitigation techniques have been proposed to address the issue of attack detection in SDNs [8]–[10]. Wei et al. [11] presented a hybrid deep learning autoencoder multilayer perceptron network with automatic feature extraction capabilities. The autoencoder is used to extract critical features by compressing and reducing the feature space.

B. Deep learning-based IDS

Convolution Neural Networks (CNNs) are particularly adept at learning complex patterns and extracting relevant features from input data, which makes them well-suited for detecting anomalies in network traffic. Long-Short Term Memory (LSTM) networks are exceptionally adept at modeling sequential data and mitigating the vanishing gradient problem prevalent in traditional RNNs. By incorporating memory cells and gated mechanisms, LSTMs effectively capture temporal dependencies and long-term patterns, making them a powerful choice for tasks such as network anomaly detection, where sequential data analysis is paramount. Sahu *et al.* [12] proposed a hybrid LSTM and fully connected network with hyperparameter tuning to classify benign and malicious network traffic activities. This approach considered the imbalanced intrusion data distributions for the majority and minority classes. Additionally, six cybersecurity datasets were utilized to evaluate binary and multiclass intrusion detection scenarios. The authors of [13] proposed an effective feature selection method using XGBoost in combination with a hybrid CNN-LSTM for DDoS attack classification. The proposed model comprises three main components: data preprocessing, feature selection, and attack classification. The model demonstrated efficient classification of network traffic with a reduced subset of reliable dataset features and successfully identified various types of attacks including DNS, UDP, and SYN attacks. Authors in [14] proposed an attention-based CNN intrusion detection model. They showed that their approach is a good performer both from the point of view of classification accuracy and from the point of view of execution speed when compared to other models.

In [15], a hybrid IDS combining CNN and LSTM was developed. The proposed model can capture both spatial and temporal features of network traffic, improving intrusion detection performance for zero-day attacks. The authors in [16] proposed a novel method for network intrusion detection. Their approach combines Q-learning-based reinforcement learning with a deep feed-forward neural network. The Deep Q-Learning model introduced in their study offers continuous learning and detection of various network intrusions. It improves its detection capabilities through an automated trial-and-error process, adapting to evolving network environments.

C. RL-based IDS

Reinforcement Learning (RL) involves agents that sense their environment, execute actions, and receive feedback in the form of rewards. By updating their policies based on this feedback, RL agents aim to optimize their decision-making for maximum defense performance in terms of detection, threat analysis, and response. RL enables adaptive sequential decision-making in security systems, leading to fast, efficient, and automated defense capabilities. Saeed *et al.* [17] reviewed multiagent IDS architectures, including several approaches that utilized RL algorithms. The adaptation capability of RL methods can help IDS to respond effectively to changes in the environment, however, obtaining the optimal solution is challenging due to the convergence of multiagent systems. Additionally, the authors of [18] proposed a novel DRL-based network intrusion detection system that incorporates feature selection methods. They investigated optimal hyper-parameter values for training DRL agents and demonstrated the effectiveness of the proposed method on various routing systems and countermeasures, integrating with different network performances. The authors in [19] presented two main security services: (1) a DRL-based mechanism for network traffic inspection to achieve scalable and intensive network traffic visibility for rapid threat detection; and (2) an address shuffling-based moving target defense technique to proactively defend against threats.

III. Adaptive DRL-based IDS-IPS (DeepIDPS)

The incorporation of LSTM is especially pertinent to network traffic analysis. This is due to the temporal characteristics of the data, which give rise to sequential traffic patterns that LSTM-based models excel in comprehending and processing. LSTM's ability to retain prior data is instrumental in assessing the current network traffic. In contrast, CNN, a deep learning model, is engineered to extract information from data through a series of hidden layers. The four pivotal layers in a CNN comprise: 1)Convolution Layer: This layer employs multiple filters to execute convolution operations, 2)ReLU Layer: It scans the original data with multiple convolutions and ReLU layers to identify distinctive features, 3)Pooling Layer: This layer conducts downsampling operations to reduce the dimensionality of the feature map. It employs filters to pinpoint specific features or segments within the input, and 4)Fully Connected Layer: Here, all resulting 2-Dimensional arrays from the pooled feature maps are transformed into a single, continuous, linear vector.

The fusion of a CNN with an LSTM has the potential to significantly boost the accuracy of an intrusion detection system by adeptly extracting both spatial and temporal characteristics from raw data. In CNN, multiple convolution layers come into play, with the initial layer focusing on extracting basic features and the subsequent layers delving into more intricate features. The pooling layer plays a role in trimming feature dimensions to enhance computational efficiency, and the final fully connected layer takes charge of classification tasks.

To enhance the learning capabilities of the proposed model, we employ reinforcement learning, which empowers agents to grasp environmental behavior to optimize reward acquisition. Additionally, we harness deep reinforcement learning, merging the perceptual strengths of deep learning with the decision-making prowess of reinforcement learning. Intrusion detection systems are pivotal in fortifying defenses against potential attacks. In our pursuit of crafting an agile and effective algorithm to counter such threats, we introduce a hybrid model that melds feature reduction techniques with deep neural networks. Our model amalgamates the merits of CNNs and LSTM networks to capture both spatial and temporal aspects within network traffic. The architectural approach embraces a hybrid CNN-LSTM framework, with CNN layers focusing on spatial feature extraction from input data, and subsequent LSTM layers capturing temporal dependencies among these features over time.

Solely employing CNN-LSTM for intrusion detection may fall short of identifying the importance of various features. Because not all features hold equal significance in representing traffic data. The Attention mechanism (AM) can employ weighted operations on target data to accentuate the most influential characteristics, yielding effective model optimization. Intrusion detection benefits from AM primarily by eliminating redundant information and reducing computational load. Following the CNN-LSTM stage, an Attention Mechanism is employed to assess the significant features of packet vectors, emphasizing the salient features for detecting malicious traffic. These features, generated by the attention mechanism, are subsequently integrated into a fully connected layer for feature fusion, culminating in the extraction of key features that aptly characterize network traffic behavior. Finally, the fused features are entered into a classifier to derive the ultimate recognition outcomes.

In the presented framework, an RL-based agent perpetually navigates the evolving environment, actively shaping real-time security policies. These policies are implemented by the SDN controller and are executed on switches, facilitating adaptation to shifting security threats. The RL process revolves around two entities: the agent and the environment. The

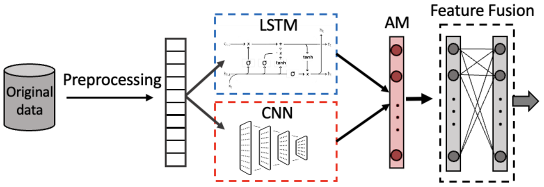


Fig. 2: Feature Fusion.

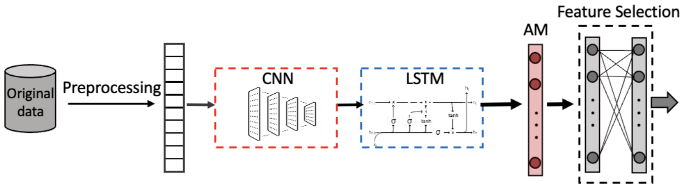


Fig. 3: Feature Selection.

environment supplies the agent with current-state information and the agent, informed by these data, formulates decisions. Subsequently, the environment assesses the agent's actions, and provides rewards or penalties accordingly. This iterative cycle fosters learning within the agent, refining its decision-making capabilities over time.

A. Model Review

The proposed architecture is composed of three distinct phases: Feature extraction, Anomaly detection, and Anomaly prevention. The feature extraction phase aims to select the most relevant and effective subsets of features from the original dataset based on specific criteria. This phase serves to eliminate redundant features, improve the classification performance of the algorithm, and reduce computational cost and time. To extract spatial features, a CNN is utilized, as it is particularly adept at this task. However, CNNs are not well suited for handling long-term time-series data.

To address this limitation, the high-dimensional features obtained from the CNN stage are then fed into the second stage, which consists of three layers: LSTM, fully connected, and output layers. The LSTM layer is employed to handle the temporal dependencies in the data, automatically constructing the state from the observations. This eliminates the need for human input and aligns with the goal of deep learning to minimize reliance on human interpretations of the problem. The fully connected layer and the output layer, employing a softmax activation function, contribute to the classification of input flow probabilities. When CNN for spatial feature extraction and LSTM for temporal feature analysis are combined, the proposed architecture leverages the strengths of both approaches, improving the overall performance of the anomaly detection system. There are three primary methods for combining a CNN and an LSTM:

1) Parallel Architecture: In this approach, CNN and LSTM operate independently on the input data. Then, their outputs are concatenated and passed to the fully connected layer for further processing. The feature fusion component combines the global and periodic features extracted by CNN and LSTM, as depicted in Fig. 2.
2) CNN to LSTM: the CNN output is used as the input to the LSTM. This enables the LSTM to learn additional features from the input data that have already been extracted by the CNN, as depicted in Fig. 3.
3) LSTM to CNN: This method involves using the output of the LSTM as the input to the CNN. By doing so, the CNN can learn features from the output of the LSTM, capturing higher-level representations.

B. RL Model

In RL model, at each state s, the agent chooses an action a, observes the reward r and the next state s'. The Q-value $\hat{Q}(s,a)$ is then updated using the equation:

$$\hat{Q}(s,a) \leftarrow (1-\alpha)\hat{Q}(s,a) + \alpha\left(r + \gamma \max\nolimits_{a'} \hat{Q}(s',a')\right), \tag{1}$$

where α is the learning rate and γ is the discount factor. This equation combines the current reward with the estimated maximum future reward to update the Q-value. To achieve convergence to the optimal policy, traditional Q-learning requires the agent to visit all states infinitely. DRL has emerged, integrating reinforcement learning with deep neural networks. DRL offers a solution by leveraging the power of deep learning to handle complex and high-dimensional state spaces, enabling more efficient and scalable learning in reinforcement learning settings. DRL holds great promise in addressing automated defense decision problems in dynamic environments with uncertain future information.

Our RL model consists of two modes: 1)Learning Model: the RL agent analyzes the current state of the network and determines the best action to take based on its learned policy. It uses reinforcement learning techniques to make informed decisions that maximize the expected rewards. 2)Detection Model: the RL agent updates the intrusion detection model based on the rewards received and the observed states of the network. It leverages the feedback from the environment to improve the accuracy and effectiveness of the intrusion detection system The first step in an RL problem is to define the state, actions, and reward. These definitions are crucial for the RL agent to learn an optimal policy. DRL agent captures its state from the network's overall state, denoted as $\mathcal{S} = (D, M)$. Here, D represents the detection state of the existing traffic in the network, and M represents the level of harm caused by malicious traffic. This combined state allows the agent to assess the network's current condition and make informed decisions to maintain network security. The actions considered in this approach are:

- a_1: BlockIP-30secs: Drop all incoming packets with the attacker's IP address for 30 seconds,
- a_2: LimitRate-25$\%$:Reduce the rate of incoming packets from attacker's IP address by 25$\%$,
- a_3: ReRoute: Redirect the attack traffic flows,
- a_4: DoNothing: No action.

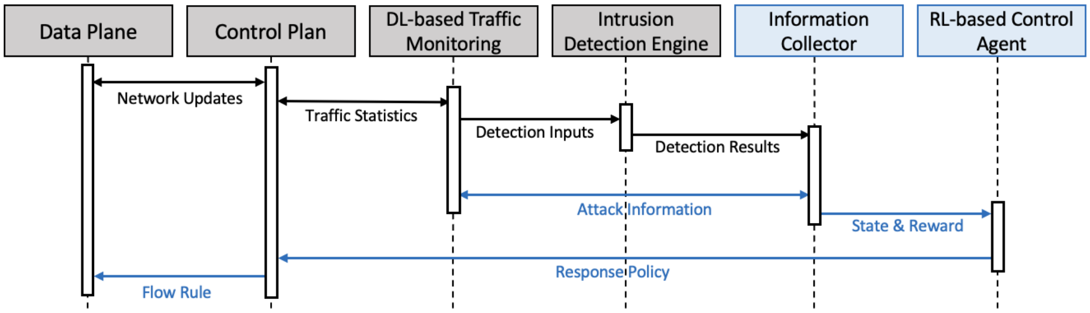


Fig. 4: Transition States.

TABLE I: Performance with different number of features

Performance Matrix	Number of Features				
	f=10	f=15	f=20	f=25	f=30
Accuracy (%)	97.70	98.24	98.60	98.64	97.81
Precision (%)	97.61	97.93	98.47	98.80	97.74
Recall (%)	97.56	97.86	98.41	98.77	98.71
F1-Score (%)	97.61	98.34	98.65	98.73	98.71
Loss	0.02	0.017	0.014	0.013	0.014
Time(ms)	10.3	12.3	18	23.9	26.5

TABLE II: CNN-LSTM vs Parallel CNN-LSTM.

Models	Accuracy(%)		Train Duration(s)	
	NSL-KDD	KDD	NSL-KDD	KDD
Feature Fusion	95.4	92.2	30.4	29.74
Feature Selection	97.42	95.26	35.26	32.2

Given the state of the system (s_t) and the actions in $A = \{a_1, a_2, a_3, a_4\}$, reward function r is defined as:

$$\mathcal{R}(s,a) = \alpha * D + \beta * U + \gamma * (1/T) + \omega * (1 - F) + \zeta * M, \quad (2)$$

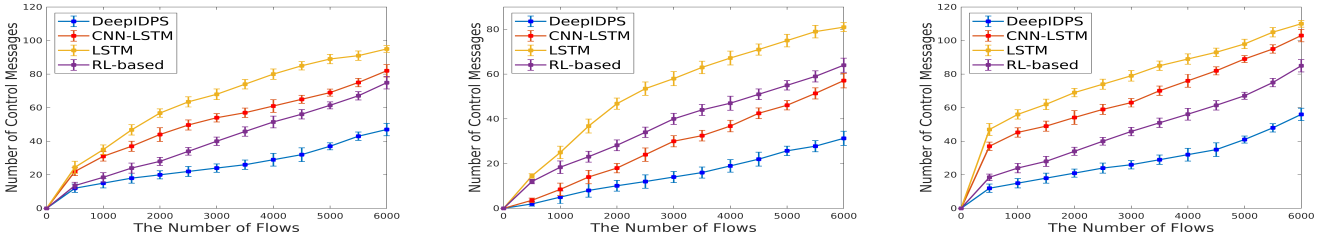
where D is the detection accuracy, ranging from 0 to 1, where 1 represents perfect accuracy. U is the resource utilization, ranging from 0 to 1, where 1 represents optimal utilization. T is the response time, measured in seconds. F is the false positive rate, ranging from 0 to 1, where 0 does not represent false positives. M is the attack mitigation, ranging from 0 to 1, where 1 represents complete mitigation. α, β, γ, ω, and ζ are the weights assigned to each component of the reward function, and they can be adjusted based on the specific requirements of the network anomaly detection system. Fig. 4 shows the timeline for message exchange between the various components of our framework in the detection period (black color) and response period (blue color).

IV. Experimental Evaluation

We implemented the model in a real SDN system and evaluated its performance using various metrics such as the total number of control messages, the capture-failure rate, accuracy, and false positive rate. We conducted tests on different datasets with varying sets of features and evaluated the model's performance against various types of attacks, including DDoS, Port Scanning, and Zero-day. To assess the effectiveness of our approach, we compared its results with those obtained using LSTM, CNN-LSTM, and RL-based approaches. The test dataset included real cyber-attacks targeted at an SDN environment. We utilized two different datasets: NSL-KDD [20] and KDD99. For the testing phase, we employed a real test bed to generate traffic and evaluate the performance of our model [8]. To generate malicious traffic, we utilized Kali Linux version 2.0, which is known for its security testing capabilities. On the other hand, we generated legitimate traffic using the Ostinato traffic generator. Ostinato provides the flexibility to generate both normal and burst-mode traffic. The initial feature set for our study consisted of the following essential features: Source IP address/port, Destination IP address/port, Duration, Source bytes/packets, Destination bytes/packets, Source TTL, Destination TTL, Source load, and Destination load.The action set that we considered in RL-model includes BlockIP-30secs, BlockIP-1min, BlockIP-3min, BlockIP-5min, LimitRate-25%, LimitRate-50%, LimitRate-75%, ReRoute, and DoNothing. The reward function is explained in Eq. (2).

Table I provides a comprehensive overview of the performance measurements obtained for DDoS detection using different numbers of selected features, ranging from 10 to 30. The table demonstrates the impact of the total number of features on DDoS detection performance. The ”Loss” represents the loss value over the training data after each epoch, which serves as a measure of how well the optimization process minimizes the training error. Lower loss values indicate better optimization and model performance. Table II compares the accuracy of different structures (Figs. 2 and 3) on different datasets. Results show that for both datasets, CNN-LSTM has better accuracy compared with parallel CNN-LSTM. Also, in the case of training during time, parallel CNN-LSTM needs more time to train data.

In Fig. 5, we depict the influence of incoming traffic scale on the *Total Number of Control Messages/overhead* for the controller. Through efficient feature selection for detection components and the effectiveness of reinforcement learning in action selection, the DeepIDPS consistently demonstrated superior performance across various attack types. Fig. 5(a) underscores the DeepIDPS's effectiveness in countering DDoS attacks, while Fig. 5(b) showcases its successful response to port scanning attacks. Furthermore, Fig. 5(c) highlights the DeepIDPS's remarkable performance in dealing with zero-day attacks. In summary, this study underscores the crucial role of powerful tools like the DeepIDPS in the detection and mitigation of cyberattacks, emphasizing the importance

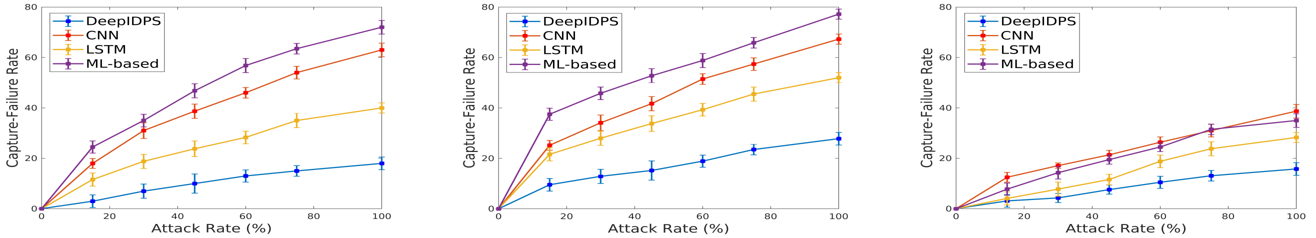


(a) DDoS Attack

(b) Port Scanning Attack

(c) Zero-day Attack

Fig. 5: Total number of flow rules and control messages.

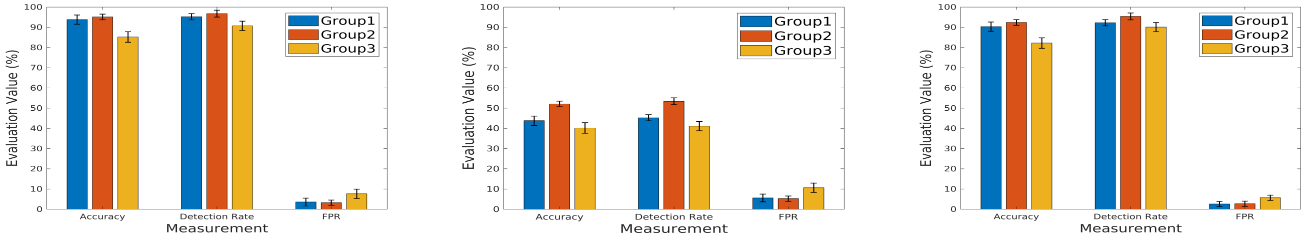


(a) DDoS Attack

(b) Port Scanning Attack

(c) Zero-day Attack

Fig. 6: Capture failure rate of malicious flows.



(a) DDoS Attack

(b) Port Scanning Attack

(c) Zero-day Attack

Fig. 7: Different feature sets.

of employing effective solutions for cybersecurity.

In Figs. 6, we evaluate the proposed method's performance by examining the *Capture-failure rate* in the context of different attack rates and attack types. The Capture-failure rate is a crucial metric that quantifies the probability of not capturing a packet from a malicious flow, calculated as the ratio of non-captured malicious packets to the total packets in the malicious flow. The results in the figures illustrate the effectiveness of the proposed approach in countering various attack types. Fig. 6(a) demonstrates its performance against DDoS attacks, while Fig. 6(b) showcases its effectiveness in dealing with port scanning attacks. Lastly, Fig. 6(c) displays its performance against zero-day attacks, exhibiting superior results compared to other attack types. The inclusion of the Capture-failure rate as a metric enables a more efficient and effective approach to detecting and mitigating a variety of cyberattacks, underscoring its significance in developing robust cybersecurity tools and strategies.

In Fig. 7, we assess our approach using different feature sets: *Group 1* containing all 41 features, *Group 2* with an initial set plus 20 selective features (33 features), and *Group 3* containing the initial 13 features. Our results demonstrate that the performance of our method is optimal with *Group 2*, which comprises a specific feature set highly effective in countering cyberattacks. Following *Group 2*, *Group 1* outperforms *Group 3*. The results indicate the significant impact of selecting the right feature sets on cybersecurity tools. Identifying the most suitable feature sets for distinct attack types enhances attack detection and response in a timely and efficient manner.

Table III compares the precision, recall, and F1-score of basic ML, CNN, LSTM, and CNN-LSTM models. The fusion of CNN and LSTM networks in the proposed model results in enhanced performance compared to other existing methods. The hybrid CNN-LSTM model attains a remarkable accuracy for intrusion detection, confirming its effectiveness and underscoring the superiority of deep learning models over conventional machine learning algorithms. Furthermore, the CNN-LSTM model surpasses alternative models in terms of F1-score for both classes, showcasing its robust predictive capabilities.

V. Conclusion

The combination of CNNs and LSTMs can be a powerful approach to anomaly detection in sequential data, allowing the detection of spatial and temporal anomalies. Applying

TABLE III: Precision, Recall, and F1-score of the different methods.

Models	**Precision**(%)		**Recall**(%)		**F1-score**(%)	
	Normal	**Attack**	**Normal**	**Attack**	**Normal**	**Attack**
ML	81.19	97.86	95.17	85.21	85.74	94.23
CNN	83.43	97.55	93.62	91.85	91.17	94.56
LSTM	85.43	97.31	93.12	93.85	89.18	95.15
CNN-LSTM	94.28	98.14	93.11	96.25	94.43	97.52

CNN to intrusion detection is an old concept that gives good accuracy, but we tried to combine LSTM with CNN to get more accuracy and make IDS more effective. Incorporating an attention mechanism into the CNN-LSTM model improves its capability to emphasize specific elements within the input sequence. This is especially valuable for tasks where certain elements of the sequence have varying levels of importance in achieving accurate predictions. The experimental findings indicate that this model will increase the precision of detection. The hybrid CNN-LSTM model has been observed to perform well compared to the rest of the DL and ML models.

References

[1] N. Z. Bawany, J. A. Shamsi, and K. Salah, "Ddos attack detection and mitigation using sdn: methods, practices, and solutions," *Arabian Journal for Science and Engineering*, vol. 42, pp. 425–441, 2017.

[2] E. Balkanli, A. N. Zincir-Heywood, and M. I. Heywood, "Feature selection for robust backscatter ddos detection," in *Proc. of the 40th IEEE Conf. on Local Computer Networks(LCN Workshops)*, 2015, pp. 611–618.

[3] J. Zhang, Y. Ling, X. Fu, X. Yang, G. Xiong, and R. Zhang, "Model of the intrusion detection system based on the integration of spatial-temporal features," *Computers & Security*, vol. 89, p. 101681, 2020.

[4] X. Han, R. Yin, Z. Lu, B. Jiang, Y. Liu, S. Liu, C. Wang, and N. Li, "Stidm: A spatial and temporal aware intrusion detection model," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2020, pp. 370–377.

[5] Y. Song, N. Luktarhan, Z. Shi, and H. Wu, "Tga: a novel network intrusion detection method based on tcn, bigru and attention mechanism," *Electronics*, vol. 12, no. 13, p. 2849, 2023.

[6] M. V. Valueva, N. Nagornov, P. A. Lyakhov, G. V. Valuev, and N. I. Chervyakov, "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation," *Mathematics and computers in simulation*, vol. 177, pp. 232–243, 2020.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] N. Niknami and J. Wu, "Enhancing load balancing by intrusion detection system chain on sdn data plane," in *Proc. of the IEEE Conf. on Communications and Network Security (CNS)*, 2022, pp. 264–272.

[9] Niknami, Nadia and Wu, Jie, "Entropy-kl-ml: Enhancing the entropy-kl-based anomaly detection on software-defined networks," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 6, pp. 4458–4467, 2022.

[10] D. Alghazzawi, O. Bamasag, H. Ullah, and M. Z. Asghar, "Efficient detection of ddos attacks using a hybrid deep learning model with improved feature selection," *Applied Sciences*, vol. 11, no. 24, p. 11634, 2021.

[11] Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu, and S. Camtepe, "Ae-mlp: A hybrid deep learning approach for ddos detection and classification," *IEEE Access*, vol. 9, pp. 146 810–146 821, 2021.

[12] S. K. Sahu, D. P. Mohapatra, J. K. Rout, K. S. Sahoo, Q.-V. Pham, and N.-N. Dao, "A lstm-fcnn based multi-class intrusion detection using scalable framework," *Computers and Electrical Engineering*, vol. 99, p. 107720, 2022.

[13] A. Zainudin, L. A. C. Ahakonye, R. Akter, D.-S. Kim, and J.-M. Lee, "An efficient hybrid-dnn for ddos detection and classification in software-defined iiot networks," *IEEE Internet of Things Journal*, 2022.

[14] Z. Wang and F. A. Ghaleb, "An attention-based convolutional neural network for intrusion detection model," *IEEE Access*, 2023.

[15] M. Abdallah, N. An Le Khac, H. Jahromi, and A. Delia Jurcut, "A hybrid cnn-lstm based approach for anomaly detection systems in sdns," in *Proc. of the 16th Intl. Conf. on Availability, Reliability and Security*, 2021, pp. 1–7.

[16] H. Alavizadeh, H. Alavizadeh, and J. Jang-Jaccard, "Deep q-learning based reinforcement learning approach for network intrusion detection," *Computers*, vol. 11, no. 3, p. 41, 2022.

[17] I. A. Saeed, A. Selamat, M. F. Rohani, O. Krejcar, and J. A. Chaudhry, "A systematic state-of-the-art analysis of multi-agent intrusion detection," *IEEE Access*, vol. 8, pp. 180 184–180 209, 2020.

[18] S. Bakhshad, V. Ponnusamy, R. Annur, M. Waqasyz, H. Alasmary, and S. Tux, "Deep reinforcement learning based intrusion detection system with feature selections method and optimal hyper-parameter in iot environment," in *Proc. of IEEE Intl. Conf. on Computer, Information and Telecommunication Systems (CITS)*, 2022, pp. 1–7.

[19] S. Kim, S. Yoon, J.-H. Cho, D. S. Kim, T. J. Moore, F. Free-Nelson, and H. Lim, "Divergence: deep reinforcement learning-based adaptive traffic inspection and moving target defense countermeasure framework," *IEEE Transactions on Network and Service Management*, 2022.

[20] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, "A hybrid deep learning model for efficient intrusion detection in big data environment," *Information Sciences*, vol. 513, pp. 386–396, 2020.

Nadia Niknami received her B.S. degree in Computer Science from University of Isfahan, Iran, in 2011, and MSc degrees From Tarbiat Modares University, Tehran, Iran in 2015. She is currently pursuing a Ph.D. degree in the Department of Computer and Information Sciences at Temple University, Philadelphia. Her current research focuses on cyber-security, attack-defense scenarios, Intrusion Detection Systems, and Neural networks.

Jie Wu is the Director of the Center for Networked Computing and Laura H. Carnell professor at Temple University. His current research interests include mobile computing and wireless networks, cloud computing, and network trust and security. Dr. Wu regularly published in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Service Computing, IEEE/ACM Transactions on Networking, and Journal of Computer Science and Technology. Dr. Wu is/was general chair/co-chair for IEEE DCOSS'09, IEEE ICDCS'13, ICPP'16, IEEE CNS'16, WiOpt'21, ICDCN'22, IEEE IPDPS'23, and ACM MobiHoc'23 as well as program chair/cochair for IEEE MASS'04, IEEE INFOCOM'11, CCF CNCC'13, and ICCCN'20.