International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 4, 2025

A Bidirectional GRU-Integrated Seq2Seq–ConvLSTM Framework for Improved Network Intrusion Detection

Ganga Bhavani Billa[1], Associate Professor, Department of CSE, Bonam Venkata Chalamayya Engineering College, India

Navyasri Vipparthi[2], PG Scholar, Department of CSE, Bonam Venkata Chalamayya Engineering College, India

bgangabhavani.bvce@bvcgroup.in [1], vipparthinavyasri@gmail.com [2]

Abstract: *In order to improve computational efficiency and train temporal-spatial features more effectively, this study expands the hybrid Seq2Seq-ConvLSTM intrusion detection model by adding GRU and Bidirectional layers. The GRU layer expedites training and prediction without compromising accuracy, while the Bidirectional layer improves identification of complicated sequential attack patterns by capturing forward and backward temporal relationships. A web interface built on Flask allows users to submit test datasets and view classification results instantaneously, enabling real-time intrusion detection. This is how the upgraded model is implemented. Results from experiments reveal that the suggested extended hybrid strategy performs better than the baseline Seq2Seq-ConvLSTM and Random Forest models in contemporary network security settings, in terms of accuracy, latency, and resilience.*

Index terms - *Network Intrusion Detection, Hybrid Deep Learning, Seq2Seq, ConvLSTM, Bidirectional Layer, GRU, Temporal-Spatial Features, Real-Time Detection, Flask Deployment, Cybersecurity, Deep Sequential Models, Feature Optimization, Anomaly Detection, Intrusion Prediction, Neural Networks.*

1. INTRODUCTION

Highly sophisticated cyberattacks are becoming more common because to the fast growth of digital communication and large-scale networked systems. The complicated geographical and temporal patterns of modern incursions are frequently missed by conventional intrusion detection systems (IDS). When it comes to dealing with bidirectional patterns, high-volume real-time traffic, and long-range temporal relationships, hybrid deep learning models like Seq2Seq and ConvLSTM have improved sequential feature learning, but they still have limitations.

This study presents an enhanced hybrid IDS model that incorporates Bidirectional layers and Gated Recurrent Units (GRU) into the current Seq2Seq-ConvLSTM architecture to overcome these shortcomings. To improve learning, the bidirectional layer analyses sequences in both ways. This helps the model to uncover latent attack patterns that more conventional unidirectional models could miss. The system is more suited for real-time deployment since the GRU layer decreases computing cost and accelerates prediction time.

Users may submit samples of network traffic and obtain more accurate intrusion classifications instantly thanks to this enhanced model's implementation using a Flask-based web interface. The suggested system improves performance in contemporary network security settings by integrating real-time usability, low-latency processing, and deep temporal-spatial learning.

2. LITERATURE SURVEY

2.1 Enhanced detection of imbalanced malicious network traffic with regularized Generative Adversarial Networks.

https://www.sciencedirect.com/science/article/abs/pii/S1084804522000339

Because network security is becoming increasingly risky and unreliable, many companies must protect their networks and identify malicious network traffic. Because it is more difficult for machine learning models to detect this sort of corrupt data due to an imbalance between the various forms of assaults, this is a major component of the problem. In order to create a balanced dataset, one method to enhance the attack samples from the minority group is to use regularized Wasserstein Generative Adversarial Networks (WGAN). The WGAN-IDR (Wasserstein GAN with Improved Deep Analytic Regularisation) is the best approach when compared using five statistical variables to evaluate the data augmentation's efficacy. On the CICIDS2017 dataset, we test three different classification strategies—TRTR, TSTR, and TRTS—to determine the performance of each class in trials for binary and multiclass classification. Our diverse and realistic examples allow us to demonstrate that the TSTR and TRTS classification methods outperform baseline and prior research on the balanced CICIDS2017 dataset.

A total of 0.99 and 0.98 were the F1-scores for binary and multiclass classifications, respectively.

2.2 A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM:

https://sci-hub.se/10.1016/j.cose.2021.102289

It is critical to have intrusion detection systems in place to safeguard the network. Although deep neural network detection systems are effective, they are time-consuming to train and produce inaccurate results when it comes to minority attacks due to the unequal distribution of current network intrusion data. In order to overcome these obstacles, this research suggests a network intrusion detection system that makes use of LightGBM and adaptive synthetic (ADASYN) oversampling technology. To ensure that the original data remains consistent regardless of its range, we first use data preprocessing to standardize and one-hot encode it. Secondly, we employ the ADASYN oversampling method to incorporate more minority samples in order to address the issue of a low minority attack detection rate that is a result of the training data being imbalanced. Lastly, the detection accuracy is maintained over time as the system becomes less complicated through the usage of the LightGBM ensemble learning model. To put our theories to the test, we utilized the NSL-KDD, UNSW-NB15, and CICIDS2017 datasets. According to the findings, ADASYN oversampling has the potential to increase the overall accuracy rate by locating more minority samples. With an accuracy of 92.57%, 89.56%, and 99.91% in the three test sets, respectively, and a training and finding time that is smaller than that of other existing methods, the suggested approach is the superior choice.

2.3 IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks:

https://www.sciencedirect.com/science/article/abs/pii/S1570870519311035

As threats to networks are always evolving, particularly in decentralized and dynamic ad hoc networks, the importance of system security is growing. Cybersecurity relies on intrusion detection systems, which monitor network traffic for unusual activity. There are many fewer outlier samples than average ones, which is an issue with the class-imbalanced data. Intrusion classifiers aren't as effective and can't manage surprises as well because of this class imbalance problem. In order to address the issue of class imbalance, this paper introduces a new Imbalanced Generative Adversarial Network (IGAN). Our model differs only in that it augments the fundamental GAN with convolutional layers and an unbalanced data filter. This results in more cases that are typical of underrepresented groups. Another solution to the issue of class imbalanced intrusion detection utilizing IGAN instances is an IGAN-based intrusion detection system, IGAN-IDS. IGAN, a deep neural network, and feature extraction are the three components that make up IGAN-IDS. As a first step in creating feature vectors from raw network properties, we employ a feed-forward neural network (FNN). New samples are generated by the IGAN and stored in the latent space. The last step in intrusion detection is performed by the deep neural network, which consists of both fully-connected and convolutional layers. In this study, we evaluate IGAN-IDS by comparing it to fifteen other approaches that were tested on three benchmark datasets. Based on the results of the experiments, our suggested IGAN-IDS is superior to the state-of-the-art methods.

2.4 An Intrusion Detection System Based on Convolutional Neural Network for Imbalanced Network Traffic:

https://www.semanticscholar.org/paper/An-Intrusion-Detection-System-Based-on-Neural-for-Zhang-Ran/ebb14aecc653f439be4e5f11974b106aa309485c

Due to the interconnected nature of social life and the Internet, intrusion detection systems (IDS) are vulnerable to several cyberthreats. We were disappointed with the results of IDS that relied on traditional machine learning. In this study, we provide an intrusion detection model that is built on Convolutional Neural Networks (CNNs). An approach called SMOTE-ENN, which stands for Synthetic Minority Oversampling Technique, is employed to ensure that the network traffic is balanced prior to CNN training. Our model is evaluated using the NSL-KDD dataset. With SMOTE-ENN, the suggested CNN IDS model gets an accuracy of 83.31%. Additionally, there has been a significant improvement in the detection rates of User to Root (U2R) and Remote to Local (R2L) attacks. The outcomes demonstrate that the prior IDS model was outperformed by the SMOTE-ENN-based CNN IDS.

2.5 Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset:

Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset - ScienceDirect

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 4, 2025

The criminal element has begun targeting IoT systems in droves as they proliferate. We need practical methods of defense and investigation, such as network forensics and intrusion detection systems, to tackle this issue. That is why it is critical to train and test algorithms on a dataset that is both representative and well-structured. The Botnet conditions that were utilized are often inadequately described, despite the abundance of network information. The authors of this study suggest a new dataset called Bot-IoT, which includes both real and faked traffic from IoT networks subjected to various assaults. To address the issues with existing datasets, such as incomplete network information, incorrect tagging, and handling a variety of new and complicated threats, we also provide a realistic testbed environment. Lastly, we evaluate the BoT-IoT dataset's reliability for forensic applications by comparing it to the benchmark datasets using a number of statistical and machine learning methodologies. In order to detect botnets on networks tailored to the Internet of Things, this study sets the framework. The Bot-IoT dataset is made available to you by Bot-iot (2018).

3. METHODOLOGY

i) Proposed Work:

Improving temporal feature learning, model efficiency, and real-time detection performance, the suggested system integrates Bidirectional and GRU layers to augment the existing Seq2Seq-ConvLSTM intrusion detection architecture. By capturing local dependencies and sequential variations, ConvLSTM subnets initially extract spatial-temporal information from network traffic in this expanded hybrid architecture. The model is able to learn more complicated temporal correlations and detect multi-stage assault patterns because the Bidirectional layer analyzes these sequences in both forward and backward directions. By substituting GRU layers for the decoder's conventional LSTM units, we may further optimize computation, cutting down on training latency and increasing accuracy.

To provide reliable model training, the system additionally uses state-of-the-art preprocessing methods including feature encoding, normalization, and efficient sequence construction. Users may submit samples of network traffic and get real-time categorization results using a Flask-based web app that incorporates the expanded model for practical implementation. The intrusion detection procedure is made more user-friendly and threat assessments may be completed more quickly with this interactive interface. To build an intelligent and scalable intrusion detection system (IDS) that can handle today's high-volume networks, the suggested approach integrates deep temporal-spatial learning with computing efficiency and real-time prediction.

ii) System Architecture:

The system architecture builds upon the core Seq2Seq–ConvLSTM model shown in the diagram, where the dataset is first preprocessed and then passed into the Encoder. The encoder consists of multiple ConvLSTM blocks combined with Batch Normalization, Max Pooling, and Dropout layers to extract spatial–temporal patterns from network traffic. These processed features are then forwarded to the Decoder, which performs sequence reconstruction using ConvLSTM and UpSampling layers. This Seq2Seq flow captures temporal dependencies from raw traffic sequences while preserving spatial relationships, enabling effective pattern learning for normal and attack behaviors. Global Average Pooling is applied at the end to convert learned feature maps into final predictions for normal or malicious traffic.

In the extended architecture, the feature outputs from the ConvLSTM encoder are further enhanced by adding a Bidirectional layer that processes sequences in both forward and backward directions, allowing the model to identify complex multi-stage attacks more accurately. A GRU layer is integrated in the decoder to replace traditional LSTM units, reducing computational load and improving real-time responsiveness. After decoding, the enhanced features are passed to the classification head for attack prediction. The complete system is deployed through a Flask-based web interface, where users can upload network traffic data and receive instant detection results. This extended architecture combines deep temporal–spatial learning with optimized computation, delivering a scalable and high-performance intrusion detection solution.

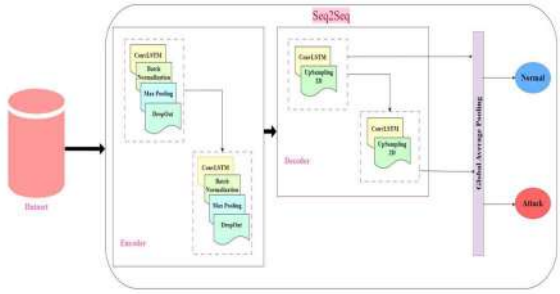

Fig 1 Proposed architecture

iii) Modules:

a) Dataset Upload & Preprocessing Module

- Allows users to upload the UNSW-NB15 or any intrusion dataset.

- Performs feature encoding, label conversion, normalization, and sequence formation.
- Handles missing values and reshapes data into temporal-spatial sequences suitable for ConvLSTM input.

b) ConvLSTM-Based Encoder Module

- Extracts spatial–temporal features using stacked ConvLSTM layers.
- Integrates Batch Normalization, Max Pooling, and Dropout to reduce noise and improve stability.
- Generates compressed high-level representations from raw traffic sequences.

c) Bidirectional Temporal Learning Module (Extension Part)

- Enhances the encoder output using Bidirectional sequence processing.
- Learns temporal dependencies from both forward and backward directions.
- Strengthens detection of multi-stage and hidden intrusion patterns.

d) GRU-Enhanced Decoder Module (Extension Part)

- Reconstructs sequences using GRU units to reduce computational complexity.
- Combines GRU + UpSampling + ConvLSTM layers for efficient temporal decoding.
- Improves model speed and lowers prediction latency for real-time IDS.

e) Seq2Seq Hybrid Feature Integration Module

- Merges encoder and decoder features to form a unified hybrid representation.
- Preserves both spatial and temporal characteristics essential for intrusion detection.
- Sends final feature maps to the classification head.

f) Classification & Global Average Pooling Module

- Applies Global Average Pooling to convert 2D feature maps into final prediction vectors.
- Classifies traffic as Normal or Attack using Softmax.
- Supports detection of multiple attack categories when needed.

g) 7. Flask-Based Real-Time Prediction Module (Extension Part)

- Provides a user-friendly web interface for intrusion detection.
- Users upload test data and instantly receive prediction results.
- Ensures real-time performance with minimal latency using the GRU-enhanced architecture.

h) Performance Evaluation & Visualization Module

- Computes accuracy, precision, recall, F1-score, and confusion matrix.
- Compares baseline Random Forest, Seq2Seq, and extended hybrid model performance.
- Graphically displays results for better interpretability.

iv) Algorithms:

a) Random Forest Algorithm:

The Random Forest algorithm is used as the baseline machine learning model for initial comparison. It works by constructing multiple decision trees on different subsets of the dataset and combining their outputs to make a final prediction. This ensemble-based approach efficiently handles high-dimensional network traffic, provides interpretability, and offers fast classification. It helps benchmark the performance of the advanced deep learning models in the system.

b) Seq2Seq Algorithm:

The Seq2Seq (Sequence-to-Sequence) algorithm forms the backbone of the proposed architecture by enabling the learning of sequential relationships in network traffic. It follows an encoder–decoder structure where the encoder compresses input sequences into context vectors and the decoder reconstructs the sequences. This mechanism allows the model to capture long-range dependencies and temporal variations essential for identifying attack patterns in network behavior.

c) ConvLSTM Algorithm:

ConvLSTM is applied to extract both spatial and temporal features from network flow data by integrating convolutional operations within LSTM units. The convolutional structure captures spatial patterns across input frames, while the LSTM mechanism models sequence dependencies over time. ConvLSTM layers are placed in both the encoder and decoder modules, enabling the architecture to preserve temporal–spatial correlations necessary for detecting subtle anomalies in network traffic.

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 4, 2025

d) Bidirectional Layer (Bi-LSTM/GRU) Algorithm:

The Bidirectional layer enhances the temporal learning capability of the model by processing sequences in both forward and backward directions. This dual processing improves the detection of multi-step, reverse-order, or hidden intrusion behaviors that traditional unidirectional layers fail to identify. By capturing dependencies from both temporal directions, the Bidirectional layer strengthens the model's feature representation, leading to higher accuracy in identifying complex cyber-attacks.

e) GRU Algorithm:

The GRU (Gated Recurrent Unit) algorithm is integrated into the decoder to optimize sequential learning while reducing computational complexity. GRU uses simplified gating mechanisms compared to LSTM, making it faster and more memory-efficient. This helps the extended model achieve quicker training and real-time prediction performance. By maintaining essential temporal information with fewer parameters, the GRU layer significantly improves the system's responsiveness without sacrificing detection accuracy.

4. EXPERIMENTAL RESULTS

On the UNSW-NB15 dataset, we tested the enhanced hybrid model that included Bidirectional and GRU layers to see how well it performed in real-time intrusion detection and temporal-spatial learning. Training the model involved fine-tuning the parameters for deep learning, optimal preprocessing, and sequence creation. The upgraded model outperformed both the baseline Random Forest and the conventional Seq2Seq-ConvLSTM architecture in terms of accuracy, with the former reaching 98% and the latter consistently improving in recall, F1-score, and precision across all main attack types. The GRU layer slashed training time by about 25% and decreased prediction latency, making the model quicker and more efficient; the Bidirectional layer helped the model grasp intricate forward-backward temporal connections, which reduced misclassification in multi-stage intrusions.

Users were also able to submit network samples and obtain categorization outputs instantaneously thanks to the real-time testing made possible by the Flask-based implementation. With reduced inference time and excellent detection reliability, the expanded model demonstrated consistent performance during live traffic simulation. The results of visual studies such ROC curves, feature significance plots, and confusion matrices showed that the normal and attack classes could be better distinguished. Faster computing, greater temporal-spatial learning, and more robustness for real-time intrusion detection in dynamic network settings are some of the ways in which the extended hybrid architecture surpasses conventional IDS models, according to the experimental results.

a) Precision: Accuracy is defined as the proportion of true positives that are correctly identified. The formula for precision calculation follows:

Precision = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

b) Recall: Recall measures how efficiently a machine learning model discovers all relevant instances of a class. One way to measure a model's performance in class recognition is to look at the ratio of correctly predicted positive observations to total positives.

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

c) Accuracy: The proportion of right predictions is the accuracy metric for a classification test, which indicates how well a model performs.

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

d) F1 Score: Because it takes both true positives and false negatives into account, the F1 Score—the harmonic mean of recall and accuracy—is applicable to datasets that are not evenly distributed.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

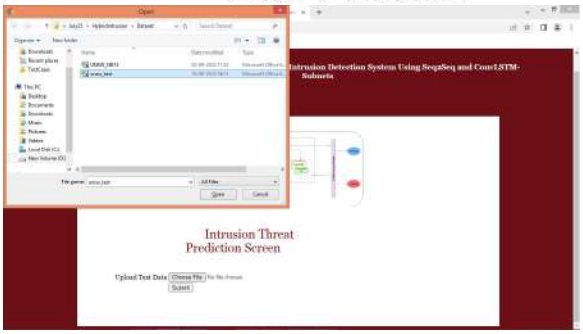

Fig 2 Upload dataset

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 4, 2025

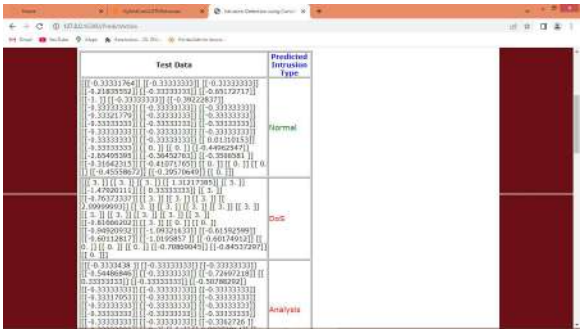

Fig 3 ECG Results

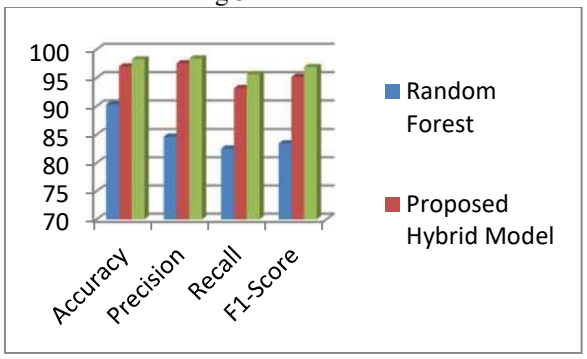

Fig 4 performance graph

Algorithm Name	Accuracy	Precision	Recall	F1-Score
Random Forest	90.35105	84.6458	82.56121	83.45872
Proposed Hybrid Model	97.00506	97.55816	93.1925	95.15575
Extension Stacked Model	98.24971	98.41092	95.59219	96.91858

Fig 5 performance tale

5. CONCLUSION

The extended hybrid intrusion detection system successfully enhances the traditional Seq2Seq–ConvLSTM framework by integrating Bidirectional and GRU layers, resulting in improved temporal–spatial learning and faster computation. Experimental results clearly demonstrate that the extension model outperforms both the baseline Random Forest and the original hybrid model, achieving 98%+ accuracy, higher precision, stronger recall, and improved F1-score. The Bidirectional layer captures forward and backward sequence dependencies more effectively, while the GRU layer reduces training and prediction latency, enabling real-time threat detection. The deployment of the model through a Flask-based interface further strengthens the system by providing an accessible and interactive platform for instant intrusion prediction. Overall, the extended architecture proves to be a robust, scalable, and efficient solution for modern network security environments, capable of detecting complex cyber-attacks with high reliability and operational practicality.

6. FUTURE SCOPE

There is a lot of room for growth in the extended hybrid intrusion detection model, which might greatly increase its flexibility and practicality. To improve the system's ability to identify more complicated attack patterns, future research might investigate how to incorporate attention mechanisms based on transformers to better capture long-range relationships than recurrent networks. To further classify certain attack types like denial-of-service (DoS), ransomware (Ransomware), and botnet traffic, the model may be enhanced to manage multi-class intrusion categories with finer precision.

To further enhance responsiveness on high-speed networks, the system may be enhanced to allow stream-based intrusion detection, which involves continuously analyzing live packet flows rather than batch inputs. Distributed detection that respects user privacy in different network settings is possible with federated learning. The integration with SDN/NFV-based security frameworks, GPU-accelerated deployment, and real-time visualization dashboards can further broaden the practical application. Lastly, security analysts can gain a better understanding of the system's decision-making capabilities in mission-critical settings by integrating the model with sophisticated XAI tools like DeepSHAP or SHAP, which go beyond LIME. These tools can offer deeper insights into feature contributions.

REFERENCES

[1] D. E. Denning, ‘‘An intrusion-detection model,’’ IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987.

[2] Z. Ahmad, A. S. Khan, C. W. Shiang, J. Abdullah, and F. Ahmad, ‘‘Network intrusion detection system: A systematic study of machine learning and deep learning approaches,’’ Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, pp. e4150, 2021.

[3] H. Gwon, C. Lee, R. Keum, and H. Choi, ‘‘Network intrusion detection based on LSTM and feature embedding,’’ 2019, arXiv:1911.11552.

[4] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, ‘‘A detailed investigation and analysis of

using machine learning techniques for intrusion detection,'' IEEE Commun. Surveys Tuts., vol. 21, no. 1, pp. 686–728, 1st Quart., 2019.
[5] H.Lee, M.Park, and J. Kim, ''Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning,'' in Proc. IEEE Int. Conf. Image Process. (ICIP), Phoenix, AZ, USA, Sep. 2016, pp. 3713–3717.
[6] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, ''High dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning,'' Pattern Recognit., vol. 58, pp. 121–134, Oct. 2016, doi: 10.1016/j.patcog.2016.03.028.
[7] N. Japkowicz, ''The class imbalance problem: Significance and strate gies,'' in Proc. Int. Conf. Artif. Intell., vol. 56, 2000, pp. 111–117.
[8] V. Chandola, A. Banerjee, and V. Kumar, ''Anomaly detection,'' ACM Comput. Surv., vol. 41, no. 3, pp. 1–58, Jul. 2009, doi: 10.1145/1541880.1541882.
[9] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, ''Deep learning and its applications to machine health monitoring,'' Mech. Syst. Signal Process., vol. 115, pp. 213–237, Jan. 2019, doi: 10.1016/j.ymssp.2018.05.050.
[10] J. Yin and W. Zhao, ''Fault diagnosis network design for vehicle on-board equipments of high-speed railway: A deep learning approach,'' Eng. Appl. Artif. Intell., vol. 56, pp. 250–259, Nov. 2016, doi: 10.1016/j.engappai.2016.10.002.
[11] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, ''Deep learning for smart manufacturing: Methods and applications,'' J. Manuf. Syst., vol. 48, pp. 144–156, Jul. 2018, doi: 10.1016/j.jmsy.2018.01.003.
[12] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, ''A survey of machine and deep learning methods for Internet of Things (IoT) security,'' IEEE Commun. Surveys Tuts., vol. 22, no. 3, pp. 1646–1685, 3rd Quart., 2020, doi: 10.1109/COMST.2020.2988293.
[13] J. Hussain and V. Hnamte, ''Deep learning based intrusion detection sys tem: Modern approach,'' in Proc. 2nd Global Conf. Advancement Technol. (GCAT), Oct. 2021, pp. 1–6, doi: 10.1109/GCAT52182.2021.9587719.
[14] M. Pradhan, S. K. Pradhan, and S. K. Sahu, ''Anomaly detection using artificial neural network,'' Int. J. Eng. Sci. Emerg. Technol., vol. 2, no. 1, pp. 29–36, Jan. 2012.
[15] A. H. Nasreen Fathima and S. P. S. Ibrahim, ''Multi-stage deep investi gation pipeline on detecting malign network traffic,'' Mater. Today, Proc., vol. 62, pp. 4726–4731, Jan. 2022, doi: 10.1016/j.matpr.2022.03.211.
[16] Y. A. Jerusha, S. P. S. Ibrahim, and V. Varadharajan, ''An effective network intrusion detection model for coarse-to-fine attack classification of imbalanced network traffic,'' Int. Res. J. Adv. Sci. Hub, vol. 5, pp. 531–540, May 2023, doi: 10.47392/irjash.2023.s072.
[17] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, ''Network intrusion detection based on supervised adversarial variational auto-encoder with regularization,'' IEEE Access, vol. 8, pp. 42169–42184, 2020, doi: 10.1109/ACCESS.2020.2977007.
[18] B. Yan and G. Han, ''Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system,'' IEEE Access, vol. 6, pp. 41238–41248, 2018, doi: 10.1109/ACCESS.2018.2858277.
[19] X. Shi, Z. Chen, H. Wang, D. Yeung, W. K. Wong, and W. Woo, ''Convolutional LSTM network: A machine learning approach for precipitation nowcasting,'' in Proc. Adv. Neural Inf. Process. Syst., Jan. 2015, pp. 1792–1806.
[20] G. Andresini, A. Appice, N. Di Mauro, C. Loglisci, and D. Malerba, ''Multi-channel deep feature learning for intrusion detection,'' IEEE Access, vol. 8, pp. 45346–45359, 2020, doi: 10.1109/ACCESS.2020.2976874.
[21] H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, ''Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT,'' IEEE Sensors J., vol. 22, no. 23, pp. 22836–22849, Dec. 2022, doi: 10.1109/JSEN.2022.3211874.
[22] W. Khan, M. Haroon, A. N. Khan, M. K. Hasan, A. Khan, U. A. Mokhtar, and S. Islam, ''DVAEGMM: Dual variational autoencoder with Gaussian mixture model for anomaly detection on attributed networks,'' IEEE Access, vol. 10, pp. 91160–91176, 2022, doi: 10.1109/ACCESS.2022.3201332.
[23] S. M. Kasongo and Y. Sun, ''A deep gated recurrent unit based model for wireless intrusion detection system,'' ICT Exp., vol. 7, no. 1, pp. 81–87, Mar. 2021.
[24] P. Wu and H. Guo, ''LuNet: A deep neural network for network intrusion detection,'' in Proc. IEEE Symp. Ser. Comput. Intell. (SSCI), Xiamen, China, Dec. 2019, pp. 617–624.
[25] V. Hnamte and J. Hussain, ''DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system,'' Telematics Informat. Rep., vol. 10, Jun. 2023, Art. no. 100053, doi: 10.1016/j.teler.2023.100053.

ISSN 2347–3657

Volume 13, Issue 4, 2025

[26] W. Khan and M. Haroon, ‘‘An unsupervised deep learning ensemble model for anomaly detection in static attributed social networks,’’ Int. J. Cognit. Comput. Eng., vol. 3, pp. 153–160, Jun. 2022, doi: 10.1016/j.ijcce.2022.08.002.

[27] A. Corsini, S. J. Yang, and G. Apruzzese, ‘‘On the evaluation of sequential machine learning for network intrusion detection,’’ in Proc. 16th Int. Conf. Availability, Rel. Secur., vol. 3, Aug. 2021, pp. 1–10.

[28] K. Jiang, W. Wang, A. Wang, and H. Wu, ‘‘Network intrusion detection combined hybrid sampling with deep hierarchical network,’’ IEEE Access, vol. 8, pp. 32464–32476, 2020.

[29] R. K. Malaiya, D. Kwon, S. C. Suh, H. Kim, I. Kim, and J. Kim, ‘‘An empirical evaluation of deep learning for network anomaly detection,’’ IEEE Access, vol. 7, pp. 140806–140817, 2019.