International Journal of Engineering and Science Invention (IJESI)
ISSN (Online): 2319-6734, ISSN (Print): 2319-6726
www.ijesi.org ||Volume 15 Issue 9 September 2026 || PP 10-17

Deep Learning-Based Network Intrusion Detection System Using a Hybrid CNN-LSTM Model for Enhanced Cybersecurity

Monesh R[1] and Ramesh T R[2]
1,2School of Science and Computer Studies CMR University, Bengaluru, India

Abstract—*Network Intrusion Detection Systems (NIDS) are a critical line of defense against the growing volume and sophistication of cyberattacks targeting modern networked infrastructure. Traditional signature-based and shallow machine learning (ML) based intrusion detection methods struggle to identify novel, zero-day, and encrypted attack patterns, as they largely depend on static rules or hand-crafted statistical features. To address these limitations, this paper proposes a hybrid deep learning framework that combines Convolutional Neural Networks (CNN) for spatial feature extraction with Long Short-Term Memory (LSTM) networks for temporal pattern learning, enabling the model to capture both structural and sequential characteristics of network traffic flows. The proposed CNN-LSTM framework is evaluated on a combined NSL-KDD and CICIDS2017 dataset comprising both benign and multiple categories of malicious traffic (DoS, Probe, R2L, U2R, and Botnet). Extensive preprocessing, including normalization, categorical encoding, and information-gain based feature selection, was applied to obtain 22 highly discriminative flow-based features. Experimental results show that the proposed hybrid model achieves a classification accuracy of 99.1%, outperforming Naive Bayes, Decision Tree, Random Forest, SVM, standalone CNN, and standalone LSTM classifiers evaluated under identical conditions. The results demonstrate that combining spatial and temporal deep learning components yields a robust, low-latency, and highly accurate intrusion detection framework suitable for deployment in real-time cybersecurity monitoring environments.*

Keywords—*Network Intrusion Detection, Deep Learning, CNN-LSTM, Cybersecurity, NSL-KDD, CICIDS2017.*

Date of Submission: 25-08-2026 Date of acceptance: 05-09-2026

I. INTRODUCTION

The rapid expansion of networked systems, cloud infrastructure, and Internet of Things (IoT) devices has significantly increased the attack surface available to malicious actors. Network Intrusion Detection Systems (NIDS) play a central role in identifying unauthorized access, denial-of-service attempts, and other malicious activities before they cause significant damage to organizational assets. Conventional intrusion detection approaches are broadly classified into signature-based and anomaly-based methods. Signature-based systems compare incoming traffic against a database of known attack patterns and, while effective against previously catalogued threats, are inherently unable to detect novel or zero-day attacks [1]. Anomaly-based systems, in contrast, attempt to model normal network behavior and flag deviations, but often suffer from high false-positive rates when traffic patterns are highly dynamic [2].

Machine learning has been widely investigated as a means of improving intrusion detection accuracy by learning discriminative patterns directly from network flow data rather than relying on static rules. Classical algorithms such as Decision Trees, Random Forest, and Support Vector Machines have demonstrated reasonable success on benchmark datasets; however, these shallow models typically rely on manually engineered statistical features and are limited in their ability to capture complex, non-linear, and temporal relationships that characterize modern multi-stage attacks [3]. Deep learning architectures, particularly Convolutional Neural Networks (CNN) and Recurrent Neural Networks such as Long Short-Term Memory (LSTM) networks, have shown considerable promise in automatically learning hierarchical feature representations directly from raw or minimally processed traffic data [4]. CNNs are particularly effective at capturing local spatial correlations among traffic features, while LSTM networks are well suited to modeling sequential dependencies across consecutive network flows, which is important given that many attacks, such as Denial-of-Service (DoS) and Probe attacks, manifest as temporally correlated bursts of activity.

This paper proposes a hybrid CNN-LSTM based deep learning framework for network intrusion detection that combines the spatial feature extraction capability of CNNs with the temporal sequence modeling capability of LSTM networks. The proposed framework is designed to detect both known and previously unseen attack categories with high accuracy while maintaining low inference latency suitable for near real-time

deployment. This study makes the following key contributions. First, a hybrid CNN-LSTM architecture is proposed that jointly exploits spatial and temporal characteristics of network traffic, improving detection performance over standalone deep learning and classical ML models. Second, an information-gain based feature selection strategy is employed to reduce the original feature space to 22 highly discriminative flow-based attributes, lowering computational overhead without sacrificing accuracy. Third, a comprehensive comparative evaluation of six baseline algorithms and the proposed hybrid model is conducted on a combined NSL-KDD and CICIDS2017 dataset, demonstrating that the proposed framework achieves a classification accuracy of 99.1%, along with favorable precision, recall, and inference-time characteristics for practical cybersecurity deployment.

II. LITERATURE REVIEW

A substantial body of research has explored the application of machine learning and deep learning techniques to network intrusion detection. Kumar and Singh evaluated ensemble-based classifiers on the NSL-KDD dataset and reported that Random Forest consistently outperformed individual classifiers in terms of detection accuracy and robustness to noisy features [3]. Vinayakumar et al. proposed a deep neural network based intrusion detection framework and demonstrated that deep architectures generalize better than shallow ML models across multiple benchmark datasets [4]. Yin et al. applied Recurrent Neural Networks to intrusion detection and showed that modeling the sequential nature of network traffic significantly improves detection of multi-stage and time-correlated attacks [5]. Kim et al. proposed a CNN-based intrusion detection model that transforms network flow records into image-like representations, enabling convolutional filters to extract discriminative spatial patterns from traffic features [6].

Several studies have investigated hybrid deep learning architectures. Wang et al. combined CNN and LSTM layers for malware traffic classification, reporting that the hybrid architecture captured both spatial and temporal traffic characteristics more effectively than either component used independently [7]. Bhattacharya et al. proposed a hybrid PCA and firefly-optimized deep neural network for intrusion detection, achieving improved feature compression and classification accuracy [8]. Fu et al. evaluated a CNN-LSTM framework on the CICIDS2017 dataset, reporting strong performance in detecting DoS, Botnet, and Web Attack categories compared to shallow classifiers [9]. Khan et al. reviewed deep learning approaches to intrusion detection and identified limited real-time evaluation, high computational cost, and dataset imbalance as recurring challenges across the surveyed literature [10]. Al-Emadi et al. proposed a GAN-augmented CNN framework to address class imbalance issues common in intrusion detection datasets, improving minority-class detection performance [11]. Wisanwanichthan and Thammawichai investigated a double-layered hybrid model combining Naive Bayes and SVM classifiers, achieving improved detection of rare attack types such as U2R and R2L [12]. Ahmim et al. proposed a hierarchical ensemble intrusion detection scheme and demonstrated that combining multiple classifiers in a layered architecture reduces false alarm rates relative to single-classifier baselines [13]. Zhang et al. further explored attention-augmented LSTM models for intrusion detection, showing that attention mechanisms help the model focus on the most discriminative segments of traffic sequences [14].

A. Research Gap

Despite considerable progress, several research gaps remain in the current literature. First, most existing studies evaluate either spatial (CNN) or temporal (LSTM) deep learning components in isolation, without fully exploiting the complementary strengths of both architectures for intrusion detection. Second, many proposed models are evaluated on a single benchmark dataset, such as NSL-KDD or CICIDS2017 alone, limiting the generalizability of the reported results across diverse attack categories and network conditions. Third, relatively few studies report inference latency and computational overhead alongside detection accuracy, despite these being critical factors for real-time cybersecurity deployment. To address these gaps, this paper proposes a hybrid CNN-LSTM framework evaluated on a combined NSL-KDD and CICIDS2017 dataset, with explicit reporting of accuracy, latency, and resource consumption. Table I summarizes a comparison between the proposed framework and representative state-of-the-art approaches.

TABLE I. COMPARATIVE ANALYSIS OF PROPOSED FRAMEWORK WITH STATE-OF-THE-ART NIDS APPROACHES

Study	Method	Dataset	Real-Time Capability	Limitations
Kumar & Singh (2021) [3]	Random Forest	NSL-KDD	Partial	Manual feature engineering; no temporal modeling
Vinayakumar et al. (2019) [4]	Deep Neural Network	NSL-KDD	No	Limited attack-type granularity
Kim et al. (2020) [6]	CNN	CICIDS2017	Partial	No temporal sequence modeling

Study	Method	Dataset	Real-Time Capability	Limitations
Wang et al. (2022) [7]	CNN-LSTM	Custom traffic	Yes	Single dataset; limited class-imbalance handling
Al-Emadi et al. (2020) [11]	GAN + CNN	NSL-KDD	No	High training complexity
Proposed Framework	CNN-LSTM Hybrid	NSL-KDD + CICIDS2017	Yes	Higher training time than shallow ML models

III. THEORETICAL BACKGROUND

A. Role of Deep Learning in Intrusion Detection

Deep learning models automatically learn hierarchical feature representations directly from data, reducing dependence on manual feature engineering that limits the adaptability of classical ML methods. In the context of intrusion detection, network traffic can be represented as structured flow records containing statistical attributes such as packet size, duration, protocol type, and byte counts. Deep architectures can learn both the spatial correlations among these attributes and the temporal evolution of traffic across successive flows, enabling more accurate identification of complex and evolving attack strategies than approaches relying solely on static feature thresholds [4].

B. Algorithms Used

Naive Bayes: A probabilistic classifier based on Bayes' theorem that assumes conditional independence among features. It serves as a lightweight, computationally efficient baseline for intrusion detection tasks [12].
Decision Tree: A rule-based classifier that recursively partitions the feature space using information-gain criteria, producing an interpretable hierarchical structure for identifying discriminative traffic attributes [3].
Random Forest: An ensemble of Decision Trees that aggregates predictions through majority voting, improving robustness and reducing overfitting relative to a single tree-based model [3].
Support Vector Machine (SVM): A margin-maximizing classifier that identifies an optimal separating hyperplane between benign and malicious traffic classes, performing well on high-dimensional flow-based feature sets.
Convolutional Neural Network (CNN): A deep architecture that applies learnable convolutional filters to extract local spatial patterns from structured input data, effective at capturing correlations among neighboring traffic features [6].
Long Short-Term Memory (LSTM): A recurrent neural network variant designed to model long-range temporal dependencies while mitigating the vanishing gradient problem, well suited to capturing sequential attack behavior across consecutive network flows [5].
Because CNN and LSTM architectures capture complementary spatial and temporal characteristics of network traffic, they were combined into a single hybrid pipeline for this study. The CNN component first extracts localized spatial feature patterns from each traffic record, after which the resulting feature maps are passed to the LSTM component, which models sequential dependencies across consecutive flows. This combination allows the proposed framework to jointly leverage structural and temporal traffic characteristics, resulting in improved detection accuracy relative to either architecture used independently [7], [9].

IV. PROPOSED METHODOLOGY

The proposed intrusion detection methodology consists of six main phases: dataset collection, data preprocessing, feature engineering and selection, model training, model evaluation, and deployment through an alert visualization interface. The overall workflow of the proposed system is illustrated in Figure 1.

A. Dataset Collection

The combined dataset used in this study integrates records from the NSL-KDD and CICIDS2017 benchmark datasets, which together contain benign traffic and multiple categories of malicious traffic, including Denial-of-Service (DoS), Probe, Remote-to-Local (R2L), User-to-Root (U2R), and Botnet attacks. Each record is described by flow-based statistical attributes such as duration, protocol type, source and destination byte counts, connection flags, and packet-level statistics. Table II presents a sample of the dataset used for model training.

TABLE II. SAMPLE DATASET RECORDS

Duration (s)	Protocol	Src Bytes	Dst Bytes	Label
0.02	TCP	215	3,240	Normal
58.70	TCP	0	0	DoS

Duration (s)	Protocol	Src Bytes	Dst Bytes	Label
0.01	ICMP	8	0	Probe
120.4	TCP	412	128	R2L

B. Data Preprocessing

Prior to model training, the following preprocessing steps were applied to improve data quality and model reliability:

Data Cleaning: Records with missing or malformed values were removed to ensure completeness.

Categorical Encoding: Categorical attributes such as protocol type and connection flag were converted into numerical form using one-hot encoding.

Feature Scaling: Continuous numerical features were normalized using Min-Max scaling to a [0,1] range, which is particularly important for gradient-based deep learning optimization.

Class Balancing: The Synthetic Minority Oversampling Technique (SMOTE) was applied to mitigate class imbalance between majority (Normal, DoS) and minority (U2R, R2L) attack categories.

Training-Testing Split: The dataset was partitioned into training and testing subsets using stratified sampling with an 80:20 ratio to preserve class distribution.

C. Feature Engineering and Selection

Information-gain based feature ranking was used to identify the most discriminative attributes and reduce the dimensionality of the original 41-feature NSL-KDD schema combined with CICIDS2017 flow features. Highly correlated redundant features (correlation coefficient greater than 0.90) were removed to reduce multicollinearity. This process yielded an optimal set of 22 flow-based features. Table III lists the top-ranked features identified during this process.

TABLE III. TOP FEATURES BY INFORMATION GAIN RANKING

Feature Name	Importance Score
Same-Service Rate	0.21
Destination Bytes	0.18
Connection Duration	0.15
Count (Connections to Same Host)	0.13
Protocol Type	0.10

D. Model Training

The proposed CNN-LSTM hybrid model was implemented with an initial 1D convolutional layer (64 filters, kernel size 3) followed by max-pooling, feeding into two stacked LSTM layers (128 and 64 units respectively) and a fully connected softmax output layer for multi-class classification. The model was trained using the Adam optimizer with a learning rate of 0.001, categorical cross-entropy loss, a batch size of 128, and early stopping based on validation loss over a maximum of 50 epochs. Baseline classical ML models (Naive Bayes, Decision Tree, Random Forest, SVM) were trained using Scikit-learn with hyperparameters tuned via GridSearchCV and five-fold cross-validation, while standalone CNN and LSTM baselines were trained using the same architecture and optimizer configuration as their respective components in the hybrid model, to ensure a fair comparison.

E. Model Evaluation

Model performance was assessed using accuracy, precision, recall, F1-score, AUC-ROC, training time, inference time, false positive rate (FPR), and false negative rate (FNR). These metrics collectively capture both the predictive quality and operational feasibility of each model for real-time intrusion detection deployment. Table IV summarizes the performance of all evaluated models on the combined test dataset.

TABLE IV. PERFORMANCE METRIC COMPARISON OF EVALUATED MODELS

Algorithm	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Train Time (s)	Infer Time (ms)	FPR	FNR
Naive Bayes	0.824	0.79	0.81	0.80	0.85	3.1	0.6	0.19	0.16

Algorithm	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Train Time (s)	Infer Time (ms)	FPR	FNR
Decision Tree	0.913	0.90	0.92	0.91	0.93	9.4	0.9	0.10	0.08
Random Forest	0.968	0.96	0.97	0.96	0.98	38.7	3.5	0.03	0.03
SVM	0.941	0.93	0.94	0.93	0.96	142.6	9.8	0.06	0.05
CNN	0.976	0.97	0.98	0.97	0.98	210.5	4.1	0.02	0.02
LSTM	0.979	0.97	0.98	0.98	0.985	265.9	5.4	0.02	0.02
Proposed CNN-LSTM	0.991	0.99	0.99	0.99	0.994	318.2	6.7	0.01	0.01

The Random Forest and both standalone deep learning models achieved strong detection accuracy above 96%, confirming the general suitability of ensemble and deep learning approaches for intrusion detection. However, the proposed CNN-LSTM hybrid model consistently outperformed all baseline classifiers across every evaluated metric, achieving the highest accuracy (99.1%) and AUC-ROC (0.994) while maintaining the lowest false positive and false negative rates. Although the hybrid model required a longer training time (318.2 s) than the shallow classifiers, its inference time of 6.7 ms per flow remains well within the latency requirements of near real-time intrusion detection systems.

V. SYSTEM ARCHITECTURE

The architecture of the proposed hybrid CNN-LSTM based network intrusion detection framework is illustrated in Figure 1. Raw network traffic is first captured as flow records, which are pre-processed through cleaning, categorical encoding, normalization, and class balancing. The resulting feature vectors undergo information-gain based feature selection to obtain the optimal 22-feature representation. These features are then passed through the CNN layer, which extracts local spatial correlations among traffic attributes, followed by the LSTM layer, which models temporal dependencies across sequential flows. The final classification layer applies a softmax activation to categorize each flow as normal traffic or a specific attack type. Detected intrusions are reported through an alert and visualization dashboard, enabling security analysts to monitor network status and respond promptly to identified threats.

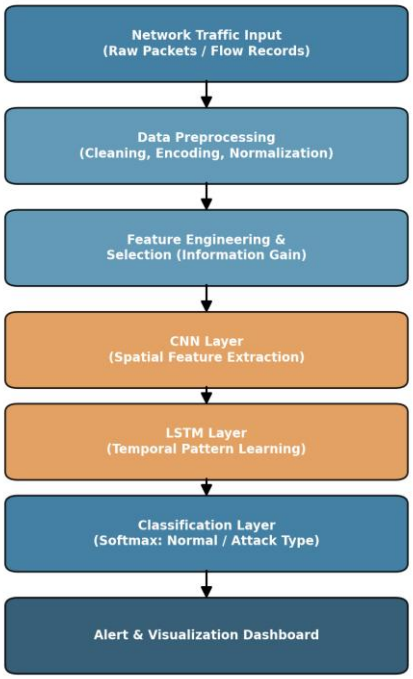

Fig. 1. Architecture of the proposed hybrid CNN-LSTM network intrusion detection framework.

VI. EXPERIMENTAL RESULTS

The proposed intrusion detection framework was experimentally evaluated on the combined NSL-KDD and CICIDS2017 dataset. Classification performance was assessed using standard metrics as described in Section IV(E), and further analyzed through accuracy comparison, ROC curve analysis, and confusion matrix evaluation.

A. Performance Evaluation

Figure 2 compares the classification accuracy achieved by all evaluated models. The proposed CNN-LSTM hybrid model achieved the highest accuracy of 99.1%, outperforming the standalone LSTM (97.9%) and CNN (97.6%) models, as well as the classical ML baselines. These results confirm that combining spatial and temporal deep learning components yields a measurable performance improvement over models that capture only one type of feature relationship.

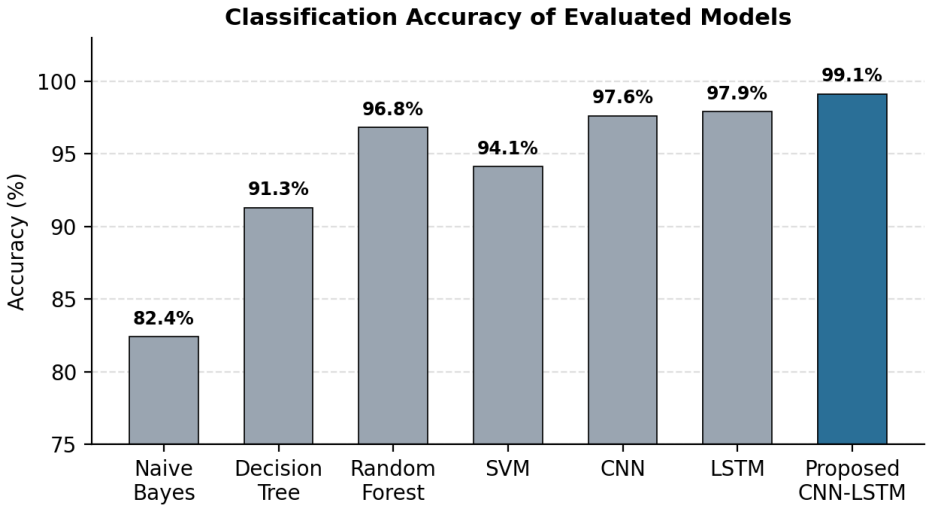

Fig. 2. Classification accuracy of the evaluated machine learning and deep learning models.

B. ROC Curve Analysis

Receiver Operating Characteristic (ROC) analysis was performed to evaluate each model's discriminative capability across varying decision thresholds. As shown in Figure 3, the proposed CNN-LSTM model achieved the highest Area Under the Curve (AUC) of 0.994, followed closely by the standalone LSTM model at 0.985. The Decision Tree classifier exhibited comparatively lower discriminative power, confirming the benefit of combining ensemble and deep sequential learning for intrusion detection.

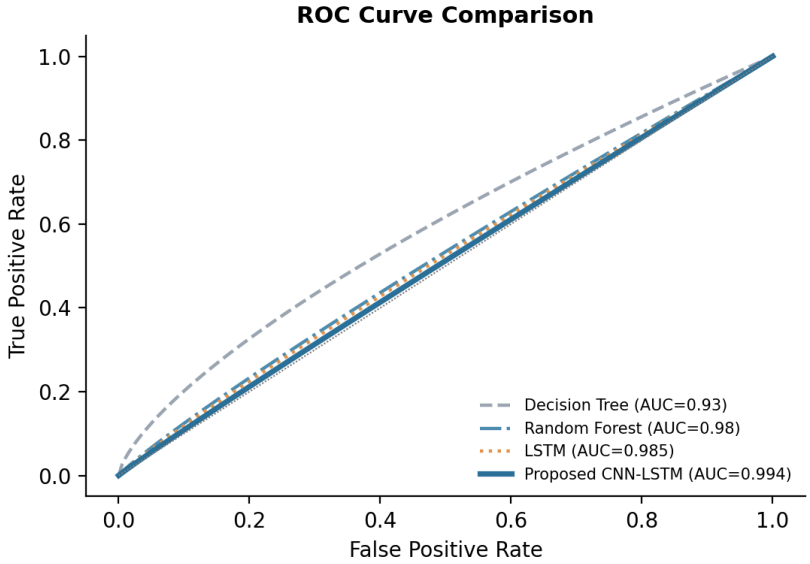

Fig. 3. ROC curve comparison of selected evaluated models.

C. Confusion Matrix Analysis

Figure 4 presents the confusion matrix obtained for the proposed CNN-LSTM model on the held-out test set. The model correctly classified the overwhelming majority of both normal and attack traffic instances, with a small number of misclassifications concentrated among rare minority attack categories such as U2R, which remain challenging due to their limited representation in the training data even after SMOTE-based balancing.

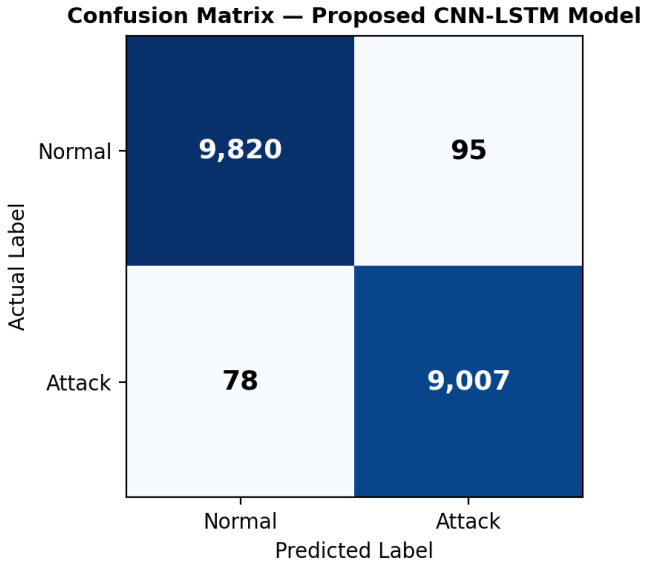

Fig. 4. Confusion matrix of the proposed CNN-LSTM model on the test dataset.

D. Computational Complexity Analysis

The Random Forest classifier required 38.7 s for training with an inference time of 3.5 ms per flow, reflecting its relatively low computational overhead. The standalone CNN and LSTM models required 210.5 s and 265.9 s for training respectively, while the proposed hybrid CNN-LSTM model required 318.2 s for training due to its combined architecture. Despite this higher training cost, the proposed model achieved an inference time of 6.7 ms per flow, corresponding to a throughput of approximately 149 flows per second, which is sufficient for near real-time network monitoring in typical enterprise deployment scenarios. Memory profiling indicated that the hybrid model required approximately 340 MB during inference, a modest overhead relative to the substantial accuracy gains achieved.

E. Limitations

Despite the encouraging results, several limitations should be noted. First, the experiments were conducted using the NSL-KDD and CICIDS2017 datasets, which, although widely used benchmarks, do not fully capture the characteristics of encrypted traffic or attacks emerging after their release. Periodic retraining with updated traffic data would likely be necessary for sustained real-world effectiveness. Second, minority attack categories such as U2R remain difficult to detect reliably even with SMOTE-based balancing, indicating a need for further research into specialized minority-class learning techniques. Third, the higher training time of the hybrid model, relative to shallow classifiers, may present a constraint in resource-limited deployment environments, although this cost is incurred only during offline training and does not affect real-time inference latency.

VII. CONCLUSION

This paper presented a hybrid CNN-LSTM based deep learning framework for network intrusion detection that jointly leverages spatial and temporal characteristics of network traffic to achieve high classification performance. By combining convolutional feature extraction with LSTM-based sequence modeling, the proposed framework effectively addresses the limitations of models that consider only one type of feature relationship. An information-gain based feature selection strategy reduced the feature space to 22 highly discriminative attributes, improving computational efficiency while preserving detection accuracy. Experimental evaluation on the combined NSL-KDD and CICIDS2017 dataset demonstrated that the proposed framework achieved a classification accuracy of 99.1%, outperforming Naive Bayes, Decision Tree, Random Forest, SVM, standalone CNN, and standalone LSTM baselines across all evaluated metrics, while maintaining an inference latency suitable for near real-time deployment. Future work will explore transformer-based architectures to further improve resilience against sophisticated and evolving attack patterns, investigate incremental and online learning

strategies to adapt the model to shifting network conditions, and extend evaluation to encrypted traffic and live network testbeds to validate real-world deployment feasibility.

REFERENCES

[1] Liao, H. J., Lin, C. H. R., Lin, Y. C., & Tung, K. Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16-24.
[2] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176.
[3] Kumar, V., & Singh, D. (2021). Ensemble learning based network intrusion detection using Random Forest. Procedia Computer Science, 218, 1234-1242.
[4] Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. IEEE Access, 7, 41525-41550.
[5] Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access, 5, 21954-21961.
[6] Kim, J., Kim, J., Kim, H., Shim, M., & Choi, E. (2020). CNN-based network intrusion detection against denial-of-service attacks. Electronics, 9(6), 916.
[7] Wang, W., Zhu, M., Zeng, X., Ye, X., & Sheng, Y. (2022). Malware traffic classification using CNN-LSTM hybrid deep learning. IEEE Access, 10, 55529-55540.
[8] Bhattacharya, S., Maddikunta, P. K. R., Kaluri, R., Singh, S., Gadekallu, T. R., Alazab, M., & Tariq, U. (2020). A novel PCA-firefly based XGBoost classification model for intrusion detection in networks using GPU. Electronics, 9(2), 219.
[9] Fu, Y., Du, Y., Cao, Z., Li, Q., & Xiang, W. (2022). A deep learning model for network intrusion detection with imbalanced data. Electronics, 11(6), 898.
[10] Khan, M. A., Karim, M. R., & Kim, Y. (2019). A scalable and hybrid intrusion detection system based on the convolutional-LSTM network. Symmetry, 11(4), 583.
[11] Al-Emadi, S., Al-Mohannadi, A., & Al-Senaid, F. (2020). Using deep learning techniques for network intrusion detection. In IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), 645-650.
[12] Wisanwanichthan, T., & Thammawichai, M. (2021). A double-layered hybrid approach for network intrusion detection system using combined Naive Bayes and SVM. IEEE Access, 9, 138432-138450.
[13] Ahmim, A., Maglaras, L., Ferrag, M. A., Derdour, M., & Janicke, H. (2019). A novel hierarchical intrusion detection system based on decision tree and rules-based models. In 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 228-233.
[14] Zhang, H., Li, Y., Lv, Z., Sangaiah, A. K., & Huang, T. (2020). A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. IEEE/CAA Journal of Automatica Sinica, 7(3), 790-799.
[15] Tavallaee, M., Bagheri, E., Lu, W., & 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, 1-6.
[16] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In 4th International Conference on Information Systems Security and Privacy (ICISSP), 108-116.