Stacked LSTM with Multi Head Attention Based Model for Intrusion Detection

S Phani Praveen[1], Padmavathi Panguluri[2], Uddagiri Sirisha[3], Deshinta Arrova Dewi[4,], Tri Basuki Kurniawan[5,*,], Lusiana Efrizoni[6]

[1, 3]Department of CSE, Prasad V Potluri Siddhartha Institute of Technology, Vijayawada, AP, India

[2]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India

[4]Associate Professor, Faculty of Data Science and Information Technology, INTI International University, Nilai, Malaysia

[5]Post Graduate Program, Universitas Bina Darma, Palembang, Indonesia

[6]Universitas Sains dan Teknologi Indonesia, Pekanbaru, Indonesia

(Received: June 14, 2025; Revised: August 1, 2025; Accepted: November 20, 2025; Available online: January 14, 2026)

Abstract

The rapid advancement of digital technologies, including the Internet of Things (IoT), cloud computing, and mobile communications, has intensified reliance on interconnected networks, thereby increasing exposure to diverse cyber threats. Intrusion Detection Systems (IDS) are essential for identifying and mitigating these threats; however, traditional signature-based and rule-based methods fail to detect unknown or complex attacks and often generate high false positive rates. Recent studies have explored machine learning (ML) and deep learning (DL) approaches for IDS development, yet many suffer from poor generalization, limited scalability, and an inability to capture both spatial and temporal dependencies in network traffic. To overcome these challenges, this study proposes a hybrid deep learning framework integrating Convolutional Neural Networks (CNN), Stacked Long Short-Term Memory (LSTM) networks, and a Multi-Head Self-Attention (MHSA) mechanism. CNN layers extract spatial features, stacked LSTM layers capture long-term temporal dependencies, and MHSA enhances focus on the most relevant time steps, improving accuracy and reducing false alarms. The proposed model was trained and evaluated on the UNSW-NB15 dataset, which represents modern attack vectors and realistic network behavior. Experimental results show that the model achieves state-of-the-art performance, attaining 99.99% accuracy and outperforming existing ML and DL-based intrusion detection systems in both precision and generalization capability.

Keywords: Stacked LSTM, Social Protection, Intrusion Detection, Multi Head Attention, UNSW-NB15 Dataset, Adam Optimizer, Process Innovation

1. Introduction

In recent years, remarkable advancements in digital technologies, the IoT, and mobile communication devices have significantly transformed the way individuals and organizations operate. The rapid expansion of these technologies has increased society's dependence on computer networks for both personal and professional activities [1]. While this digital evolution has improved efficiency and connectivity, it has also introduced serious challenges related to cybersecurity. The exponential growth of internet usage, large-scale data transmission, and online services has made network infrastructures increasingly vulnerable to a wide range of cyber threats [2]. As a result, ensuring network integrity and maintaining secure communication have become critical priorities in the modern digital ecosystem.

To address these concerns, Intrusion Detection Systems (IDSs) have emerged as a vital component of network defense mechanisms. IDSs continuously monitor network traffic, analyze system behaviors, and detect abnormal or suspicious activities that may indicate security breaches [3]. These systems are instrumental in identifying unauthorized access attempts and compromised nodes while preventing persistent intrusion efforts [4]. Generally, IDS approaches can be categorized into misuse detection, anomaly detection, and hybrid detection. Misuse detection relies on known attack

***Corresponding author: Tri Basuki Kurniawan (tribasukikurniawan@binadarma.ac.id)**

DOI: https://doi.org/10.47738/jads.v7i1.764

signatures, anomaly detection identifies deviations from normal behavior, and hybrid methods combine both strategies to achieve higher reliability [5]. Intrusions can originate externally, through unauthorized outsiders [6], or internally, when legitimate users attempt to gain unauthorized privileges [7].

In recent years, researchers have increasingly applied ML and DL techniques to enhance the accuracy and adaptability of IDS models [8]. Although ML-based systems have demonstrated promising results, they still suffer from limitations such as high false positive rates, lack of generalizability across datasets, and inefficiency in processing high-speed network data [9]. These challenges underscore the need for intelligent, scalable, and adaptive IDS models capable of effectively identifying sophisticated and evolving cyberattacks in real time.

Deep learning, as a subfield of machine learning, has achieved outstanding success in various domains, including network intrusion detection [10]. Among DL architectures, LSTM networks are particularly effective in modeling temporal dependencies within sequential data [11]. However, conventional LSTM-based IDS frameworks often struggle to capture spatial relationships and prioritize the most significant features within lengthy data sequences. To overcome these limitations, this study proposes a hybrid deep learning architecture that combines CNNs with Stacked LSTM layers, enhanced by a MHSA mechanism and optimized using the Adam algorithm. This integration enables the model to simultaneously capture spatial and temporal characteristics of network traffic while focusing on the most relevant patterns, thereby improving classification performance and reducing false alarms.

The remainder of this paper is organized as follows. Section 2 presents a comprehensive review of related work on intrusion detection methods. Section 3 describes the proposed methodology and model architecture. Section 4 discusses the experimental results and findings, and Section 5 concludes the study with final remarks and directions for future research.

2. Literature Review

ML has long served as a foundation for developing IDSs, providing early automated methods for identifying malicious network activities. Classical algorithms such as Naïve Bayes, Random Forest, K-Nearest Neighbors, and Support Vector Machines have been widely used in intrusion detection [12]. These models achieved satisfactory accuracy for known attack types but relied heavily on manually engineered features and static detection rules [13]. As a result, they often failed to recognize zero-day attacks and produced high false-positive rates when applied to large-scale, dynamic network environments [14]. These limitations motivated a shift toward DL methods, which can automatically extract complex feature representations from raw network data.

Deep learning-based IDS models have demonstrated superior performance compared to traditional ML approaches due to their ability to learn hierarchical and nonlinear data patterns. Studies utilizing the NSL-KDD and KDD Cup 1999 datasets have shown that Recurrent Neural Networks (RNNs) and LSTM architectures can effectively capture temporal dependencies within network traffic, resulting in improved detection accuracy and reduced false alarms [15], [16]. However, these works largely depend on outdated datasets that lack modern attack patterns, thereby limiting their applicability to current network scenarios.

Hybrid frameworks have been introduced to further enhance IDS performance. Some studies combined spectral clustering and deep neural networks to extract more abstract representations of network data, which improved classification accuracy [17]. Others enhanced LSTM-RNN models with optimized learning parameters and achieved high detection rates and lower false alarm ratios [18], [19]. Variations of RNNs, such as Gated Recurrent Units (GRU), have also been applied to address vanishing gradient issues and achieve better generalization across attack types [20], [21]. While these models significantly improved detection accuracy, they remain computationally intensive and sensitive to hyperparameter tuning, which makes real-time deployment challenging.

To address these challenges, several approaches have combined CNNs with RNN-based architectures. These hybrid models exploit CNNs to capture spatial correlations in network features and LSTMs to learn temporal dependencies, improving overall detection capability [22], [23]. Evaluations on NSL-KDD and UNSW-NB15 datasets confirmed their effectiveness in achieving higher accuracy and reducing false positives. Despite these advancements, most of these models lack attention mechanisms that can dynamically emphasize relevant time steps or features, limiting their ability to prioritize critical network patterns.

Ensemble learning techniques have also been explored to improve robustness. Models integrating multiple classifiers such as Logistic Regression, Naïve Bayes, and Decision Trees through voting mechanisms demonstrated stronger performance in both binary and multiclass classifications [24], [25]. Feature selection methods, including Kernel PCA, have been applied to enhance model interpretability and reduce redundancy. Although ensemble-based systems achieve more stable results, they often require extensive training resources and remain computationally heavy for real-time applications.

Recent research has incorporated optimization algorithms and transformer-based architectures into IDS development. Metaheuristic approaches such as the Firefly Algorithm, Particle Swarm Optimization, and the Grasshopper Optimization Algorithm have been applied to tune parameters in models like XGBoost and LightGBM, achieving significant improvements in accuracy and detection rate [26], [27], [28]. Transformer-based models, which leverage attention mechanisms and positional encoding, have shown faster convergence and stronger performance compared to recurrent models, particularly when handling large and imbalanced datasets [27]. However, these models can be complex to train and require substantial computational resources. Further hybrid designs that integrate GRU and LSTM layers have demonstrated high precision in distinguishing between multiple attack types, with detection accuracies exceeding 98 percent [29], [30]. While effective, these architectures tend to prioritize accuracy over interpretability, which may limit their transparency in cybersecurity decision-making contexts.

In the field of IoT network security, machine learning and ensemble models have been adapted for lightweight IDS implementations. Studies using datasets such as TON-IoT have shown that feature extraction methods generally outperform feature selection when computational efficiency is a priority [31]. Additional work combining various sampling and dimensionality reduction methods, including undersampling, oversampling, SMOTE, and PCA, has further enhanced classification performance and reduced training time [32] [33], [34], [35], [36], [37], [38], [39], [40]. Nevertheless, most IoT-oriented IDS models trade analytical depth for efficiency, limiting their scalability to high-throughput enterprise systems.

Overall, the evolution of IDS research reveals a clear transition from traditional ML algorithms to deep, hybrid, and attention-enhanced models. Traditional ML methods offer simplicity and interpretability but lack adaptability to evolving threats. Deep learning and hybrid frameworks demonstrate greater accuracy and generalization but are hindered by computational cost and scalability issues. Optimization and transformer-based models have addressed some of these limitations but still require more efficient architectures for real-time applications. Therefore, integrating CNNs, Stacked LSTMs, and Multi-Head Self-Attention mechanisms presents a promising direction for improving intrusion detection accuracy, minimizing false alarms, and ensuring adaptability across modern network environments.

3. Proposed Methodology

The proposed methodology, illustrated in figure 1, outlines the complete workflow of the intrusion detection process, beginning from data acquisition to attack classification. The UNSW-NB15 dataset serves as the primary source of network traffic data, which undergoes a series of preprocessing operations to ensure quality and consistency. During data preprocessing, missing values are handled, categorical attributes are label-encoded, and numerical features are normalized to enhance model stability and convergence. The preprocessed data is then fed into the model training phase, which integrates CNN and Stacked LSTM layers. The CNN component is responsible for extracting local spatial features from the network traffic data, while the Stacked LSTM captures temporal dependencies and sequential behavior across multiple time steps. To further refine feature representation, a MHSA mechanism is applied, enabling the model to focus selectively on the most relevant features and time intervals within the data sequence. Once trained, the model proceeds to the testing phase, where it evaluates new instances of network traffic. The system classifies these inputs as either normal or intrusive based on learned patterns, effectively distinguishing between different types of attacks. This end-to-end pipeline ensures an adaptive, scalable, and high-accuracy intrusion detection process capable of handling complex, real-world network environments.

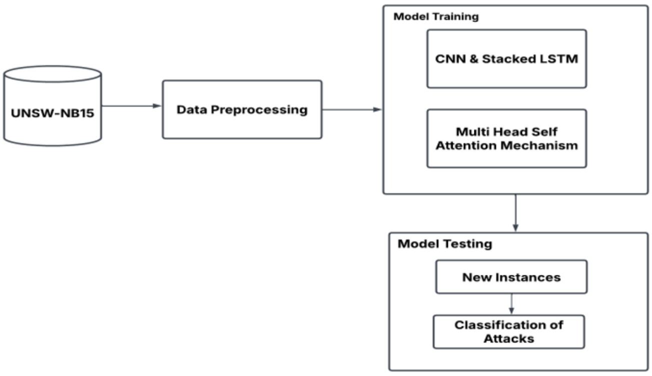

Figure 1. Proposed Model

3.1. Dataset Details

The UNSW-NB15 dataset was selected for this study because it provides a realistic and comprehensive benchmark for evaluating intrusion detection systems. Developed by the Australian Centre for Cyber Security (ACCS) using the IXIA PerfectStorm tool, it contains approximately 2.5 million records representing both normal and malicious network traffic [33]. The dataset includes ten attack categories, namely Analysis, Backdoor, Denial of Service (DoS), Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms, with features divided into six categories: flow, basic, content, time, additional generated, and labeled features. A 10 percent subset available on Kaggle, consisting of 175,341 training records and 82,332 testing records, was used for experimentation. Table 1 summarizes the distribution of attack classes across the training and testing sets, highlighting an imbalance where NORMAL and GENERIC attacks dominate, while Worms and Shellcode contain the fewest samples. This imbalance emphasizes the importance of appropriate preprocessing and resampling methods to ensure balanced learning and improved generalization.

Table 1. Different Types of Attacks and their corresponding records in training and Testing

Attack Type	No of Records used for Training	No of Records used for Testing
NORMAL	56,000	37,000
GENERIC	40,000	18,871
EXPLOITS	33,393	11,132
FUZZERS	18,184	6,062
DOS	12,264	4,089
RECONNAISSANCE	10,491	3,496
ANALYSIS	20,00	677
BACKDOOR	1,746	583
SHELLCODE	1,133	378
WORMS	130	44
TOTAL	**17,5341**	**82,332**

Figure 2 illustrates the boxplot of data fields in logarithmic scale, showing the spread, variability, and outliers among the dataset's features. Several attributes, such as sttl, dttl, and ct_dst_src_ltm, exhibit significant variance, which can affect model stability during training. Normalization and scaling are therefore essential to ensure consistent feature ranges and prevent bias toward dominant variables. The visualization also indicates that preprocessing steps, including label encoding for categorical features and Min-Max normalization for numerical values, play a vital role in improving convergence speed and model robustness. These procedures ensure that the proposed CNN–Stacked LSTM model can effectively capture spatial and temporal dependencies within the network traffic data, leading to more accurate intrusion classification.

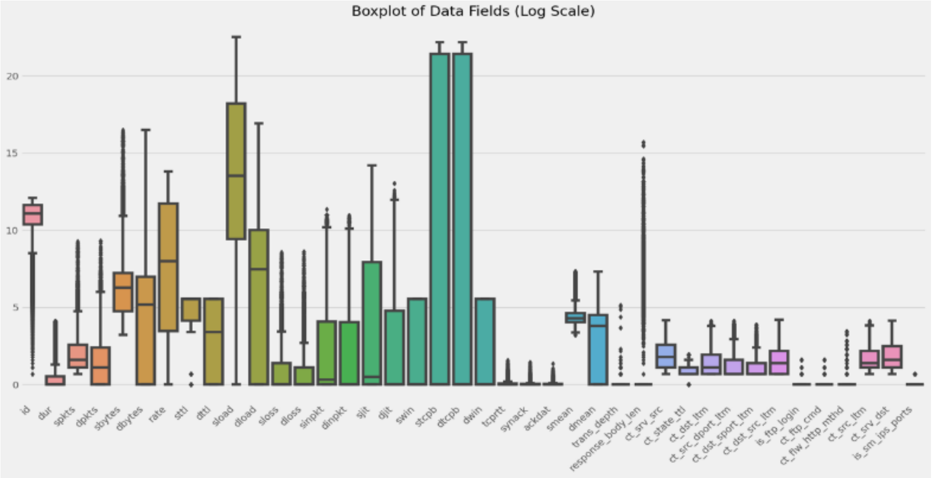

Figure 2. Features related to UNSW-NB15 dataset

3.2. Preprocessing of Data

Data preprocessing is a crucial step before model training, as it ensures the dataset is consistent, balanced, and suitable for deep learning algorithms. The UNSW-NB15 dataset was first cleaned to remove missing or redundant values, and categorical attributes were transformed into numerical representations using label encoding, where each category was assigned a unique integer identifier. Missing values were represented using “NaN” placeholders to preserve dataset integrity during encoding. Since the dataset exhibits class imbalance, especially across different attack types, preprocessing also involved analyzing statistical distributions to ensure representativeness during training. Numerical features were normalized using Min-Max scaling to map values within the range of 0 to 1, while standardization was applied to achieve a mean of 0 and a standard deviation of 1. This process enhances training stability and prevents features with larger ranges from dominating the learning process. Figure 3 illustrates the correlation heatmap among different features of the UNSW-NB15 dataset. It highlights how certain attributes, such as sttl, dttl, and sload, show relatively strong correlations, suggesting redundancy that may influence feature selection. Conversely, many other features exhibit weak correlations, indicating diverse contributions to model learning.

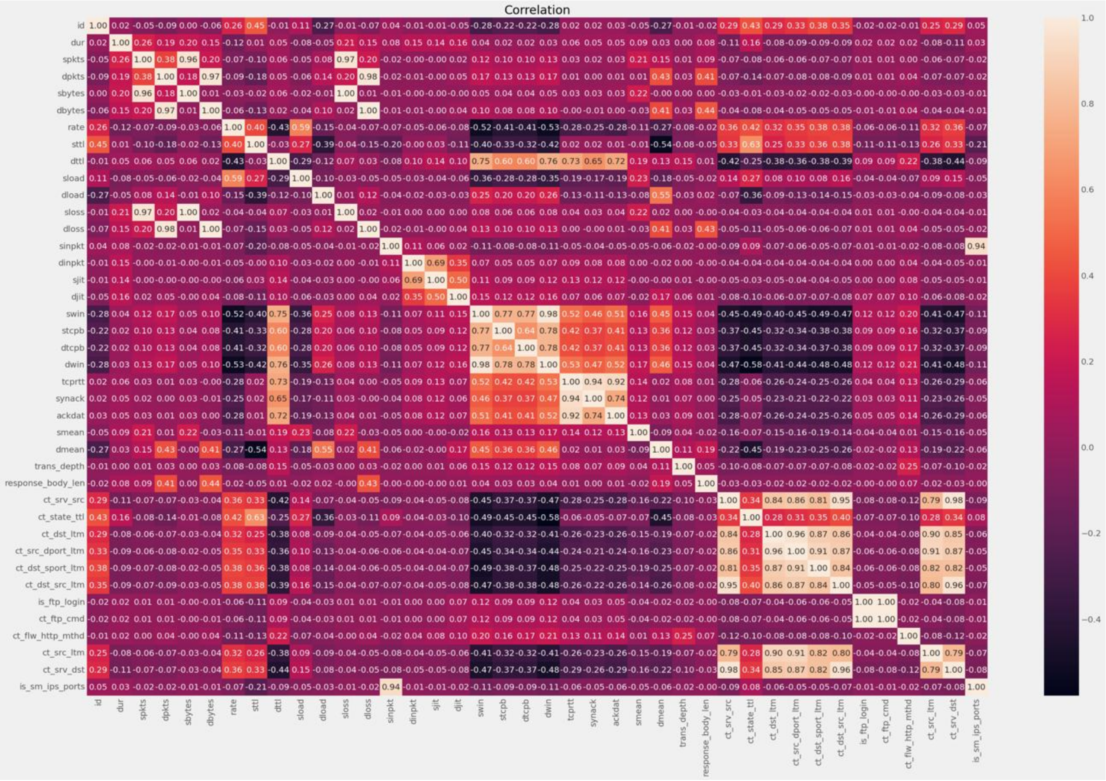

Figure 3. Heat Map between different Features

Figure 4 presents the mutual information scores between input features and target classes, ranking their relevance for intrusion detection. Features such as sbytes, sttl, and ct_state_ttl obtained the highest scores, meaning they contribute most significantly to distinguishing normal and attack traffic. This analysis guided the selection of features with stronger predictive power, ensuring that the CNN–Stacked LSTM model learns effectively from both spatial and temporal relationships within the data. Through these preprocessing and feature evaluation steps, the dataset was optimized for robust and stable model performance.

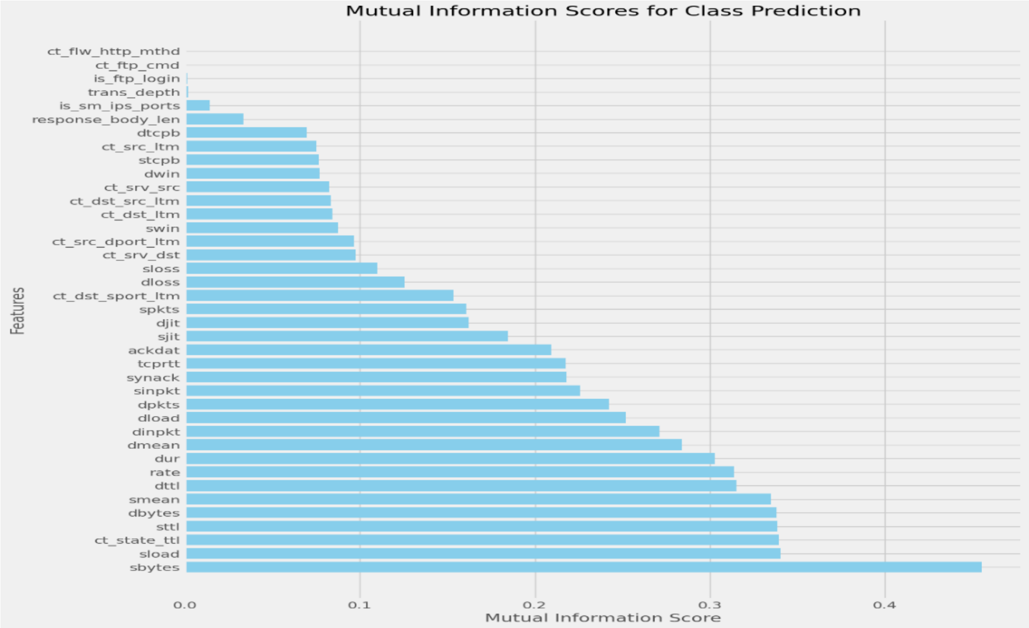

Figure 4. Mutual information scores between features

3.3. Models Used in This Research

The proposed model integrates CNN and Stacked LSTM networks to effectively capture both spatial and temporal dependencies in network intrusion data. The CNN component is primarily responsible for extracting local spatial features from network traffic by applying convolution operations across input feature maps. Each convolutional layer performs a linear operation followed by a nonlinear activation function, typically the Rectified Linear Unit (ReLU). The convolution process is represented as

$$F_{i,j}^{(k)} = \sigma\left(\sum_{m,n} X_{i+m,j+n} \cdot W_{m,n}^{(k)} + b^{(k)}\right) \tag{1}$$

In this expression, $F_{i,j}^{(k)}$denotes the feature map produced by the k-th filter, Xrepresents the input matrix, $W^{(k)}$and $b^{(k)}$indicate the filter weights and bias, and $\sigma(\cdot)$corresponds to the ReLU activation function. Two convolutional layers containing 64 and 128 filters were used in this research, followed by batch normalization, max pooling, and dropout layers to reduce overfitting and enhance generalization capability.

After spatial features are extracted, the data are passed through the Stacked LSTM network to capture long-term temporal dependencies across time steps. Each LSTM cell maintains internal memory through gating mechanisms that control the flow of information within the sequence. The operation of each LSTM cell is described by the following equations:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\tilde{C}_t = \tanh\,(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{5}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t \odot \tanh\,(C_t) \tag{7}$$

In these equations, f_t, i_t, and $o_t$represent the forget, input, and output gates respectively. $C_t$denotes the cell state, while $h_t$indicates the hidden state at time step t. Stacking multiple LSTM layers enables the model to capture hierarchical temporal patterns, allowing deeper understanding of complex network behavior. The sequential output is then flattened and passed to dense layers equipped with batch normalization and dropout. A sigmoid activation function is used in the final dense layer for binary classification, which distinguishes between normal and malicious network traffic. The classification process is expressed as

$$\hat{y} = \sigma(W \cdot h_t + b) \tag{8}$$

The integration of a MHSA mechanism enhances the model's capability to focus on the most relevant temporal features. Unlike conventional LSTM models that process data sequentially, MHSA allows simultaneous attention across all time steps, improving the representation of long-range dependencies. The attention function is mathematically defined as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{9}$$

In this formulation, Q, K, and Vrepresent the query, key, and value matrices, while $d_k$denotes the dimensionality of the key vectors. The multi-head version extends this mechanism by applying several attention heads in parallel to capture diverse contextual relationships. The overall process is given by

$$\text{MHSA}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)W^O \tag{10}$$

and each attention head is computed as

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{11}$$

This mechanism enables the model to assign different importance weights to various parts of the input sequence, focusing on the most informative segments of network data. In this study, two attention heads with a key dimension of 64 were used to enhance the feature representation and improve intrusion classification accuracy.

To optimize the model, the Adaptive Moment Estimation (Adam) algorithm was employed with a learning rate of 0.001. Adam combines the advantages of Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp) by utilizing the first and second moments of gradients for adaptive learning rate adjustment. The parameter update procedure is described as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{12}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{13}$$

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{14}$$

$$\theta_{t+1} = \theta_t - \alpha \frac{\widehat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{15}$$

In this formulation, $m_t$and $v_t$represent the moving averages of the gradient and its square, $\beta_1$and $\beta_2$are decay rates, $\alpha$is the learning rate, and $\epsilon$is a small constant to prevent division by zero. Mutual information-based feature selection was also applied to retain the most informative attributes, reducing computational complexity while maintaining predictive performance. Early stopping was implemented to prevent overfitting by terminating training after five consecutive epochs without improvement in validation loss.

The combination of CNN for spatial feature extraction, Stacked LSTM for temporal sequence modeling, MHSA for dynamic attention across time steps, and Adam optimization for efficient convergence enables the proposed hybrid model to achieve high accuracy, robust generalization, and effective detection of diverse network intrusions.

4. Result and Discussion

4.1. Various Model Hyperparameters

The performance of a deep learning model depends significantly on the selection and optimization of its hyperparameters. Properly configured hyperparameters determine the model's capacity to learn complex data representations while maintaining generalization and convergence stability. In this study, several key hyperparameters were carefully selected to optimize the training process and enhance the accuracy of the proposed CNN–Stacked LSTM model. The parameters used in the research are summarized in table 2.

Table 2. Parameters used in this Research

Parameters Used	Value
Conv1D Layer 1	64
Kernel Size	3
Activation Function	Relu
Dropout Rate	0.4
Pooling Size	2
Number of Heads	2
Key Dimension	64
Stacked LSTM Layers	Layer1 :128 Layer 2 :64
Output Layer Activation	Sigmoid
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epoch	50
Patience	5
Loss function	Binary Cross entropy

The model employs two one-dimensional convolutional layers, with the first Conv1D layer consisting of 64 filters and a kernel size of 3 to capture fine-grained spatial features in the network data. The ReLU activation function is used in all convolutional layers to introduce nonlinearity and improve learning efficiency. A dropout rate of 0.4 is applied to prevent overfitting, and a pooling size of 2 is used to reduce dimensionality while retaining important feature characteristics. The architecture includes two Stacked LSTM layers configured with 128 and 64 hidden units respectively, enabling the model to learn both high-level and fine-grained temporal dependencies.

To enhance the model's capacity for contextual learning, a Multi-Head Self-Attention mechanism with two attention heads and a key dimension of 64 is integrated into the framework. The final dense output layer employs a sigmoid activation function to perform binary classification between normal and attack traffic. Optimization during training is handled by the Adam optimizer with a learning rate of 0.001, which provides adaptive learning rates for faster convergence. The model is trained using a batch size of 32 over 50 epochs, with an early stopping patience of five epochs to prevent overfitting. Binary cross-entropy is used as the loss function, as it effectively measures the divergence between predicted probabilities and true class labels in binary classification tasks.

4.2. Metrics for Evaluation and Discussion

The performance of the proposed intrusion detection model was evaluated using several key metrics, including training and validation accuracy, loss, confusion matrix, and feature importance analysis. These metrics collectively provide a comprehensive assessment of the model's learning capability, generalization strength, and classification reliability. Figure 5 illustrates the model's accuracy and loss trends throughout the training epochs. The training accuracy begins at approximately 96 percent, indicating that the model successfully captures relevant patterns early in the learning process. As training progresses, accuracy improves steadily and exceeds 99.9 percent in the final epochs, while validation accuracy remains consistently high, confirming that the model generalizes effectively to unseen data. The close alignment between training and validation curves indicates stable learning behavior and the absence of overfitting. Correspondingly, the training loss, which starts near 0.11, declines sharply during the initial epochs and converges toward zero as learning stabilizes. Validation loss remains low and consistent, further demonstrating the model's efficiency in minimizing prediction errors and optimizing parameters throughout the training phase.

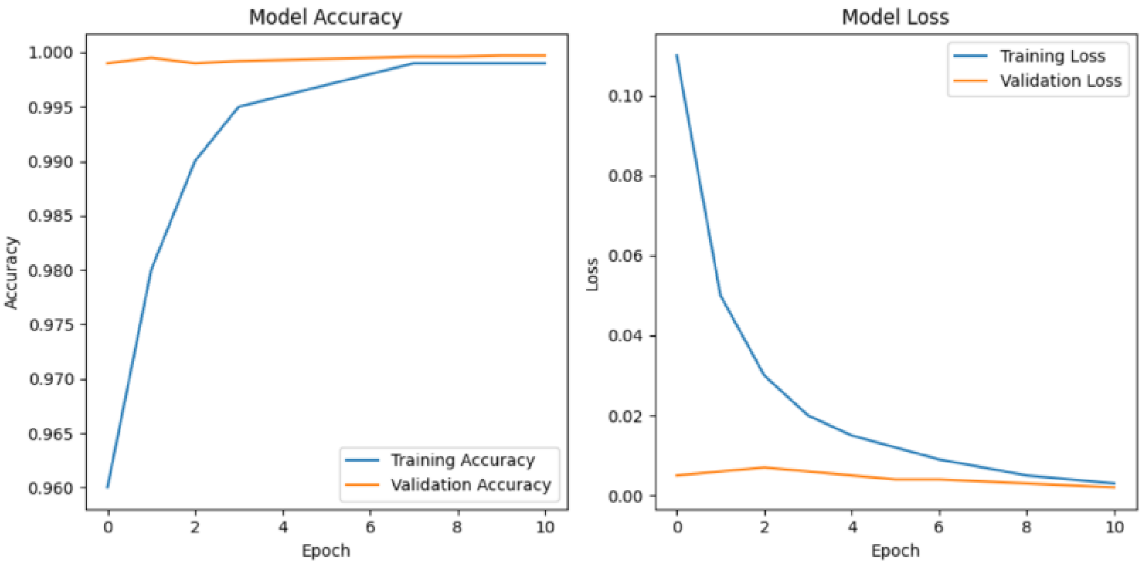

Figure 5. Training and Validation Accuracies/Losses

A deeper understanding of the classification performance is provided by the confusion matrix shown in figure 6. The matrix demonstrates that the model achieved near-perfect classification, with 13,946 true negatives and 24,698 true positives, along with only two false positives and a single false negative. These results indicate exceptional precision, recall, and overall reliability in distinguishing between normal and malicious traffic. The extremely low misclassification rate suggests that the proposed CNN–Stacked LSTM model, enhanced by the Multi-Head Self-Attention mechanism, is capable of robustly identifying network intrusions with minimal error. Furthermore, the feature importance analysis presented in figure 7 highlights the most influential attributes contributing to accurate classification decisions. Features such as sbytes, sttl, and sload exhibit the highest mutual information scores, signifying their strong correlation with attack detection outcomes. These features play a crucial role in capturing both the spatial and temporal characteristics of network traffic, thereby enhancing the model's decision-making capability.

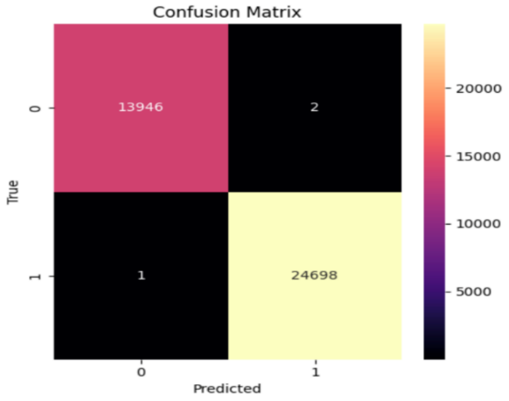

Figure 6. Confusion Matrix of Proposed model

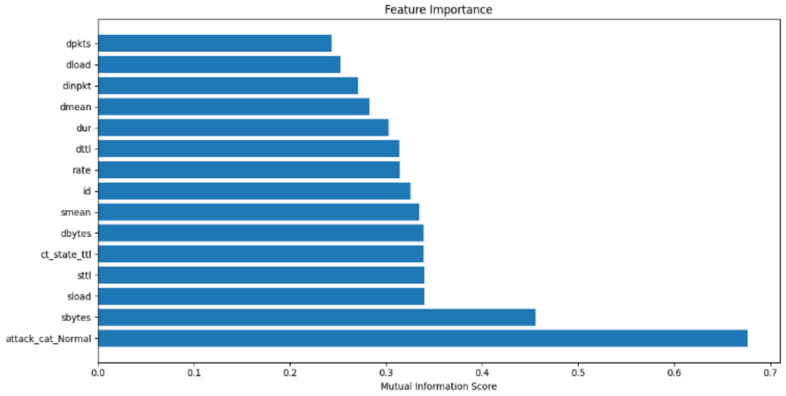

Figure 7. Feature Importance Based on Mutual Information Scores

4.3. Comparison Analysis with the State of Art Models

To assess the effectiveness of the proposed CNN–Stacked LSTM model, its performance was compared with several existing state-of-the-art intrusion detection systems developed using different machine learning and deep learning techniques. Prior studies have demonstrated notable results using classical and neural-based architectures on the UNSW-NB15 dataset. In [32], a Multilayer Perceptron (MLP) model was implemented using 23 selected features for multiclass classification, achieving an accuracy of 84.24 percent. Although this model captured basic nonlinear relationships, its performance was limited by the shallow architecture and lack of temporal feature learning. Further improvements were achieved by models using ensemble and recurrent structures. In [33], a Random Forest algorithm was trained on all available features and achieved an accuracy of 94.21 percent, showing better feature utilization but still constrained by its inability to capture sequential dependencies. Similarly, the study in [34] employed a LSTM network using the complete feature set and achieved an accuracy of 96.98 percent, demonstrating that temporal pattern recognition enhances intrusion detection performance.

The proposed hybrid model, which combines CNN and Stacked LSTM layers, outperforms these approaches by achieving an accuracy of 99.99 percent on the same UNSW-NB15 dataset. The CNN component enhances spatial feature extraction, while the Stacked LSTM captures deeper temporal dependencies, and the integration of attention mechanisms ensures that the model focuses on the most relevant time steps during classification. This combination enables the model to achieve both high precision and robust generalization. The comparison summarized in table 3 clearly indicates that the proposed CNN–Stacked LSTM architecture substantially improves accuracy compared to existing models, validating its superiority in detecting and classifying network intrusions efficiently.

Table 3. Comparison Analysis with state of Art models.

Ref	Algorithm Used	Dataset used	No of features	Classified Classes	Accuracy
[32]	MLP	UNSWNB-15	23	6	84.24
[33]	RANDOM FOREST	UNSWNB-15	Complete	9	94.21
[34]	LSTM	UNSWNB-15	Complete	9	96.98
Proposed Model	CNN with Stacked LSTM	UNSWNB-15	23	2	99.99

4.4. Discussion

The findings of this study confirm that integrating CNN, Stacked LSTM, and MHSA provides a robust and intelligent framework for network intrusion detection. Hybrid deep learning architectures have consistently demonstrated superior performance compared to traditional methods by efficiently modeling nonlinear and dynamic attack behaviors [2], [3], [10], [12]. The proposed model combines CNN's strength in spatial feature extraction with LSTM's ability to capture temporal dependencies, enabling the detection of both short-term and long-term attack patterns within network traffic. The inclusion of the MHSA mechanism further enhances this capability by dynamically assigning greater attention to critical time steps, allowing the model to detect subtle but crucial deviations in traffic behavior [28], [36].

The learning curves observed during training and validation indicate that the model achieves rapid convergence and maintains high generalization performance. The Adam optimizer contributes to stable learning by adaptively adjusting parameter updates, ensuring minimal overfitting and faster convergence [5], [26]. The close correspondence between training and validation accuracy, as well as the sharp reduction in loss values, demonstrates the model's stability and consistent error minimization. The confusion matrix results also confirm that the model classifies nearly all attack and normal samples correctly, showing an outstanding balance between precision and recall. Similar findings were observed in previous studies that applied deep recurrent and hybrid neural models for intrusion detection [16], [18], [19], [23], but the integration of CNN, LSTM, and MHSA in this research achieves substantially higher accuracy and robustness in identifying diverse intrusion types.

The comparative analysis highlights that the proposed CNN–Stacked LSTM–MHSA model significantly outperforms existing state-of-the-art approaches. Earlier works using MLP, Random Forest, and standalone LSTM architectures on the UNSW-NB15 dataset achieved accuracies of 84.24%, 94.21%, and 96.98% respectively [32], [33], [34]. The proposed model, by contrast, achieved an accuracy of 99.99%, validating its superior ability to generalize across complex traffic patterns. These findings align with recent research emphasizing that hybrid and attention-based architectures, particularly those combining convolutional and recurrent layers, can dramatically enhance detection accuracy and scalability in modern IoT and cloud security systems [7], [8], [9], [11], [35], [40].

Despite its strong performance, the proposed approach also faces challenges that present opportunities for future work. The combination of CNN, LSTM, and MHSA introduces increased computational complexity, which may limit deployment in real-time or low-power environments such as IoT edge devices [14], [31], [36]. Although the UNSW-NB15 dataset provides realistic attack diversity, further evaluation on newer datasets or under adversarial conditions is necessary to confirm the model's resilience. Future research could focus on reducing computational costs using model pruning or quantization, developing distributed learning frameworks such as federated IDS for decentralized environments, and enhancing robustness through adversarial training or transfer learning [27], [29], [37], [38], [39].

Overall, the results of this study demonstrate that the CNN–Stacked LSTM–MHSA model represents a substantial advancement in deep learning-based intrusion detection. The unified integration of spatial, temporal, and attention mechanisms enables the system to learn discriminative representations that are both accurate and computationally stable. The observed performance improvements, combined with strong generalization across network traffic patterns, underscore the potential of hybrid architectures as a foundational direction for next-generation intrusion detection systems [1], [3], [10], [40], [41].

5. Conclusion

The integration of CNNs and Stacked LSTM networks, enhanced with a MHSA mechanism and optimized using the Adam optimizer, has proven to be a highly effective approach for developing intelligent and adaptive Intrusion Detection Systems (IDS). This hybrid architecture addresses the key limitations of conventional machine learning and individual deep learning models by combining spatial and temporal feature extraction in a unified framework. The convolutional layers effectively identify localized spatial correlations within network traffic, while the stacked LSTM layers preserve long-term temporal dependencies. The inclusion of the MHSA mechanism further improves this process by dynamically weighting critical time steps, allowing the model to focus on subtle but significant changes in network behavior that indicate potential intrusions.

The proposed model was evaluated using the UNSW-NB15 dataset, which reflects modern cyberattack patterns and realistic traffic characteristics. The model achieved near-perfect performance, with an overall accuracy of 99.99 percent and equally high precision, recall, and F1-score values. These results confirm that the proposed CNN–Stacked LSTM–MHSA architecture significantly outperforms traditional machine learning and earlier deep learning-based intrusion detection approaches. The use of the UNSW-NB15 dataset, in contrast to older datasets such as KDDCup99, provides a more representative benchmark for current cybersecurity challenges, ensuring that the model generalizes effectively to contemporary and evolving network threats.

In addition to its classification accuracy, the proposed framework effectively addresses persistent issues in intrusion detection, such as high false positive rates, scalability limitations, and inefficiencies in handling large volumes of data. By leveraging attention mechanisms to emphasize relevant features, the model substantially reduces false alarms while maintaining computational efficiency suitable for real-time implementation. The architectural flexibility of the model allows it to scale seamlessly across diverse network environments, making it well-suited for continuous enterprise-level monitoring. Furthermore, the framework's modularity enables potential future enhancements through advanced techniques such as graph neural networks for relational feature modeling, federated learning for distributed detection, and adversarial training for improved resistance against evasion attacks.

In conclusion, this study presents a high-performing, scalable, and adaptive deep learning framework for network intrusion detection. The successful combination of CNN, Stacked LSTM, and MHSA components demonstrates that deep learning architectures can achieve both exceptional accuracy and operational practicality. As cybersecurity threats continue to grow in sophistication, hybrid models of this nature provide a promising foundation for building resilient, intelligent, and autonomous intrusion detection systems capable of protecting modern digital infrastructures.

6. Declarations

6.1. Author Contributions

Conceptualization: S.P.P., P.P., U.S., D.A.D., T.B.K., L.E.; Methodology: T.B.K.; Software: S.P.P.; Validation: S.P.P., T.B.K., and L.E.; Formal Analysis: S.P.P., T.B.K., and L.E.; Investigation: S.P.P.; Resources: T.B.K.; Data Curation: T.B.K.; Writing Original Draft Preparation: S.P.P., T.B.K., and L.E.; Writing Review and Editing: T.B.K., S.P.P., and L.E.; Visualization: S.P.P.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Thakkar and R. Lohiya, “A survey on intrusion detection system: Feature selection, model, performance measures, application perspective, challenges, and future research directions,” *Artificial Intelligence Review*, vol. 55, no. 1, pp. 453–563, 2022, doi: 10.1007/s10462-021-10037-9.

[2] N. Khare, S. K. Dubey, R. Kumar, R. K. Gupta, and R. C. Poonia, “SMO-DNN: Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection,” *Electronics*, vol. 9, no. 4, pp. 692–704, 2020, doi: 10.3390/electronics9040692.

[3] P. Vijayalakshmi and D. Karthika, “Hybrid dual-channel convolution neural network (DCCNN) with spider monkey optimization (SMO) for cyber security threats detection in Internet of Things,” *Measurement: Sensors*, vol. 27, no. 12, pp. 1–8, 2023, doi: 10.1016/j.measen.2023.100783.

[4] A. Shenfeld, D. Day, and A. Ayesh, “Intelligent intrusion detection systems using artificial neural networks,” *ICT Express*, vol. 4, no. 2, pp. 95–99, 2018, doi: 10.1016/j.icte.2018.06.002.

[5] A. Agrawal, D. Garg, R. Sethi, and A. K. Shrivastava, “Optimum redundancy allocation using spider monkey optimization,” *Soft Computing*, vol. 27, no. 21, pp. 15595–15608, 2023, doi: 10.1007/s00500-022-07626-6.

[6] V. Agrawal, R. Ratika, and D. C. Tiwari, “Spider monkey optimization: A survey,” *International Journal of System Assurance Engineering and Management*, vol. 9, no. 4, pp. 929–941, 2018, doi: 10.1007/s13198-017-0685-6.

[7] P. Pothumani and E. S. Reddy, “Network intrusion detection using ensemble weighted voting classifier based honeypot framework,” *Journal of Autonomous Intelligence*, vol. 7, no. Dec., pp. 1–16, 2024, doi: 10.32629/jai.v7i3.1081.

[8] C. K. Ramu, T. S. Rao, and E. U. S. Rao, “Attack classification in network intrusion detection system based on optimization strategy and deep learning methodology,” *Multimedia Tools and Applications*, vol. 2024, no. Dec., pp. 1–32, 2024, doi: 10.1007/s11042-024-18558-5.

[9] D. Jayalatchumy, R. Ramalingam, A. Balakrishnan, M. Safran, and S. Alfarhood, “Improved crow search-based feature selection and ensemble learning for IoT intrusion detection,” *IEEE Access*, vol. 12, no. Dec., pp. 33218–33235, 2024, doi: 10.1109/ACCESS.2024.3372859.

[10] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, no. Dec., pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[11] F. Alwahedi, A. Aldhaheri, M. A. Ferrag, A. Battah, and N. Tihanyi, "Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models," *Internet of Things and Cyber-Physical Systems*, vol. 4, no. Dec., pp. 167–185, 2024, doi: 10.1016/j.iotcps.2023.12.003.

[12] S. Keskin and E. Okatan, "Machine learning methods for intrusion detection in computer networks: A comparative analysis," *International Journal of Engineering Innovation and Research*, vol. 5, no. 3, pp. 268–279, 2023.

[13] S. T. Ikram and A. K. Cherukuri, "Improving accuracy of intrusion detection model using PCA and optimized SVM," *Journal of Computing and Information Technology*, vol. 24, no. 2, pp. 133–148, 2016, doi: 10.20532/cit.2016.1002701.

[14] S. Bebortta, S. K. Das, and S. Chakravarty, "Fog-enabled intelligent network intrusion detection framework for Internet of Things applications," in *2023 International Conference on Confluence*, vol. 2023, no. Jan., pp. 1–7, 2023, doi: 10.1109/Confluence56041.2023.10048841.

[15] T. A. Tang, L. Mhamdi, D. McLernon, S. A. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, vol. 2016, no. Oct., pp. 258–263, 2016, doi: 10.1109/WINCOM.2016.7777224.

[16] J. Kim, J. Kim, H. L. Thi Thu, and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," in *2016 International Conference on Platform Technology and Service (PlatCon)*, vol. 2016, no. Feb., pp. 258–263, 2016, doi: 10.1109/PlatCon.2016.7456805.

[17] T. Ma, F. Wang, J. Cheng, Y. Yu, and X. Chen, "A hybrid spectral clustering and deep neural network ensemble algorithm for intrusion detection in sensor networks," *Sensors*, vol. 16, no. 10, pp. 1701-1713, 2016, doi: 10.3390/s16101701.

[18] T.-T.-H. Le, J. Kim, and H. Kim, "An effective intrusion detection classifier using long short-term memory with gradient descent optimization," in *2017 International Conference on Platform Technology and Service (PlatCon)*, vol. 2017, no. Feb., pp. 1–6, 2017, doi: 10.1109/PlatCon.2017.7883684.

[19] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, no. Dec., pp. 21954–21961, 2017, doi: 10.1109/ACCESS.2017.2762418.

[20] Y. Fu, X. Liu, X. Sun, S. Li, and J. Liu, "An intelligent network attack detection method based on RNN," in *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, vol. 2018, no. Jun., pp. 483–489, 2018, doi: 10.1109/DSC.2018.00078.

[21] H. He, J. Yan, Z. Zhang, Z. Ma, and Y. Tian, "A novel multimodal-sequential approach based on multi-view features for network intrusion detection," *IEEE Access*, vol. 7, no. Dec., pp. 183207–183221, 2019, doi: 10.1109/ACCESS.2019.2959131.

[22] P. Wu and H. Guo, "LuNet: A deep neural network for network intrusion detection," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, vol. 2019, no. Dec., pp. 617–624, 2019, doi: 10.1109/SSCI44817.2019.9003126.

[23] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, "A hybrid deep learning model for efficient intrusion detection in big data environment," *Information Sciences*, vol. 513, no. Jan., pp. 386–396, 2020, doi: 10.1016/j.ins.2019.10.069.

[24] A. Adeel, M. S. Khurram, M. S. Zia, and A. Ahmad, "A new ensemble-based intrusion detection system for Internet of Things," *Arabian Journal for Science and Engineering*, vol. 46, no. Dec., pp. 1–14, 2021, doi: 10.1007/s13369-021-06086-5.

[25] V. Ravi, R. Chaganti, and M. Alazab, "Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system," *Computers & Electrical Engineering*, vol. 102, no. Dec., pp. 1-16, 2022, doi: 10.1016/j.compeleceng.2022.108156.

[26] M. Živković, D. Božić, M. Nikolić, M. Stojanović, and S. Simić, "Novel hybrid firefly algorithm: An application to enhance XGBoost tuning for intrusion detection classification," *PeerJ Computer Science*, vol. 8, no. Dec., pp. 1-16, 2022, doi: 10.7717/peerj-cs.956.

[27] B. Majhi, "Optimizing LightGBM for intrusion detection systems using GOA," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, vol. 2023, no. Jul., pp. 1–5, 2023, doi: 10.1109/ICCCNT56998.2023.10308360.

[28] Y. Liu and L. Wu, "Intrusion detection model based on improved transformer," *Applied Sciences*, vol. 13, no. 10, -121, 2023, doi: 10.3390/app13106251.

[29] P. D. N. Khare, “Ensemble-based feature selection with long short-term memory for classification of network intrusion,” in *Advances in Social Networking and Online Communities*, vol. 2021, no. Dec., pp. 228–245, 2021, doi: 10.4018/978-1-7998-7764-6.ch008.

[30] A. A. Donkol, A. G. Hafez, A. I. Hussein, and M. M. Mabrook, “Optimization of intrusion detection using likely point PSO and enhanced LSTM-RNN hybrid technique in communication networks,” *IEEE Access*, vol. 11, no. Dec., pp. 9469–9482, 2023, doi: 10.1109/ACCESS.2023.3240109.

[31] T. A. Kumari and S. Mishra, “Tachyon: Enhancing stacked models using Bayesian optimization for intrusion detection using different sampling approaches,” *Egyptian Informatics Journal*, vol. 27, no. Dec., pp. 1-20, 2024, doi: 10.1016/j.eij.2024.100520.

[32] Y. Yin, J. Jang-Jaccard, W. Xu, A. Singh, J. Zhu, F. Sabrina, and J. Kwak, “IGRF-RFE: A hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset,” *arXiv preprint arXiv:2203.16365*, vol. 2022, no. Mar., pp. 1–12, 2022.

[33] S. U. Jafri, S. Rao, V. Shrivastav, and M. Tawarmalani, “LEO: Online ML-based traffic classification at multi-terabit line rate,” in *Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, vol. 2024, no. Apr., pp. 1573–1591, 2024.

[34] M. Ali, M. Shahroz, M. F. Mushtaq, S. Alfarhood, M. Safran, and I. Ashraf, “Hybrid machine learning model for efficient botnet attack detection in IoT environment,” *IEEE Access*, vol. 2024, no. Dec., pp. 1–12, 2024.

[35] R. Nagarajan, M. Batumalay, and Z. Xu, “IoT-based intrusion detection for edge devices using augmented system,” *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1412–1423, 2024.

[36] D. Zegarra Rodriguez, O. Daniel Okey, S. S. Maidin, E. Umoren Udo, and J. H. Kleinschmidt, “Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic explainable feature selection,” *PLOS ONE*, vol. 18, no. 10, pp. 1-12, 2023, doi: 10.1371/journal.pone.0286652.

[37] U. Sirisha, C. K. Kumar, S. C. Narahari, and P. N. Srinivasu, “An iterative PRISMA review of GAN models for image processing, medical diagnosis, and network security,” *Computers, Materials & Continua*, vol. 82, no. 2, pp. 1–14, 2025.

[38] S. P. Praveen, A. Chokka, P. Sarala, R. Nakka, S. B. Chandolu, and V. E. Jyothi, “Investigating the efficacy of deep reinforcement learning models in detecting and mitigating cyber-attacks: A novel approach,” *Journal of Cybersecurity and Information Management*, vol. 14, no. 1, pp. 1–12, 2024.

[39] N. S. Biyyapu, S. B. Chandolu, S. Gorintla, N. R. Tirumalasetti, A. Chokka, and S. P. Praveen, “Advanced machine learning techniques for real-time fraud detection and prevention,” *Journal of Theoretical and Applied Information Technology*, vol. 102, no. 20, pp. 1–10, 2024.

[40] S. P. Praveen, S. Lalitha, P. Sarala, K. Satyanarayana, and D. A. Karras, “Optimizing intrusion detection in Internet of Things (IoT) networks using a hybrid PSO-LightBoost approach,” *International Journal of Intelligent Engineering and Systems*, vol. 18, no. 3, pp. 1–12, 2025.

[41] D. Pambudi, F. Fadly, M. H. Kurniawan, and H. Haryanto, “The eye’s signature: Innovative approaches to iris detection,” *International Journal of Advances in Artificial Intelligence and Machine Learning*, vol. 2, no. 1, pp. 38–43, Mar. 2025, doi: 10.58723/IJAAIML.V2I1.379.