International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 20 No. 4 (2024)



ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 20, No.4, 2024
© 2024 IJERST. All Rights Reserved

Research Paper

An Optimized Spatial-Temporal Deep Learning Model for Network Intrusion Detection Using ConvLSTM and Bidirectional GRU

1. Rajaram Jatothu , Professor, Dept of Computer Science and Engineering ,Malla Reddy Engineering College for Women, Hyderabad, Telangana, India ,drjrajaram81@gmail.com.
2. K Venkata Naganjaneyulu ,Professor, Dept of Computer Science and Engineering ,Malla Reddy Engineering College for Women, Hyderabad, Telangana, India, kvnaganjaneyulu75@gmail.com
3. A.Sireesha,Assistant Professor , Dept of Computer Science and Engineering Teegala Krishna Reddy College of Engineering & Technology, Hyderabad, Telangana, India luckyreddysireesha1987@gmail.com

Abstract: The increasing sophistication of cyber-attacks and the rapid expansion of network communication have created significant challenges for conventional Network Intrusion Detection Systems (NIDS). Traditional machine learning approaches often struggle to identify complex intrusion patterns due to their limited ability to capture spatial and temporal dependencies in network traffic. To overcome these limitations, this work proposes an enhanced hybrid deep learning framework integrating Sequence-to-Sequence (Seq2Seq), Convolutional Long Short-Term Memory (ConvLSTM), Bidirectional Long Short-Term Memory (BiLSTM), and Gated Recurrent Unit (GRU) architectures for accurate network intrusion detection and classification. Initially, Random Forest is implemented as a baseline classification algorithm, followed by a hybrid Seq2Seq-ConvLSTM model for extracting spatial and sequential features from network traffic. As an extension, Bidirectional and GRU layers are incorporated to improve temporal dependency learning, enhance feature representation, and reduce computational latency. The proposed framework is evaluated using the UNSW-NB15 dataset through performance metrics including accuracy, precision, recall, and F1-score. Experimental results demonstrate that the Random Forest model achieves 90.31% accuracy, while the hybrid Seq2Seq-ConvLSTM model achieves 97.00% accuracy. The extended stacked model achieves the highest accuracy of 98.24%, with 98.41% precision, 95.59% recall, and 96.91% F1-score. Furthermore, Explainable Artificial Intelligence (XAI) using Local Interpretable Model-Agnostic Explanations (LIME) provides insights into important features influencing intrusion predictions. Finally, the trained model is deployed through a Flask-based web application, enabling users to upload network traffic datasets and obtain intrusion predictions

through an interactive interface. The proposed extension improves classification performance and computational efficiency, providing an effective approach for intelligent network intrusion detection.

Keywords: Network Intrusion Detection, Deep Learning, Seq2Seq, ConvLSTM, Bidirectional LSTM, GRU, Spatial-Temporal Feature Extraction, Explainable Artificial Intelligence, LIME, UNSW-NB15

1. INTRODUCTION

The rapid growth of internet connectivity, cloud computing, and digital communication has significantly increased the vulnerability of network infrastructures to various cyber threats. Modern networks continuously generate large volumes of traffic containing complex patterns associated with normal activities and malicious behaviors. Cyber-attacks such as Denial-of-Service (DoS), distributed denial-of-service (DDoS), malware, reconnaissance, and unauthorized access pose serious risks to data security, system availability, and network reliability. Therefore, developing an intelligent and efficient Network Intrusion Detection System (NIDS) has become essential for identifying and preventing malicious network activities.

Traditional Network Intrusion Detection Systems primarily depend on signature-based detection, predefined rules, and conventional machine learning algorithms. Although these approaches can identify previously known attack patterns, they often face difficulties in detecting sophisticated, unknown, and evolving cyber threats. Furthermore, conventional classification techniques have limited capabilities in learning complex spatial and temporal relationships from high-dimensional network traffic data. These limitations may lead to incorrect classifications, increased false alarm rates, and reduced detection efficiency in dynamic network environments.

With advancements in Artificial Intelligence (AI) and Deep Learning (DL), intelligent intrusion detection approaches have gained significant attention in cybersecurity research. Deep learning models can automatically extract relevant features from network traffic and identify complex patterns without extensive manual feature engineering. In particular, Sequence-to-Sequence (Seq2Seq) architectures are effective in learning sequential relationships, while Convolutional Long Short-Term Memory (ConvLSTM) networks combine convolutional feature extraction with temporal sequence learning. Integrating these architectures provides an opportunity to capture both spatial and temporal characteristics of network traffic, improving the identification of complex intrusion patterns.

In this work, a hybrid deep learning framework combining Seq2Seq and ConvLSTM architectures is considered for network intrusion detection. Random Forest is also employed as a baseline classification algorithm to evaluate the effectiveness of the proposed hybrid approach. The framework is evaluated using the UNSW-NB15 dataset, which contains different categories of normal and malicious network activities. Data preprocessing techniques, including categorical feature encoding, normalization, duplicate removal, and dataset splitting, are

applied to prepare the network traffic data for effective model training and evaluation.

As an extension, the existing Seq2Seq-ConvLSTM architecture is enhanced by incorporating Bidirectional and Gated Recurrent Unit (GRU) layers. The Bidirectional layer captures sequential dependencies in both forward and backward directions, enabling improved temporal feature representation. The GRU layer helps optimize sequential learning while reducing computational complexity and prediction latency. This combination aims to improve intrusion classification performance and computational efficiency compared with the existing hybrid architecture.

Furthermore, Explainable Artificial Intelligence (XAI) using Local Interpretable Model-Agnostic Explanations (LIME) is incorporated to identify important features influencing the model's intrusion detection decisions. The extended model is deployed through a Flask-based web application, allowing users to log in, upload network traffic test datasets, and obtain predicted intrusion categories through an interactive interface.

The experimental results demonstrate that the Random Forest model achieves 90.31% accuracy, whereas the existing Seq2Seq-ConvLSTM hybrid model achieves 97.00% accuracy. The extended Seq2Seq-ConvLSTM-Bidirectional-GRU model achieves an improved accuracy of 98.24%, demonstrating the effectiveness of incorporating advanced recurrent learning components for network intrusion detection.

Overall, the proposed extension provides an enhanced deep learning framework for identifying complex network intrusions with improved classification performance, temporal feature learning, computational efficiency, and prediction interpretability. The integration of Flask-based deployment further supports practical utilization in intelligent and real-time network security applications

2. LITERATURE SURVEY

1. Harshavardhan et al. (2025)

In their study, "Hybrid Deep Learning Framework for Intrusion Detection: Integrating CNN, LSTM, and Attention Mechanisms to Enhance Cybersecurity," the authors proposed a hybrid deep learning framework combining CNN, LSTM, and attention mechanisms. CNN layers were used to extract spatial features, while LSTM captured temporal dependencies in network traffic. The framework also incorporated Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) for feature reduction and selection. Experimental results on the UNSW-NB15 dataset achieved 97.89% accuracy. However, the model involves considerable computational complexity and may face challenges in real-time deployment.

2. Yadulla et al. (2023)

In their research, "Enhancing Cybersecurity with AI: Implementing a Deep Learning-Based Intrusion Detection System Using Convolutional Neural Networks," the authors investigated deep learning techniques for identifying malicious network activities. The approach utilized CNN-based feature extraction and hybrid CNN-LSTM architectures to capture spatial and sequential characteristics of network traffic. The system demonstrated improved detection accuracy, precision, and reduced

false-positive rates compared with conventional machine learning approaches. However, the framework requires significant computational resources and may experience limitations when processing highly imbalanced datasets. A Hybrid Deep Learning Model for Network Intrusion Detection System Using Seq2Seq and ConvLSTM-Subnets - PPT 3.pptx

3. Moukhafi et al. (2024)

In their work, "Intelligent Intrusion Detection Through Deep Autoencoder and Stacked Long Short-Term Memory," the authors introduced an intrusion detection framework combining a deep autoencoder and stacked LSTM networks. The autoencoder was employed to extract meaningful features and reduce noise from network traffic data, while stacked LSTM layers captured complex temporal dependencies for intrusion classification. The proposed approach achieved approximately 90.59% accuracy on the UNSW-NB15 dataset. However, the model requires considerable training time and computational resources, particularly when processing large-scale network traffic.

4. Nazre et al. (2024)

In their study, "A Temporal Convolutional Network-Based Approach for Network Intrusion Detection," the authors proposed a Temporal Convolutional Network (TCN) incorporating residual blocks and dilated convolutions to capture long-range temporal dependencies in network traffic. The architecture enables parallel sequence processing and improves computational efficiency compared with conventional recurrent models. Experimental evaluation using the Edge-IIoTset dataset achieved 96.72% accuracy. However, the approach depends on sufficient labeled training data and may face challenges in generalizing across different network environments and detecting previously unseen attack patterns.

5. Liu et al. (2024)

In their research, "EESNN: Hybrid Deep Learning Empowered Spatial-Temporal Features for Network Intrusion Detection System," the authors proposed a hybrid deep learning architecture to improve spatial and temporal feature extraction. The framework integrated DenseNet169 for spatial feature extraction, a self-attention Transformer for temporal dependency learning, and an Enhanced Elman Spiking Neural Network for final classification. Data imbalance was addressed using sampling and data augmentation techniques. Experimental results demonstrated improved accuracy, precision, recall, and reduced false alarm rates across multiple intrusion detection datasets. However, the combination of multiple deep learning components increases computational complexity and makes real-time deployment challenging.

3. METHODOLOGY

i) Proposed Work:

The proposed work focuses on developing an enhanced hybrid deep learning framework for Network Intrusion Detection Systems (NIDS) to accurately identify and classify malicious network activities. The framework integrates Sequence-to-Sequence (Seq2Seq) and Convolutional Long Short-Term Memory (ConvLSTM) architectures to capture spatial and temporal dependencies in network traffic data. Initially, Random Forest is implemented as a baseline classification algorithm, followed by the

hybrid Seq2Seq-ConvLSTM model to evaluate the effectiveness of deep learning-based intrusion detection.

As an extension, the existing hybrid architecture is enhanced by incorporating Bidirectional and Gated Recurrent Unit (GRU) layers. The Bidirectional layer processes sequential network traffic information in both forward and backward directions, enabling the extraction of more comprehensive temporal features. The GRU layer improves sequential learning efficiency and helps reduce computational overhead and prediction latency. By integrating these components, the extended architecture aims to improve classification performance and efficiently identify complex network intrusion patterns.

The UNSW-NB15 dataset is utilized for training and evaluating the proposed framework. Initially, data preprocessing techniques such as missing-value handling, categorical feature encoding, duplicate removal, normalization, and train-test splitting are performed to prepare the dataset. The processed network traffic features are then supplied to the baseline and hybrid deep learning models for training and classification.

Furthermore, Explainable Artificial Intelligence (XAI) using Local Interpretable Model-Agnostic Explanations (LIME) is incorporated to identify the important features influencing intrusion detection predictions. This improves the interpretability and transparency of the classification process. Finally, the trained extended model is deployed using a Flask-based web application, allowing authenticated users to upload test datasets and obtain predicted intrusion categories through an interactive interface.

The experimental evaluation demonstrates that the Random Forest model achieves 90.31% accuracy, while the existing Seq2Seq-ConvLSTM model achieves 97.00% accuracy. The extended Seq2Seq-ConvLSTM-Bidirectional-GRU architecture achieves 98.24% accuracy, with 98.41% precision, 95.59% recall, and 96.91% F1-score. These results demonstrate the improvement achieved by the extended architecture over the existing hybrid model.

ii) System Architecture:

The proposed system architecture presents an enhanced hybrid deep learning framework for Network Intrusion Detection using the UNSW-NB15 dataset. Initially, network traffic data is collected and preprocessed through missing-value handling, categorical feature encoding, duplicate removal, normalization, and train-test splitting. Random Forest is implemented as a baseline classifier, followed by a hybrid Seq2Seq-ConvLSTM model to capture sequential and spatial-temporal features from network traffic. As an extension, Bidirectional LSTM and Gated Recurrent Unit (GRU) layers are integrated into the existing hybrid architecture to enhance temporal dependency learning in both forward and backward directions while improving computational efficiency. The extracted features are passed through dense layers for final intrusion classification into normal and malicious traffic categories. Furthermore, Explainable Artificial Intelligence (XAI) using LIME is employed to identify important features influencing prediction outcomes. The trained extended

model is evaluated using accuracy, precision, recall, and F1-score and deployed through a Flask-based web application, enabling users to upload network traffic datasets and obtain intrusion prediction results through an interactive interface. The extended model achieves 98.24% accuracy, demonstrating improved classification performance compared with the existing Seq2Seq-ConvLSTM model..

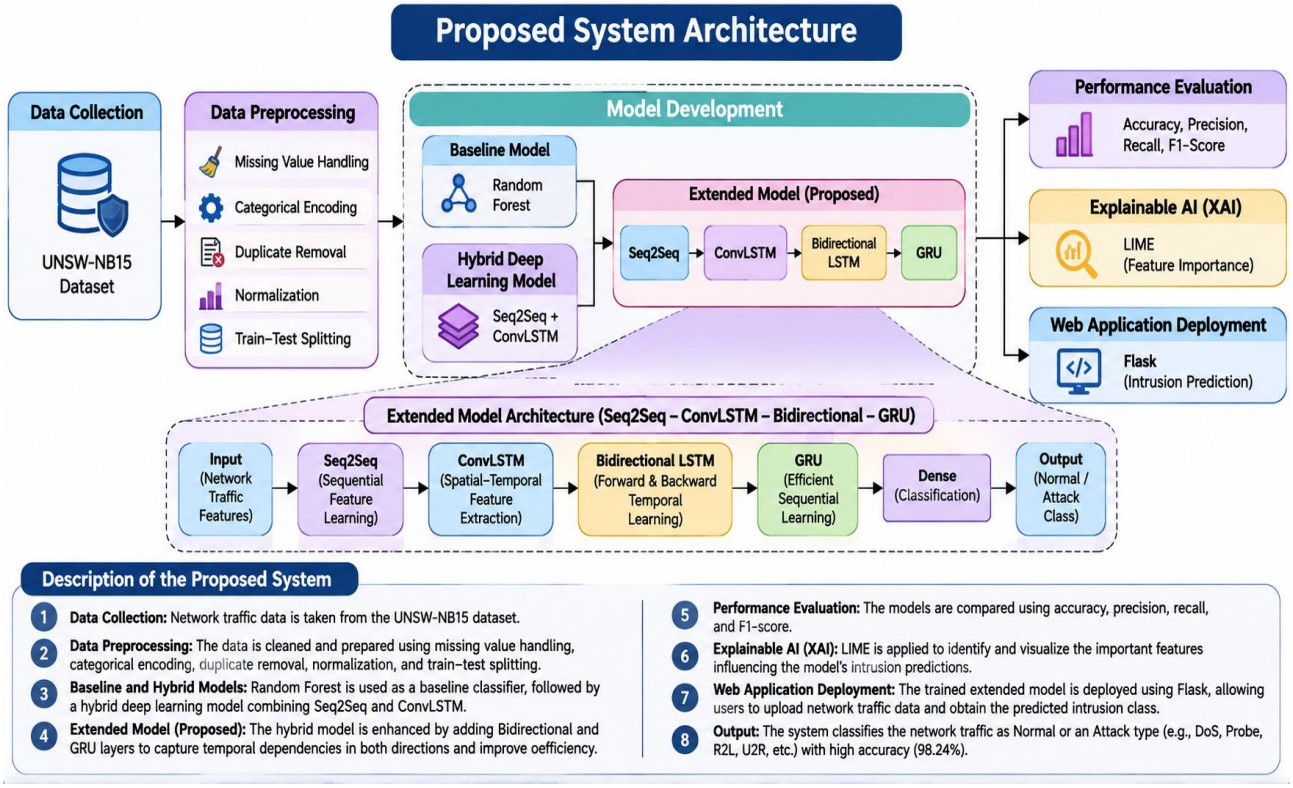


Fig 1 System Architecture

iii) MODULES:

1. Importing Required Packages

This module imports the necessary Python libraries and frameworks required for data processing, visualization, deep learning model development, performance evaluation, explainability, and web application deployment.

2. Dataset Collection and Exploration

This module loads the UNSW-NB15 dataset and examines its structure, attributes, data types, and class distribution. It helps identify network traffic characteristics and understand different categories of normal and malicious activities.

3. Data Visualization

This module generates graphical representations of network traffic features, attack distributions, and feature correlations. Visualization helps understand dataset characteristics and identify important patterns associated with different intrusion categories.

4. Data Preprocessing

This module prepares the collected network traffic data by handling missing values, converting categorical attributes into numerical representations using encoding techniques, removing duplicate records, and eliminating irrelevant features.

5. Data Normalization

This module scales numerical network traffic features into a consistent range to prevent large-valued attributes from dominating the learning process. Normalization improves model training stability and supports efficient feature learning.

6. Training and Testing Data Splitting

This module divides the preprocessed dataset into training and testing subsets, typically using an 80:20 ratio. The training data is used for model development, while the testing data evaluates classification performance on unseen samples.

7. Baseline Model Training

This module implements the Random Forest algorithm as a baseline classifier for network intrusion detection. It uses multiple decision trees to classify network traffic and provides a performance reference for evaluating the proposed deep learning architectures.

8. Hybrid Deep Learning Model Development

This module develops the existing Seq2Seq-ConvLSTM architecture by combining sequential learning and convolutional feature extraction. Seq2Seq captures sequential dependencies, while ConvLSTM extracts spatial-temporal features from network traffic data.

9. Extended Hybrid Model Development

This module enhances the existing Seq2Seq-ConvLSTM architecture by incorporating Bidirectional LSTM and GRU layers. Bidirectional LSTM captures temporal dependencies in both forward and backward directions, while GRU improves sequential learning efficiency and helps reduce computational complexity and prediction latency.

10. Performance Evaluation

This module evaluates and compares Random Forest, the existing hybrid model, and the extended hybrid model using accuracy, precision, recall, and F1-score. The extended model achieves 98.24% accuracy, demonstrating improved classification performance.

11. Explainable Artificial Intelligence (LIME)

This module integrates Local Interpretable Model-Agnostic Explanations (LIME) to identify and analyze important network traffic features influencing intrusion classification decisions. It improves the interpretability and transparency of the prediction process.

12. Flask Web Application

This module deploys the trained extended deep learning model using the Flask framework. It provides an interactive web interface through which users can access the intrusion detection functionalities.

13. User Authentication

This module manages secure user login by validating usernames and passwords. It restricts unauthorized access and allows authenticated users to access the intrusion detection application.

14. Intrusion Threat Prediction

This module allows users to upload network traffic test datasets through the web interface. The uploaded data is processed by the trained extended hybrid model, and the predicted intrusion categories are displayed for user analysis.

15. Logout

This module securely terminates the user's session after completing the prediction

process and prevents unauthorized access to the application

v) ALGORITHMS:

1. Random Forest Algorithm (Baseline Model)

Random Forest is an ensemble machine learning algorithm that combines multiple decision trees to perform classification tasks. In the proposed network intrusion detection system, Random Forest is used as a baseline model to classify network traffic into normal and malicious categories. It works by constructing multiple decision trees using different subsets of training data and features, and the final prediction is determined through majority voting. The algorithm is suitable for handling high-dimensional network traffic data, reducing overfitting, and identifying important features influencing intrusion classification. It provides a reference for comparing the performance of the proposed hybrid deep learning models.

2. Sequence-to-Sequence (Seq2Seq) Algorithm

Sequence-to-Sequence (Seq2Seq) is a deep learning architecture designed to process sequential input data and learn relationships between different time steps. It generally consists of an encoder and a decoder, where the encoder captures relevant information from the input sequence and the decoder generates the corresponding output representation. In the proposed system, Seq2Seq is used to learn sequential dependencies in network traffic data and identify patterns associated with malicious activities. It helps the intrusion detection framework understand complex traffic sequences and supports accurate classification of different attack categories.

3. Convolutional Long Short-Term Memory (ConvLSTM) Algorithm

ConvLSTM is a hybrid deep learning architecture that combines convolutional operations with Long Short-Term Memory networks. Unlike conventional LSTM, ConvLSTM performs convolutional operations within its recurrent structure to capture spatial and temporal information simultaneously. In the proposed system, ConvLSTM extracts spatial-temporal features from network traffic representations and identifies complex relationships among traffic attributes. By integrating convolutional feature extraction with sequential learning, it improves the ability to recognize sophisticated intrusion patterns.

4. Bidirectional Long Short-Term Memory (BiLSTM) Algorithm

Bidirectional Long Short-Term Memory (BiLSTM) is an extension of the traditional LSTM architecture that processes sequential information in both forward and backward directions. It consists of two LSTM networks that learn dependencies from past-to-future and future-to-past contexts. In the extended intrusion detection framework, BiLSTM is integrated into the existing Seq2Seq-ConvLSTM architecture to capture additional temporal information from network traffic sequences. This improves feature representation and helps the model recognize complex sequential attack behaviors more effectively.

5. Gated Recurrent Unit (GRU) Algorithm

Gated Recurrent Unit (GRU) is a recurrent neural network architecture designed to

learn sequential dependencies with fewer gating components than traditional LSTM networks. It uses update and reset gates to control information flow and retain relevant information across time steps. In the proposed extension, GRU is integrated after the Bidirectional layer to improve sequential feature learning and computational efficiency. It helps reduce model complexity and prediction latency while maintaining effective temporal information processing for network intrusion classification.

6. Local Interpretable Model-Agnostic Explanations (LIME)

LIME is an Explainable Artificial Intelligence (XAI) technique used to interpret machine learning and deep learning predictions. It explains individual predictions by approximating the behavior of a complex model using an interpretable local model. In the proposed system, LIME identifies the network traffic features that contribute to intrusion classification decisions. It provides insights into why a particular traffic record is classified as normal or malicious, improving model transparency and supporting cybersecurity analysis.

7. Extended Hybrid Deep Learning Algorithm

The extended hybrid algorithm integrates Seq2Seq, ConvLSTM, Bidirectional LSTM, and GRU architectures into a unified network intrusion detection framework. Initially, Seq2Seq learns sequential relationships from network traffic data, while ConvLSTM extracts spatial-temporal features. The extracted representations are further processed using Bidirectional LSTM to capture dependencies in both directions, followed by GRU for efficient sequential learning. Finally, dense layers perform the classification of network traffic into different intrusion categories. The extended model achieves 98.24% accuracy, with 98.41% precision, 95.59% recall, and 96.91% F1-score on the reported experimental evaluation

4. EXPERIMENTAL RESULTS

The proposed network intrusion detection framework is evaluated using the UNSW-NB15 dataset to analyze the effectiveness of the baseline machine learning model, existing hybrid deep learning model, and extended hybrid architecture. The experiments are conducted by applying data preprocessing, categorical encoding, normalization, and train-test splitting before model training. Random Forest is initially implemented as a baseline classifier, followed by the Seq2Seq-ConvLSTM hybrid model and the extended Seq2Seq-ConvLSTM-Bidirectional-GRU architecture. The performance of these models is measured using accuracy, precision, recall, and F1-score to evaluate their classification effectiveness

Accuracy: A test's accuracy is defined as its ability to recognize debilitated and solid examples precisely. To quantify a test's exactness, we should register the negligible part of genuine positive and genuine adverse outcomes in completely examined cases. This might be communicated numerically as:

Accuracy = TP + TN TP + TN + FP + FN.

$$Accuracy = \frac{(TN + TP)}{T}$$

Precision: Precision measures the proportion of properly categorized occurrences or samples among the positives.

As a result, the accuracy may be calculated using the following formula:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$Accuracy = \frac{(TN + TP)}{T}$$

Recall: Recall is a machine learning metric that surveys a model's capacity to recognize all pertinent examples of a particular class. It is the proportion of appropriately anticipated positive perceptions to add up to real up-sides, which gives data about a model's capacity to catch instances of a specific class.

$$Accuracy = \frac{(TN + TP)}{T}$$

F1-Score: The F1 score is a machine learning evaluation measurement that evaluates the precision of a model. It consolidates a model's precision and review scores. The precision measurement computes how often a model anticipated accurately over the full dataset.

$$Accuracy = \frac{(TN + TP)}{T}$$

Test Data	Predicted Intrusion Type
[[[-0.33331764]] [[-0.33333333]] [[-0.33333333]] [[-0.21835552]] [[-0.33333333]] [[-0.65172717]] [[-3.]] [[-0.33333333]] [[-0.39222837]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33321779]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[0.01310153]] [[-0.33333333]] [[0.]] [[0.]] [[-0.44962547]] [[-2.65495395]] [[-0.36452763]] [[-0.3506581]] [[-0.31642315]] [[-0.41071765]] [[0.]] [[0.]] [[0.]] [[-0.45558672]] [[-0.39570649]] [[0.]]]	Normal
[[[3.]] [[3.]] [[3.]] [[1.31217385]] [[3.]] [[-1.47920111]] [[0.33333333]] [[3.]] [[-0.76373337]] [[3.]] [[3.]] [[3.]] [[2.99999993]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[3.]] [[-0.81666202]] [[3.]] [[0.]] [[0.]] [[-0.94920932]] [[-1.09321633]] [[-0.61592599]] [[-0.60112817]] [[-1.0195857]] [[-0.60174912]] [[0.]] [[0.]] [[0.]] [[-0.70869045]] [[-0.84537297]] [[0.]]]	DoS
[[[-0.3333438]] [[-0.33333333]] [[-0.33333333]] [[-0.54486846]] [[-0.33333333]] [[-0.72697218]] [[0.33333333]] [[-0.33333333]] [[-0.50788292]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33317053]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.33333333]] [[-0.3362726]]	Analysis

Fig 2 The above figure illustrates the output of the proposed network intrusion detection system after processing the uploaded test dataset. The system analyzes the input network traffic features using the trained extended hybrid deep learning model and displays the corresponding predicted intrusion types. The results show different classifications, including Normal, DoS, and Analysis, indicating the model's ability to distinguish legitimate network traffic from malicious activities. The test data values are displayed in the left column, while the predicted intrusion categories are presented in the right column. Different text colors are used to distinguish the prediction results, providing a clear and user-friendly representation of intrusion detection outcomes. This demonstrates the successful integration of the trained model with the Flask-based web application for network traffic classification

5. CONCLUSION

The proposed work presents an enhanced hybrid deep learning framework for Network Intrusion Detection by integrating Seq2Seq, ConvLSTM, Bidirectional LSTM, and GRU architectures. The framework effectively captures spatial and temporal dependencies in network traffic data to identify and classify different types of cyber-attacks. Initially, Random Forest is implemented as a baseline model, followed by the existing Seq2Seq-ConvLSTM architecture and the extended hybrid model. Experimental evaluation using the UNSW-NB15 dataset demonstrates that the Random Forest model achieves 90.31% accuracy, while the existing hybrid model achieves 97.00% accuracy. The extended model achieves the highest accuracy of 98.24%, with 98.41% precision, 95.59% recall, and 96.91% F1-score. The integration of Bidirectional and GRU layers enhances temporal feature learning and improves overall classification performance. Furthermore, LIME provides explainable insights into important features influencing intrusion predictions, while Flask-based deployment enables users to upload network traffic datasets and obtain predicted intrusion categories through an interactive web interface. Overall, the experimental results demonstrate that the extended hybrid framework provides an effective approach for accurate and interpretable network intrusion detection, supporting improved cybersecurity monitoring and threat classification.

REFERENCES

[1] A. Harshavardhan, M. Sree Vani, A. Patil, N. Yamsani, and K. Archana, “Hybrid Deep Learning Framework for Intrusion Detection: Integrating CNN, LSTM, and Attention Mechanisms to Enhance Cybersecurity,” Journal of Theoretical and Applied Information Technology, vol. 103, no. 1, 2025. [Online]. Available: https://www.jatit.org/volumes/Vol103No1/5Vol103No1.pdf

[2] A. R. Yadulla, V. K. Kasula, M. Yenugula, and B. Konda, “Enhancing Cybersecurity with AI: Implementing a Deep Learning-Based Intrusion Detection System Using Convolutional Neural Networks,” European Journal of Advances in Engineering and Technology, vol. 10, no. 12, pp. 89–98, 2023. [Online]. Available: https://ejaet.com/PDF/10-12/EJAET-10-12-89-98.pdf

[3] M. Moukhafi, M. Tantaoui, I. Chana, and A. Bouazi, "Intelligent Intrusion Detection Through Deep Autoencoder and Stacked Long Short-Term Memory," International Journal of Electrical and Computer Engineering, vol. 14, no. 3, pp. 2908–2917, 2024. DOI: 10.11591/ijece.v14i3.pp2908-2917.

[4] R. Nazre, R. Budke, O. Oak, S. Sawant, and A. Joshi, "A Temporal Convolutional Network-Based Approach for Network Intrusion Detection," in Proceedings of the 2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS), 2024. DOI: 10.1109/ICIICS63763.2024.10860234.

[5] J. Saikam and K. Ch, "EESNN: Hybrid Deep Learning Empowered Spatial-Temporal Features for Network Intrusion Detection System," IEEE Access, vol. 12, pp. 15930–15945, 2024. DOI: 10.1109/ACCESS.2024.3350197.

[6] N. Moustafa and J. Slay, "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), 2015.

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?' Explaining the Predictions of Any Classifier," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144, 2016. DOI: 10.1145/2939672.2939778.