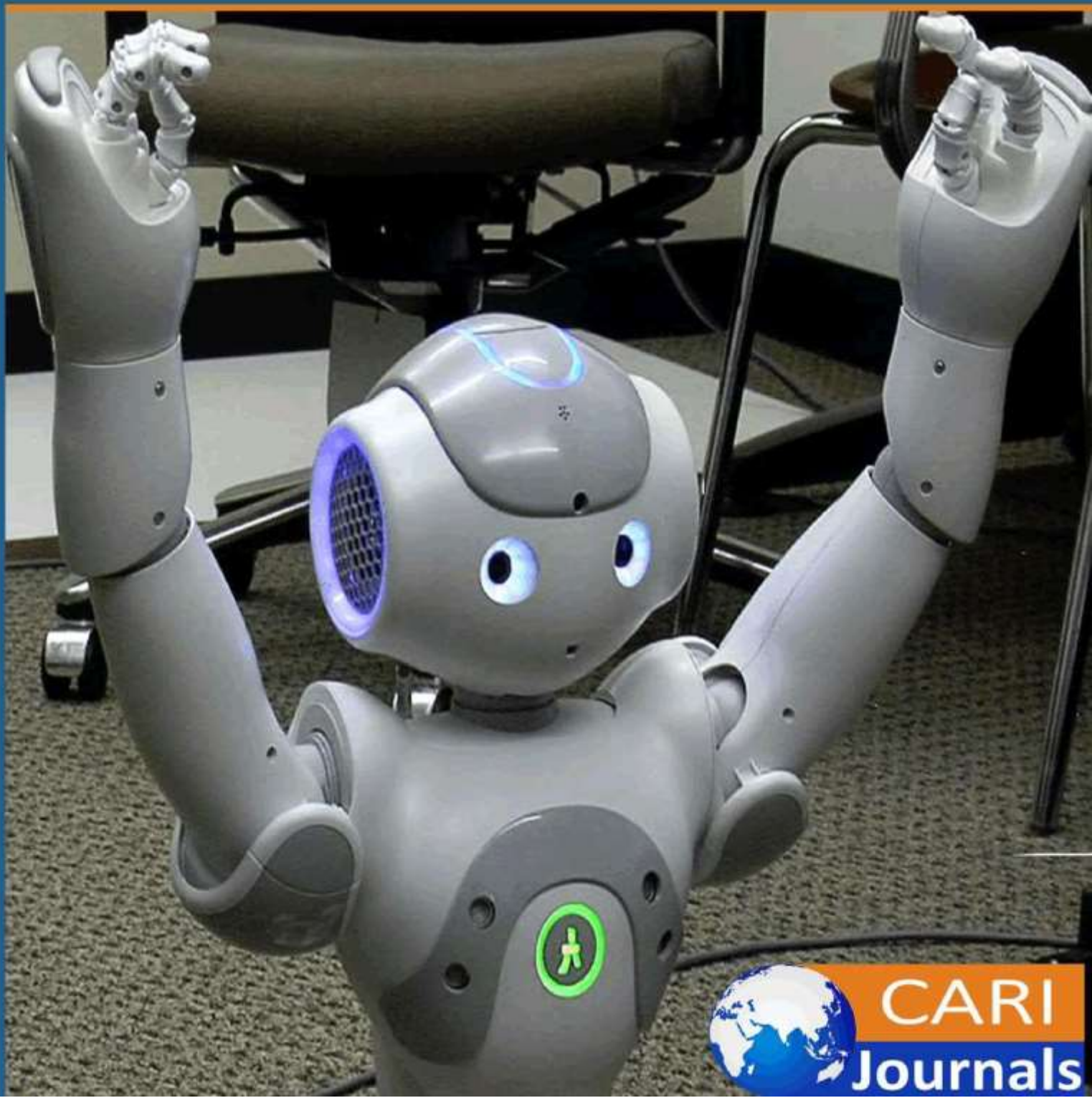
International Journal of
Computing and
Engineering
(IJCE)

A Deep Learning Driven Cloud Edge Intelligence Framework for Real-Time Big Data Based Cyber-Security Threat Detection



CARI Journals

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

A Deep Learning Driven Cloud Edge Intelligence Framework for Real-Time Big Data Based Cyber-Security Threat Detection

[1*]Dr. Abdinasir Ismael Hashi, [2]Abdirizak Mohamed Hashi

[1]Somali National university

[2]Jazeera university

https://orcid.org/0009-0009-0635-2609

Accepted: 26th Dec, 2025, Received in Revised Form: 9th Jan, 2026, Published: 15th Jan, 2026

Abstract

Purpose: The rapid expansion of cloud computing, edge intelligence, and big data environments has increased the scale and sophistication of cybersecurity threats. This study aims to develop an intelligent real-time cyber-threat detection framework based on deep learning within a cloud–edge intelligence architecture, capable of effectively analyzing large-scale network traffic data.

Methodology: The proposed framework employs the CICIDS2017 benchmark dataset and applies multiple stages of data preprocessing, feature engineering, and class distribution balancing. Several deep learning models—Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU)—are designed to capture spatial and temporal attack patterns. In addition, a hybrid CNN–LSTM–GRU model is developed to leverage the complementary strengths of these architectures. Model performance is evaluated using both binary and multiclass classification tasks.

Findings: Experimental results demonstrate that the hybrid CNN–LSTM–GRU model outperforms individual models across all evaluation metrics. In binary classification, the hybrid model achieves an accuracy of 99.24%, surpassing CNN (99.10%), GRU (99.00%), and LSTM (98.95%). For multiclass classification, the hybrid model attains an accuracy of 93.35%, exceeding CNN (91.93%), GRU (92.36%), and LSTM (80.28%). These results confirm the framework's strong capability for accurate and real-time cyber-threat detection.

Unique Contribution to Theory, Practice and Policy: This study contributes theoretically by demonstrating the effectiveness of integrating spatial and temporal deep learning models for cyber-threat detection. Practically, it provides a high-performance cloud–edge intelligence framework suitable for real-time deployment in complex network environments. From a policy perspective, the findings support the adoption of advanced AI-driven security mechanisms to enhance national and organizational cybersecurity resilience in cloud and edge computing infrastructures.

Keywords: *Cyber-security, Cloud–Edge Intelligence, Deep Learning, Intrusion Detection, Big Data Analytics*

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

1. Introduction

The rapid expansion of cloud computing, edge intelligence, and big data technologies has fundamentally transformed modern digital infrastructures, enabling scalable computation, real-time analytics, and ubiquitous connectivity across diverse application domains [1]. As much as these innovations are helpful in supporting smart cities, health care systems, automation of industries, and Internet of Things (IoT) networks and ecosystems, they have also dramatically widened the attack surface of cyber threats [2]. The large scale, speed, and amount of data at cloud and edge layers render standard security systems unsuitable in detecting and responding to advanced and novel cyber-attacks [3]. The consequence is an increase in demand of smart, dynamic and real time cyber security systems that are able to perform effectively at distributed computing environment. The various typical attacks or threats with respect to cybersecurity are depicted in figure 1.

Conventional cyber security approaches, such as signature-based intrusion detection systems and rule-based firewalls, depend on predefined patterns and centralized processing [4]. These methods are effective against known attacks but fall short in recognizing zero-day exploits, advanced persistent threats, and polymorphic malware that are continuously changing their structure and behavior [5]. Additionally, the centralized nature of traditional cloud-based security solutions creates latency, bandwidth overheads, and single points of failure - all critical limitations for time-sensitive applications [6]. Edge computing has recently emerged as a paradigm that brings computation close to data sources; hence it can be an opportunity to overcome these constraints by facilitating faster response times while reducing the amount of data transmitted to centralized cloud servers [7].

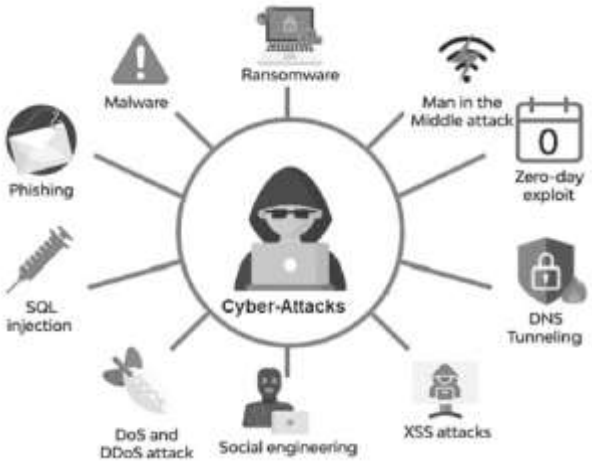


Figure 1: Several common attacks or threats in the context of cybersecurity [8].

Cloud-edge intelligence, in this regard, has become one of the promising paradigms involving the combination of the high-computation capabilities of cloud platforms and the storage capabilities of cloud platforms with the low-latency and context-aware processing of edge

devices [9]. Cloud-edge systems can promote scaleable, robust, and real-time cyber security functions by allocating analytics and decision-making both to cloud and edge layers [10]. Nevertheless, the non-uniformity of devices, dynamic network profiles and persistent streams of data make it extremely difficult to establish effective threats detection mechanisms in such contexts. It is then necessary to apply intelligent automation and learning methods to handle this complexity and provide a high level of security over the entire infrastructure [11].

DL has already proven itself to be exceptionally effective at identifying intricate patterns in massive amounts of data, and has become a major facilitator to the next generation of cyber security solutions (Figure 2) [12]. DL models in contrast to the traditional approaches of machine learning are able to learn hierarchical representations automatically of raw data and are thus specifically suited to analyzing high dimensional, and unstructured cyber security data including network traffic, system logs and user behavioral patterns [13,14]." Convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory (LSTM)" networks have demonstrated the great potential in the detection of intrusions, malware, and anomalous activities at a high rate of accuracy [15,16].

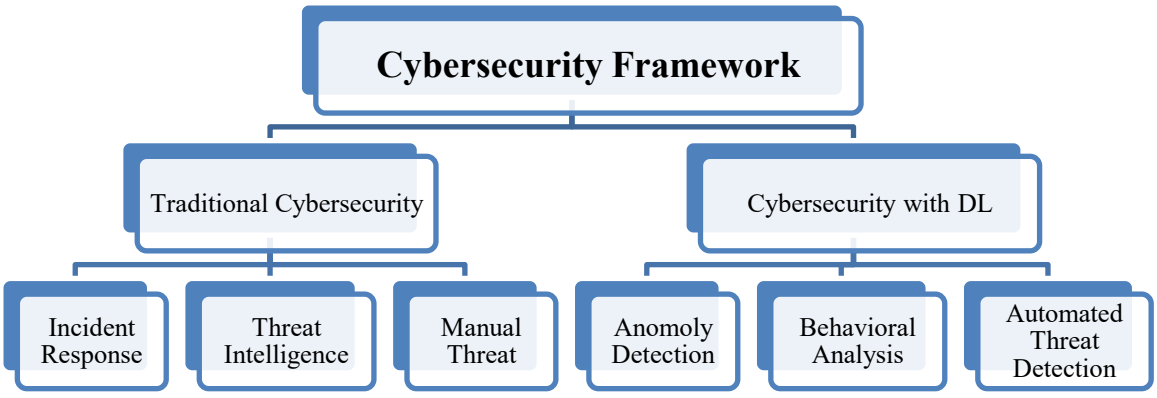


Figure 2: Deep Learning in Cyber-security **[17].**

DL integration into a cloud-edge intelligence framework supports real-time threat detection while satisfying latency and scalability requirements [18]. Edge nodes can conduct preliminary analysis and anomaly detection on local data streams for immediate responses to potential threats, whereas the cloud layer can manage more complex model training, global threat correlation, and long-term analysis using aggregated big data [19,20]. Such collaborative processing not only improves detection efficiency but also enhances system resilience by distributing security intelligence across multiple layers [21].

The study presents an intelligent system of cloud-edge cyber security threat detection by using big data analytics which is proposed to be based on DL. The dataset used in the study is CICIDS2017 as it is used to simulate realistic attack and network traffic and subsequently,

extensive data preprocessing such as cleaning, encoding, feature engineering, scaling, and class imbalance management is conducted. The hybrid CNN-LSTM-GRU model is created to concurrently identify both spatial and temporal correlations of network flows. The edge node does low-latency anomaly detection and the cloud manages model training and analysis on a global level. The framework is tested with the standard performance measures in order to show accurate scalable and timely threat detection. Here are the research objectives of the study follows as:

- To design a DL–driven cloud–edge intelligence framework for real-time cybersecurity threat detection in distributed environments.
- To utilize the CICIDS2017 benchmark dataset to model realistic network traffic and diverse contemporary cyber-attack scenario.
- To develop a hybrid CNN–LSTM–GRU model capable of capturing both spatial and temporal attack patterns.
- To study the impact of feature engineering, scaling, and class imbalance handling on key performance parameters such as accuracy, precision, recall, and F1-score.

2. Literature Review

The latest developments in the field of cybersecurity of IoT, cloud and critical infrastructure settings have focused more on the application of AI and DL algorithms to tackle the increasing complexity, multifacetedness and volume of cyber threats. Awan et al. (2025) [22] introduced a framework called SecEdge that is a transformer- and GNN-based framework with federated learning, and it has the highest real-time detection rates of over 98% on various benchmark datasets (NSL-KDD, UNSW-NB15, CICIDS2017). Likewise, Khalaf et al. (2025) [23] noted the constraints of the traditional rule-based systems and indicated that AI-based threat detection is more accurate, flexible, and has automated response functions in critical infrastructure settings with the performance measures as accuracy of 0.95, precision 0.93, and recall 0.92. The study by Hussein et al. (2023) [24] involved the approach to real-time intrusion detection based on the application of a Fully Streaming Big Data Framework (FSBDL) and hyper-parallel optimization of CNNs in order to obtain accuracy reaching over 99.9%. In [25], Malik et al. expanded AI-based cybersecurity designs to incorporate supervised, unsupervised and reinforcement learning models, federated learning, and the Explainable AI (XAI), offering adaptive, scalable and transparent threat alleviation solutions to organization settings. The comparison of literature review is presented in table 1.

Table 1: Comparison of literature Review

Author(s), Year	Domain / Focus	Dataset / Environment	Model / Technique	Key Contributions	Performance Metrics
Awan et al., 2025	Mobile IoT cybersecurity	NSL-KDD, UNSW-NB15, CICIDS2017	Transformer + GNN + Federated Learning	Real-time threat detection with adaptive learning; handles relational data	DoS detection: 98.8%, MitM: 98.5%, Data injection: 98.7%
Khalaf et al., 2025	Critical infrastructure threat detection	Simulated environment	AI-based ML system with anomaly detection & automated response	Adaptive real-time detection; reduces false positives; automated mitigation	Accuracy: 0.95, Precision: 0.93, Recall: 0.92, F1-score: 0.92
Hussen et al., 2023	Real-time intrusion detection	Various network datasets	FSBDL framework with hyper-parallel optimized CNN (Adam + RMSprop)	Real-time detection with high stability and reduced overfitting	Accuracy: >99.9%
Malik et al., 2025	AI-driven cybersecurity architecture	Industry-level surveys & case studies	Supervised, unsupervised, RL + ANN-ISM + XAI + Federated Learning	Multi-layered, adaptive threat detection; scalable & explainable	Significant improvement over traditional systems in accuracy, adaptability, and response time
Farzaan et al., 2025	Cloud cybersecurity / incident response	NSL-KDD, UNSW-NB15, CIC-IDS-2017	Random Forest, Neural Network, Deep Learning + Containerized Deployment	Automated incident response pipeline; scalable cloud integration	Random Forest: Accuracy 90–99%, Malware NN: 99% accuracy, Precision 96%
Ezeh et al., 2025	Network traffic threat mitigation	NSK-DD dataset	LSTM + Autoencoder + Cross-correlation feature extraction	Real-time feature evaluation and mitigation; low latency	Accuracy: 98.6%, Precision: 97.9%, Recall: 98.1%, F1-score: 98.0%, Mitigation latency <1.5s
Adeniyi et al., 2024	MEC DDoS detection	NF-UQ-NIDS-V2	Hybrid AE–MLP	Combines feature extraction and DL for DDoS detection	Accuracy: 99.98%
Sathupadi et al., 2024	Predictive maintenance	Edge–cloud sensor data	KNN (edge) + LSTM (cloud)	Real-time anomaly detection and predictive failure analysis	Latency ↓35%, Energy ↓28%, Bandwidth ↓60%
Areghan et al., 2024	Cloud threat detection	AWS, Azure, GCP logs (~1.2M entries)	RF, SVM, XGBoost, CNN, LSTM	Multi-model evaluation for cloud platforms; risk scoring & real-time alerting	CNN ROC-AUC: 0.94, LSTM: 0.91, XGBoost: 0.87, Precision: 92%, Recall: 89%
Saxena et al., 2023	Cloud VM threat prediction	Google Cluster & OpenNebula VM traces	MR-TPM (Multiple Risk Analysis + ML classifier)	Proactive VM threat estimation; reduces cybersecurity risks	Threat reduction: 88.9%
Al-Ghuwairi et al., 2023	Cloud intrusion detection	Time series cloud data	Collaborative Feature Selection + Facebook Prophet	Early detection using time series anomalies; reduces false positives	Reduced training, prediction, cross-validation time by 85%, 15%, 97% respectively
Tyagadurgam et al., 2022	Cloud IDS	CICIDS2017	Bi-LSTM	Captures forward & backward dependencies; handles class imbalance	Accuracy: 98.51%, Precision: 99%, Recall: 98%, F1-score: 99%

Other studies have looked at hybrid and domain-specific strategies to improve detection performance. Farzaan et al. (2025) [26] proposed an AI-based cyber incident response system for cloud environments, which combines DL with Random Forest models and containerized deployment, achieving up to 99% accuracy in malware analysis. Ezeh et al. (2025) [27] used LSTM in conjunction with an autoencoder for feature extraction to provide end-to-end real-time mitigation with low latency. Edge and hybrid frameworks were discussed by Adeniyi et al. (2024) [28] and Sathupadi et al. (2024) [29], who showed that DDoS detection and predictive maintenance could be improved through AE–MLP and KNN–LSTM models, respectively. Areghan et al. (2024) [30], Saxena et al. (2023) [31], and Al-Ghuwairi et al. (2023) [32] worked on detecting threats in the cloud using ML, assessing the risks of VMs, and modeling anomalies in time series data, respectively. Finally, Tyagadurgam et al., in 2022 applied Bi-LSTM models for advanced intrusion detection with almost 99% performance metrics across various indicators being reported. These studies together emphasize the effectiveness of AI-driven frameworks in improving real-time cybersecurity in varied environments while tackling issues like scalability, adaptability, and low-latency detection.

3. Research Methodology

The research methodology for the DL Driven Cloud–Edge Intelligence Framework for Real-Time Big Data Based Cyber security Threat Detection is designed to ensure accurate threat identification, low-latency response, and scalable deployment across distributed environments. The methodology integrates big data processing, DL models, and cloud–edge collaboration in a structured and systematic manner. Figure 3 shows the flowchart of the suggested work.

3.1 Dataset Used

The CICIDS2017 dataset [34], which was created by the Canadian Institute for Cybersecurity, is a benchmark dataset that is very popular in cyber-security research. The dataset was created to address the limitations of the old datasets by including realistic network traffic and up-to-date attack scenarios. There was both benign and malicious traffic in the dataset generated in a controlled environment, but it was like the real world. The dataset features a vast number of attacks that include brute force (FTP and SSH), DoS and DDoS, botnet activity, web attacks (SQL injection and XSS), infiltration, and port scanning. Besides PCAP files, CICIDS2017 also offers flow-based features that were extracted with the help of CIC-Flow-Meter, and these features account for more than 84 statistical attributes per flow (as shown in Table 3). Due to its size, variety, and lifelike characteristics, it is an excellent dataset for testing DL-based cloud–edge cybersecurity threat detection frameworks. The Training and Testing Split of the proposed dataset is given in Table 2.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

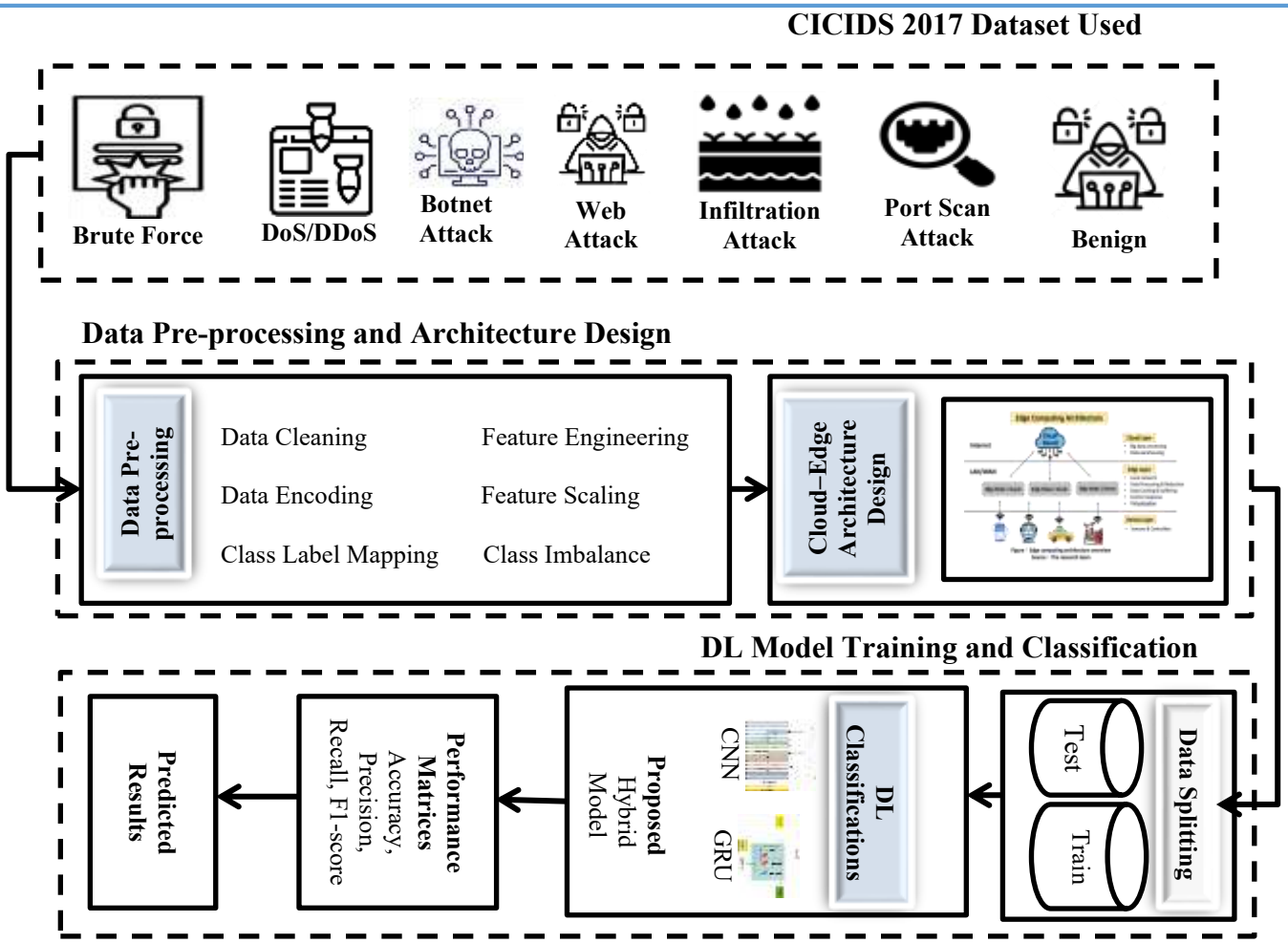


Figure 3: Flowchart of proposed work

Table 2: Training and Testing Split of CICIDS2017 Dataset

Category	Class	Flow Count	Percentage (%)	Training (70%)	Testing (30%)
Benign	Benign	2,273,097	76.75	1,591,167	681,929
DDoS	DDoS	231,073	7.802	161,751	69,321
DoS	Heartbleed	11	0.0003	7	3
DoS	DoS Slowloris	5,796	0.1957	4,057	1,738
DoS	DoS GoldenEye	10,293	0.3475	3,087	7,205
DoS	DoS SlowHTTPTest	5,499	0.1856	3,849	1,649
DoS	DoS Hulk	231,073	0.0392	161,751	69,321
Web Attack	SQL Injection	5,796	0.2121	4,057	1,738
Web Attack	Brute Force	7,938	0.2906	5,556	2,381
Web Attack	XSS	5,897	0.2158	4,127	1,769
Infiltration	Infiltration	10,293	0.3768	7,205	3,087
Port Scan	Port Scan	158,930	5.8184	111,251	47,679
Brute Force	FTP-Patator	1,769	0.2906	1,238	530
Brute Force	SSH-Patator	5,897	0.2158	4,127	1,769
Bot	Bot	1,966	0.0719	1,376	589

3.2 Data Pre-processing

The Data Pre-processing Layer plays a critical role in the deep learning pipeline by transforming raw inputs into a suitable form for training, evaluation, and real-time inference [35]. The structure and quality of the input data strongly influence the overall performance of deep learning models. Effective preprocessing enhances model accuracy and reduces the risk of overfitting, thereby improving generalization to unseen data. In the Smart-Trust Framework, this layer processes raw network traffic, user activity records, and contextual data into meaningful features that can be utilized by deep learning models, including hybrid architectures such as CNN, LSTM, and GRU. The preprocessing stage involves operations such as data normalization, feature extraction, sequential data arrangement, and encoding of categorical attributes [36,37]. Figure 4 illustrates the architecture of the data pre-processing layer.

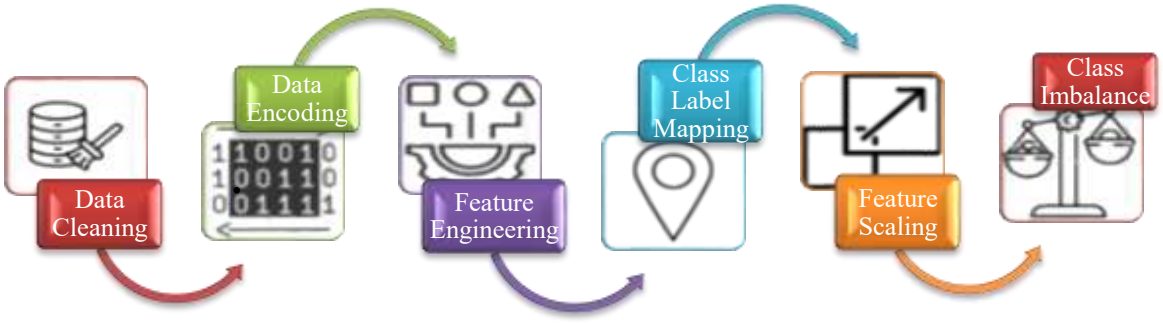


Figure 4: Data-preprocessing layer

- **Data Cleaning**

Data cleaning steps remove noise, duplicates, and missing values with the goal of improving data quality. Incomplete entries are changed via statistical imputation, thus ensuring that DL models can be trained effectively in a consistent and reliable manner.

$$x_i = \begin{cases} \bar{x}, & x_i \; missing \\ x_i, & otherwise \end{cases}$$

- **Data Encoding**

Data encoding changes categorical attributes into numerical representations through one-hot or label encoding, which makes it possible for neural networks to process protocol types, services, and flags without the need for ordinal bias to be introduced.

$$c_j \rightarrow [0,0,\ldots,1,\ldots,0]$$

- **Feature Engineering**

Feature engineering takes relevant features like packet rate and flow duration, selects and transforms them to lower the dimensionality while increasing the discriminative power for accurate cyber-security threat detection.

$$MI(f,y) = \sum p(f,y) log \frac{p(f,y)}{p(f)p(y)}$$

Table 3: 84 Feature attributes

Feature Name	Feature Name	Feature Name	Feature Name
Flow ID	Flow Packets/s	Fwd Packets/s	Fwd Avg Packets/Bulk
Source IP	Flow IAT Mean	Bwd Packets/s	Fwd Avg Bulk Rate
Source Port	Flow IAT Std	Min Packet Length	Bwd Avg Bytes/Bulk
Destination IP	Flow IAT Max	Max Packet Length	Bwd Avg Packets/Bulk
Destination Port	Flow IAT Min	Packet Length Mean	Bwd Avg Bulk Rate
Protocol	Fwd IAT Total	Packet Length Std	Subflow Fwd Packets
Timestamp	Fwd IAT Mean	Packet Length Variance	Subflow Fwd Bytes
Flow Duration	Fwd IAT Std	FIN Flag Count	Subflow Bwd Packets
Total Fwd Packets	Fwd IAT Max	SYN Flag Count	Subflow Bwd Bytes
Total Backward Packets	Fwd IAT Min	RST Flag Count	Init_Win_bytes_forward
Total Length of Fwd Packets	Bwd IAT Total	PSH Flag Count	Init_Win_bytes_backward
Total Length of Bwd Packets	Bwd IAT Mean	ACK Flag Count	act_data_pkt_fwd
Fwd Packet Length Max	Bwd IAT Std	URG Flag Count	min_seg_size_forward
Fwd Packet Length Min	Bwd IAT Max	CWE Flag Count	Active Mean
Fwd Packet Length Mean	Bwd IAT Min	ECE Flag Count	Active Std
Fwd Packet Length Std	Fwd PSH Flags	Down/Up Ratio	Active Max
Bwd Packet Length Max	Bwd PSH Flags	Average Packet Size	Active Min
Bwd Packet Length Min	Fwd URG Flags	Avg Fwd Segment Size	Idle Mean
Bwd Packet Length Mean	Bwd URG Flags	Avg Bwd Segment Size	Idle Std
Bwd Packet Length Std	Fwd Header Length	Fwd Header Length	Idle Max
Flow Bytes/s	Bwd Header Length	Fwd Avg Bytes/Bulk	Idle Min

- **Class Label Mapping**

Class label mapping takes fine-grained attack labels and puts them into a standardized category in order to simplify the classification task and enhance model generalization over different types of attacks as well as more normal patterns of network traffic.

$$y' = g(y), \quad y' \in \{1, \dots, C\}$$

- **Feature Scaling**

Feature scaling normalizes numerical attributes to a common range through either standardization or min–max normalization, thus avoiding the problem of features with high magnitudes dominating the others and making the neural network converge faster when training.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- **Class Imbalance Handling**

Class imbalance handling uses resampling or cost-sensitive methods to make the minority and majority classes equal, thereby increasing the identification of rare cyber-attacks and lessening the bias towards normal traffic samples.

$$x_{new} = x_i + \lambda(x_j - x_i), \lambda \in (0,1)$$

3.3 Deep Learning Model

- **CNN, LSTM and GRU**

Fundamentally, "Convolutional Neural Networks (CNN)", "Long Short-Term Memory (LSTM)", and "Gated Recurrent Units (GRU)" are DL architectures of significant power and complexity, which are mainly utilized in threats to Upgrade Security in the network [38]. CNNs are very effective in pulling out spatial features from network traffic data through the identification of local patterns and correlations in input sequences like packet headers or flow characteristics. As a result, it becomes on the lookout for abnormal behaviors and attack signatures. Moreover, LSTM networks, which are a form of recurrent neural networks (RNNs), are capable of remembering long-term relationships in the temporal side of sequential data, hence they are able to recognize the patterns of network intrusions which are gradually changing over time [39]. Memory cells address the vanishing gradient problem and can accurately learn from past dependencies. "Gated recurrent unit (GRU)" networks further simplify the memory cell mechanism by combining the forget gate and the input gate into a single update gate, which allows for faster computation without sacrificing accuracy. GRU networks are highly beneficial in dynamic and complex network settings with limited computational resources [40]. In this way, when they are used together, CNN extracts spatial features, while LSTM and GRU provide temporal and sequential relationships. In this way, the hybrid model can detect both known and unknown intrusions to a high degree of accuracy. In combination, they can improve intrusion

detection accuracy, reduce false alarms, and increase the resilience of cyber security against threats to an evolving environment [41]. Figure 5 shows the architecture of CNN, LSTM, and GRU.

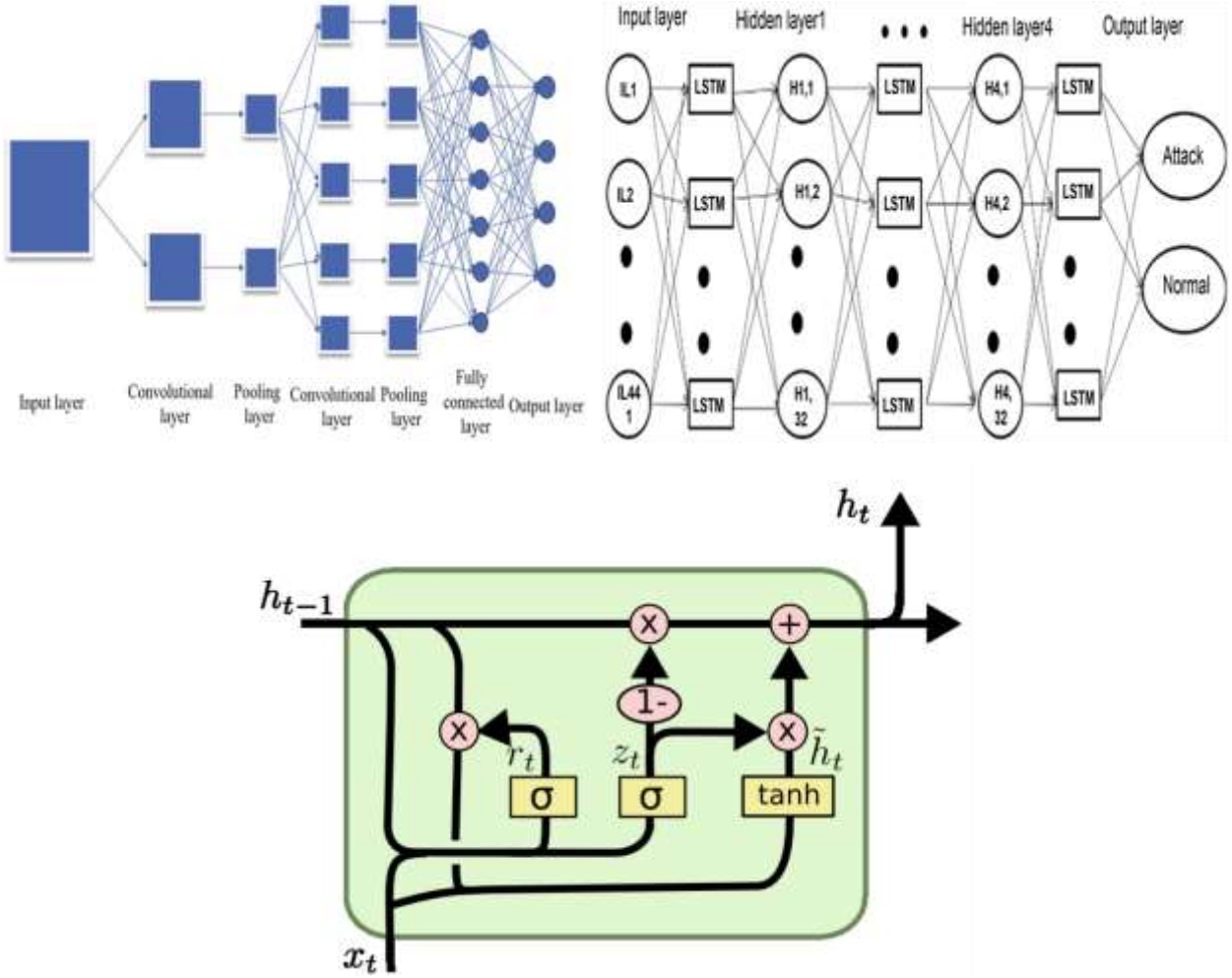


Figure 5: Architecture of CNN, LSTM, and GRU

- **CNN-LSTM-GRU**

The CNN-LSTM-GRU is a hybrid model that is effective in capturing both spatial and temporal information in sequential data. This model has the advantages of convolutional and recurrent neural networks. The CNN layer finds significant features and local spatial patterns based on input data such as signals, pictures or time series sequences. Afterward, the LSTM-layer acquires long-term dependencies and thus maintains sequential relationships but reduces the problem of vanishing gradients. GRU layer further simplifies temporal learning by requiring less parameters, training better and generalizing better. The CNN-LSTM-GRU pipeline can be formally described as:

LSTM cell equations:

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

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

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

$$\widetilde{c_t} = \tanh\,(W_c x_t + U_c h_{t-1} + b_c)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t}$$

$$h_t = o_t \odot \tanh\,(c_t)$$

GRU cell equations:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$

$$\widetilde{h_t} = \tanh\,(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t}$$

Tasks like intrusion detection, voice recognition, healthcare monitoring, and financial time-series forecasting are much improved by this architecture, which also improves sequence modeling and prediction accuracy [42].

3.4 Evaluation Metrics

The model's performance was evaluated using Equations (9)–(12), where $\boldsymbol{A_{accuracy}}$, $\boldsymbol{F1_{score}}$, $\boldsymbol{P_{precision}}$, and $\boldsymbol{R_{recall}}$ are the relevant variables [43,44].

- "TP (true positive): If the model predicts Norm, it is the accurate response.
- FP (false positive): If the model predicts Norm, the accurate response is Attack.
- TN (true negative): If the model predicts Attack and this is the right response.
- FN (false negative): If the model predicts Attack, the accurate response is Norm".

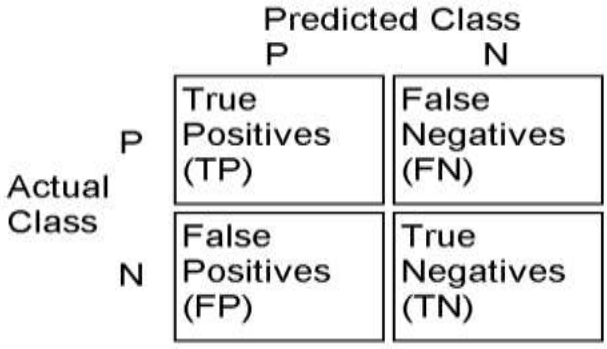


Figure 6: Confusion Matrix

$$\text{Accuracy} = \frac{TN+TP}{TP+FN+FP+TN} \quad (7)$$

$$\text{Precision} = \frac{NA}{NA+FP} \quad (8)$$

$$\text{Recall} = \frac{NA}{FN+NA} \quad (9)$$

$$\text{F1 score} = \frac{2\times\text{precision}\times\text{recall}}{\text{recall}+\text{precision}} \quad (10)$$

4. Results and Discussion

In this section, the experimental results of the proposed DL-based cloud-edge cyber-security framework, and discuss them. Each of the models was executed and tested in Python programming environment, and the data was preprocessed and model train and trained with the help of the following libraries: TensorFlow, Keras, NumPy, and Scikit-learn. Binary and multiclass intrusion detection was evaluated using the CICIDS2017. Accuracy, precision, recall, F1-score, and confusion matrices, ROC, and Precision Recall curves were calculated to give a detailed analysis. The findings are thoroughly discussed to bring out effectiveness of models, comparative performance and robustness under varying attack conditions.

4.1 CNN

Figure 7 shows the performance of the CNN-based binary classifier training with the CICIDS2017 dataset. The subfigure of the left gives the trends of accuracy with five epochs. Accuracy of training is steadily growing, 0.9870 at epoch 0 and 0.9902 at epoch 4, and the validation accuracy is growing as well, 0.9885 to 0.9899, which means that the learning progresses steadily and the results are beneficial in terms of generalization. The right subfigure shows the loss convergence, the training loss is reduced to 0.027, and the validation loss is reduced to 0.023. The tight convergence between the training and validation curve indicates that the model has been optimized and the performance is steady in all the epochs.

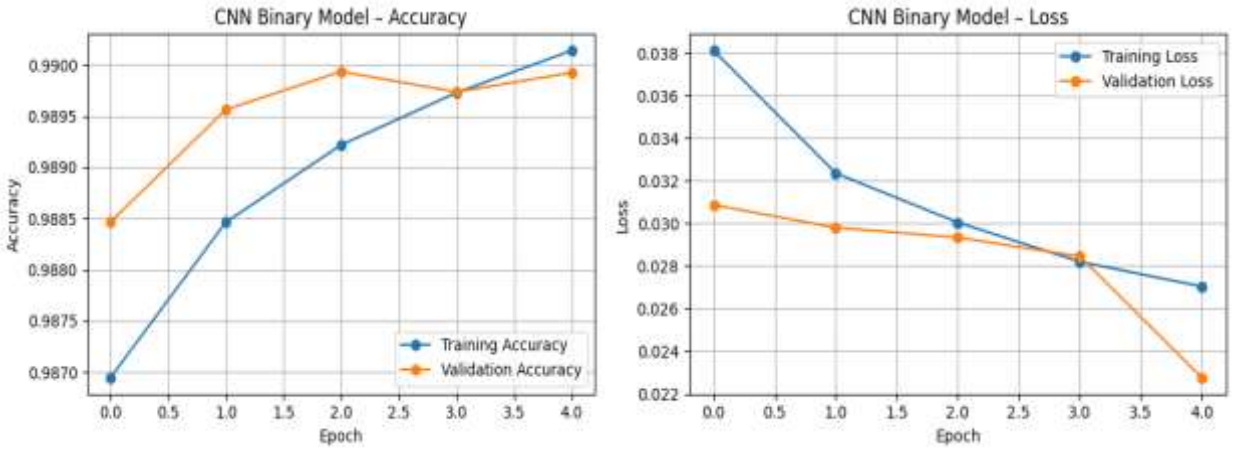


Figure 7: CNN binary model accuracy and loss across training epochs..

The Figure 8 shows how CNN-based multiclass classification model performs over 5 training epochs. The accuracy plot shows that training accuracy is increasing with the epoch starting at 0.84 up to 0.89 and validation accuracy is also increasing starting at 0.89 up to 0.92, and this implies that the model is learning effectively and generalizing better. The loss plot has steadily decreased initially by 0.27 to 0.10 at the final epoch in an indication of a successful model optimization. The loss in validation reduces to 0.22, with slight changes in mid epochs. Training and validation curve disparity is also not high which indicates that convergence is stable and overfitting decreases. In general, the findings qualify the strength and ability of CNN model in multiclassification tasks.

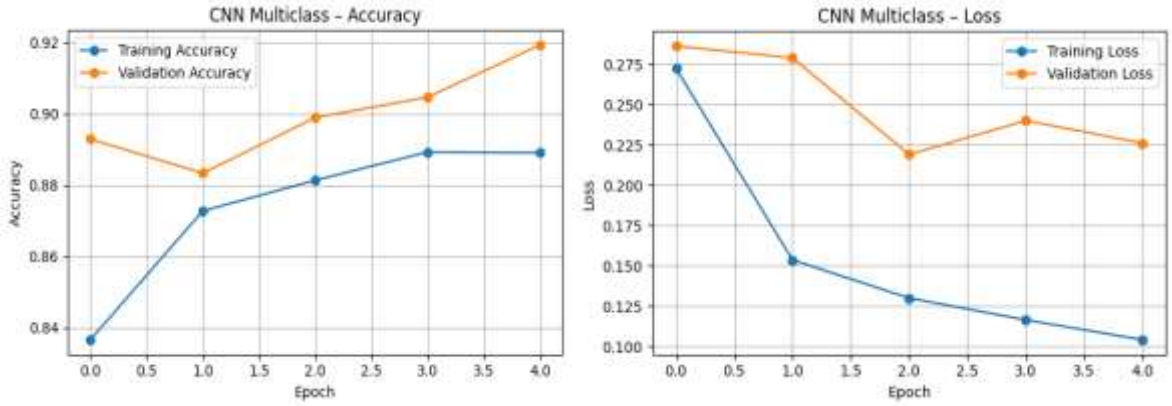


Figure 8: CNN multiclass model accuracy and loss across training epochs.

The figure 9 shows the results of a CNN model with binary and multiclass confusion matrices. In the binary classification scenario, the model correctly labels 394833 benign samples and 63379 attack samples. False alarms are also quite minimal with 631 benign cases falsely predicted as attack (false positive) and 3,919 attack cases falsely predicted as benign (false negative), which shows great detection ability and discrimination between the classes. In the multiclass confusion matrix, the diagonal dominance is an indication of true classification of multiple attack categories and the highest value of true positive is benign class with 3.5 × 10 5 (approximately). Some minor confusion is identified between close forms of attacks like DoS and Brute Force, yet, overall, the misclassification is not significant, which proves the restrictiveness and scalability of the CNN to multi-category intrusion detection.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

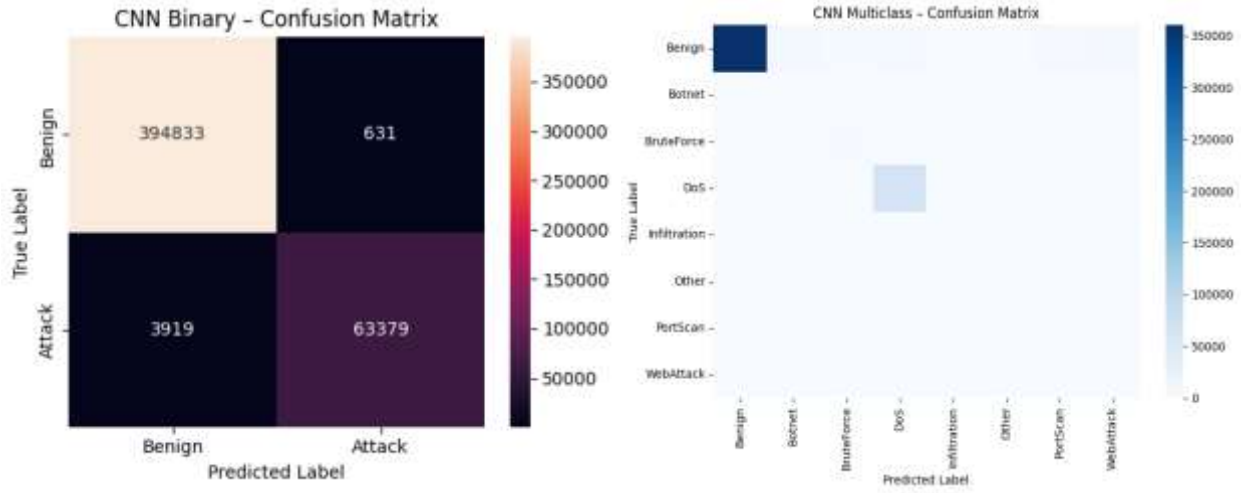


Figure 9: CNN confusion matrices for binary and multiclass intrusion detection.

The performance evaluation of the CNN-based binary intrusion detection model is illustrated in figure 10 through Precision-Recall (PR) and Receiver Operating Characteristic (ROC) curves. The PR curve shows that the model consistently achieves high precision across almost the whole range of recall, meaning that it has a very low false-positive rate even when the recall is high. The average precision (AP) score of 0.9967 is an indication of the model's excellent performance in detecting attack instances in the case of an imbalanced dataset. The ROC curve also supports the strong classifier, gaining an AUC of 0.9993, which is very close to 1, the ideal value. The curve is consistently near the top-left corner, showing that there is an outstanding separation between the two classes (benign and attack) with a very low number of false alarms and a high true positive rate.

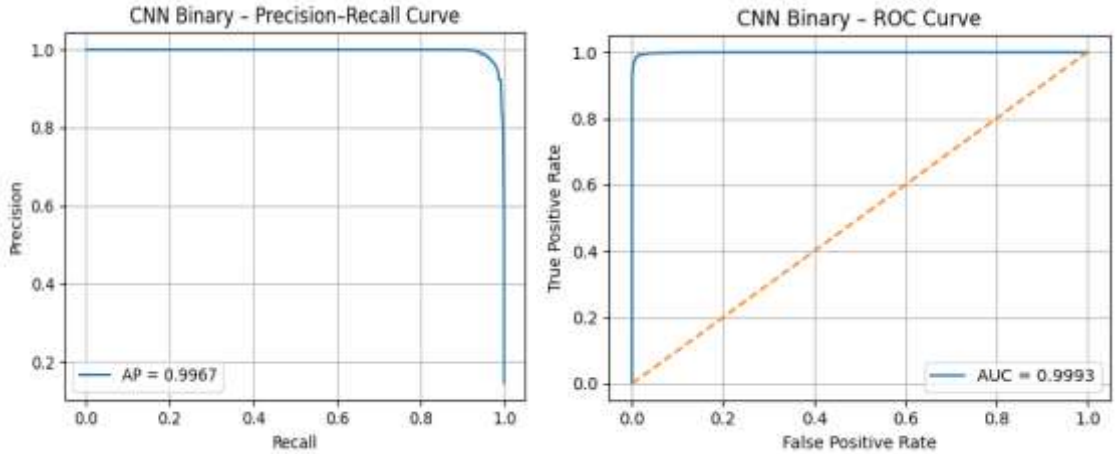


Figure 10: PR and ROC curves of the CNN binary intrusion detection model.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

4.2 LSTM

The training and validation results of a binary classification LSTM model in terms of loss and accuracy during various epochs are depicted in the figure 11. The accuracy curve shows regular increase and the accuracy of training improved with time; at epoch 0, the accuracy was 96.5 percent but at epoch 4 the accuracy has improved to approximately 98.8 percent. In the same manner, the accuracy of validation becomes 97.1 to almost 99.0, which exhibits high generalization. The loss plots show that the training loss decreases steadily with increasing epochs, the training loss declines to about 0.115 and validation loss declines to about 0.034. The fact that the training and validation curves come close indicates that there is no drastic change in learning behavior and there is low overfitting. On the whole, the findings demonstrate good model convergence with a high level of accuracy and minimal loss in a limited number of training cycles.

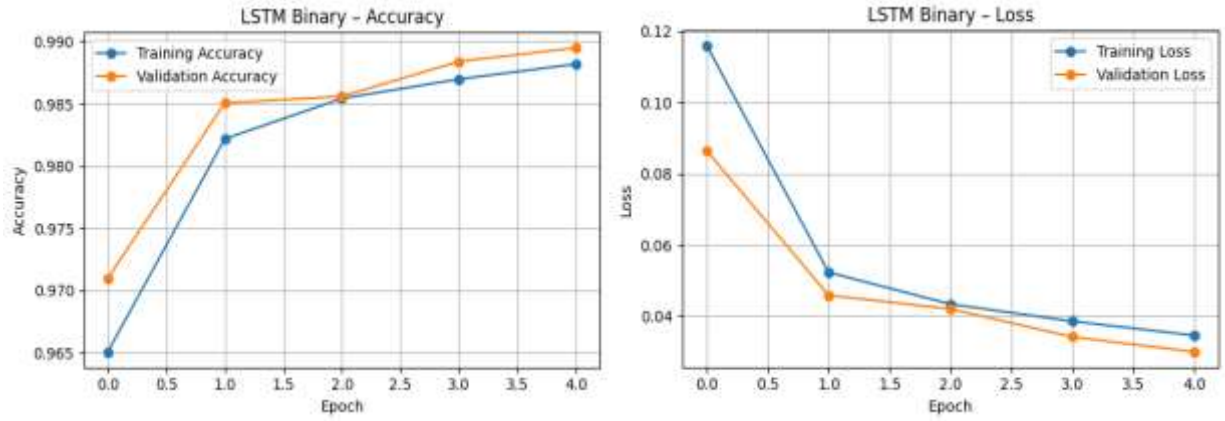


Figure 11: Training and validation accuracy and loss of the LSTM binary classification model.

Figure 12 shows the performance of the LSTM-based multiclass classification model in training and validation over five epochs. The training accuracy rises steeply from 70% at epoch 0 to about 81% by epoch 1, peaking at roughly 84% in epoch 2 with some minor fluctuations afterward and settling close to 83% at epoch 4. Validation accuracy starts relatively high, around 84%, then slightly drops to about 83% by epoch one before more significantly declining down to roughly 76% at epoch two; it recovers almost fully back up to near 80% by the fourth epoch. The loss curves indicate that training loss decreases from approximately 0.55 through the second epoch but then increases slightly up to 0.31 while validation loss generally trends downwards from 0.61 toward something close to 0.47, indicating learning has taken place with moderate class-wise variability.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

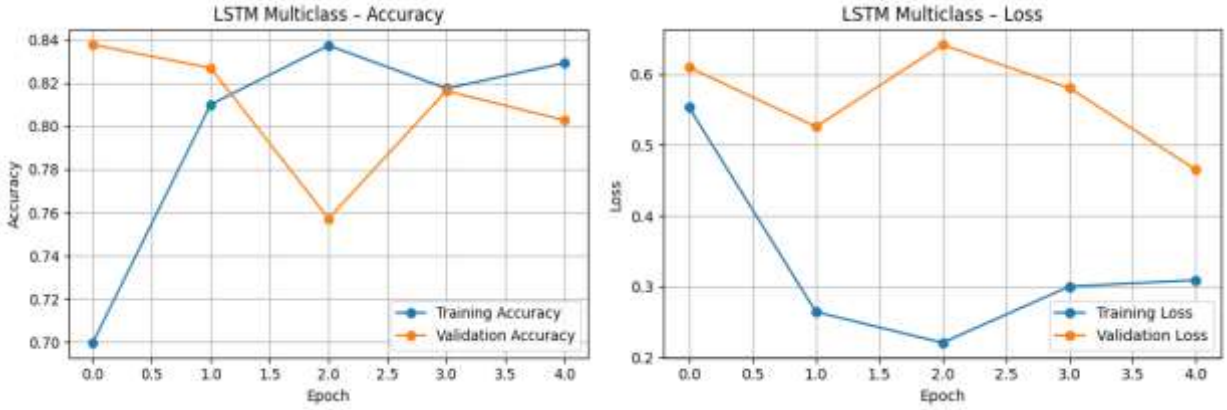


Figure 12: Training and validation accuracy and loss of the LSTM multiclass classification model

The illustration in Figure 13 shows the LSTM model performance through binary and multiclass confusion matrices. The binary classification matrix shows the model achieves perfect identification for 394,468 benign samples and 63,468 attack samples which demonstrates its high classification performance. The system makes two types of errors: 996 benign cases get flagged as attacks (false positives) while 3,830 attack cases pass through as non-intrusions (false negatives). The multiclass confusion matrix shows the prediction results for each class on a logarithmic scale which includes Benign, Botnet, Brute Force, DoS, Infiltration, PortScan and WebAttack categories. Strong diagonal dominance is observed, particularly for Benign and DoS classes, signifying high correct classification rates. The model shows slight confusion between Botnet and Brute Force and PortScan attacks but its overall performance stays strong for multiclass intrusion detection.

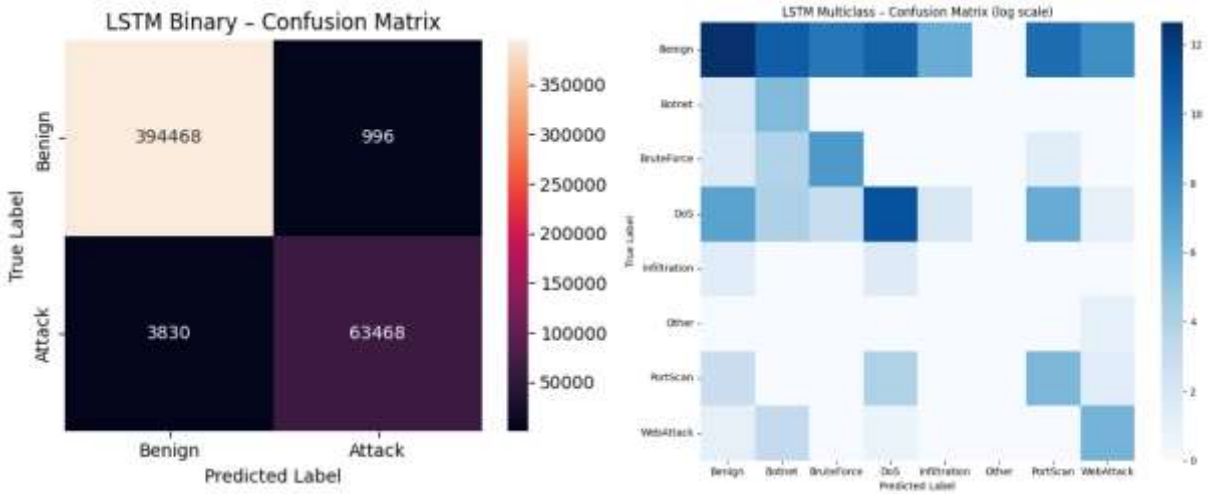


Figure 13: Binary and multiclass confusion matrices of the LSTM model for intrusion detection.

The values 14 show the analysis of the LSTM-based binary classification model on PR and ROC curves. The PR curve is steady in high precision throughout nearly the most of the recall which is

only slightly reduced at the full recall which is a sign of effective attack detection with minimum false positive rate. The value of the Average Precision (AP) of 0.9936 proves that the model is working well with a lack of balance in data. The ROC curve also exhibits a high level of discriminatory ability with an Area Under the Curve (AUC) of 0.9986 that is very near to the theoretical figure of 1. The curve is still close to the top-left side indicating a high true positive and a very low rate of false positive. In general, the findings suggest that the LSTM mechanism is able to deliver very much accurate and strong binary classification results.

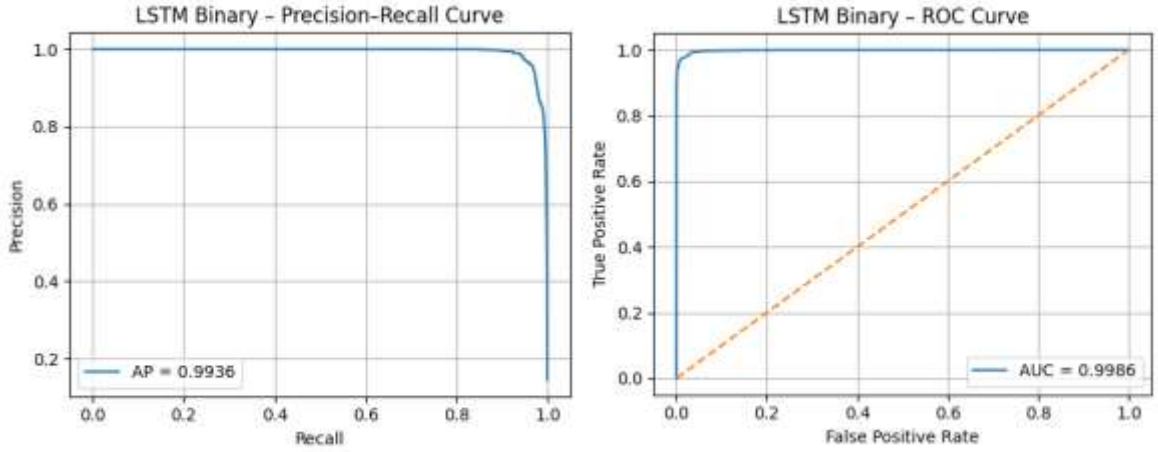


Figure 14: PR and ROC curves of the LSTM binary classification model.

4.3 GRU

Figure 15 shows the training and validation results of the binary classification using GRU model in accuracy and loss per epoch. Accuracy plot indicates that convergence is rapid where the training accuracy at epoch 0 is around 96.2% which is raised to 99.0% at epoch 4. Similar is the case of validation accuracy which increased to about 97.6 to almost 99.0 and this shows high generalization. The loss curves show a steady decrease with epochs, with training loss falling between 0.112 and 0.028, and validation loss falling between 0.072 and 0.025. The fact that the training and validation measures are close implies that learning behavior is stable, and overfitting is minimal. In general, the GRU model obtains high precision and low loss in few training epochs, which show that it is an efficient model to use in a binary classification task.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

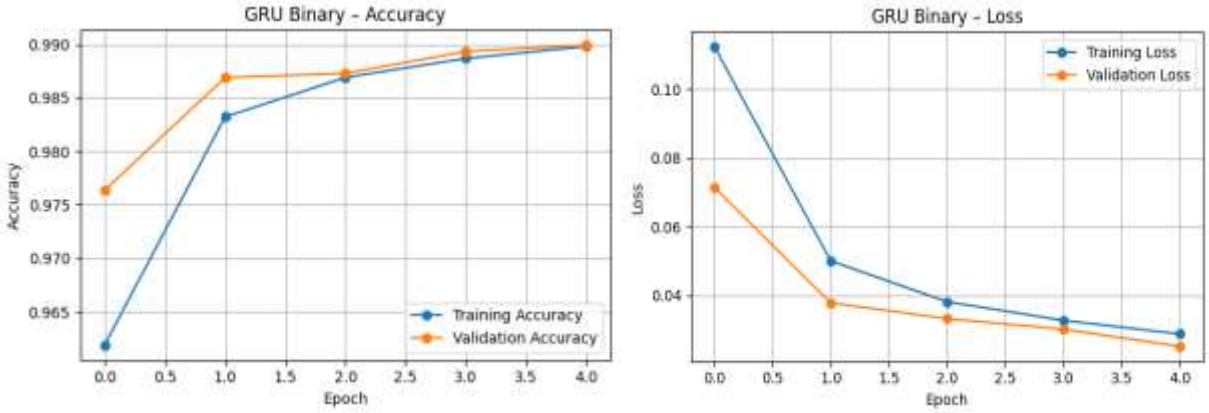


Figure 15: Training and validation accuracy and loss of the GRU binary classification model.

The training and validation performance of the GRU based multiclass classification model is provided in the figure 16 in terms of five epochs. The accuracy of training is continuously improving with an overall process of improving to a high level of about 59 percent at epoch 0 improving to about 75 percent at epoch 1, and 82 percent at epoch 2 and then finally 87 percent at epoch 4. The validation accuracy begins approximately at 66% and increases sharply to almost 88% at epoch 1 and slightly to approximately 91% at epoch 4 which means that there is good generalization to various classes. The loss curves show a steady decline whereby the training loss declines as well as validation loss declines respectively as 0.67 and 0.92. The small changes in validation loss indicate that there is variability in the classification per class, but the convergence in general proves the power of the GRU model to classify in multiclass classification.

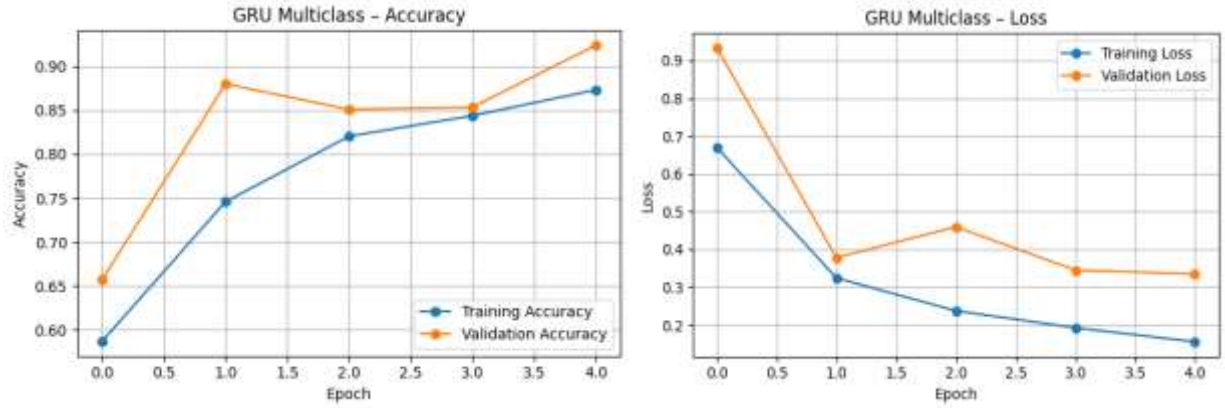


Figure 16: Training and validation accuracy and loss of the GRU multiclass classification model.

This figure 17 shows the results of the GRU-based model in binary and multiclass confusion matrices. The model is very accurate on detecting benign samples since it correctly classifies 394,420 samples and 63,836 attack samples in the binary confusion matrix. There is also a low level of misclassification (1,044 benign false positives and 3,462 attack false positives and false negatives). The multiclass confusion matrix presented in a logarithmic scale, illustrates the

behavior of the prediction by classes on a category of Benign, Botnet, Brute Force, DoS, Infiltration, PortScan, and WebAttack. Good diagonal dominance is detected especially in the Benign and DoS classes and this means that the classification is reliable. There is some confusion that is observed between the related types of attack such as Botnet, Brute force and PortScan, the overall misclassification is not very high, which confirms that the GRU model is beneficial and efficient in intrusion detection both binary and multi-class.

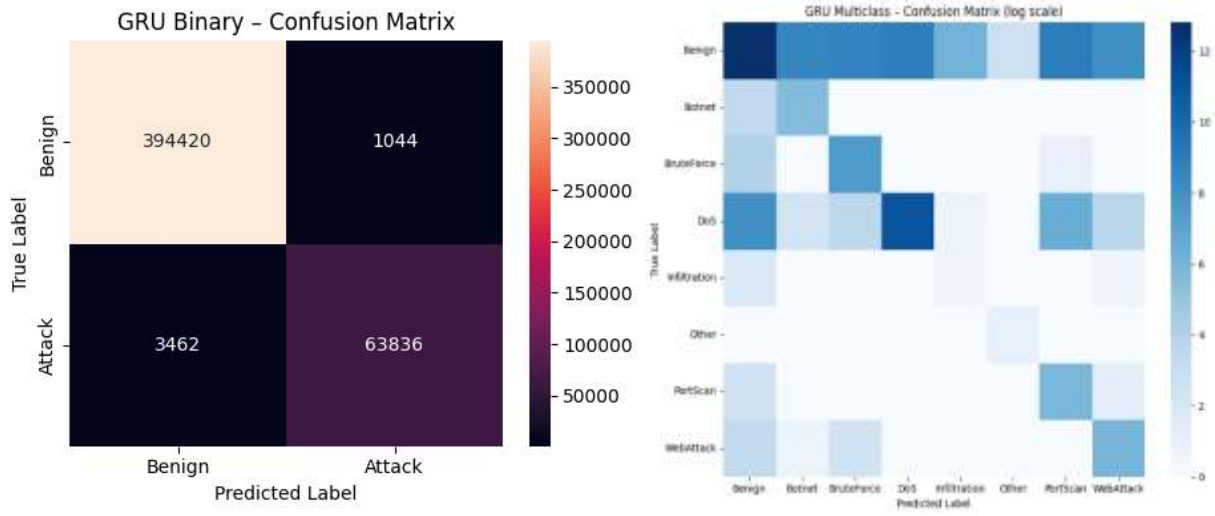


Figure 17: Binary and multiclass confusion matrices of the GRU model for intrusion detection.

Figure 18 shows how well the GRU-based binary classification model works using PR and ROC curves. The PR curve shows that the precision is high over almost all the recall range, with just a small drop near full recall, meaning it can reliably detect attack instances with very few false positives. The Average Precision score of 0.9955 means strong classification performance, especially under class imbalance. The ROC curve further illustrates this by giving an AUC of 0.9990, which is very close to the ideal value of 1; here, the curve stays concentrated near the top-left corner, indicating a high true positive rate at a very low false positive rate. Results confirm the robustness and effectiveness of the GRU model for binary intrusion detection tasks.

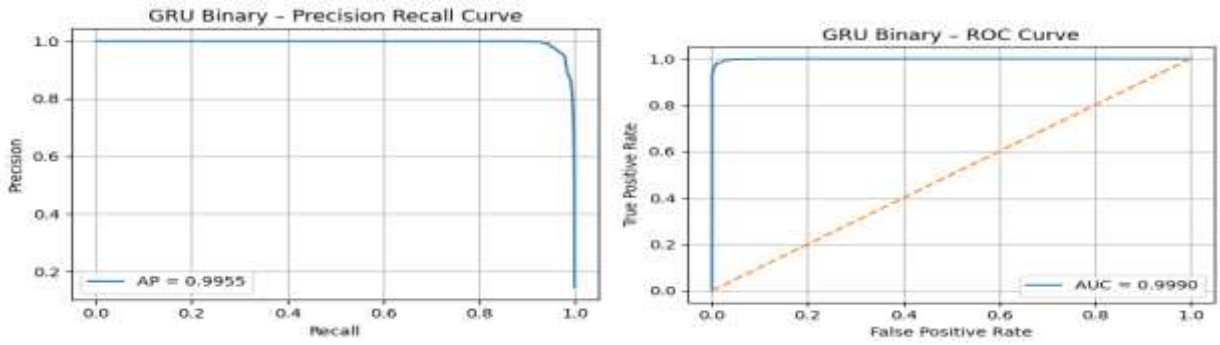


Figure 18: PR and ROC curves of the GRU binary classification model.

4.4 Hybrid Model of CNN+LSTM+GRU

The graph 19 shows the training and validation performance of the hybrid CNNLSTMGRU binary classification model in terms of loss and accuracy in five epochs. The loss curves indicate a quick decrease in training and validation loss values over the 4 epochs, training loss at epoch 0 is approximately 0.060 and at the 4th epoch it is approximately 0.023, and validation loss at epoch 0 is approximately 0.039 and at epoch 4 is approximately 0.020 which indicate that learning has been stabilized and overfitting is limited. The accuracy plot indicates steady improvement where the training accuracy grows by approximately 98.1 percent to almost 99.2 percent, and validation accuracy by approximately 98.7 percent to close to 99.3 percent across the training epochs. High generalization is emphasized by the fact that training and validation measures are closely correlated. In general, the hybrid CNN-LSTM-GRU model has high precision and the loss is low, and it proves to be effective in the task of binary classification.

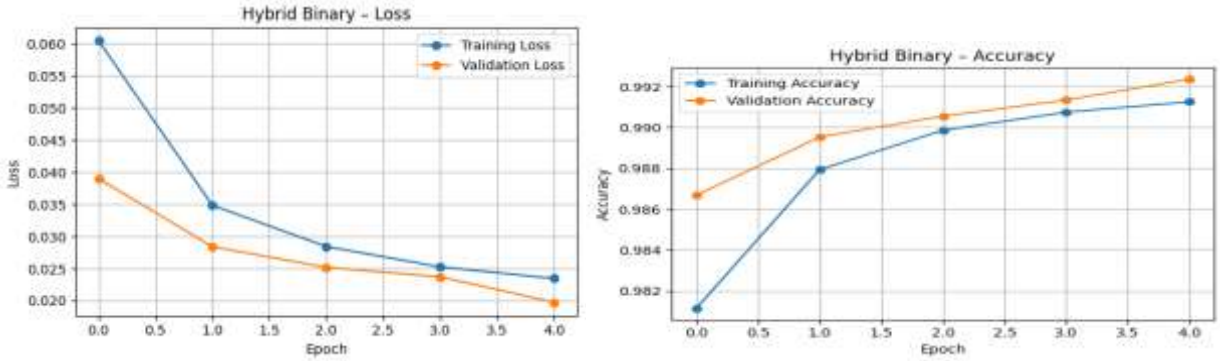


Figure 19: Training and validation accuracy and loss of the hybrid CNN–LSTM–GRU binary model.

The figure 20 presents the training and validation performance of the hybrid CNN–LSTM–GRU multiclass classification model across five epochs. The training accuracy shows a steady increase from approximately 73% at epoch 0 to about 86% at epoch 1, reaching nearly 89% at epoch 3 and stabilizing around 89–90% by epoch 4. Validation accuracy starts high at around 91%, peaks at approximately 95% at epoch 1, slightly decreases to 91% at epoch 2, and then recovers to nearly 93% at epoch 4, indicating good generalization across multiple classes. The loss curves demonstrate consistent convergence, with training loss decreasing from roughly 0.43 to 0.14, while validation loss reduces from about 0.33 to approximately 0.22. Minor fluctuations in validation loss suggest class-wise variability, but overall trends confirm effective learning and stable multiclass performance of the hybrid model.

International Journal of Computing and Engineering
ISSN 2958-7425 (online)
Vol. 8, Issue No. 1, pp 13 - 44, 2026
www.carijournals.org

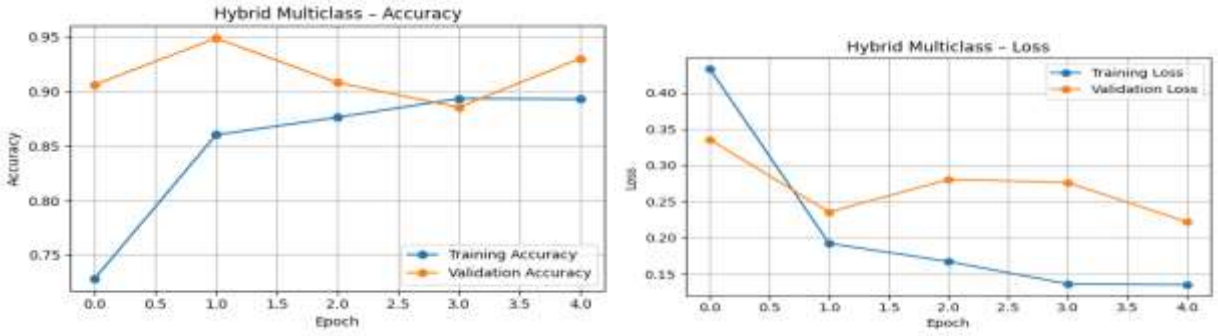


Figure 20: Training and validation accuracy and loss of the hybrid CNN–LSTM–GRU multiclass model.

This figure 21 shows the performance of the hybrid CNN-LSTM-GRU model on binary and multiclass confusion matrices. The model distinguishes correctly 394,924 benign samples and 64,367 attack samples in the binary classification matrix which means that it has high detection accuracy. There is also low misclassification with only 540 benign cases being wrongly predicted as attacks (false positives) and 2,931 attack cases being wrongly predicted as benign cases (false negatives). The multiclass confusion matrix, which is presented in the form of a logarithmic-scaled table, depicts great dominance of the diagonal among classes and Benign, Botnet, Brute Force, DoS, Infiltration, PortScan, and WebAttack, which are correct predictions made by classes. There are some high counts of correct classification in the DoS and Benign categories. Minimal confusion is witnessed between related types of attacks, particularly between Botnet and Brute Force, but generally the error rates are low, which proves the strength and stability of the hybrid model in binary and multiclass intrusion detection.

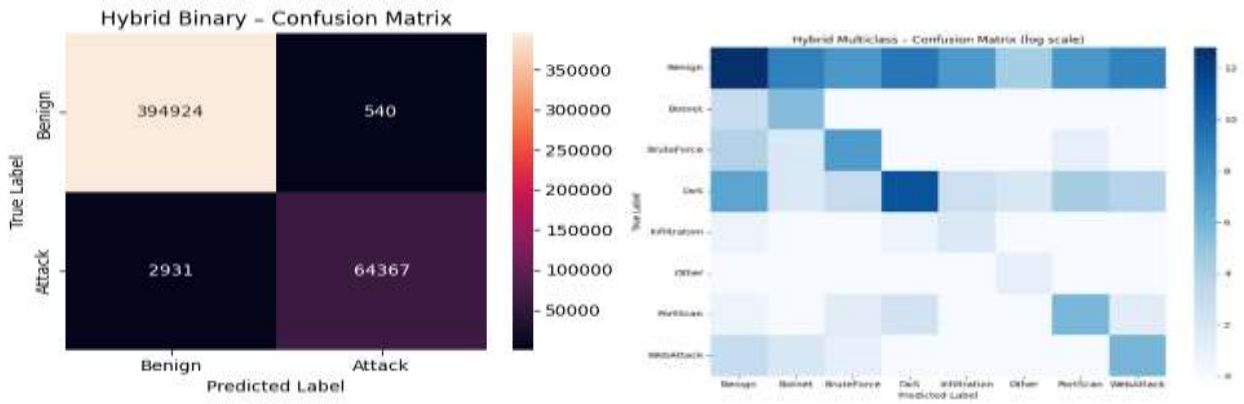


Figure 21: Binary and multiclass confusion matrices of the hybrid CNN–LSTM–GRU model.

The figure 22 provides the PR curve of the hybrid CNN-LSTM-GRU binary classification. Most recall values lie close to the upper limit of the curve thus meaning that the precision is also high over a large degree of recall. This performance shows that the model was highly accurate in detecting attack cases and it has a very low false positive. The aforementioned classifier results of 0.9972 on the reported Average Precision (AP) demonstrate that the classification is excellent, especially in the conditions of class imbalance. Minor loss in accuracy occurs only at test limits

of full recall which is common at detection sensitivity maximization. All in all, the PR curve shows that the hybrid model is robust and reliable in detecting binary intrusions.

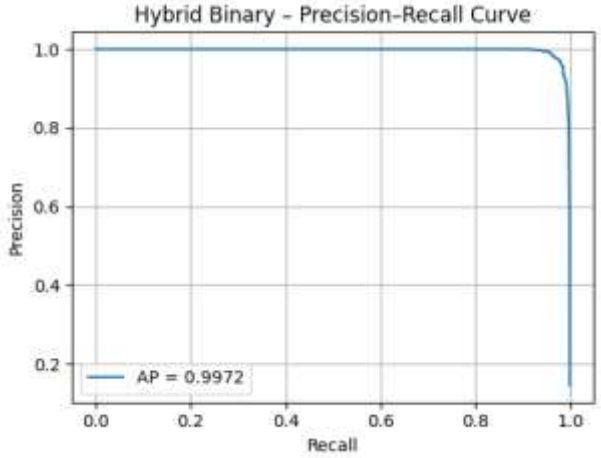
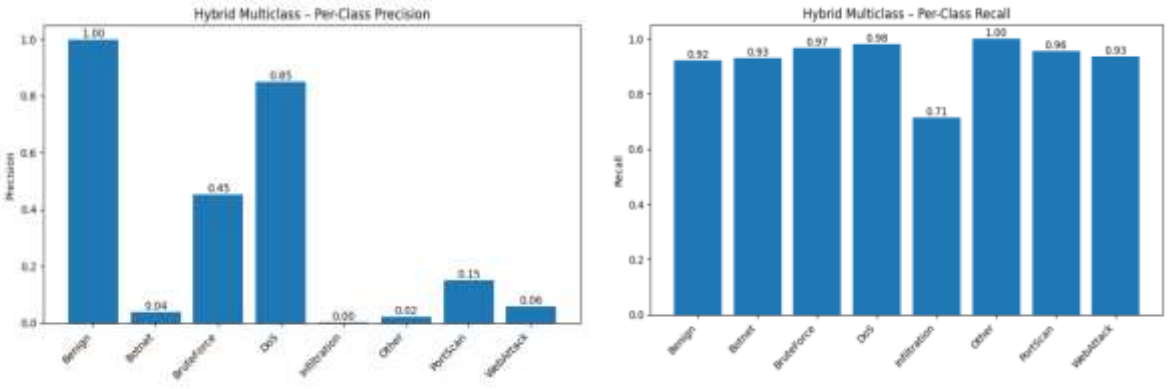


Figure 22: Precision–Recall curve of the hybrid CNN–LSTM–GRU binary model.

The chart 23 below shows how well the hybrid multiclass model did for each class using precision, recall, and F1-score. For precision, the Benign class had a perfect score of 1.00; DoS had 0.85 and Brute Force has moderate precision at 0.45. Botnet (0.04), PortScan (0.15), WebAttack (0.06), and very low values for Infiltration and Other have near zero values which means more false positives for these classes are indicated by lower precision values. Recall values are high with other (1.00), DoS (0.98), Brute Force (0.97), PortScan (0.96), and WebAttack (0.93) while Infiltration has relatively lower recall at 0.71; this is reflected in the F1-scores that show good balance with strong performances from Benign (0.96) and DoS (0.91), moderate from Brute Force (0.62), and lower scores from minority attack classes indicating effects of class imbalances.



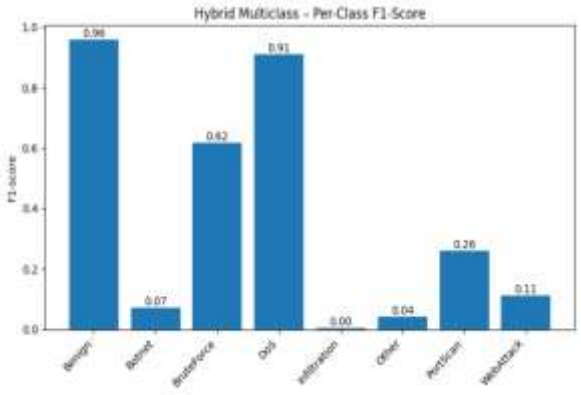


Figure 23: Per-class precision, recall, and F1-score of the hybrid multiclass model.

The given figure 24 represents the ROC curve for the binary classification CNN-LSTM-GRU hybrid model. The curve lies close to the top-left corner, thereby substantiating robust discrimination between the classes of benign examples and attack examples. The obtained Area Under Curve (AUC) is 0.9994, which is nearly equal to 1. It confirms near-excellent performance of classification. The high value of true positive rate is preserved even for a low value of false positive rate, thereby validating effective detection of attacks. The diagonal line in the figure denotes random classification. The large gap between the ROC curve of the developed model and the diagonal line for random classification ensures robustness of the model for binary classifications in intrusion detection.

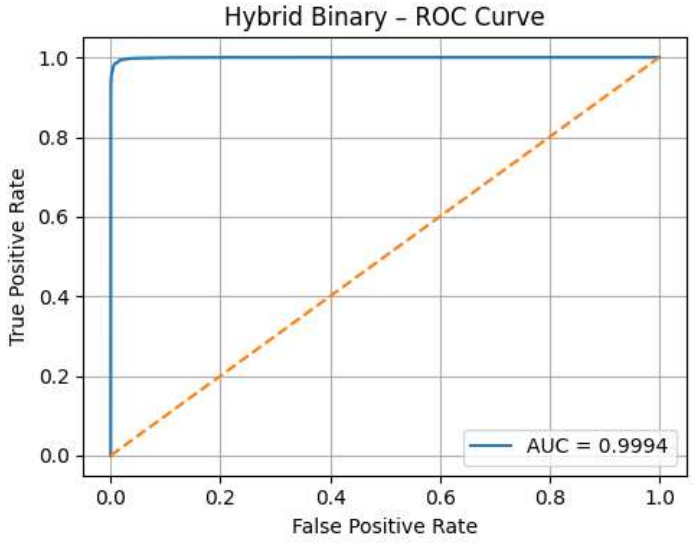


Figure 24: ROC curve of the hybrid CNN–LSTM–GRU binary classification model.

The bar graph 25 illustrates the evaluation of AUC values for different models employed in binary intrusion detection. The CNN model receives a very high score of 0.99931 in terms of AUC, denoting a great ability to differentiate. The LSTM model is not far behind with an AUC

of 0.99856, which means that there is still a strong but less powerful performance than that of the previous model. The GRU model, taking advantage of the LSTM model, the learning of time-based features, and thus being able to get an AUC of 0.99902, is a little stronger than LSTM. The hybrid CNN-LSTM-GRU model gets the highest AUC of 0.99941, beating all single architectures. The close spread of AUC values, which are all above 0.998, ensures that the performance of each model is excellent; however, the hybrid technique gives the most powerful classification with the best true positive and false positive balance. This comparison emphasizes the benefit of using a combination of spatial and temporal feature extraction for binary intrusion detection.

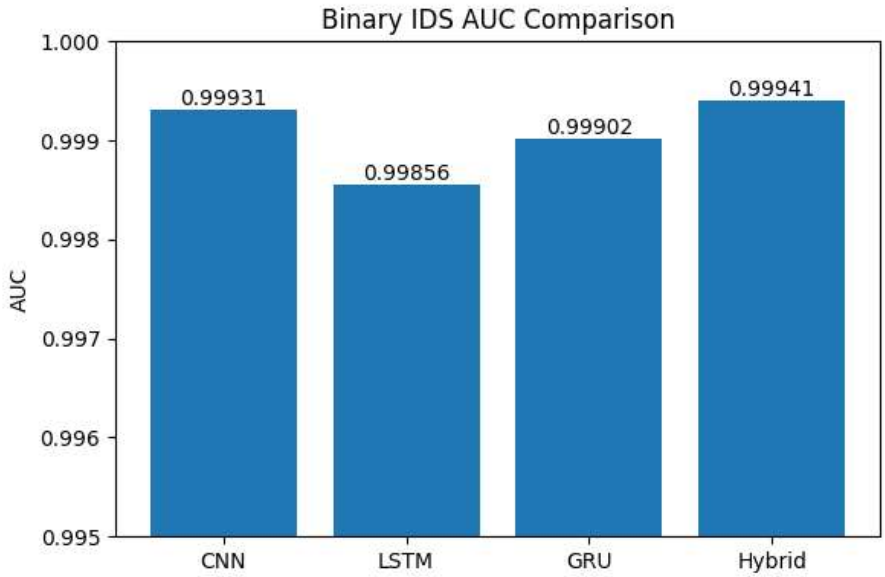


Figure 25: Comparison of AUC values for binary intrusion detection models.

The figure 26 presents model accuracies for binary and multiclass classification tasks in intrusion detection scenarios. For binary classification, model accuracies are nearly equal to 1, with CNN at 99.10%, LSTM at 98.95%, GRU at 99.00%, and the hybrid CNN-LSTM-GRU model achieving the highest accuracy of 99.24%. In the multiclass scenario, accuracy values are lower due to increased class complexity: CNN attains 91.93%, LSTM has the lowest multiclass accuracy of 80.28%, GRU reaches up to 92.36%, and again, the hybrid model leads all others with an accuracy of 93.35%. Results indicate both the hybrid approach's superiority as a practical choice and how different binary versus multiclass intrusion detection tasks affect performances.

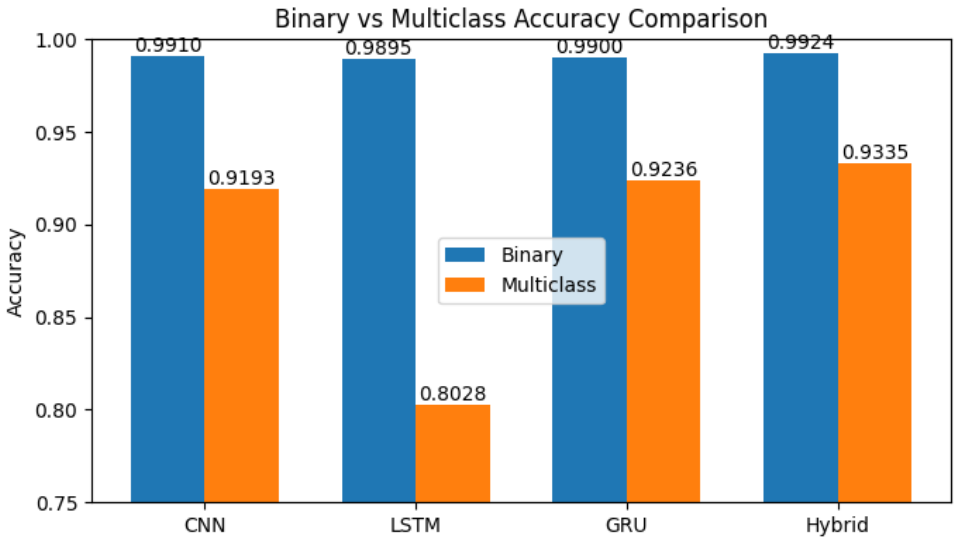


Figure 26: Binary and multiclass accuracy comparison of intrusion detection models.

The table 4 below presents a summary of recent intrusion detection studies by model, dataset, and accuracy. Santhadevi et al. (2023) applied a CNN–Stacked LSTM model to the NBaIoT dataset and achieved 97.39% accuracy, which demonstrates good feature learning for IoT traffic. Akinbolaji et al. (2024) used hybrid CNN and RNN architecture on the KDD Cup 1999 dataset with reported accuracy of 95%. Farzaan et al. (2025) implemented a Random Forest classifier on the NSL-KDD dataset to achieve 90% accuracy. Ethan et al. (2024) fused CNN, LSTM, and Transformer models using the CICIDS2017 dataset with an attained accuracy of 97.2%. The proposed CNN–LSTM–GRU model tested over the same CICIDS2017 dataset outperforms all previous proposals at a maximum recorded accuracy of 99.24%, proving its usability in intrusion detection applications. The figure illustrates the accuracy comparison of various intrusion detection models, with the CNN–LSTM–GRU model achieving the highest performance.

Table 4: Comparison of intrusion detection models and their classification accuracy.

Authors [Reference]	**Model**	**Datasets**	**Accuracy**
Authors (Year) [Ref.]	Model	Dataset	Accuracy (%)
Santhadevi et al. (2023) [45]	CNN–Stacked LSTM	NBaIoT	97.39
Akinbolaji et al. (2024) [46]	CNN and RNN	KDD Cup 1999	95.00
Farzaan et al. (2025) [47]	Random Forest	NSL-KDD	90.00
Ethan et al. (2024) [48]	CNN + LSTM + Transformer	CICIDS2017	97.20

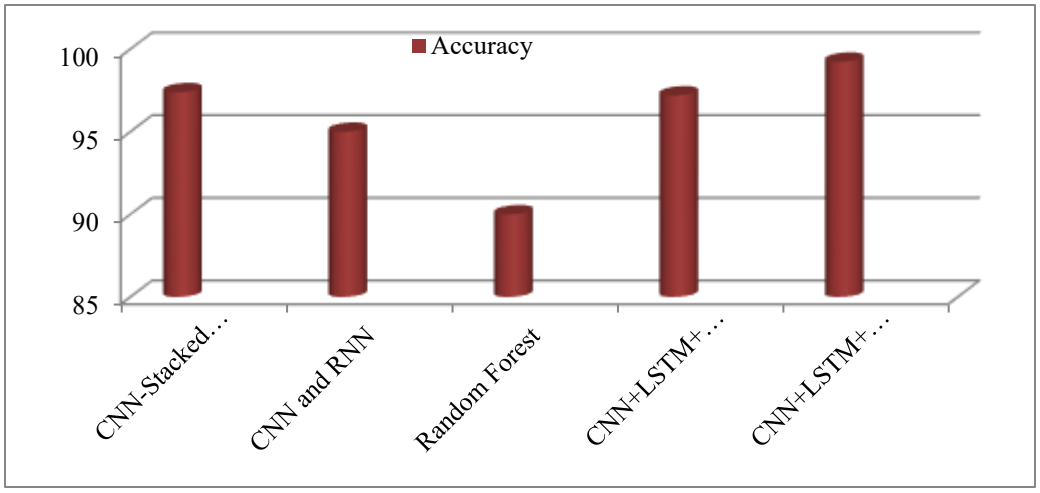


Figure 27: Comparison graph of different intrusion detection models.

5. Conclusion and Future Scope

The findings of this study provide important theoretical contributions to the field of cybersecurity and intelligent systems. From a practical perspective the proposed cloud edge deep learning framework offers a viable and deployable solution for real time cybersecurity threat detection in modern network environments. The study also carries important policy implications for cybersecurity governance and digital infrastructure regulation. Policymakers and regulatory bodies can leverage the findings to promote the adoption of artificial intelligence driven security mechanisms as part of national and organizational cybersecurity strategies. Based on the findings future research should focus on integrating federated learning techniques to enhance data privacy and reduce dependency on centralized data storage. Incorporating explainable deep learning methods would further improve transparency and trust in automated intrusion detection decisions. Additionally, evaluating the framework on more recent and diverse datasets as well as testing its performance in real world operational environments would strengthen its applicability and robustness. These enhancements would further position the proposed framework as a scalable trustworthy and adaptive solution for next generation cloud edge cybersecurity systems..

References

1. Sundaramurthy, Senthil Kumar, Nischal Ravichandran, Anil Chowdary Inaganti, and Rajendra Muppalaneni. "AI-Driven Threat Detection: Leveraging Machine Learning for Real-Time Cybersecurity in Cloud Environments." *Artificial Intelligence and Machine Learning Review* 6, no. 1 (2025): 23-43.
2. Ofoegbu, Kingsley David Onyewuchi, Olajide Soji Osundare, Chidiebere Somadina Ike, Ololade Gilbert Fakeyede, and Adebimpe Bolatito Ige. "Real-Time Cybersecurity threat detection using machine learning and big data analytics: A comprehensive approach." *Computer Science & IT Research Journal* 4, no. 3 (2024).

3. Dey, Saswata, Writuraj Sarma, and Sundar Tiwari. "Deep learning applications for real-time cybersecurity threat analysis in distributed cloud systems." *World Journal of Advanced Research and Reviews* 17, no. 3 (2023): 1044-1058.
4. Ameedeen, Mohamed Ariff, Rula A. Hamid, Theyazn HH Aldhyani, Laith Abdul Khaliq Mohammed Al-Nassr, Sunday Olusanya Olatunji, and Priyavahani Subramanian. "A framework for automated big data analytics in cybersecurity threat detection." *Mesopotamian Journal of Big Data* 2024 (2024): 175-184.
5. Jha, Krishna Madhav, Varun Bodepudi, Suneel Babu Boppana, Niharika Katnapally, Srinivasa Rao Maka, and Manikanth Sakuru. "Deep Learning-Enabled Big Data Analytics for Cybersecurity Threat Detection in ERP Ecosystems." (2023).
6. Santhadevi, D., and B. Janet. "Stacked deep learning framework for edge-based intelligent threat detection in IoT network." *The Journal of Supercomputing* 79, no. 11 (2023): 12622-12655.
7. Andrés, Pereira, Ivanov Nikolai, and Wang Zhihao. "Real-Time AI-Based Threat Intelligence for Cloud Security Enhancement." *Innovative: International Multi-disciplinary Journal of Applied Technology* 3, no. 3 (2025): 36-54.
8. Sarker, Iqbal H. "Machine learning for intelligent data analysis and automation in cybersecurity: current and future prospects." *Annals of Data Science* 10, no. 6 (2023): 1473-1498.
9. Biswas, Shaikat. "Artificial Intelligence–Enhanced Cybersecurity Frameworks for Real-Time Threat Detection In Cloud And Enterprise." *ASRC Procedia: Global Perspectives in Science and Scholarship* 1, no. 01 (2025): 737-770.
10. Kumar, Busireddy Hemanth, Sai Teja Nuka, Murali Malempati, Harish Kumar Sriram, Someshwar Mashetty, and Sathya Kannan. "Big Data in Cybersecurity: Enhancing Threat Detection with AI and ML." *Metallurgical and Materials Engineering* 31, no. 3 (2025): 12-20.
11. Sivaprasad Yerneni, K., A. Ravi Teja, K. Sri Harsha, and Y. Naresh Kiran Kumar Reddy. "Towards Proactive Cloud Security: A Survey on ML and Deep Learning-Based Intrusion Detection Systems." *J Contemp Edu Theo Artific Intel: JCETAI-116* (2025).
12. Okafor, Maureen Oluchukwuamaka. "Deep learning in cybersecurity: Enhancing threat detection and response." *World Journal of Advanced Research and Reviews* 24, no. 3 (2024): 1116-1132.
13. Sarker, Iqbal H. "Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective." *SN Computer Science* 2, no. 3 (2021): 154.
14. Okoli, Ugochukwu Ikechukwu, Ogugua Chimezie Obi, Adebunmi Okechukwu Adewusi, and Temitayo Oluwaseun Abrahams. "Machine learning in cybersecurity: A review of threat detection and defense mechanisms." *World Journal of Advanced Research and Reviews* 21, no. 1 (2024): 2286-2295.

15. Khaleel, Yahya Layth, Mustafa Abdulfattah Habeeb, A. S. Albahri, Tahsien Al-Quraishi, O. S. Albahri, and A. H. Alamoodi. "Network and cybersecurity applications of defense in adversarial attacks: A state-of-the-art using machine learning and deep learning methods." *Journal of Intelligent Systems* 33, no. 1 (2024): 20240153.
16. Tirulo, Aschalew, Siddhartha Chauhan, and Kamlesh Dutta. "Machine learning and deep learning techniques for detecting and mitigating cyber threats in IoT-enabled smart grids: a comprehensive review." *International Journal of Information and Computer Security* 24, no. 3-4 (2024): 284-321.
17. https://www.xenonstack.com/blog/deep-learning-in-cybersecurity
18. Santhadevi, D., and B. Janet. "Stacked deep learning framework for edge-based intelligent threat detection in IoT network." *The Journal of Supercomputing* 79, no. 11 (2023): 12622-12655.
19. Vankayalapati, Ravi Kumar. "Unifying Edge and Cloud Computing: A Framework for Distributed AI and Real-Time Processing." *Available at SSRN 5048827* (2023).
20. Lilhore, Umesh Kumar, Sarita Simaiya, Yogesh Kumar Sharma, Anjani Kumar Rai, S. M. Padmaja, Khan Vajid Nabilal, Vimal Kumar, Roobaea Alroobaea, and Hamed Alsufyani. "Cloud-edge hybrid deep learning framework for scalable IoT resource optimization." *Journal of Cloud Computing* 14, no. 1 (2025): 5.
21. Cao, Zhigang, Bo Liu, Dongzhan Gao, Ding Zhou, Xiaopeng Han, and Jiuxin Cao. "A Dynamic Spatiotemporal Deep Learning Solution for Cloud–Edge Collaborative Industrial Control System Distributed Denial of Service Attack Detection." *Electronics* 14, no. 9 (2025): 1843.
22. Awan, Kamran Ahmad, Ikram Ud Din, Ahmad Almogren, Ali Nawaz, Muhammad Yasar Khan, and Ayman Altameem. "SecEdge: A novel deep learning framework for real-time cybersecurity in mobile IoT environments." *Heliyon* 11, no. 1 (2025).
23. Khalaf, Noora Zidan, Israa Ibraheem Al Barazanchi, A. D. Radhi, Sushma Parihar, Pritesh Shah, and Ravi Sekhar. "Development of real-time threat detection systems with AI-driven cybersecurity in critical infrastructure." *Mesopotamian Journal of CyberSecurity* 5, no. 2 (2025): 501-513.
24. Hussen, Noha, Sally M. Elghamrawy, Mofreh Salem, and Ali I. El-Desouky. "A fully streaming big data framework for cyber security based on optimized deep learning algorithm." *IEEE Access* 11 (2023): 65675-65688.
25. Malik, Anum, Kaleem Arshid, Nooruddin Noonari, and Rizwan Munir. "Artificial Intelligence-Driven Cybersecurity Framework Using Machine Learning for Advanced Threat Detection and Prevention." *Sch J Eng Tech* 6 (2025): 401-423.
26. Farzaan, Mohammed AM, Mohamed Chahine Ghanem, Ayman El-Hajjar, and Deepthi N. Ratnayake. "AI-powered system for an efficient and effective cyber incidents detection and response in cloud environments." *IEEE Transactions on Machine Learning in Communications and Networking* (2025).

27. Ezeh Ebere, M., and T. C. Asogwa. "A DEEP LEARNING FRAMEWORK FOR REAL-TIME CYBER THREAT DETECTION AND MITIGATION IN NETWORKED ENVIRONMENTS." *International Journal Of Real-Time Applications And Computing Systems* 4, no. 1 (2025).
28. Adeniyi, Olusola, Ali Safaa Sadiq, Prashant Pillai, Mohammad Aljaidi, and Omprakash Kaiwartya. "Securing mobile edge computing using hybrid deep learning method." *Computers* 13, no. 1 (2024): 25.
29. Sathupadi, Kaushik, Sandesh Achar, Shinoy Vengaramkode Bhaskaran, Nuruzzaman Faruqui, M. Abdullah-Al-Wadud, and Jia Uddin. "Edge-cloud synergy for AI-enhanced sensor network data: A real-time predictive maintenance framework." *Sensors* 24, no. 24 (2024): 7918.
30. Areghan, Edoise, and Osondu Onwuegbuchi. "Predictive Cyber Threat Analysis in Cloud Platforms Using Artificial Intelligence and Machine Learning Algorithms." *Applied Sciences, Computing, and Energy* 1, no. 1 (2024): 197-205.
31. Saxena, Deepika, Ishu Gupta, Rishabh Gupta, Ashutosh Kumar Singh, and Xiaoqing Wen. "An AI-driven VM threat prediction model for multi-risks analysis-based cloud cybersecurity." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 53, no. 11 (2023): 6815-6827.
32. Al-Ghuwairi, Abdel-Rahman, Yousef Sharrab, Dimah Al-Fraihat, Majed AlElaimat, Ayoub Alsarhan, and Abdulmohsen Algarni. "Intrusion detection in cloud computing based on time series anomalies utilizing machine learning." *Journal of Cloud Computing* 12, no. 1 (2023): 127.
33. Tyagadurgam, Mukund Sai Vikram, Venkataswamy Naidu Gangineni, Sriram Pabbineedi, Mitra Penmetsa, Jayakeshav Reddy Bhumireddy, and Rajiv Chalasani. "Designing an Intelligent Cybersecurity Intrusion Identify Framework Using Advanced Machine Learning Models in Cloud Computing." *Universal Library of Engineering Technology* Issue (2022).
34. https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset
35. Khan, Attiya, Muhammad Rizwan, Ovidiu Bagdasar, Abdulatif Alabdulatif, Sulaiman Alamro, and Abdullah Alnajim. "Deep Learning-Driven Anomaly Detection for IoMT-Based Smart Healthcare Systems." *CMES-Computer Modeling in Engineering & Sciences* 141, no. 3 (2024).
36. Ni, C., and S. C. Li. "Machine learning enabled Industrial IoT Security: Challenges, Trends and Solutions. Journal of Industrial Information Integration." (2024).
37. Saini, Dinesh Kumar, Krishan Kumar, and Punit Gupta. "[Retracted] Security Issues in IoT and Cloud Computing Service Models with Suggested Solutions." *Security and Communication Networks* 2022, no. 1 (2022): 4943225.
38. Kilichev, Dusmurod, Dilmurod Turimov, and Wooseong Kim. "Next–generation intrusion detection for iot evcs: Integrating cnn, lstm, and gru models." *Mathematics* 12, no. 4 (2024): 571

39. Deore, Bhushan, and Surendra Bhosale. "Hybrid optimization enabled robust CNN-LSTM technique for network intrusion detection." *Ieee Access* 10 (2022): 65611-65622
40. Imrana, Yakubu, Yanping Xiang, Liaqat Ali, Adeeb Noor, Kwabena Sarpong, and Muhammed Amin Abdullah. "CNN-GRU-FF: a double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units." *Complex & Intelligent Systems* 10, no. 3 (2024): 3353-3370.
41. Noor, Ayman. "Cloud-Based Deep Learning for Real-Time URL Anomaly Detection: LSTM/GRU and CNN/LSTM Models." *Comput. Syst. Sci. Eng* 49 (2025): 259-286
42. Afraji, Doaa Mohsin Abd Ali, Jaime Lloret, and Lourdes Peñalver. "An Integrated Hybrid Deep Learning Framework for Intrusion Detection in IoT and IIoT Networks Using CNN-LSTM-GRU Architecture." *Computation* 13, no. 9 (2025): 222.
43. Dini, Pierpaolo, and Sergio Saponara. "Analysis, design, and comparison of machine-learning techniques for networking intrusion detection." Designs 5, no. 1 (2021): 9.
44. Dini, Pierpaolo, Andrea Begni, Stefano Ciavarella, Emiliano De Paoli, Giuseppe Fiorelli, Carmelo Silvestro, and Sergio Saponara. "Design and testing novel one-class classifier based on polynomial interpolation with application to networking security." IEEE Access 10 (2022): 67910-67924.
45. Santhadevi, D., and B. Janet. "Stacked deep learning framework for edge-based intelligent threat detection in IoT network." The Journal of Supercomputing 79, no. 11 (2023): 12622-12655.
46. Akinbolaji, TAIWO JOSEPH. "Advanced integration of artificial intelligence and machine learning for real-time threat detection in cloud computing environments." Iconic Research and Engineering Journals 6, no. 10 (2024): 980-991.
47. Farzaan, Mohammed AM, Mohamed Chahine Ghanem, Ayman El-Hajjar, and Deepthi N. Ratnayake. "AI-powered system for an efficient and effective cyber incidents detection and response in cloud environments." IEEE Transactions on Machine Learning in Communications and Networking (2025).
48. Ethan, Amelia, and Motohisa Osaka. "Deep Learning-Powered Cyber Threat Intelligence for Multi-Cloud Environments." (2024).