Open Access ISSN: 3007-1437 (Online), ISSN: 3007-1429 (Print)

IRAQI Academic Scientific Journals

Alkadhim Journal for Computer Science (KJCS)

Journal Homepage: https://alkadhum-col.edu.iq/JKCEAS

KJCS ALKADHIM JOURNAL FOR COMPUTER SCIENCE

Optimized Hybrid CNN-LSTM Framework with Multi-Feature Analysis and SMOTE for Intrusion Detection in SDN

[1]Dalia Shihab Ahmed, [1]Abbas Abdulazeez Abdulhameed, [1]Methaq T. Gaata

[1] Computer Science Department, College of Science, Mustansiriyah University, Baghdad-Iraq

Article information

Article history:

Received: November, 4, 2025

Accepted: November, 22, 2025

Available online: December, 25, 2025

Keywords:

Software-Defined Networking,
Intrusion Detection System,
Deep Learning,
CNN,
LSTM,
Feature Selection,
SMOTE

Corresponding Author:

Dalia Shihab Ahmed

dalia_shihab@uomustansiriyah.edu.iq

DOI:

https://doi.org/10.61710/kjcs.v3i4.132

This article is licensed under:

Creative Commons Attribution 4.0 International License.

Abstract

There is a growing trend toward the use of software-defined networks (SDN), which presents new security challenges requiring advanced intrusion detection systems (IDS). This paper proposes a deep learning-based hybrid system combining convolutional neural networks (CNNs) and long-term short-term memory networks (LSTMs) that can be used for effective intrusion detection in SDN environments. The model uses CNNs to extract spatial features from network traffic data and LSTMs to learn temporal patterns, enabling the identification of complex attack patterns. We evaluate our model using an In SDN dataset and test its performance using various feature sets, ranging from 6 to 83 features in our model. Experimental results indicate that our model has a high multi-class classification accuracy of 99.63% when using all 83 features in Group 1. Furthermore, we utilize the Synthetic Minority Over-sampling Technique (SMOTE) to address the issue of class imbalance which considerably enhances the detection accuracy of minority attack classes which is reaching 99.76%. It is established

that the presented hybrid CNN-LSTM model is a powerful and effective solution for improving SDN security.

1. Introduction

Software-defined networking (SDN) has transformed networking management by decoupling the control plane from data plane and has made it more flexible and programmable. However, the centralized architecture is also a new source of vulnerabilities, attack points and it becomes an easy target of cyber-attacks such as DDoS, brute force, and web attacks. Traditional intrusion detection mechanisms struggle to keep up with advanced threat types in SDN environments. Consequently, this is why intelligent, adaptive and effective intrusion detection systems are urgently needed to be able to acquire the complex patterns of the network traffic[1][2].

Although conventional machine learning (ML) has been applied successfully to detecting static threats[3]. It fails to capture the complex spatiotemporal dependencies of contemporary cyber-attacks, particularly in the case of the specific characteristics of SDN networks. The methodologies of DL have shown a high potential in solving complex pattern recognition problems in the context of cyber-security applications [4]. The convolutional neural networks (CNNs) [5] are also good at extracting hierarchical spatial features from network traffic data, whereas the long short-term memory (LSTM) networks [6] are good at modeling the temporal dependencies required to understand attack sequences and behavioral anomalies. However, effective hybrid architecture design has to consider numerous architectural components and hyper parameters, such as layer designs, activation functions, regularization strategies, and optimization parameters [2]. The paper suggests a DL model which is a combination of CNNs to extract spatial features and LSTMs to learning temporal sequences. This proposed model is meant to extract both spatial and temporal features of network traffic flows to offer a more detailed analysis towards intrusion detection.

The choice of the CNN-LSTM hybrid architecture stems from its ability to take both spatial and temporal patterns inherent in network traffic data which is a crucial demand for efficient intrusion detection in software-defined network (SDN) environments. CNNs excel at extracting hierarchical spatial features from data streams, which identify positional patterns and structural correlations indicative of malicious activity. LSTMs on the other hand are particularly adept at modeling sequential dependencies and long-term temporal behaviors that are critical for recognizing multi-stage attacks and behavioral anomalies over time. While standalone models like CNNs, RNNs and traditional machine learning algorithms have shown effectiveness in specific scenarios but they often struggle to simultaneously learn spatial features and temporal dynamics. By combining CNN for spatial representation with LSTM for sequential learning, our hybrid model offers a comprehensive and adaptable framework capable of accurately detecting complex and sophisticated threats.

This work has several contributions. First, a hybrid CNN-LSTM network is constructed and trained strategically, utilizing dropout and batch normalization as stability and generalization tools. Second, we compare this model's performance on In SDN data across six feature sets, providing an idea of the balance between feature dimensions

and detection performance. Lastly, the critical class imbalance is handled by employing SMOTE [7] method which proves to be an important enhancement for detecting rare and severe attacks. The findings indicate that our framework is a highly valid and robust solution for securing SDN environments.

The rest of this paper is structured as follows: Section 2 provides a review of related work in SDN intrusion detection. Section 3 explains an In SDN dataset, pre-processing and the proposed model structure. Section 4 shows the performance evaluation and experimental results. Section 5 presents the findings and limitations of this work and Section 6 concludes the paper with future research.

2. Related Work

The field of SDN intrusion detection has seen substantial advancements with the application of several ML and DL methods, with recent studies showing increasingly sophisticated and effective approaches.

A comprehensive multi-layered security framework was presented that combined MAC address authentication with a dual-discriminator conditional generative adversarial network (DDcGAN). This system utilizes Four-Q curve cryptography for authentication, univariate ensemble feature selection for switch optimization and Sheep Flock Optimization Algorithm (SFOA) to develop DDcGAN performance. The framework obtained remarkable results involving accuracy of 98.29%, true positive rate of 99.04% and false alarm rate of 2.05% while showing 4.5% energy savings contrast to existing methods [8]. Similarly, a Graph Residual Attention Network (GRAN) was presented that integrates attention mechanisms and residual learning into graph neural networks for SDN intrusion detection and attained accuracy of 97.1% in multi-class attack classification [9]. Recent studies have increasingly adopted graph-based learning for SDN security. For example, Graph SAGE was employed within SDN framework to detect DoS attacks, showing increased accuracy and scalability by structural feature learning[10].

Notable contributions have been made in hybrid models and generative approaches. One study advanced DAERF which is a hybrid framework that combines a deep Autoencoder for feature learning with random forest classifier and obtains 98% accuracy on benchmark datasets while preserving low false positives and minimal controller overhead [11]. Another comprehensive comparison of diverse GAN architectures involving traditional GAN, DCGAN and WGAN-GP for anomaly detection in SDN environments showed that even simpler GAN models can effectively detect attacks with decreased latency [12].

For instance, a study evaluating various algorithms found that the Deep CNN model obtained a classification accuracy of 99.85% which is outstanding traditional ML methods [13]. Similarly, the LSTM-based system (SATIDS) demonstrated a high accuracy rate for both binary and multi-class classification [14]. However, challenges such as data imbalance and over fitting in distributed non-IID environments remain as shown in studies that integrate federated learning (FL) with Generative Adversarial Networks (GANs)[15].

Traditional ML approaches have also demonstrated competitive performance, especially when integrated with feature selection. A main study revealed that Decision Tree (DT) models could achieve an accuracy of 99.8% with small number of features from the original features which significantly reduces computational load [16]. Furthermore, the combination of optimized feature selection like modified Grey Wolf Optimizer (GWO) with a Light GBM classifier has achieved accuracies reached to 99.8% [17]. Ensemble methods and hierarchical architectures like hierarchical multiclass (HMC) with SMOTE technique have been effective in handling class imbalance which improves the detection of rare attacks like U2R and Botnet [18]. The value of feature engineering is further highlighted by lightweight models that use a small number of well-designed features for real-time DDoS detection and achieving accuracies of 97-99.4% [19][20].

Hybrid and specialized models show great promise in balancing performance and efficiency. LSTM-Autoencoder have proven powerful in anomaly detection with lower computational costs which led to makes them proper for resource-constrained environments [21]. Integrating CNNs with ML classifiers has also been successful, one study obtained high accuracy while introducing novel regularization technique (SD-Reg) to prevent and avoid over fitting [22]. Despite these improvements is still a notable limitation of many models is poor generalization to unseen data sources which highlights the need for robust and cross-domain solutions [23].

Recurrent architectures like RNNs, LSTMs and GRUs are inherently well-suited for detecting temporal patterns in network attacks [24]. Based on this, hybrid CNN-LSTM models have also been prepared to learn spatial and temporal features concurrently. However, one of those models obtained accuracy of 96.32% [25]. It has also reported a false-positive rate of 6% which refers to improvements in model design and regularization. Other methods like ensemble methods [26] unsupervised learning with LSTM Autoencoder and one-class SVM [27] present various ways to attain successful intrusion detection under accuracy, resource and data availability constraints.

While recent approaches like transformers, GNNs, Autoencoder and GANs give advantages for intrusion detection but they face challenges like high computational costs, dependency on network topology or need for balanced data. In contrast, our CNN-LSTM hybrid effectively combines spatial and temporal learning with greater computational efficiency, making it adaptable for both high-performance and resource-limited SDN deployments.

Despite these improvements, various critical constraints persist in existing SDN-focused IDS models. First, many approaches rely on either spatial or temporal modeling in isolation and employ CNNs for feature extraction or LSTMs for sequence analysis but fail to combine both capabilities into a cohesive architecture. This restricts their capability to detect complex and multi-stage attacks that display both spatial correlations and temporal dependencies [13, 14, 21]. Second, while feature selection is widely specified as important, most studies assess models on fixed or narrowly defined feature sets without systematically exploring the trade-off between feature richness and computational efficiency over a range of feature dimensions [16, 17, 19]. Third,

class imbalance remains a pervasive issue, although techniques like SMOTE have been used, many models still struggle to detect rare but critical attack classes like U2R and Web Attacks, resulting in poor recall for minority categories [18, 22, 27]. Fourth, generalization remains a concern with many models over fitting to specific datasets or failing to execute consistently in non-IID or developing network environments [15, 23]. Finally, even hybrid CNN-LSTM models reported in the literature often exhibit relatively high false-positive rates (e.g., 6%) which refers to the need for improved regularization and architectural optimization [25].

To handle these gaps, this paper suggests an optimized hybrid CNN-LSTM framework to overcome the aforementioned limitations. Our model combines CNN and LSTM layers in a complementary architecture that simultaneously captures spatial patterns and temporal sequences, allowing more robust detection of sophisticated attacks. We systematically assess the model across six feature sets ranging from 6 to 83 features which gives empirical insights into the balance between detection performance and computational efficiency, allowing flexible deployment in both resource-constrained and full-feature scenarios. To tackle class imbalance, we explicitly apply the Synthetic Minority Over-sampling Technique (SMOTE) to the training data which notably enhancing the detection of minority attack classes without compromising overall accuracy. Furthermore, we incorporate batch normalization, dropout and stratified sampling to improve generalization and mitigate over fitting. Experimental results show that our framework obtains an accuracy of 99.63% with a false-positive rate of 0.05% in the full-feature configuration, outperforming existing hybrid models across both overall and per-class metrics particularly for rare attack types.

3. Adopted Approach

This section depicts the overall methodology utilized in our proposed model. Figure 1 details the framework's overall workflow for guiding the reader via the systematic process from raw data input to the final intrusion detection mechanism.

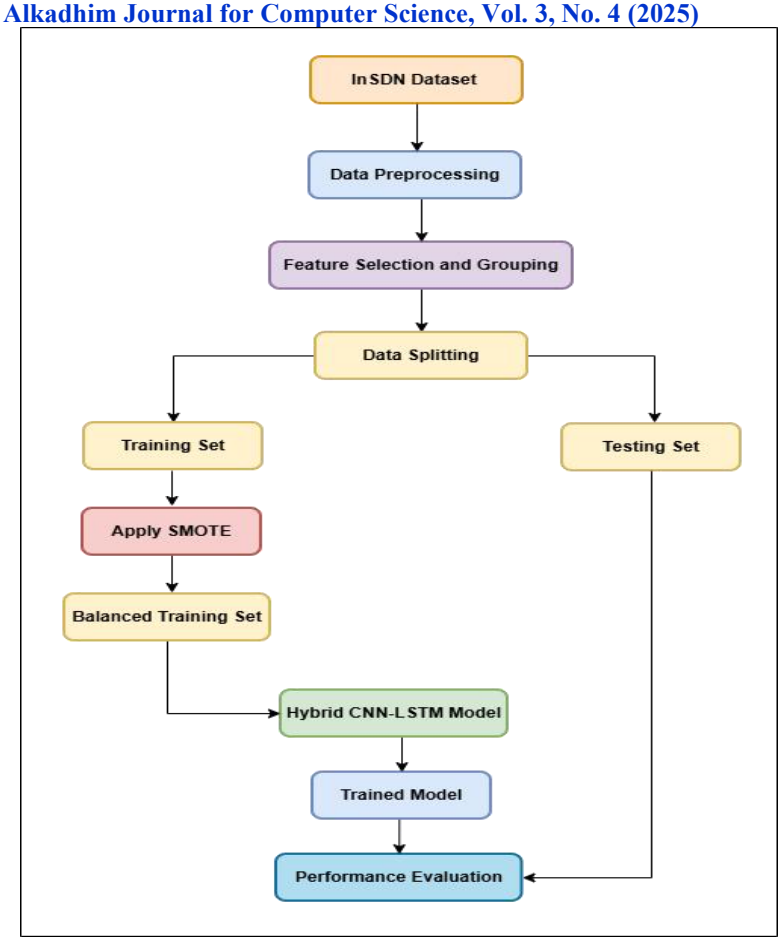


Figure (1): The Overall Framework of the Proposed Intrusion Detection System.

3.1 Dataset

We utilize the In SDN dataset [28], a comprehensive cyber-security dataset specifically prepared for SDN networks. This dataset handles constraints of traditional networking datasets by incorporating attack vectors with characteristics of SDN architectures. Utilizing datasets prepared for traditional networks rather than SDN networks can lead to compatibility issues and inaccurate performance like some attacks behave differently in SDN environments. The In SDN dataset involves a variety of attacks related to SDN networks like DDoS, Probe, DoS, Brute force, web attack, botnet and U2R attacks, in addition to normal traffic. This dataset is available in PCAP and CSV formats and involves 343,889 samples with 84 features generated utilizing the CIC Flow Meter tool. The distribution of data instances is detailed in Table 1.

Table (1): Details of The Data Instance of In SDN.

Class label		Samples
Normal		68,424
Attacks	Attacks	121,942
Probe		98,129
DoS		53,616
Brute force		1,405
Web-attack		192
Botnet		164
U2R		17
Total		343,889

3.2 Data preparation

Preprocessing the data is crucial to ensuring optimal model performance. The process starts by integrating all CSV files into a single data frame. Data cleaning involved systematically removing null, infinite values, and duplicate records to keep data integrity.

Next, feature engineering was used, and labels were converted to a numerical format utilizing one-hot encoding. The features were standardized using StandardScaler to ensure measurement consistency through all input dimensions based on the following formula:

$$z = ((x - \mu))/\sigma \quad (1)$$

Where z is normalized value, x is original value, μ is mean and σ is standard deviation. This step handles the large variations in feature ranges.

Finally, the dataset was divided utilizing stratified sampling to preserve a consistent distribution of categories across the training set (70%), validation set (10%), and testing set (20%). Table 2 shows the detailed distribution of the samples after preprocessing.

Table (2): Distribution of Samples Across Dataset Splits.

Class	Training Samples	Testing Samples
DDoS	85,359	36583
Prob	68,690	29439
Normal	47,897	20527
DoS	37,531	16085
BFA	984	421
Web-Attack	134	58
Botnet	115	49

U2R	12	5

3.3 Proposed Model

We propose DL-based hybrid architecture combining CNNs and LSTM systems to improve performance of NIDS. The block diagram of the proposed hybrid architecture is depicted in Figure 2 which is designed to classify sequential network traffic data by leveraging complementary strengths of CNNs and LSTM systems.

The architecture starts with input layer that receives a pre-processed 1D-feature vector and is then followed by two subsequent blocks of 1D-CNN layers for spatial feature extraction. The first block has two convoluted layers with 32 and 64 filters, respectively, while the second block has layers with 128 and 256 filters. Each layer utilizes kernel size of 3, stride value of 1, 'same' padding and Re LU activation function. A Max Pooling layer is used after each CNN block to decrease spatial dimensions, maintain critical information and minimize computational complexity.

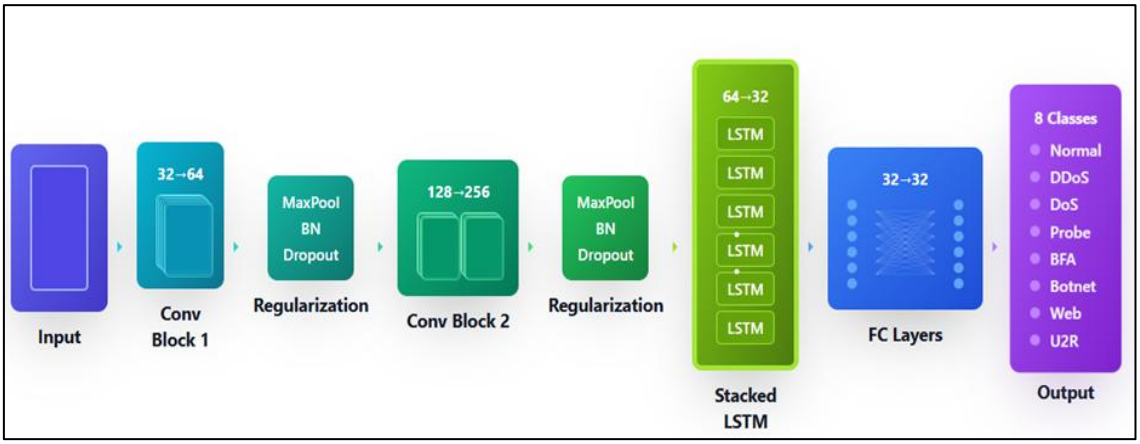


Figure (2): Block Diagram of the Proposed Hybrid CNN-LSTM Architecture.

We employ batch normalization and dropout to improve model training and generalization. The training is stabilized by utilizing batch normalization after each CNN block to avoid gradient problems. Dropout layers randomly deactivate neurons and reduce risk of over-fitting.

The feature maps from the convolutional base structure are then fed into a sequential modeling unit containing two LSTM layers. The first LSTM layer contains 128 units and returns the complete sequence for the next layer. Its output is processed by batch normalization and a dropout layer (rate = 0.3). A second LSTM layer, containing 32 units, produces a compressed context vector, which is also applied and normalized using dropout.

The model then passes through a fully connected (dense) 32-unit layer with Re LU activation, followed by further batch normalization and a final dropout layer (rate = 0.2). The network terminates with an output layer that uses the Softmax activation function for multiclass classification across the eight attack classes.

This model was constructed with the Adam optimizer and a learning rate of 0.001. Multi-class classification was performed with categorical cross-entropy as the loss function. The process of training was carried out using a batch size of 128 and 100 epochs. Dropout layers with rates of 0.3 and 0.2 were applied after the first LSTM layer and before the output layer, respectively, to mitigate over-fitting. Table 3 summarizes the hyper parameter settings.

Table (3): Parameter Settings of The Proposed Model.

Parameter	Value
Batch Size	128
Epoch Size	100
Optimizer	Adam
Learning Rate	0.001
Dropout Rate	0.2, 0.3
Loss Function	categorical cross-entropy

4 Evaluation Results

This section introduces a comprehensive evaluation of the proposed hybrid CNN-LSTM model's performance.

4.1 Experimental Setup

The proposed model was designed and executed using the Python programming language, where the Keras API and Tensor Flow backend library were utilized for all proposed approaches. All experiments were performed on an NVIDIA GeForce RTX 3080 GPU.

The main libraries involved Pandas and NumPy for handling data, Sickie-learn for data preprocessing like Standard Scalar for feature standardization, stratified data splitting and Imbalanced-learn for applying the SMOTE technique only to the training data.

4.2 Evaluation Criteria

The hybrid CNN-LSTM model was estimated by use of conventional classification metrics[29]:

Accuracy (AC): The proportion of accurate predictions over the total traffic.

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Precision (P): Fraction of the attacks correctly facilitated overall the instances of attacks (false alarm rate).

$$P = \frac{TP}{TP + FP} \quad (3)$$

Recall (R): The proportion of actual attacks that were identified properly.

$$R = \frac{TP}{TP + FN} \quad (4)$$

F1-Score (F1): The harmonic mean of Precision and Recall which gives a single balanced metric.

$$F1 = \frac{2 \times Pre \times Rec}{Pre + Rec} \quad (5)$$

These metrics are computed on the grounds of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN).

4.3 Experimental Results

The experimental evaluation of our proposed model shows critical insights into feature selection, architectural interaction and detection performance across various attack categories. There were six different groups of features tested systematically and represent various approaches to selecting features as follows:

Group 1: Contains all feature sets of the In SDN dataset which consists of 83 features obtained on each flow [28].

Group 2: Comprised of 48 sets of features [28] in the In SDN dataset.

Group 3: Comprised of the 18 group of features [30] in the In SDN dataset.

Group 4: There are 14 feature sets [31] in the In SDN dataset.

Group 5: Consists of 9 feature sets [17] in the In SDN dataset.

Group 6: It has 6 feature sets [32] in the In SDN dataset.

The overall multi-classification measures obtained in Table 4 indicate that Group 1 which implemented all 83 features, performed optimally with an accuracy of 99.63%, a precision of 99.58%, and an F1-score of 99.61%. The complete feature representation in Group 1 allowed the model to acquire complex patterns critical to the efficient detection of intrusions. In particular, Group 6, which had only 6 features, still scored highly at 98.78%, indicating that this group can be deployed efficiently in a resource-constrained environment.

Table (4): The overall Multi-classification Metrics.

Metric	TPR	TNR	FPR	FNR	Accuracy	Precision	F1 -Score
Group 1	99.63%	99.95%	0.05%	0.37%	99.63%	99.58%	99.61%
Group 2	99.06%	99.87%	0.13%	0.94%	99.06%	99.04%	99.04%
Group 3	98.44%	99.78%	0.22%	1.56%	98.44%	98.42%	98.41%
Group 4	93.51%	99.07%	0.93%	6.49%	93.51%	93.68%	93.38%
Group 5	98.15%	99.74%	0.26%	1.85%	98.15%	98.14%	98.12%
Group 6	98.78%	99.83%	0.17%	1.22%	98.78%	98.72%	98.75%

Table 5 shows per-class performance, where it can be observed that there are notable differences in the various types of attacks. majority classes such as DDoS and Normal traffic have good accuracy rate of more than 99% over all feature subset groups, whereas minority classes such as U2R and Web-Attack always have notably

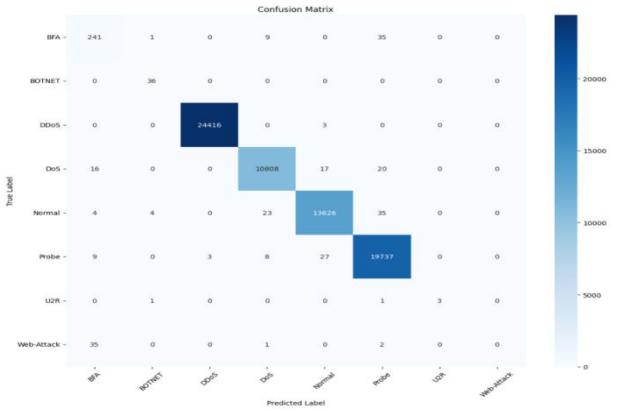
lower detection effectiveness in terms of precision and recall.

Table (5): Detailed Per-Class Classification Performance.

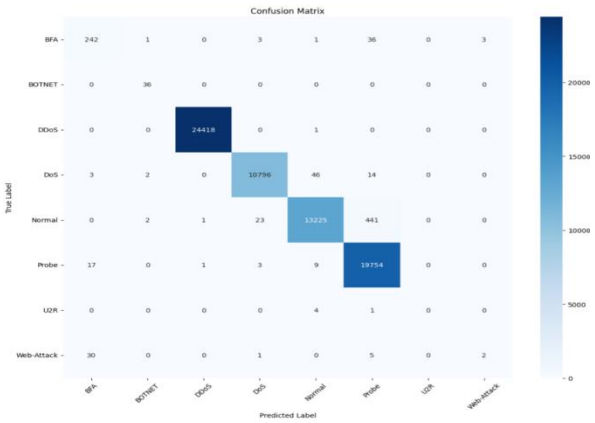
Metric	BFA	BOTNET	DDoS	DoS	Normal	Probe	U2R	Web-Attack
Accuracy								
Group 1	99.84%	99.99%	99.99%	99.86%	99.84%	99.80%	100.00%	99.95%
Group 2	99.86%	99.99%	100.00%	99.86%	99.24%	99.24%	99.99%	99.94%
Group 3	99.83%	99.99%	99.98%	99.31%	98.64%	99.18%	99.99%	99.94%
Group 4	99.74%	99.98%	95.84%	98.50%	98.24%	94.79%	99.99%	99.95%
Group 5	99.84%	99.98%	99.96%	99.03%	98.39%	99.15%	99.99%	99.96%
Group 6	99.89%	100.00%	99.93%	99.01%	99.36%	99.44%	99.99%	99.95%
Precision								
Group 1	79.02%	85.71%	99.99%	99.62%	99.66%	99.53%	100.00%	0.00%
Group 2	82.88%	87.80%	99.99%	99.72%	99.54%	97.55%	0.00%	40.00%
Group 3	91.71%	85.71%	99.97%	98.28%	97.55%	97.42%	0.00%	46.15%
Group 4	95.61%	69.23%	89.73%	97.25%	96.32%	94.99%	0.00%	0.00%
Group 5	96.74%	77.27%	99.90%	97.10%	97.07%	97.39%	0.00%	85.71%
Group 6	94.04%	94.74%	99.99%	97.03%	98.25%	98.69%	0.00%	0.00%
Recall								
Group 1	84.27%	100.00%	99.99%	99.51%	99.52%	99.76%	60.00%	0.00%
Group 2	84.62%	100.00%	100.00%	99.40%	96.59%	99.85%	0.00%	5.26%
Group 3	65.73%	100.00%	99.98%	97.34%	95.51%	99.79%	0.00%	31.58%
Group 4	38.11%	100.00%	99.62%	93.09%	94.72%	86.34%	0.00%	0.00%
Group 5	62.24%	94.44%	99.98%	96.74%	94.73%	99.72%	0.00%	31.58%
Group 6	77.27%	100.00%	99.82%	96.65%	98.55%	99.36%	0.00%	0.00%
F1-score								
Group 1	81.56%	92.31%	99.99%	99.57%	99.59%	99.65%	75.00%	0.00%
Group 2	83.74%	93.51%	99.99%	99.56%	98.04%	98.68%	0.00%	9.30%
Group 3	76.58%	92.31%	99.98%	97.81%	96.52%	98.59%	0.00%	37.50%
Group 4	54.50%	81.82%	94.42%	95.13%	95.51%	90.46%	0.00%	0.00%
Group 5	75.74%	85.00%	99.94%	96.92%	95.88%	98.54%	0.00%	46.15%
Group 6	84.84%	97.30%	99.91%	96.84%	98.40%	99.03%	0.00%	0.00%

These findings are graphically confirmed in the confusion matrices in Figure 3 which clearly show strong diagonal concentrations on most of the classes and lowly predicted minority classes in all groups of feature set. Similarly, the receiver operating characteristic (ROC) curves in Figure 4 indicate the ability of the model to be discriminative with the curves of the majority classes moving towards an optimal top-left corner, but the minority classes have reduced reparability. The Area Under the Curve (AUC) for each class exceeded 99% for majority classes and showed significant improvement for minority classes after SMOTE application, with U2R

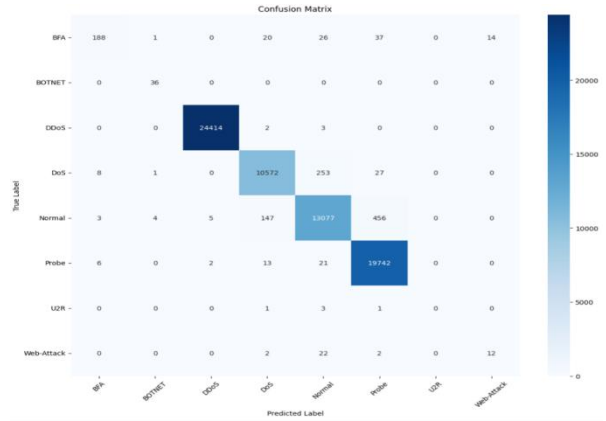
and Web-Attack AUC rising from ~70% to 95%.



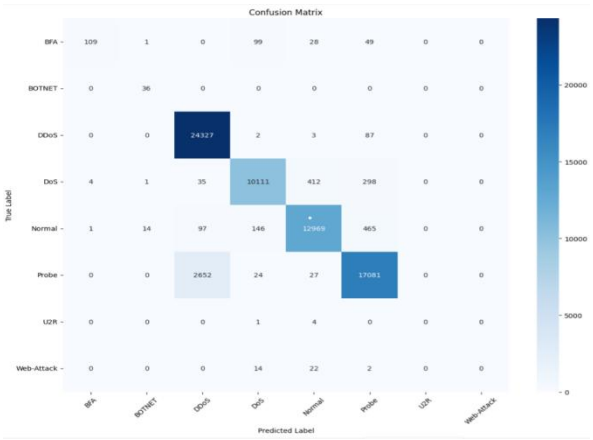
(a): Group1



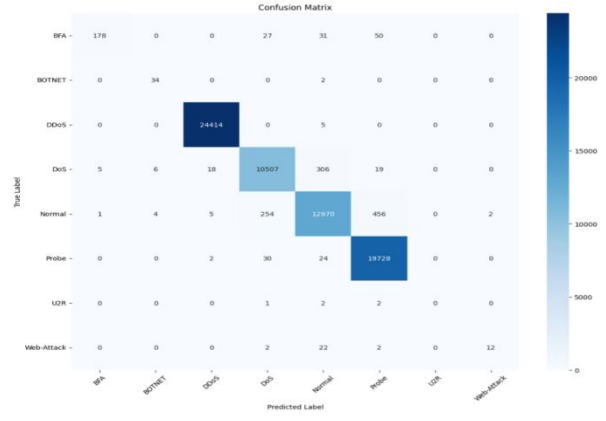
(b): Group2



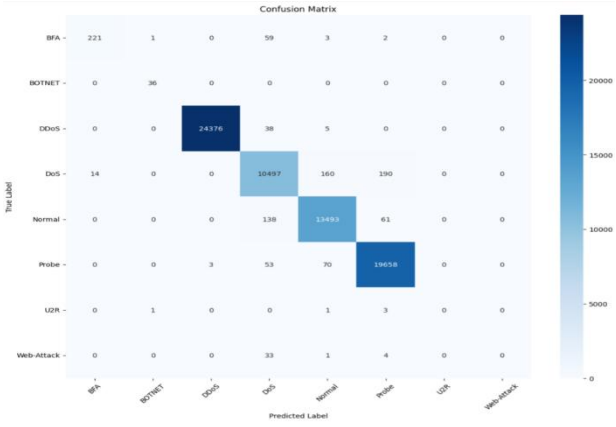
(c): Group3



(d): Group4



(e): Group5



(f): Group6

Figure (3): Confusion Matrix for each Group.

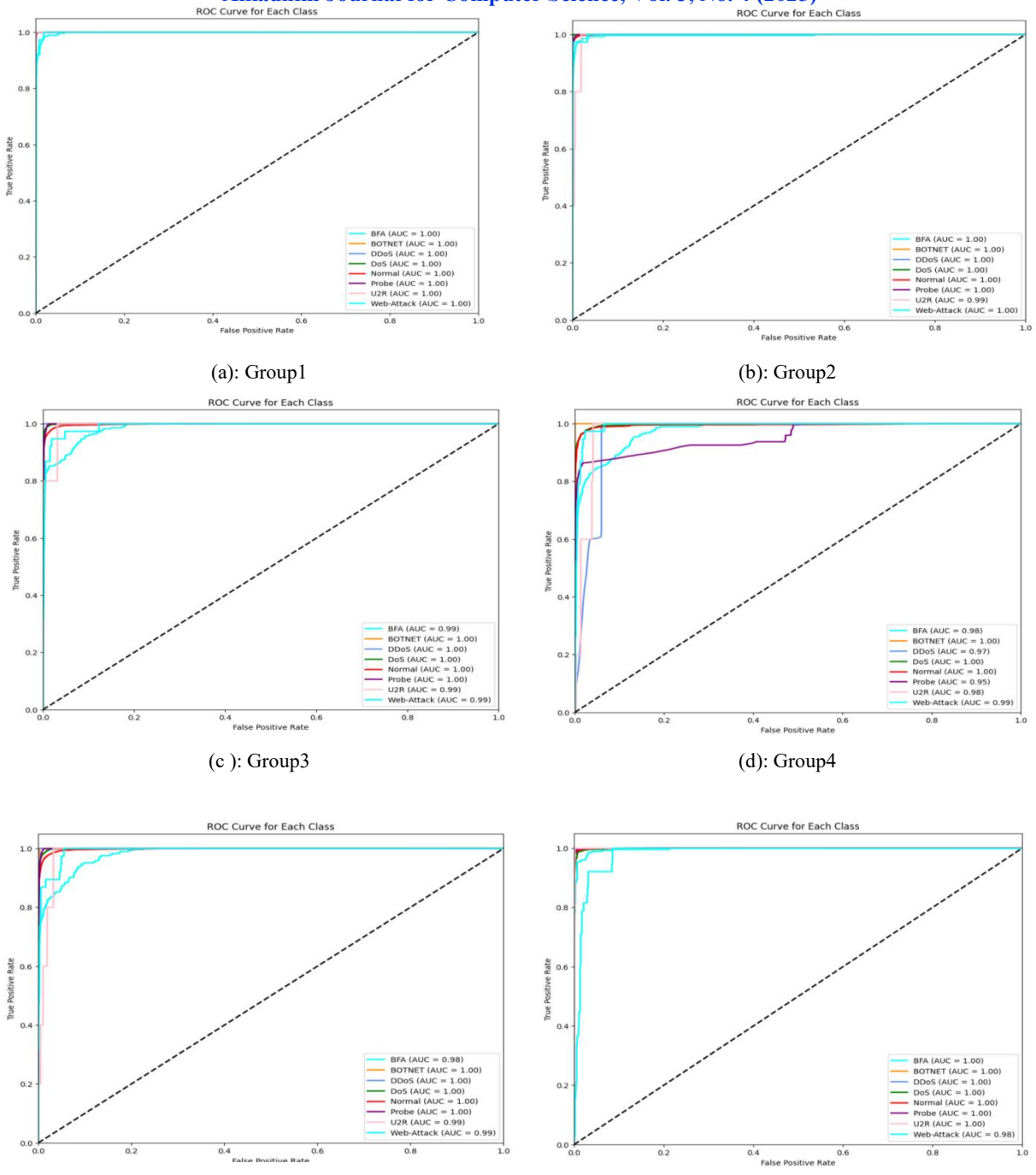
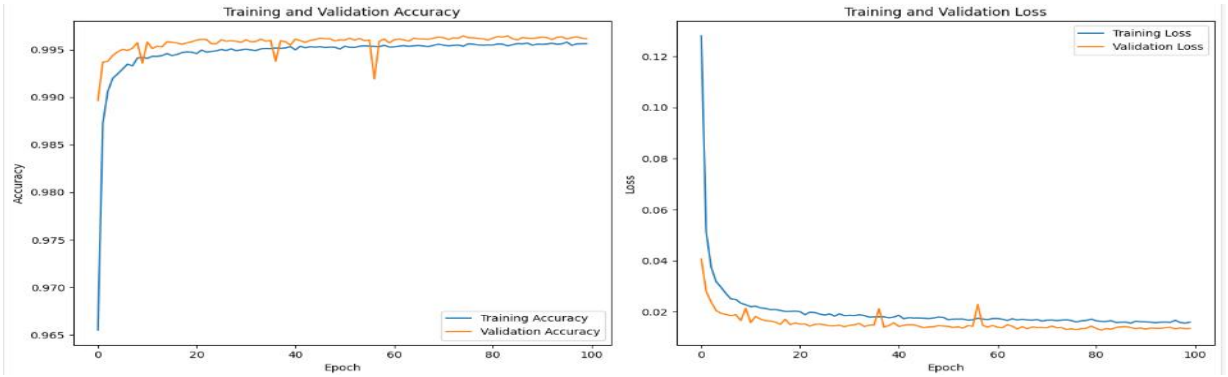
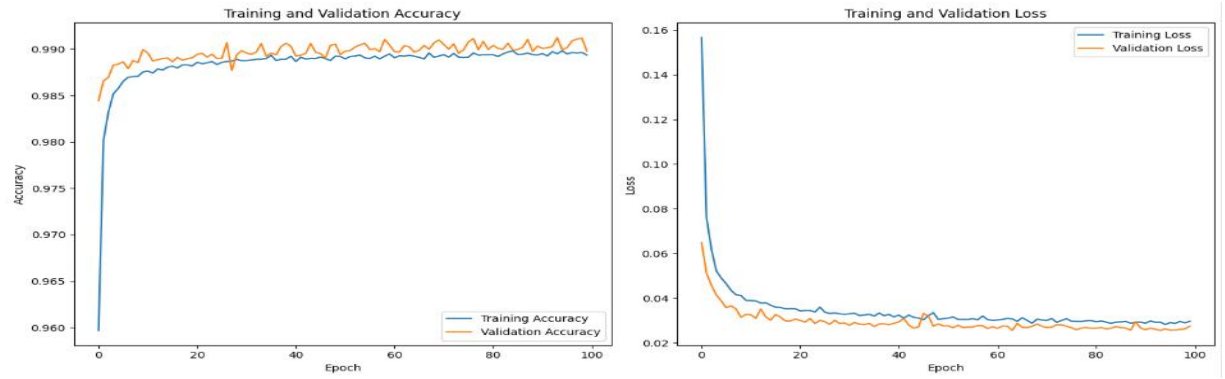


Figure (4): ROC for each Class of each Group.

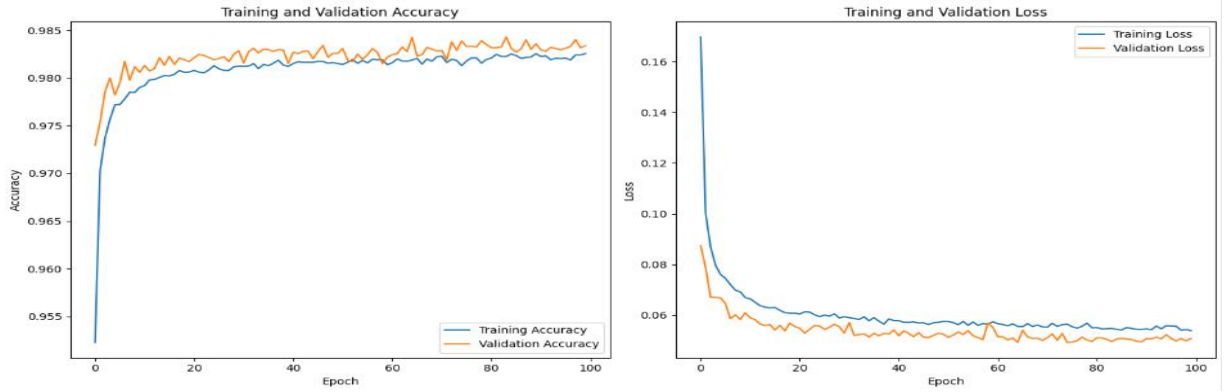
Figure 5 illustrates the training dynamics under which all feature set groups converge steadily with consistent performance in accuracy and loss metrics. The fact that there is a minimal gap between training and validation curves indicates that it generalizes effectively and eliminates any chance of over-fitting, thus justifying the architectural design choices, especially the dropout and batch normalization.



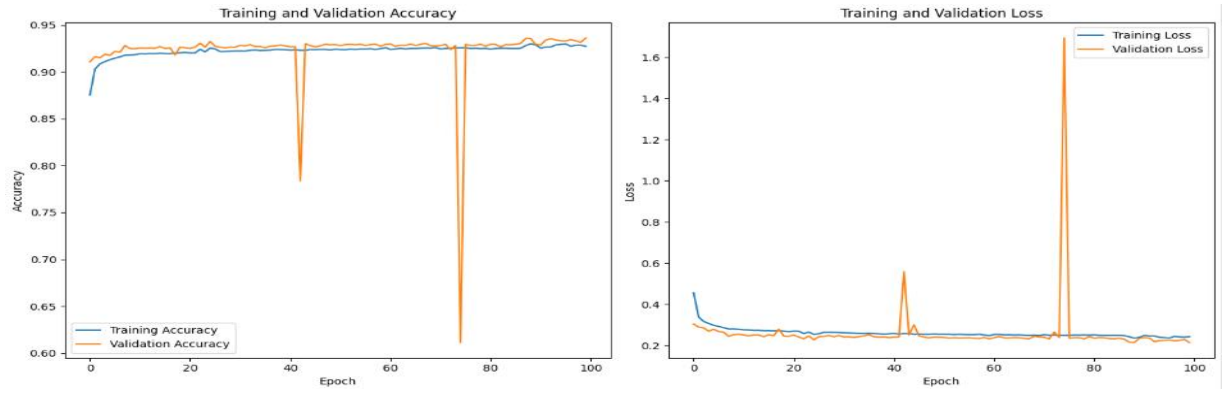
(a): Group 1



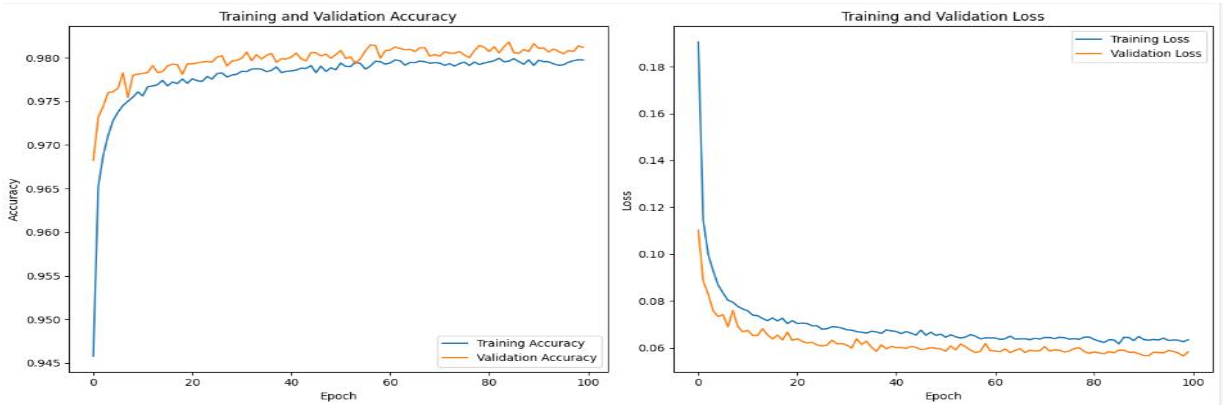
(b): Group 2



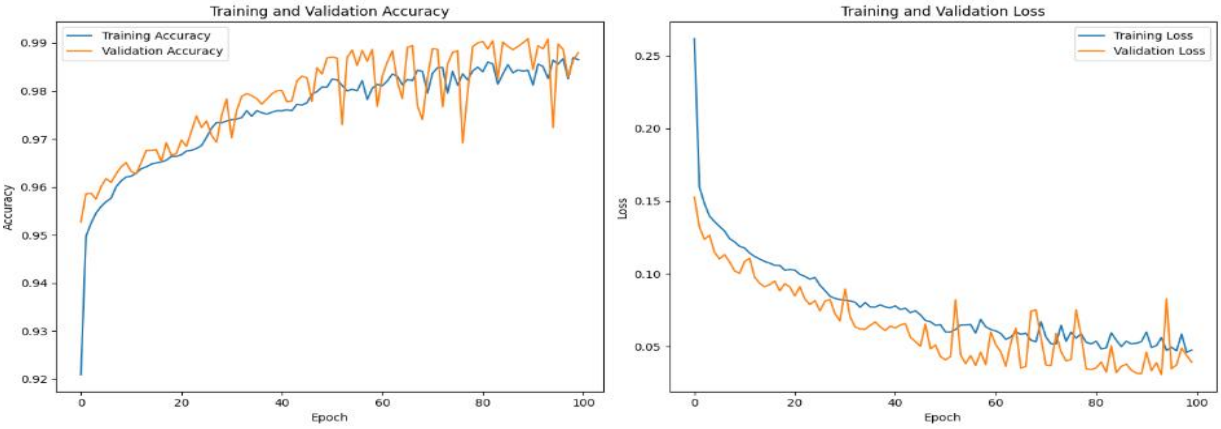
(c): Group 3



(d): Group 4



(e): Group 5



(f): Group 6

(a) Training and Validation Accuracy. (b) Training and Validation Loss Curve.

Figure (5): Training, Validation and Loss of each Group.

To mitigate the high-class imbalance problem that we have observed during the initial results, we used the SMOTE technique on Group 1. As the results summarized in Tables 6 and 7 show, the utility of the minority class increased significantly while maintaining excellent overall performance results.

Table (6): Detailed SMOTE Per-Class Classification Performance on Group 1.

Metric	BFA	BOTNET	DDoS	DoS	Normal	Probe	U2R
Accuracy	99.99%	100%	99.99%	99.90%	99.83%	99.85%	100%
Precision	98.28%	100%	99.99%	99.68%	99.71%	99.76%	71.43%
Recall	99.65%	100%	99.99%	99.71%	99.43%	99.72%	100%
F1 Score	98.96%	100%	99.99%	99.69%	99.57%	99.74%	83.33%

Table (7): The overall SMOTE Multi-classification Metrics on Group 1.

Metric	TPR	TNR	FPR	FNR	Accuracy	Precision	F1-Score
SMOTE	99.76%	99.97%	0.03%	0.24%	99.76%	99.78%	99.76%

The visual evidence of the performance enhancement resulting from the introduction of SMOTE is presented in Figure 6, which shows the confusion matrix of the balanced dataset, and in Figure 7, which illustrates the ROC curves for each class in the balanced dataset. The confusion matrix shows increased accuracy across all attack categories, with the most significant improvement in the minority classes which had been problematic to classify earlier. The ROC curves indicate improved class discrimination, as they now exhibit significantly stronger discriminatory power.

Based on Figure 8, it can be observed that the training dynamics of the balanced dataset have steady convergence patterns which confirms the idea that the model has already learned the information of the synthetic samples, and the model has retained its generalization capacity.

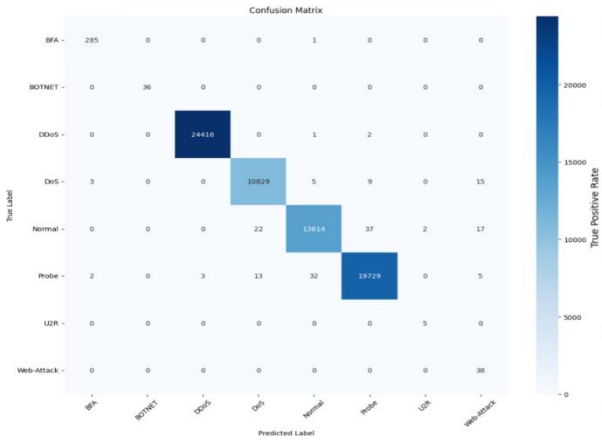


Figure (6): Confusion Matrix for Balance Dataset.

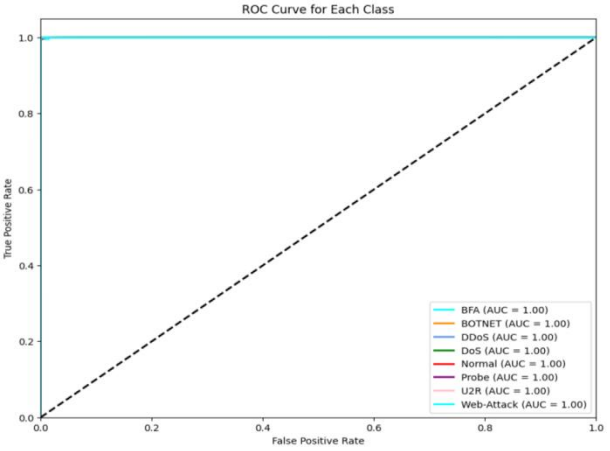
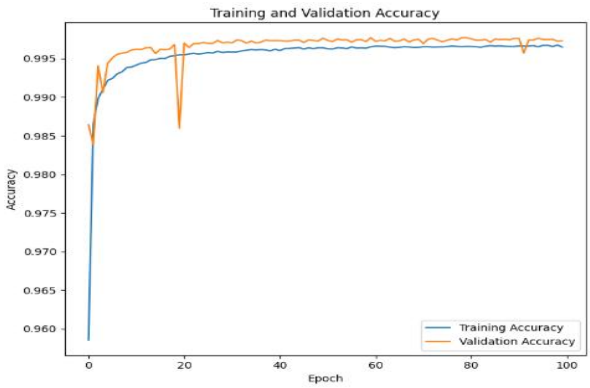
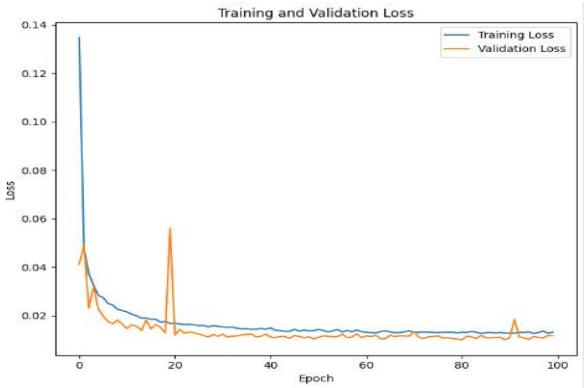


Figure (7): ROC for each Class of Balance Dataset.





(a) Training and Validation Accuracy. (b) Training and Validation Loss Curve.

Figure (8): Training, Validation and Loss of Balance Dataset.

4. Discussion

The comprehensive experimental evaluation provides valuable insights into the adequate performance of the proposed hybrid CNN-LSTM design for SDN intrusion detection. The results, presented in Table 4, confirm that the model achieves excellent performance across a wide range of feature combinations, with Group 1 achieving the highest accuracy of 99.63%. This best performance underlines the importance of comprehensive feature representation for effective spatial and temporal pattern learning in network intrusion detection.

Analysis of feature selection reveals notable performance of model. Although Group 1 yields best performance, the high performance of Group 6 with six features only and accuracy of 98.78% which shows that well-selected feature subsets can maintain high detection rates. This result is especially valuable in real-time SDN implementations where constraints on resources are of special concern. Nonetheless, the considerable decrease in the performance of Group 4 which received accuracy of 93.51% highlights the fact that the quality and relevance of the chosen features are more important than their number.

Critical observations we made from analyzing the performance of each category in Table 5 involve notable impact of category imbalances on detection capability. Initial results show significant challenges in detecting minority attack categories like Web attack and U2R attacks, exhibiting recall rates of 0% and 60% respectively, in Group 1. This limitation highlights fundamental challenge in network security systems where rare but potentially damaging attacks may go undetected due to imbalances in training data.

The SMOTE application shows notable improvements in handling class imbalances as shown in Tables 6 and 7. The balanced dataset achieves an overall accuracy of 99.76% with notable improvements in minority class detection. Most notably detection of web attacks and U2R shows tremendous improvements and attains recall of 100% with F1-scores of 67.26% and 83.33%, respectively. This notable improvement confirms that hybrid architecture has the inherent capability to learn complex patterns of rare attacks when provided with balanced training data. The application of SMOTE notably improved detection of minority attack classes with U2R recall increasing from 60% to 100% and Web Attack from 0% to 100%, showing critical role of data balancing in practical IDS deployment.

The training dynamics noticed through the experiments especially in Figures 5 and 8, show consistent and stable convergence patterns across all feature set groups. The minimal difference between training and validation metrics with the smoothness of the loss curves which indicates effective generalization and absence of overfitting. This training stability which was achieved through the careful implementation of the batch normalization and dropout layers which demonstrates the robustness of the architectural design.

The visual evidence presented in Figures 3 and 6 for confusion matrices and Figures 4 and 7 for ROC curves confirms the model's performance. The confusion matrices show strong diagonal concentrations which indicate accurate classification for most attack types, while the ROC curves demonstrate excellent discrimination with areas close to the ideal upper left corner.

5. Conclusion

In conclusion, the current study has demonstrated and proven the usefulness of hybrid CNN-LSTM model to detect intrusion in SDN environments. Experimental testing has shown that the suggested architecture will provide even better performance, reaching the accuracy of 99.63% at the complete feature size, and that it maintains a high detection accuracy rate even on reduced feature dimensionality. The current study underscores the need to address class imbalance in cyber-security applications, demonstrating the high returns of applying SMOTE, minority-class detection reached 100% recall for previously undetectable attack types. The architecture combines spatial feature extraction via CNN layers with temporal pattern recognition via LSTM networks, providing a powerful framework for detecting advanced cyber threats in SDN environments. The framework's flexibility across feature configurations offers practical deployment solutions for a range of operational scenarios, from resource-limited environments to critical infrastructure protection. The study, therefore, offers an important contribution to the development of intelligent security solutions to SDN that can provide high detection rates as well as computational efficiency whilst facing the major issues related to optimization of features as well as data imbalance. The future work will focus on the real-time implementation of the system, adaptive learning mechanisms for evolving threats and integration with SDN controllers to facilitate automated security response systems.

Acknowledgement: The Author would like to thank Mustansiriyah University (https://uomustansiriyah.edu.iq/) Baghdad –Iraq for its support in the present work.

Conflict of Interest: The author's disclosure statement confirms the absence of any conflicts of interest.

References

[1] M. Hussain, N. Shah, R. Amin, S. S. Alshamrani, A. Alotaibi, and S. M. Raza, "Software-Defined Networking: Categories, Analysis, and Future Directions," *Sensors 2022, Vol. 22, Page 5551*, vol. 22, no. 15, p. 5551, Jul. 2022, doi: 10.3390/S22155551.

[2] H. A. Suleiman and Z. H. Ali, "The Extremism Detection Using Hybrid Deep Learning," *Mustansiriyah J. Pure Appl. Sci.*, vol. 3, no. 4, pp. 173–189, Sep. 2025, doi: 10.47831/MJPAS.V3I4.292.

[3] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1,

pp. 1–54, Dec. 2019, doi: 10.1186/S40537-019-0192-5/TABLES/18.

[4] S. Salman, S. Salman, and J. H. Soud, "Deep Learning Machine using Hierarchical Cluster Features," *Al-Mustansiriyah J. Sci.*, vol. 29, no. 3, pp. 82–93, Mar. 2019, doi: 10.23851/mjs.v29i3.625.

[5] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/S13244-018-0639-9.

[6] "Long Short-Term Memory." Accessed: Sep. 09, 2025. [Online]. Available: https://www.researchgate.net/publication/13853244_Long_Short-Term_Memory

[7] C. V., B. W., H. O., and K. Philip, "SMOTE," *J. Artif. Intell. Res.*, Jun. 2002, doi: 10.5555/1622407.1622416.

[8] N. K. S. Nayak and B. Bhattacharyya, "Multilayered SDN security with MAC authentication and GAN-based intrusion detection," PLoS One, vol. 20, no. 9, p. e0331470, Sep. 2025, doi: 10.1371/JOURNAL.PONE.0331470.

[9] Y. Zhang, C. Jue, W. Liu, and Y. Ma, "GRAN: a SDN intrusion detection model based on graph attention network and residual learning," Comput. J., vol. 68, no. 3, pp. 241–260, Mar. 2025, doi: 10.1093/COMJNL/BXAE108.

[10] A. Dadhania et al., "Software defined network and graph neural network-based anomaly detection scheme for high speed networks," Cyber Secur. Appl., vol. 3, p. 100079, Dec. 2025, doi: 10.1016/J.CSA.2024.100079.

[11] L. Mhamdi and M. M. Isa, "Securing SDN: Hybrid autoencoder-random forest for intrusion detection and attack mitigation," J. Netw. Comput. Appl., vol. 225, p. 103868, May 2024, doi: 10.1016/J.JNCA.2024.103868.

[12] A. M. Zacaron, D. M. B. Lent, V. G. da Silva Ruffo, L. F. Carvalho, and M. L. Proença, "Generative Adversarial Network Models for Anomaly Detection in Software-Defined Networks," J. Netw. Syst. Manag. 2024 324, vol. 32, no. 4, pp. 93-, Sep. 2024, doi: 10.1007/S10922-024-09867-Z.

[13] H. A. Hassan, E. El-Din Hemdan, M. Shokair, F. E. A. El-Samie, and W. El-Shafai, "An Efficient Attack Detection Framework in Software-Defined Networking using Intelligent Techniques," *ICEEM 2023 - 3rd IEEE Int. Conf. Electron. Eng.*, no. October, 2023, doi: 10.1109/ICEEM58740.2023.10319575.

[14] R. A. Elsayed, R. A. Hamada, M. I. Abdalla, and S. A. Elsaid, "Securing IoT and SDN systems using deep-learning based automatic intrusion detection," *Ain Shams Eng. J.*, vol. 14, no. 10, Oct. 2023, doi: 10.1016/j.asej.2023.102211.

[15] S. H. A. Kazmi, F. Qamar, R. Hassan, K. Nisar, D. P. B. Dahnil, and M. A. Al-Betar, "Threat Intelligence with Non-IID Data in Federated Learning enabled Intrusion Detection for SDN: An Experimental Study," *2023 24th Int. Arab Conf. Inf. Technol. ACIT 2023*, pp. 1–6, 2023, doi: 10.1109/ACIT58888.2023.10453867.

[16] A. Mzibri, R. Benaini, and M. Ben Mamoun, "Case Study on the Performance of ML-Based Network Intrusion Detection Systems in SDN," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 90–95. doi: 10.1007/978-3-031-37765-5_7.

[17] A. Almazyad, L. Halman, and A. Alsaeed, "Probe Attack Detection Using an Improved Intrusion Detection System," *Comput. Mater. Contin.*, vol. 74, no. 3, pp. 479–4784, 2023, doi: 10.32604/cmc.2023.033382.

[18] H. M. Chuang, F. Liu, and C. H. Tsai, "Early Detection of Abnormal Attacks in Software-Defined Networking Using Machine Learning Approaches," *Symmetry (Basel).*, vol. 14, no. 6, Jun. 2022, doi: 10.3390/sym14061178.

[19] S. Wang *et al.*, "Detecting flooding DDoS attacks in software defined networks using supervised learning techniques," *Eng. Sci. Technol. an Int. J.*, vol. 35, Nov. 2022, doi: 10.1016/j.jestch.2022.101176.

[20] J. Wang and L. Wang, "SDN-Defend: A Lightweight Online Attack Detection and Mitigation System for DDoS Attacks in SDN," *Sensors*, vol. 22, no. 21, 2022, doi: 10.3390/s22218287.

[21] M. S. El Sayed, N. A. Le-Khac, M. A. Azer, and A. D. Jurcut, "A Flow-Based Anomaly Detection Approach With Feature Selection Method Against DDoS Attacks in SDNs," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 4, pp. 1862–1880, 2022, doi: 10.1109/TCCN.2022.3186331.

[22] M. S. ElSayed, N. A. Le-Khac, M. A. Albahar, and A. Jurcut, "A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique," *J. Netw. Comput. Appl.*, vol. 191, Oct. 2021, doi: 10.1016/j.jnca.2021.103160.

[23] Q. V. Dang, "Intrusion Detection in Software-Defined Networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 356–371. doi: 10.1007/978-3-030-91387-8_23.

[24] A. S. Alshra'A, A. Farhat, and J. Seitz, "Deep Learning Algorithms for Detecting Denial of Service Attacks in Software-Defined Networks," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 254–263. doi: 10.1016/j.procs.2021.07.032.

[25] 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 *ACM International Conference Proceeding Series*, Association for Computing Machinery, Aug. 2021. doi: 10.1145/3465481.3469190.

[26] O. E. Tayfour and M. N. Marsono, "Collaborative detection and mitigation of DDoS in software-defined networks," *J. Supercomput.*, vol. 77, no. 11, pp. 13166–13190, Nov. 2021, doi: 10.1007/s11227-021-03782-9.

[27] M. Said Elsayed, N. A. Le-Khac, S. Dev, and A. D. Jurcut, "Network Anomaly Detection Using LSTM Based Autoencoder," in *Q2SWinet 2020 - Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, Association for Computing Machinery, Inc, Nov. 2020, pp. 37–45. doi: 10.1145/3416013.3426457.

[28] M. S. Elsayed, N. A. Le-Khac, and A. D. Jurcut, "InSDN: A novel SDN intrusion dataset," *IEEE Access*, vol. 8, pp. 165263–165284, 2020, doi: 10.1109/ACCESS.2020.3022633.

[29] L.-P. Chen, "Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar: Foundations of machine learning, second edition," *Stat. Pap. 2019 605*, vol. 60, no. 5, pp. 1793–1795, Jul. 2019, doi: 10.1007/S00362-019-01124-9.

[30] S. H. A. Kazmi, F. Qamar, R. Hassan, K. Nisar, D. P. B. Dahnil, and M. A. Al-Betar, "Threat Intelligence with Non-IID Data in Federated Learning enabled Intrusion Detection for SDN: An Experimental Study," in *2023 24th International Arab Conference on Information Technology, ACIT 2023*, Institute of Electrical and Electronics

Engineers Inc., 2023. doi: 10.1109/ACIT58888.2023.10453867.

[31] N. Abbas, Y. Nasser, M. Shehab, and S. Sharafeddine, “Attack-Specific Feature Selection for Anomaly Detection in Software-Defined Networks,” in *2021 3rd IEEE Middle East and North Africa COMMunications Conference, MENACOMM 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 142–146. doi: 10.1109/MENACOMM50742.2021.9678279.

[32] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, “Deep Recurrent Neural Network for Intrusion Detection in SDN-based Networks,” *2018 4th IEEE Conf. Netw. Softwarization Work. NetSoft 2018*, pp. 462–469, Sep. 2018, doi: 10.1109/NETSOFT.2018.8460090.