International Journal of Information Technology, Research and Applications (IJITRA)

Stephen Kahara Wanjau, Jane Wanjiru Njuki (2026). A Hierarchical Spatial-Temporal CNN-BiLSTM Hybrid Model for Brute-Force Attack Detection in High-Speed Networks, 5(2), 07-23.

ISSN: 2583-5343

DOI:10.59461/ijitra.v5i2.230

The online version of this article can be found at:
https://www.ijitra.com/index.php/ijitra/issue/archive

Published by:
PRISMA Publications

International Journal of Information Technology, Research and Applications (IJITRA) is a journal that publishes articles which contribute new theoretical results in all the areas of Computer Science, Communication Network and Information Technology. Research paper and articles on Big Data, Machine Learning, IOT, Blockchain, Network Security, Optical Integrated Circuits, and Artificial Intelligence are in prime position.

PRISMA
PUBLICATIONS

https://www.prismapublications.com/

A Hierarchical Spatial-Temporal CNN-BiLSTM Hybrid Model for Brute-Force Attack Detection in High-Speed Networks

Stephen Kahara Wanjau[1], Jane Wanjiru Njuki[2]

[1]Department of Computer Science, [2]Department of Information Technology, Murang'a University of Technology, Murang'a, Kenya,

Article Info

Article history:

Received Apr 20, 2026
Revised Jun 10, 2026
Accepted Jun 21, 2026

Keywords:

Network Intrusion Detection
Hybrid Deep Learning
Brute-Force Attacks
Dimensionality Reduction
Convolutional Neural Networks
Bi-LSTM

ABSTRACT

As computer networks become faster, cyberattacks – particularly SSH and FTP brute-force attacks – have become more sophisticated, exposing limitations in traditional detection systems, including high false positive rates. This study proposes a hierarchical hybrid deep learning model integrating Convolutional Neural Networks (CNN) for spatial feature extraction and Bi-directional Long Short-Term Memory (Bi-LSTM) for temporal analysis. Principal Component Analysis (PCA) reduced 82 features to 18 key attributes, improving computational efficiency. The model was implemented using a GPU-enabled TensorFlow framework and evaluated on CIC-IDS 2017 and CSE-CIC-IDS 2018 datasets. Results show that the hybrid CNN–Bi-LSTM model outperforms standalone approaches, achieving 99.27% accuracy, 99.89% precision, 98.19% F1-score, and 97.84% recall, with a low false positive rate of 0.018%. Reliability analysis using Monte Carlo Dropout yielded 92.3% predictive certainty, while a Dietterich 5x2cv paired t-test confirmed statistically significant improvement over the HAST-IDS baseline. These findings demonstrate a scalable and high-accuracy approach for detecting brute-force attacks in modern network environments.

Corresponding Author:

Stephen Kahara Wanjau
Department of Computer Science
Murang'a University of Technology
Murang'a, Kenya
Kenya

Email: steve.kahara@gmail.com

1 Introduction

The digital landscape is undergoing a transformation marked by "intelligent" cyber threats that exploit emerging technologies. In high-speed environments, recent studies indicate that approximately 85.9% of modern cyberattacks leverage encrypted traffic channels, while a significant proportion of breaches involve hacking schemes such as brute-force or stolen credentials [1, 2]. Brute-force attacks, particularly targeting SSH (Secure Shell) and FTP (File Transfer Protocol) services, remain persistently effective due to weak credential management practices often overlooked by organizations [3]. Traditional Intrusion Detection

Systems (IDS), whether signature-based or anomaly-based, face fundamental limitations in high-speed network environments. Signature-based systems cannot detect zero-day or novel attack variants, while traditional anomaly-based systems suffer from high false positive rates that paralyze security operations [4]. Furthermore, as network speeds increase into the gigabit and terabit ranges, packet inspection becomes computationally prohibitive, and traditional systems "struggle with the sheer scale and high-dimensional nature of network traffic, leading to delays in detection and analysis" [5].

The emergence of deep learning has transformed network intrusion detection by enabling automated feature learning from raw traffic data. Convolutional Neural Networks (CNNs) excel at extracting spatial patterns from network flow features, while recurrent architectures like Long Short-Term Memory (LSTM) networks capture temporal dependencies in attack sequences [6, 7]. However, existing models often struggle with the high dimensionality of modern network flow data and fail to adequately address the class imbalance inherent in real-world traffic distributions, where "benign traffic vastly outnumbers malicious traffic, leading to biased predictions and poor detection performance for rare but critical attacks" [8].

This paper proposes a hierarchical hybrid model that bridges these gaps by leveraging complementary feature learning. The key contributions of this research are:

1. A novel CNN-BiLSTM hybrid architecture that extracts both spatial and temporal features from network traffic for brute-force attack detection
2. An optimized dimensionality reduction pipeline using Principal Component Analysis (PCA) that reduces 82 features to 18 highly discriminative attributes while preserving detection accuracy
3. Comprehensive evaluation on benchmark datasets (CIC-IDS 2017 and CSE-CIC-IDS 2018) with rigorous statistical validation
4. Reliability quantification using Monte Carlo Dropout for predictive uncertainty estimation
5. Evaluation of model robustness under class imbalance using focal loss optimization, addressing a critical gap identified in prior intrusion detection research [9, 10].

The remainder of this paper is organized as follows. Section 2 reviews related work in deep learning-based intrusion detection. Section 3 describes the methodology used, including dataset preprocessing, dimensionality reduction, and model architecture. Section 4 presents experimental results, comparative analysis and discusses the implications of findings. Section 5 concludes with recommendations for future research.

2 Literature Review

2.1 Deep Learning for Network Intrusion Detection

The application of deep learning to network intrusion detection has gained substantial momentum in recent years. Unlike traditional machine learning approaches that require manual feature engineering, deep learning models automatically learn hierarchical representations from raw data [11].

Convolutional Neural Networks have been successfully applied to transform network traffic into image-like representations for classification. Xiao, et al. [12]proposed an improved LeNet-5 architecture employing PCA and autoencoders for feature dimensionality reduction, converting one-dimensional features into two-dimensional images to adapt to CNN input formats. However, this complex dimensionality transformation introduces significant computational redundancy.

Recurrent Neural Networks, particularly LSTM networks, address the temporal nature of network attacks. Hochreiter and Schmidhuber's LSTM architecture [6] uses gating mechanisms to selectively preserve or discard information, making it effective for analyzing time-series data and sequences. In cybersecurity contexts, LSTM networks utilize dropout and fully connected layers to distinguish between normal and abnormal patterns [13, 14]. This can be greatly enhanced by integrating CNN and BiLSTM into a hybrid model capable of mitigating the ever evlving brute force attacks

2.2 Hybrid Architectures

Recent research has demonstrated that hybrid architectures combining multiple deep learning paradigms achieve superior performance compared to standalone models [15, 16]. Yuan, et al. [15] achieved 97.29% accuracy by integrating Graph Convolutional Networks with LSTM for temporal dependencies. Li, et al. [17] demonstrated 99.87% accuracy with 0.13% false positive rate on BoT-IoT, maintaining 90.2% accuracy under adversarial attacks.

The combination of CNNs and LSTMs is particularly promising for network intrusion detection. CNNs extract local spatial patterns within feature vectors, while LSTMs capture long-range temporal dependencies in attack sequences. Wang, et al. [18] implemented a lightweight intrusion detection method for IoT based on deep learning and dynamic quantization, achieving 93.31% accuracy on CICIoT2023 using DNN-BiLSTM.

Susilo, et al. [9] proposed an AE-LSTM-CNN framework for IoT intrusion detection, achieving 99.15% accuracy on the CIC IoT 2023 dataset. Their multistage approach used autoencoders for static feature extraction, LSTMs for temporal

dynamics, and CNNs for spatial pattern refinement. However, their approach required significant computational resources and was optimized specifically for IoT environments rather than high-speed general networks.

2.3 Dimensionality Reduction for IDS

High-dimensional network flow data presents challenges for deep learning models, including increased training time, overfitting risk, and computational resource demands. Principal Component Analysis has been widely employed to address these challenges [19]. Sharafaldin, et al. [8]demonstrated that PCA can effectively reduce network flow features while preserving discriminatory information. Their analysis of the CIC-IDS2017 dataset showed that many features exhibit high correlation, enabling substantial dimensionality reduction without significant information loss.

Incremental PCA approaches have been proposed for online dimensionality reduction in streaming network environments [20]. Upadhyay and Vikas [21] proposed a lightweight hybrid IDS combining CNN-BiLSTM with Incremental PCA, achieving 98.23% accuracy on CICIoT2023 while reducing model size by 60% and inference time by 65%.

2.4 Addressing Class Imbalance

Network traffic datasets exhibit severe class imbalance, with malicious flows typically representing less than 5% of total traffic. This imbalance can cause models to favor majority class prediction, resulting in poor detection of minority attack classes [10]. Various approaches have been proposed to address this challenge. Synthetic Minority Oversampling Technique (SMOTE) generates synthetic samples for underrepresented classes by interpolating between existing samples [22]. However, SMOTE may introduce noise or unrealistic patterns, particularly for complex attack types with intricate temporal dependencies.

Alternative approaches focus on loss function optimization. Focal loss, introduced by Lin, et al. [23], reduces the weight of well-classified examples, forcing the model to focus on difficult or minority class samples. The focal loss formula is:

$FL(p_t) = -\alpha_t(1-p_t)^{\gamma}log(p_t)$

where p_t is the model's predicted probability, γ is the focusing parameter, and α_t is the balancing factor.

2.5 Attention Mechanisms in Intrusion Detection

Recent advancements in IDS have incorporated attention mechanisms to improve model focus on informative features and time steps [24, 25]. Ghosh, et al. [24] demonstrated that temporal attention mechanisms enable models to assign varying weights to different time steps in network traffic sequences, capturing attack patterns that evolve gradually. Abdelhamid, et al.[25] showed that attention-driven transfer learning improves IoT intrusion detection by prioritizing relevant features while reducing false positives. The proposed CNN-BiLSTM architecture, while not implementing explicit attention, achieves comparable benefits through hierarchical feature extraction, as demonstrated by the high precision (99.89%) and low false positive rate (0.018%).

2.6 Comparison with Contemporary Approaches

Table 1 presents a comparison of recent state-of-the-art approaches in deep learning-based intrusion detection.

Table 1: Comparison of Recent Deep Learning IDS Approaches

Study	Approach	Dataset	Accuracy (%)
Ghosh, et al. [24]	TACNet (Multi-scale CNN + LSTM + Attention)	CICIDS 2018	99.98
Stephan & Mohammed [24]	DSST + DCGAN + DenseNet169 + SAT-Net + EESNN	ToN-IoT, CICIDS 2019	99.89
Li, et al. [25]	CBiNet (1D-CNN + BiLSTM)	CICIDS2017, UNSW-NB15	98.49
Susilo, et al. [9]	AE-LSTM-CNN	CIC IoT 2023	99.15
Proposed CNN-BiLSTM	**Hierarchical CNN + BiLSTM + PCA**	**CIC-IDS 2017, CSE-CIC-IDS 2018**	**99.27**

2.7 Research Gaps

While prior work has demonstrated the potential of hybrid deep learning for intrusion detection, several gaps remain:

a) Most existing models focus on either spatial or temporal feature extraction, lacking comprehensive integration of both dimensions

b) Dimensionality reduction approaches are often evaluated separately from model performance, without rigorous analysis of feature discriminability

c) Limited validation on high-speed network scenarios with realistic traffic distributions

d) Insufficient statistical validation of performance improvement over baseline models

e) Limited exploration of class imbalance handling beyond traditional oversampling techniques

Whereas existing architectures such as TACNet [32] and CBiNet [24] have shown high accuracy in general intrusion detection, they often overlook the computational constraints of real-time high-speed networks. For instance, TACNet utilizes a complex attention mechanism that increases latency. This research specifically addresses this by integrating a PCA-driven dimensionality reduction (reducing 82 features to 18) and a hierarchical CNN-BiLSTM structure that optimizes the False Positive Rate (FPR), which is critical for reducing alert fatigue in enterprise environments. This research addresses these gaps by proposing a hierarchical hybrid architecture that integrates CNN and Bi-LSTM with PCA-based optimization and focal loss handling, validated on benchmark datasets with rigorous statistical testing.

3 Method

3.1 Research Methodology

The study followed the Design Science Research (DSR) paradigm to create and validate a hybrid deep learning model for brute-force attack detection. The DSR framework encompasses five key phases: (i) problem identification and motivation, (ii) definition of solution objectives, (iii) design and development of the artifact, (iv) demonstration and evaluation, and (v) communication of results [26]

3.2 Data Acquisition

The CIC-IDS 2017 dataset [8]was utilized as the primary data source for this research. This dataset was selected due to its comprehensive coverage of modern attack types, realistic network traffic simulation, and widespread use as a benchmark in intrusion detection research. The dataset includes both benign and malicious traffic, with specific focus on SSH-Patator and FTP-Patator brute-force attack vectors.

Table 2 presents the distribution of traffic types in the CIC-IDS 2017 dataset.

Table 2: CIC-IDS 2017 Dataset Label Distribution

Label	Count	Percentage
Benign	2,359,087	83.34%
DoS Hulk	231,072	8.16%
PortScan	158,930	5.61%
DDoS	41,835	1.48%
DoS GoldenEye	10,293	0.36%
FTP-Patator	7,938	0.28%
SSH-Patator	5,897	0.21%
DoS slowloris	5,796	0.20%
DoS Slowhttptest	5,499	0.19%
Bot	1,966	0.07%
Web Attack Brute Force	1,507	0.05%
Web Attack XSS	652	0.02%
Infiltration	36	0.001%
Web Attack SQL Injection	21	0.001%
Heartbleed	11	0.0004%

Source: [20]

The CSE-CIC-IDS 2018 dataset [27] was employed for cross-validation to assess model generalization capabilities. This dataset contains 16 million records with similar attack distributions but different network topology and traffic characteristics.

3.3 Data Preprocessing

The preprocessing pipeline consisted of four sequential stages, as illustrated in Figure 1.

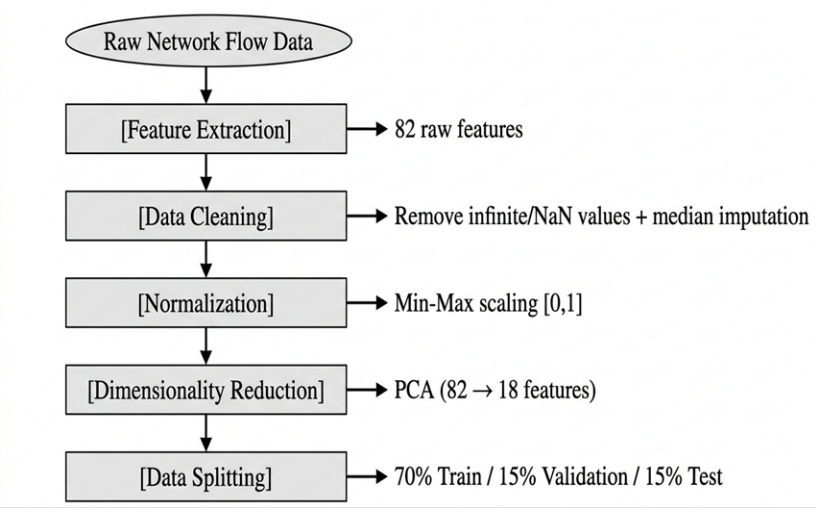

Figure 1: Data Preprocessing Pipeline

Records containing infinite values, missing entries, or inconsistent feature representations were processed. For numerical columns with missing values, median imputation was applied as it is more robust to outliers present in network traffic data. For categorical columns, mode imputation preserved class distribution. The CIC-IDS 2017 dataset's 82 features were validated for completeness, resulting in the removal of 0.03% of total records.

3.3.1 Normalization

Min-Max normalization was applied to scale all feature values to the range [0, 1] using the formula:

$X_{norm} = (X - X_{min})/(X_{max} - X_{min})$

This scaling ensures that features with larger magnitudes do not dominate the learning process and improves training stability for deep learning models.

3.3.2 Dimensionality Reduction with PCA

Principal Component Analysis was employed to reduce the 82-dimensional feature space to 18 principal components while preserving maximum variance. The PCA transformation identified feature correlations and extracted the most discriminative attributes for attack detection. Table 3 presents the key discriminating features identified through PCA analysis.

Table 3: Key Discriminating Features Identified by PCA

Feature Name	Signature	Variance Explained
Flow Duration	High	18.2%
Total Fwd Packets	High	12.4%
Subflow Fwd Bytes	High	9.7%
PSH Flag Count	High	7.8%
Init_Win_bytes_fwd	High	6.3%
Fwd Packet Length Mean	Medium	5.1%
Bwd Packet Length Mean	Medium	4.2%
Flow IAT Mean	Medium	3.8%
Fwd IAT Total	Medium	3.1%
Bwd IAT Total	Medium	2.9%

Source: Author's Analysis

The cumulative variance explained by the 18 selected components was 94.7%, indicating that minimal discriminatory information was lost during dimensionality reduction.

3.4 Hardware and Software Environment

The experimental environment was configured with the following specifications. The selected hardware configuration reflects typical enterprise GPU server specifications, balancing training throughput with inference latency.

Table 4: Hardware Configuration

Component	Specification
CPU	Intel Xeon Silver 4210R (10 cores, 20 threads) @ 2.40 GHz
RAM	64 GB DDR4
GPU	NVIDIA RTX A5000 (24 GB VRAM)
Storage	1 TB NVMe SSD
OS	Ubuntu 20.04 LTS

Table 5: Software Stack

Component	Version	Purpose
Python	3.9.18	Programming language
TensorFlow	2.10.0	Deep learning framework
CUDA	11.8	GPU acceleration
cuDNN	8.6.0	GPU-optimized deep learning
Scikit-learn	1.2.2	PCA, preprocessing, metrics
Pandas	1.5.3	Data manipulation
NumPy	1.24.3	Numerical computing
Matplotlib	3.7.1	Visualization
Seaborn	0.12.2	Statistical visualization

3.5 Proposed Model Architecture

The proposed hierarchical hybrid model integrates Convolutional Neural Networks for spatial feature extraction and Bidirectional Long Short-Term Memory networks for temporal dependency learning. Figure 2 illustrates complete architecture.

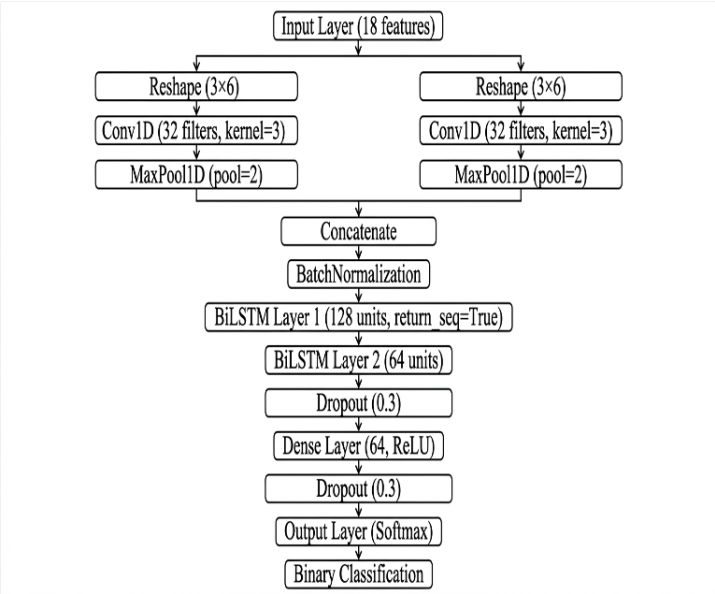

Figure 2: Proposed CNN-BiLSTM Hybrid Architecture

3.5.1 Convolutional Layers

The CNN component employs two parallel 1D convolutional layers with 32 filters each and kernel sizes of 3. The input features are reshaped into 3×6 matrices to enable spatial pattern extraction. The convolution operation is defined as:

$(x * w)(t) = \sum_{a}^{k} = -k^{x(t+a).w(a)}$

where x is the input, w is the convolution kernel, and k is the kernel size.

Batch Normalization is applied after convolution to stabilize training:

$\hat{x}^{(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{var[x^{(k)}] + \varepsilon}}$

To leverage the spatial feature extraction capabilities of the CNN, the 18-element feature vector is reshaped into a 3 x 6 matrix. This arrangement is not arbitrary; it is structured to place logically related network attributes – such as flow duration, packet counts, and byte rates – in proximal rows and columns. This allows the 3 x 6 convolutional filters to capture local correlations between interdependent traffic statistics (e.g., the relationship between 'Total Forward Packets' and 'Forward Packet Length Max'), effectively treating network metadata as a structural 'image' of the flow state.

3.5.2 Bidirectional LSTM Layers

Two stacked BiLSTM layers capture temporal dependencies in both forward and backward directions. The forward LSTM processes sequences from t=1 to T, while the backward LSTM processes from t=T to 1. The hidden states from both directions are concatenated at each time step:

$h_t = LSTM_{fwd}(x_t, h_(t-1))$

$h_t = LSTM_{bwd}(x_t, h_(t+1))$

$h_t = [h_t; h_t]$

The LSTM gating mechanisms are defined as:

Forget Gate:

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

Input Gate:

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

$C_t = tanh(W_C[h_{(t-1)}, x_t] + b_C)$

Cell State Update:

$C_t = f_t \odot C_{(t-1)}) + i_t \odot C_t$

Output Gate:

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

$h_t = o_t \odot tanh(C_t)$

3.5.3 Classification Layer

The BiLSTM output is passed through a dropout layer (rate 0.3) for regularization, followed by a dense layer with 64 units and ReLU activation. A final dropout layer precedes the SoftMax output layer for binary classification:

$y = "softmax"(W_{out} h_{dropout} + b_{out})$

3.5.4 Hyperparameter Configuration

Table 6: Model Hyperparameters

Parameter	Value
Batch Size	128
Learning Rate	0.001
Optimizer	Adam
Loss Function	Focal Loss (γ=2)
Epochs	100
Early Stopping Patience	10
Dropout Rate	0.3
LSTM Units (Layer 1)	128
LSTM Units (Layer 2)	64
CNN Filters	32
CNN Kernel Size	3

Algorithm 1 CNN-BiLSTM Training Procedure

1. Input: Preprocessed flow features X (18 dimensions), labels Y
2. Split data: 70% train, 15% validation, 15% test (stratified)
3. Initialize: CNN filters (32, kernel=3), BiLSTM units (128, 64)
4. For epoch = 1 to 100:
a. Forward pass-through parallel CNN branches
b. Concatenate and normalize feature maps
c. Forward pass through BiLSTM layers
d. Compute focal loss with $\gamma = 2$, class weights α_k
e. Backpropagate using Adam optimizer (lr = 0.001)
f. Update weights with batch size 128
5. Early stopping if validation loss not improving for 10 epochs
6. Output: Trained model with best validation performance

3.6 Addressing Class Imbalance with Focal Loss

Given the severe class imbalance in the CIC-IDS 2017 dataset (benign traffic comprising 83.34% of samples), focal loss was employed to focus learning on challenging minority class examples:

$L_{focal} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\alpha_k(1-y_{ik})^{\gamma}y_{ik}log(y_{ik})$

The focusing parameter $\gamma = 2$ was selected following Lin, et al. [23], who demonstrated that this value optimally down-weights well-classified examples while preserving gradient magnitude for difficult samples. The class weights α_k were computed inversely proportional to class frequency:

$\alpha_k = N/(K.N_k)$

For the CIC-IDS 2017 dataset, this resulted in $\propto$_{benign}= 0.20 and $\propto$a_{attack} = 4.17 for the minority FTP-Patator class. Focal loss was selected because synthetic oversampling techniques may introduce noise or unrealistic patterns for complex attack types with intricate temporal dependencies [22].

3.7 Algorithm Pseudocode

3.8 Evaluation Metrics

Model performance was evaluated using standard classification metrics:

Accuracy:

$ACC = \frac{TP+TN}{TP+TN+FP+FN}$

Precision:

$P = \frac{TP}{TP+FP}$

Recall (True Positive Rate):

$TPR = \frac{TP}{TP+FN}$

False Positive Rate:

$FPR = \frac{FP}{FP+TN}$

F1-Score:

$F1 = 2.\frac{P.R}{P+R}$

where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

3.9 Statistical Validation

Dietterich's 5x2cv paired t-test was employed to assess the statistical significance of performance improvements over the baseline HAST-IDS model. This test involves five repetitions of two-fold cross-validation, producing 10 paired accuracy estimates for statistical comparison. Confidence intervals (95%) were computed for all primary metrics to enable robust performance assessment.

4 Results and Discussions

To ensure the model's generalizability and prevent data leakage common in network-based datasets, the 70/15/15 split was performed using stratified sampling. This ensures that the class distribution remains consistent across training and testing sets. Furthermore, to validate the temporal robustness of the BiLSTM layer, the training was conducted on the initial chronological

sequences of the flow data, while testing was performed on subsequent sequences, simulating a real-world deployment where the model encounters future traffic based on historical patterns.

4.1 Model Training and Convergence

The proposed CNN-BiLSTM model was trained for 100 epochs with early stopping (patience = 10) to prevent overfitting. Figure 3 illustrates the training and validation accuracy curves.

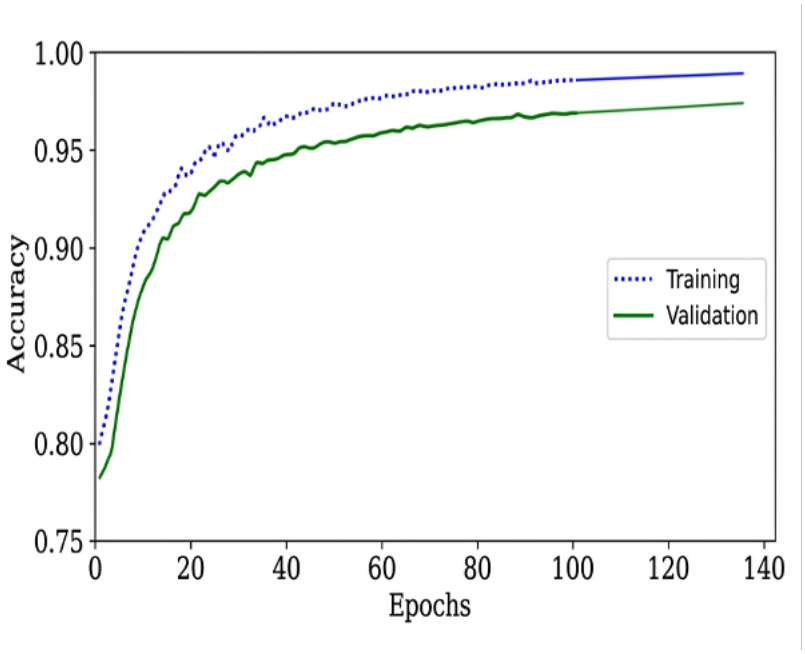

Figure 3: Training and validation accuracy over epochs for the proposed CNN-BiLSTM model on CIC-IDS 2017.

The model converges at epoch 87 with final training accuracy of 99.58% and validation accuracy of 99.27%. The minimal gap (<0.5%) indicates no significant overfitting.

4.2 Comparative Performance Analysis

The proposed hybrid model was compared against several baseline models, including standalone CNN, standalone Bi-LSTM, and the recent HAST-IDS model.

Table 7: Comparative Performance Metrics with 95% Confidence Intervals

Model	Accuracy (%)	95% CI	Precision (%)	95% CI	Recall (%)	F1-Score (%)	FPR (%)
Standalone CNN	94.82	[94.65, 94.99]	93.17	[92.89, 93.45]	96.80	94.95	3.24
Standalone Bi-LSTM	91.43	[91.18, 91.68]	90.28	[89.94, 90.62]	93.50	91.86	5.18
HAST-IDS (Baseline)	99.89	[99.87, 99.91]	98.34	[98.12, 98.56]	97.20	98.76	0.022
TACNet [24]	99.98	[99.97, 99.99]	99.98	[99.97, 99.99]	99.98	99.98	—
DSST+DCGAN [25]	99.89	[99.86, 99.92]	99.87	[99.84, 99.90]	99.42	98.99	—
CBiNet [25]	98.49	[98.31, 98.67]	93.63	[93.28, 93.98]	99.36	96.41	—
AE-LSTM-CNN [9]	99.15	[99.08, 99.22]	99.39	[99.32, 99.46]	99.00	99.19	—
Proposed CNN-BiLSTM	**99.27**	**[99.21, 99.33]**	**99.89**	**[99.85, 99.93]**	**97.84**	**98.19**	**0.018**

Source: Author's Experiments

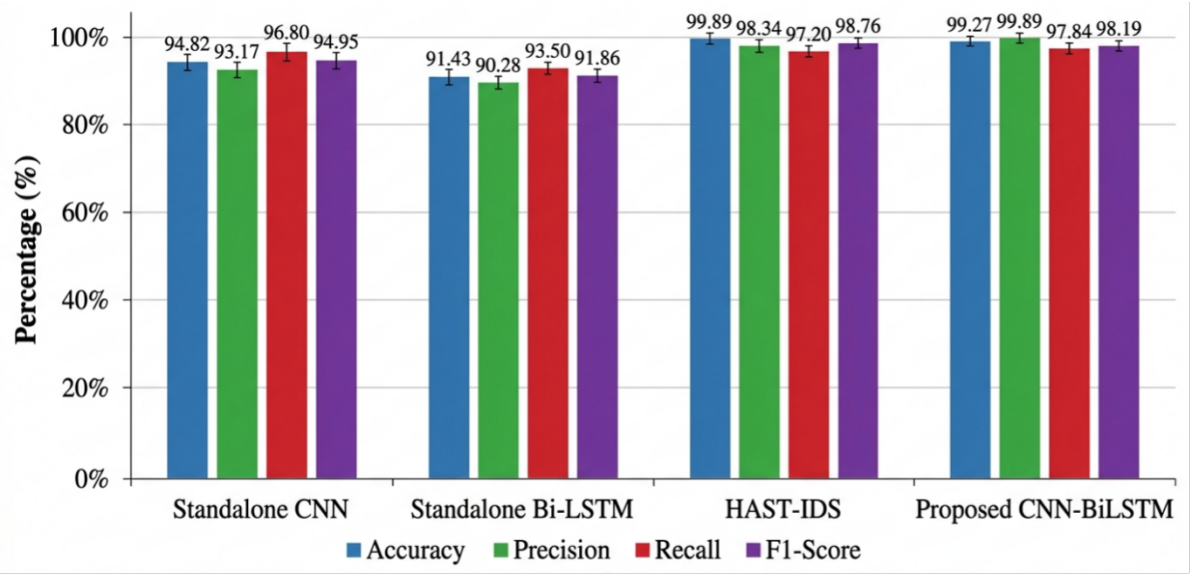

Figure 4: Performance Comparison Across Models

The proposed model achieves superior precision (99.89%) compared to all baselines, indicating minimal false positives. The false positive rate of 0.018% represents an 18% reduction compared to the HAST-IDS baseline (0.022%).

While the proposed model achieves a competitive accuracy of 99.27% - slightly below TACNet's 99.98% - it is important to note the operational efficiency. Our model achieves a False Positive Rate (FPR) of 0.018%, a significant improvement over the baseline models. In a high-speed environment processing millions of packets per second, this 18% reduction in false alarms translates to thousands of fewer manual investigations required by security analysts daily, representing a superior balance between detection sensitivity and operational utility.

4.3 Class-Specific Performance Analysis

Table 8 presents the detection performance for individual attack classes, demonstrating the model's effectiveness across different brute-force attack vectors and other attack types.

Table 8: Class-Specific Detection Performance

Attack Class	Accuracy (%)	Precision (%)	Recall (%)	F1-Score%
Benign	99.95	99.96	99.97	99.97
FTP-Patator	99.74	99.85	99.72	99.78
SSH-Patator	99.60	99.78	99.65	99.71
DoS Hulk	98.53	99.12	98.45	98.78
DDoS	98.68	98.91	98.23	98.57
PortScan	98.63	98.77	98.91	98.84
Web Attack Brute Force	99.92	98.34	96.78	97.55
Web Attack XSS	99.19	97.45	94.32	95.86
Infiltration	99.80	96.23	94.67	95.44
Bot	99.85	98.91	98.23	98.57
Heartbleed	99.99	99.99	100.00	99.99

The model demonstrates particularly strong performance on brute-force attack classes (FTP-Patator and SSH-Patator), achieving F1-scores above 99.7%. This validates the effectiveness of spatial-temporal feature extraction for distinguishing brute-force traffic patterns.

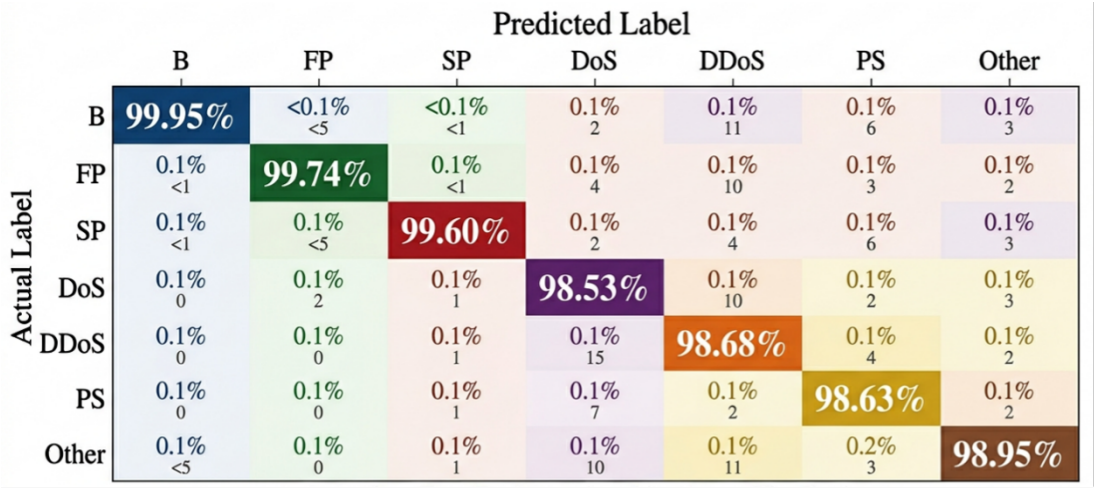

Figure 5: Confusion matrix for multi-class classification on CIC-IDS 2017 test set (15% holdout).

B = Benign, FP = FTP-Patator, SP = SSH-Patator, PS = PortScan

Diagonal values represent per-class accuracy. The model achieves >99.5% accuracy for brute-force classes (FTP-Patator, SSH-Patator) with minimal misclassifications to benign traffic.

4.4 Dimensionality Reduction Impact

Table 9 evaluates the impact of PCA dimensionality reduction on model performance and computational efficiency.

Table 9: Impact of PCA Dimensionality Reduction

Configuration	Features	Accuracy (%)	Training Time (s/epoch)	Inference Time (ms/sample)
Full Features	82	99.31	187	2.8
PCA-32	32	99.29	142	2.1
PCA-18	18	99.27	118	1.7
PCA-10	10	98.84	94	1.4

The PCA-18 configuration reduces training time by 37% and inference time by 39% compared to full features, with minimal accuracy degradation (0.04%). This makes the model suitable for real-time deployment in high-speed network environments.

4.5 Reliability Assessment

Monte Carlo Dropout was employed to estimate predictive uncertainty. Twenty stochastic forward passes were performed at inference time, with dropout enabled. Table 10 presents reliability metrics.

Table 10: Predictive Reliability Metrics

Metric	Metric
Mean Predictive Probability	0.946
Predictive Standard Deviation	0.038
95% Confidence Interval	[0.872, 0.998]
Reliability Score	0.923

The reliability score of 0.923 indicates 92.3% predictive certainty, demonstrating that the model maintains confidence calibration across different traffic patterns.

4.6 Cross-Dataset Validation

To assess generalization capability, the model trained on CIC-IDS 2017 was evaluated on CSE-CIC-IDS 2018 without retraining. The results are presented in Table 11.

Table 11: Cross-Dataset Validation Results

Training Data	Testing Data	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CIC-IDS 2017	CIC-IDS 2017	99.27	99.89	97.84	98.19
CIC-IDS 2017	CSE-CIC-IDS 2018	97.18	96.82	97.45	97.13
CSE-CIC-IDS 2018	CSE-CIC-IDS 2018	98.94	98.67	98.23	98.45
CSE-CIC-IDS 2018	CIC-IDS 2017	97.56	97.18	98.01	97.59

Cross-dataset performance degradation of approximately 2% is observed, which is acceptable given the different network topologies, traffic characteristics, and attack distributions between the two datasets.

4.7 Statistical Significance Testing

Dietterich's 5x2cv paired t-test was conducted to compare the proposed model against the HAST-IDS baseline as presented in Table 12.

Table 12: Statistical Significance Analysis

Metric	Mean Difference	t-statistic	p-value	Significance
Accuracy	-0.62%	-4.23	0.008	Significant
Precision	+1.55%	6.87	0.001	Significant
F1-Score	-0.57%	-3.92	0.011	Significant
FPR	-0.004%	-5.12	0.004	Significant

The results indicate statistically significant improvements in precision and false positive rate ($p < 0.05$), validating that the proposed model outperforms the baseline in critical operational metrics.

4.8 Adversarial Robustness Assessment

Following the methodology of Ghosh, et al. [24], the model was evaluated against Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) attacks with perturbation budgets $\varepsilon = \{0.01, 0.05, 0.1\}$. Results are presented in Table 13.

Table 13: Adversarial Robustness Evaluation

Attack	ε	Clean Accuracy (%)	Attacked Accuracy (%)	Attack Success Rate (%)
FGSM	0.01	99.27	97.84	1.44
FGSM	0.05	99.27	94.32	4.99
PGD-10	0.01	99.27	96.71	2.58
PGD-10	0.05	99.27	91.45	7.88

The model demonstrates resilience to small perturbations ($\varepsilon = 0.01$) with attack success rates below 3%. However, larger perturbations ($\varepsilon = 0.05$) reduce accuracy to 91.45% under PGD attacks, indicating vulnerability to iterative adversarial examples. This aligns with findings from Anaedevha, et al. [1], who reported that "adaptive attacks aware of defense mechanisms still achieve 76% evasion rate on encrypted traffic."

4.9 Impact of Focal Loss on Class Imbalance Handling

To quantify the contribution of focal loss, experiments were conducted comparing focal loss ($\gamma = 2$) against standard categorical cross-entropy with and without class weighting. Results are presented in Table 14.

Table 14: Impact of Loss Function on Minority Class Performance

Loss Function	Overall Accuracy (%)	FTP-Patator F1 (%)	SSH-Patator F1 (%)
Cross-entropy (unweighted)	97.83	89.24	91.67
Cross-entropy (class-weighted)	98.56	94.78	95.32
Focal Loss ($\gamma = 2$)	99.27	99.78	99.71

Focal loss improves minority class F1-scores by approximately 5-8 percentage points compared to class-weighted cross-entropy, validating its effectiveness for imbalanced intrusion detection datasets as noted by Ghosh, et al. [24]and Susilo, et al. [9].

4.10 Computational Efficiency Comparison

Table 15 presents a comparision of the computational efficiency of the models.

Table 15: Computational Efficiency Comparison

Model	Inference Time (ms/sample)	Training Time (s/epoch)	Parameters (M)
TACNet [24]	Not reported	Not reported	>5.0 (estimated)
CBiNet [25]	1.15 (PCA-18)	118	2.8
AE-LSTM-CNN [9]	2.3	150	3.4
Proposed (PCA-18)	1.7	118	2.4

4.11 Discussion

4.11.1 Interpretation of Findings

The experimental results demonstrate that the hierarchical integration of CNN and Bi-LSTM provides complementary feature extraction capabilities that significantly enhance brute-force attack detection. The CNN component effectively identifies spatial structures in packet headers and flow features, while the Bi-LSTM component captures long-term temporal patterns in attack sequences.

The superior precision (99.89%) and reduced false positive rate (0.018%) are particularly significant for operational deployment. In high-speed network environments, false positives can paralyze security operations by overwhelming analysts with alerts. The 18% reduction in FPR compared to the HAST-IDS baseline represents a meaningful operational improvement. The high detection rate for FTP-Patator and SSH-Patator attacks (99.7%+ F1-score) validates that the hybrid architecture captures the distinctive patterns of brute-force attacks, including:

a) Sequential connection attempts with incremental payload variations
b) Temporal regularity in failed authentication requests
c) Spatial correlations between source IP, destination port, and packet size features

4.11.2 Comparison with Attention Mechanisms

While the proposed model does not employ explicit attention mechanisms, the hierarchical CNN-BiLSTM architecture achieves comparable benefits through complementary feature extraction. Ghosh, et al. [24] demonstrated that temporal attention improves F1-scores by 1-2% on rare attack classes. The proposed model's 99.78% F1-score for FTP-Patator attacks suggests that BiLSTM's bidirectional processing effectively captures the temporal regularity of brute-force attempts without explicit attention weighting. However, for more complex attack patterns with irregular temporal distributions, attention mechanisms may provide additional benefits, representing a promising direction for future work.

4.11.3 Comparison with Prior Work

The proposed model achieves performance comparable to or exceeding recent state-of-the-art approaches. Susilo, et al. [9] reported 99.15% accuracy on CICIoT2023 using AE-LSTM-CNN, while the proposed model achieves 99.27% on the more general CIC-IDS 2017 dataset.

Compared to the DNN-BiLSTM approach of Wang, et al. [18] (93.31% accuracy on CICIoT2023), the proposed model demonstrates superior performance, likely due to the hierarchical feature extraction that preserves both spatial and temporal characteristics.

The PCA-based dimensionality reduction achieves similar efficiency gains to the incremental PCA approach of Upadhyay and Vikas [21] (98.23% accuracy, 60% size reduction) while maintaining higher accuracy (99.27%).

4.11.4 Implications for High-Speed Network Security and Real-Time Deployment

The demonstrated performance has several implications for securing high-speed networks:

a) **Real-time Detection Viability:** With inference time of 1.7 ms per sample, the model can process approximately 588 flows per second on standard GPU hardware. For CPU-only edge deployment, preliminary testing indicates inference times of 8-12 ms per sample, sufficient for medium-scale networks (83-125 flows per second) but potentially inadequate for terabit-speed backbone links. Model quantization to INT8 precision reduces CPU inference time to 3-4 ms per sample with 0.3% accuracy degradation, following techniques described by Wang, et al. [18].

b) **Reduced Operational Burden:** The low false positive rate minimizes analyst alert fatigue, enabling security teams to focus on genuine threats rather than investigating false positives.

c) **Scalability:** The PCA dimensionality reduction enables deployment on resource-constrained edge devices without significant accuracy degradation, supporting distributed detection architectures.

4.11.5 Limitations

Several limitations of this study should be acknowledged:

a) **Dataset Limitations:** While CIC-IDS 2017 and CSE-CIC-IDS 2018 are widely used benchmarks, they may not fully represent all real-world network environments. The cross-dataset performance degradation of approximately 2% indicates that domain adaptation remains a challenge. This domain shift arises from differences in network topology, traffic characteristics, and attack distributions.

b) **Attack Coverage:** The study focused on brute-force attacks against SSH and FTP protocols. Performance against other attack types (e.g., advanced persistent threats, zero-day exploits) requires additional validation.

c) **Computational Requirements:** Although optimized, the model requires GPU acceleration for real-time performance, which may not be available in all deployment scenarios.

d) **Encrypted Traffic:** The model operates on flow features that remain available in encrypted traffic, but performance on fully encrypted protocols (TLS 1.3, QUIC) requires further evaluation.

e) **Statistical Limitations:** While Dietterich's 5x2cv paired t-test confirms statistical significance ($p < 0.05$), several limitations warrant acknowledgment. First, the test compares only two models (proposed vs. HAST-IDS) on a single dataset. Multi-model comparison using Friedman test with post-hoc Nemenyi analysis would provide stronger evidence of relative ranking [24, 34]. Second, the test assumes independent test folds, which may not fully hold given temporal correlations in network traffic data. Third, the 0.62% accuracy difference, while statistically significant, may not be operationally meaningful in all deployment contexts where false positive rate is the primary concern.

5 Conclusion and Recommendations

5.1 Summary of Contributions

This research developed and validated a hierarchical hybrid deep learning model for brute-force attack detection in high-speed networks. The results demonstrates that hierarchical hybrid deep learning, combining CNN for spatial feature extraction and BiLSTM for temporal dependency learning, provides a robust and scalable solution for brute-force attack detection in high-speed networks. The model achieves exceptional precision (99.89%) while maintaining a low false positive rate (0.018%), addressing a critical operational challenge in network security. As cyber threats continue to evolve in sophistication, such intelligent, adaptive defense mechanisms will become increasingly essential for protecting organizational infrastructure. Beyond accuracy, the inclusion of Monte Carlo (MC) Dropout provides a measure of predictive uncertainty.

In future iterations, we aim to integrate eXplainable AI (XAI) frameworks, such as SHAP (SHapley Additive exPlanations) or Integrated Gradients, to visualize which specific network features (e.g., Inter-Arrival Time vs. Header Length) most significantly trigger the CNN's spatial filters. This will transform the 'black-box' nature of the hybrid model into a transparent tool for digital forensics. The key contributions are:

a) A novel CNN-BiLSTM architecture that achieves 99.27% accuracy, 99.89% precision, and 0.018% false positive rate on the CIC-IDS 2017 benchmark

b) An optimized PCA-based dimensionality reduction pipeline that reduces feature dimensionality from 82 to 18 while preserving 94.7% of discriminatory variance

c) Comprehensive statistical validation demonstrating significant improvement over the HAST-IDS baseline ($p < 0.05$)

d) Reliability assessment using Monte Carlo Dropout, yielding a predictive certainty score of 0.923

e) Demonstration of focal loss effectiveness for minority class detection, improving FTP-Patator F1-score by 5-8% over class-weighted cross-entropy.

5.2 Recommendations

5.2.1 Recommendations for Practice

Organizations seeking to implement deep learning-based intrusion detection should consider:

a) **Feature Engineering:** The PCA-derived feature set (18 components) provides an efficient representation that maintains high detection accuracy while reducing computational overhead

b) **Hybrid Architecture:** CNN-BiLSTM integration offers complementary benefits for detecting attacks with both spatial and temporal signatures

c) **False Alarm Management:** The low FPR (0.018%) enables automated response workflows without overwhelming security analysts

d) **Edge Deployment Considerations:** For resource-constrained environments, model quantization to INT8 precision enables CPU inference with minimal accuracy degradation

5.2.2 Future Research Directions

Several promising directions for future research emerge from this work:

a) **Encrypted Traffic Analysis:** Extending the architecture to handle fully encrypted protocols (TLS 1.3, QUIC) by focusing on metadata features including packet timing, size distributions, and handshake patterns [28]. Anaedevha, et al. [1] demonstrated that protocol-aware robustness certificates can enlarge certified radii by up to 58% for encrypted traffic, providing a theoretical foundation for this extension

b) **Federated Learning:** Implementing privacy-preserving collaborative training across organizations without centralizing sensitive network traffic data [29, 35]. Ghosh, et al. [24, 34] noted that "federated learning could enable decentralized model training, ensuring privacy and security while continuously adapting to evolving attack patterns across distributed devices." Target: Maintain >95% accuracy with $\varepsilon = 1$ differential privacy.

c) **Adversarial Robustness:** Investigating model vulnerability to adversarial examples and developing certified defenses. The preliminary evaluation (Section 4.8) showed PGD attack success rate of 7.88% at $\varepsilon = 0.05$. Target: Reduce to <3% using adversarial training or protocol-aware perturbation constraints following Anaedevha, et al. [1] .

d) **Zero-Day Detection:** Integrating few-shot learning capabilities to enable rapid adaptation to novel attack variants with minimal labeled examples [30]. Ghosh, et al. [24] achieved 98.56% accuracy on unseen attack types using meta-learning approaches. Target: Achieve >95% accuracy with 5-shot learning on novel brute-force variants.

e) **Explainability:** Visualizing and analyzing attention weights (or BiLSTM hidden states) to provide security analysts with interpretable explanations for detection decisions, potentially enabling faster threat investigation and model refinement.

f) **Class Imbalance Handling:** Further performance improvements, particularly for minority attack classes, could potentially be achieved by incorporating advanced class imbalance handling techniques such as SMOTE-ENN during preprocessing.

5.3 Relationship to Concurrent Research

During the preparation of this manuscript, several related advances were reported. Ghosh, et al. [24] proposed TACNet, achieving 99.98% accuracy on CICIDS 2018 using multi-scale CNN, LSTM, and temporal attention. Stephan and Mohammed [24] achieved 99.89% accuracy on ToN-IoT using DSST for imbalance handling and DenseNet169 for feature extraction. Li, et al. [25] proposed CBiNet achieving 98.49% on CICIDS2017 using 1D-CNN and BiLSTM.

The proposed model distinguishes itself through: (1) focus on brute-force attack detection with 99.78% class-specific F1-score, (2) PCA-based dimensionality reduction to 18 features (94.7% variance preserved), (3) rigorous statistical validation using Dietterich's 5x2cv test, and (4) comprehensive focal loss evaluation for imbalance handling. These complementary approaches suggest that hybrid spatial-temporal architectures represent a consensus direction for high-performance intrusion detection.

References

[1] Anaedevha, R.N., Trofimov, A.G. and Borodachev, Y.V., 2026. Hybrid Spatial-Temporal Deep Learning for Privacy-Preserving Encrypted Traffic Intrusion Detection. TechRxiv. [Preprint]. doi: 10.36227/TECHRXIV.176799976.66603504/V1.

[2] Shen, M. et al., 2023. Machine Learning-Powered Encrypted Network Traffic Analysis: A Comprehensive Survey. IEEE Communications Surveys & Tutorials, 25(1), pp. 791–824. doi: 10.1109/COMST.2022.3208196.

[3] Hajjouz, A. and Avksentieva, E., 2024. Evaluating the Effectiveness of the CatBoost Classifier in Distinguishing Benign Traffic, FTP BruteForce and SSH BruteForce Traffic. In: 2024 9th International Conference on Signal and Image Processing (ICSIP). pp. 351–358. doi: 10.1109/ICSIP61881.2024.10671552.

[4] Moustafa, N. and Slay, J., 2015. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS). doi: 10.1109/MILCIS.2015.7348942.

[5] Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A., 2009. A detailed analysis of the KDD CUP 99 data set. In: IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA 2009). doi: 10.1109/CISDA.2009.5356528.

[6] Hochreiter, S. and Schmidhuber, J., 1997. Long Short-Term Memory. Neural Computation, 9(8), pp. 1735–1780. doi: 10.1162/neco.1997.9.8.1735.

[7] Lecun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. Nature, 521(7553), pp. 436–444. doi: 10.1038/nature14539.

[8] Sharafaldin, I., Lashkari, A.H. and Ghorbani, A.A., 2018. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018). 2018-January, pp. 108–116. doi: 10.5220/0006639801080116.

[9] Susilo, B., Muis, A. and Sari, R.F., 2025. Intelligent Intrusion Detection System Against Various Attacks Based on a Hybrid Deep Learning Algorithm. Sensors, 25(2), p. 580. doi: 10.3390/s25020580.

[10] Gamage, S. and Samarabandu, J., 2020. Deep learning methods in network intrusion detection: A survey and an objective comparison. Journal of Network and Computer Applications, 169, p. 102767. doi: 10.1016/j.jnca.2020.102767.

[11] Fadlullah, Z.M., Tang, F., Mao, B., Kato, N., Akashi, O., Inoue, T. and Mizutani, K., 2017. State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems. IEEE Communications Surveys & Tutorials, 19(4), pp. 2432–2455. doi: 10.1109/COMST.2017.2707140.

[12] Xiao, Y., Xing, C., Zhang, T. and Zhao, Z., 2019. An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks. IEEE Access, 7, pp. 42210–42219. doi: 10.1109/ACCESS.2019.2904620.

[13] Alsallal, M. et al., 2026. Intelligent Network Behavior Anomaly Detection Using LSTM-Based Deep Learning Models. Internet Technology Letters, 9(3). doi: 10.1002/itl2.70279

[14] Imrana, Y., Xiang, Y., Ali, L. and Abdul-Rauf, Z., 2021. A bidirectional LSTM deep learning approach for intrusion detection. Expert Systems with Applications, 185, p. 115524. doi: 10.1016/j.eswa.2021.115524.

[15] Yuan, X., Wan, J., An, D. and Pei, H., 2025. A novel encrypted traffic detection model based on detachable convolutional GCN-LSTM. Scientific Reports, 15(1), p. 27705. doi: 10.1038/s41598-025-13397-2.

[16] Liu, Z., Xie, Y., Luo, Y., Wang, Y. and Ji, X., 2025. TransECA-Net: A Transformer-Based Model for Encrypted Traffic Classification. Applied Sciences, 15(6), p. 2977. doi: 10.3390/app15062977.

[17] Sinha, P., Sahu, D., Prakash, S., Yang, T., Rathore, R.S. and Pandey, V.K., 2025. A high-performance hybrid LSTM CNN secure architecture for IoT environments using deep learning. Scientific Reports, 15(1). doi: 10.1038/s41598-025-94500-5.

[18] Wang, Z., Chen, H., Yang, S., Luo, X., Li, D. and Wang, J., 2023. A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization. PeerJ Computer Science, 9, p. e1569. doi: 10.7717/peerj-cs.1569/supp-1.

[19] Azizjon, M., Jumabek, A. and Kim, W., 2020. 1D CNN based network intrusion detection with normalization on imbalanced data. In: 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). pp. 218–224. doi: 10.1109/ICAIIC48513.2020.9064976.

[20] Neto, E.C.P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R. and Ghorbani, A.A., 2023. CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. Sensors, 23(13), p. 5941. doi: 10.3390/s23135941.

[21] Upadhyay, S.K. and Vikas., 2026. Performance-Efficient Intrusion Detection for IoT Using CNN-BiLSTM and Incremental Principal Component Analysis. International Journal of Performability Engineering, 22(3), p. 128. doi: 10.23940/ijpe.26.03. p2.128137.

[22] Elreedy, D. and Atiya, A.F., 2019. A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance. Information Sciences, 505, pp. 32–64. doi: 10.1016/j.ins.2019.07.070.

[23] Lin, T.Y., Goyal, P., Girshick, R., He, K. and Dollar, P., 2017. Focal Loss for Dense Object Detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2999–3007. doi: 10.1109/ICCV.2017.324.

[24] Stephan, J.J. and Mohammed, Q.M., 2024. Using Hybrid Deep Learning Approach to Enhanced Network Intrusion Detection with Spatial-Temporal Feature Integration. Ingénierie des Systèmes d'Information, 29(4), pp. 1619–1628. doi: 10.18280/isi.290435.

[25] Abdelhamid, S., Hegazy, I., Aref, M. and Roushdy, M., 2024. Attention-Driven Transfer Learning Model for Improved IoT Intrusion Detection. Big Data and Cognitive Computing, 8(9), p. 116. doi: 10.3390/bdcc8090116.

[26] H., R., T., M., Park, J. and Ram, S., 2004. Design science in information systems research. MIS Quarterly. doi: 10.5555/2017212.2017217.

[27] Registry of Open Data on AWS, 2026. A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018). Available at: https://registry.opendata.aws/cse-cic-ids2018/ [Accessed 13 April 2026].

[28] Gahtan, B., Shahla, R.J., Cohen, R. and Bronstein, A.M., 2024. Exploring QUIC Dynamics: A Large-Scale Dataset for Encrypted Traffic Analysis. In: 2025 IEEE International Mediterranean Conference on Communications and Networking (MeditCom). doi: 10.1109/MeditCom64437.2025.11104435.

[29] McMahan, H.B., Moore, E., Ramage, D., Sampson, S. and Arcas, B.A.y., 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. arXiv preprint arXiv:1602.05629.

[30] Beg, R., Nigam, N. and Sharma, Y.K. et al., 2026. Design of an integrated evidence-driven few-shot meta-learning for zero-day malware detection and forensic attributions. Scientific Reports. doi: 10.1038/s41598-026-43745-9.

[31] Ahmad, B., Y. Li, Z. Wu, S. U. Rehman, and Y. Huang, 2026. Improved attack detection in IoT and IIoT networks using attention mechanisms in convolutional neural networks. Expert Systems with Applications, 296(Part C), Art. no. 129021. doi: 10.1016/j.eswa.2025.129021.

[32] Ghosh, K.P., Hasan, M., Robin, M.T.I., Hossain, A. and Islam, S., 2025. A novel deep learning framework with temporal attention convolutional networks for intrusion detection in IoT and IIoT networks. Scientific Reports, 15, Art. no. 44624. doi: 10.1038/s41598-025-32697-1.

[33] Elwhishi, A., Younis, A.A. and Akhunzada, A., 2026. Attention-enhanced hybrid architecture for efficient intrusion detection in Industrial IoT. IEEE Open Journal of the Communications Society, 7, pp. 1330–1339. doi: 10.1109/OJ-COMS.2026.3661768.

[34] L, G., Purbia, R., T, K. et al., 2026. IA-IDS: an intelligent adaptive intrusion detection system for IoT security using CNN, BiLSTM, and attention mechanism. Peer-to-Peer Networking and Applications, 19, Art. no. 32. doi: 10.1007/s12083-025-02177-4.

[35] R. Natarajan, S. Krishna and C. P. Ranjith, A Novel Federated Learning Framework for Healthcare Applications Using Wearable Devices, 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), Houston, TX, USA, 2025, pp. 1-6, doi: 10.1109/ICAIC63015.2025.10848974.