International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2s) 2026 pp. 1194-1204
https://doi.org/10.70917/ijcisim-2026-2223
Article

A HYBRID DEEP LEARNING FRAMEWORK FOR ENHANCED THREAT DETECTION IN WIRELESS SENSOR NETWORKS

Bikash Kalita[1], Satyajit Sarmah[2]*, Bijay Kumar Singh[3], Surajit Medhi[4], Deepjyoti Kalita[5], Amkar Brahma[6]

[1]Department of Information Technology, Gauhati University, Guwahati, India, bikax99@gmail.com
[2]Department of Information Technology, Gauhati University, Guwahati, India, ss@gauhati.ac.in
[3]Department of Information Technology, Assam Skill University, Mangaldai, India, bijay.kumar.singh@outlook.com
[4]Department of Information Technology, B. Borooah College, Guwahati, India, surajitmdh@gmail.com
[5]Department of Computer Science & Information Technology, Mangaldai College, Mangaldai, India, deepjyoti111@gmail.com
[6]Department of Computer Science & Information Technology, Bodoland University, Kokrajhar, India, amkarbrahma9773@gmail.com
Corresponding Author: Satyajit Sarmah (Email: (ss@gauhati.ac.in))

Abstract: Wireless Sensor Networks (WSN) have become a crucial technology in various areas. They are adopted across a wide range of distributed and data-driven applications. Due to the decentralized and resource-constrained nature of WSN, these systems are vulnerable to various security threats. These systems are susceptible to multiple forms of malicious activities that disrupt communication and degrade network performance. Traditional methods are unable to recognize the intricate temporal and spatial patterns found in network traffic data. So, these systems frequently fail to detect advanced threats. For effective intrusion detection, this paper presents a hybrid deep learning framework that combines Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and an attention mechanism. High-level spatial features are extracted from network traffic via the CNN component. Additionally, temporal dependencies between sequential data are captured by the LSTM. By giving important time steps adaptive priority, the attention mechanism improves the model's detection accuracy for dynamic attack behaviors. The WSN-DS dataset is used in this experiment. The model obtains an overall accuracy of 98.75% along with high performance in other metrics across several attack classes. The research results show that the proposed design performs noticeably better than both independent Deep Learning (DL) models and traditional Machine Learning (ML) techniques.

Keywords: WSN, Network Threats, Deep Learning, CNN-LSTM, Cybersecurity.

1. INTRODUCTION

There are Several security threats occurring from the rapid growth of Wireless Sensor Networks (WSNs) deploying in various applications (Sekhar & Sarvabhatla, 2012). WSN are very vulnerable to different network security threats like jamming, spoofing, sinkhole, and blackhole attacks because they are tiny, energy-constrained sensor nodes that communicate wirelessly (Pundir & Sandhu, 2021). Due to the resource constraints and distributed nature of the WSNs, traditional security systems like encryption and authentication are typically insufficient to protect these networks from advanced attack types (Pourrahmani et al., 2023) Because they can identify only known attacks. So the network security threat detection systems, or IDS, are essential to secure WSN.

In the network security threat detection systems, deep learning techniques have boosted up the efficiency of security threat detection systems by enabling the efficient and precise detection of threats (Aldhaheri et al., 2024).

Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) model can be a great system for detecting security threats. The LSTM system enable the detection of temporal dependencies in sequential data, while the CNN are primarily useful for extracting high-level spatial features (Yamashita et al., 2018). So this hybrid model can be the best model as the use of a hybrid model is the best way to comprehend WSN traffic behaviour, particularly when various attack types are present (Al-Selwi et al., 2024).

In order to address these limitations, this study proposes an enhanced deep learning-based intrusion detection framework that focuses on improving the intrinsic learning capability of the model rather than relying on external preprocessing techniques. The proposed approach integrates a multi-scale convolutional feature extraction mechanism with a Long Short-Term Memory network to capture both spatial and temporal characteristics of WSN traffic. Additionally, an attention mechanism is incorporated to dynamically focus on critical temporal features. Here a focal loss function is employed to improve the learning of hard-to-classify instances. This combination enables the model to effectively detect complex and stealthy attacks in WSN environments. **SIGNIFICANCE AND CONTRIBUTION**

The major contributions of this research are summarized as follows:

a) A hybrid CNN–LSTM IDS with an attention mechanism is proposed for WSN environments.

b) Sequential feature structuring is employed to preserve temporal relationships in WSN traffic data.

c) The attention mechanism enhances temporal feature prioritization for improved attack detection.

d) Comprehensive experimental analysis is performed using the WSN-DS dataset.

e) Comparative evaluation demonstrates superior performance over conventional machine learning and existing deep learning approaches.

2. RELATED WORK

For cybersecurity, the network security threat detection systems have been extensively developed to protect WSN from a variety of cyber-related attacks (Sivagaminathan et al., 2023). This section reviews the current developments and state-of-the-art methods of IDSs along with the recent advancements of deep learning-based approaches.

2.1 TRADITIONAL INTRUSION DETECTION APPROACHES

The traditional IDSs for WSNs were primarily signature-based or anomaly-based (Khraisat & Alazab, 2021).Signature-based IDSs (SIDS) recognise attacks by applying pre-designed signatures or rules. SIDSs have high accuracy in recognising known attacks; however they do not discover unknown attacks (Asad et al., 2024).Anomaly-based intrusion detection systems (AIDS) use statistical models or machine learning models to identify differences from typical network behaviour (Kushal et al., 2024). and efficient in discovering zero-day attacks. Since WSN traffic is changing on a regular basis, the basic problem with AIDS is the high false-positive rate. (Bhavsar et al., 2023).

Many machine learning techniques have been developed for anomaly detection, such as Support Vector Machines (SVM) (Akpinar et al., 2021), Random Forest (RF) (Pekar & Jozsa, 2024), and K-Nearest Neighbors (KNN) (Halder et al., 2024). The basic problem with these models discussed is

that they were primarily designed and developed using hand-crafted features, limiting their designed adaptability to address advancements.

2.2 DEEP LEARNING-BASED IDS

Implementation of DL techniques in network security threat detection systems has been an area of recent research. It uses neural networks to automatically extract important features from network traffic data. The CNN, RNN, and their hybrids are some examples of deep learning models that have demonstrated positive effects in identifying sophisticated temporal and spatial patterns in network behaviour that conventional approaches might miss. The ability of these models is that it rapidly acquire complex representations from unprocessed or minimally processed input, improving their accuracy in detecting both known and unknown attack types. They are ideal for dynamic and evolving cybersecurity environments because of their versatility and capacity to make predictions from vast amounts of data.

2.2.1 CONVOLUTIONAL NEURAL NETWORKS (CNNS):

The CNN models have shown strong feature extraction capabilities in deployment as an IDS (Alars & Kurnaz, 2024). It identifies complex spatial dependencies and subtle feature interactions within data representations such as traffic matrices, payload patterns, or network flow visualizations as their primary focus is on spatial patterns, this makes them less effective for sequential data.

2.2.2 LONG SHORT-TERM MEMORY (LSTM):

LSTM models can be used widely for identifying evolving attack patterns because they are primarily focused to identify temporal relationships in network traffic data. These models are especially good at identifying hidden or slowly evolving threats that change over time because of their memory-based architecture, that enables them to remember and leverage long-term dependencies in sequential data (Al-Selwi et al., 2024). Also, they can adapt with evolving traffic conditions. The capacity of the LSTM models to learn from the timing and sequence of events provides a security against increasingly complex cyberattacks.

2.2.3 HYBRID DEEP LEARNING MODELS:

In recent times, many researchers are considering CNN-LSTM hybrid systems that integrate the spatial feature extraction skills of CNN with the progressive learning abilities of LSTM. On the otherhand, CNN layers are typically used to extract data patterns and hierarchical features from raw network traffic data. The LSTM layers in this hybrid method finds the temporal relationships across the extracted features. By understanding both the existing features and their evolution over time, thus the integration of these benefits, it allows the model to gain a more comprehensive understanding of the behavior of the network (Nazir et al., 2024). So the deployment of these models have shown significant improvements in IDS accuracy compared to standalone CNN or LSTM models (Bamber et al., 2025).

2.3 IDS IN WIRELESS SENSOR NETWORKS

Though there are different techniques used to protect the network system, the security challenges in WSNs differ from those in traditional networks due to the resource-constrained nature of sensor nodes and dynamic network topologies (Ahmed et al., 2021). Existing network threat detection system in WSNs can be categorized into the following:

2.3.1 HOST-BASED IDS (HIDS): In this type of system, it monitors activities on individual sensor nodes but incurs high energy consumption (Ghosal & Halder, 2017).

2.3.2 NETWORK-BASED IDS (NIDS): Here the system analyses network traffic flow across the entire network, offering broader attack coverage but facing scalability challenges (Holdbrook et al., 2024).

2.3.3 HYBRID IDS: It is a combination of both the HIDS and NIDS that can enhance detection efficiency using the benefits of both models (Wang & Zhang, 2012).

Several rechearchers have proposed deep learning-based IDS solutions for WSNs in receent times. For instance, (Salmi & Oughdir, 2022) introduced an LSTM-based anomaly detection system for WSN security, and achieved a good result. However, these models often lack of robust feature extraction, which makes them susceptible to high false positives. But, the combination of CNN and LSTM has shown potential in addressing these limitations by improving both spatial and temporal feature learning.

Existing hybrid models do not explicitly model feature importance across temporal sequences, which motivates the integration of an attention mechanism.

3. PROPOSED METHODOLOGY

This section presents the proposed Hybrid CNN–LSTM model with an Attention Mechanism for intrusion detection in Wireless Sensor Networks (WSNs). The model is designed to effectively capture both spatial correlations among features and temporal dependencies in network traffic data.

The Convolutional Neural Network (CNN) component is employed for extracting high-level spatial features, while the Long Short-Term Memory (LSTM) network models sequential behavior in the data. To further enhance performance, an attention mechanism is integrated over the LSTM outputs, enabling the model to focus on the most relevant time steps contributing to intrusion detection.

The proposed model is trained and evaluated on the WSN-DS dataset, which contains labeled instances of normal and attack traffic.

3.1 DATASET AND PREPROCESSING

The WSN-DS dataset is used for training and evaluation in this study. It contains multiple categories of network behavior, including normal and anomalous instances, characterized by features such as communication patterns, routing information, and energy consumption metrics.

The following are the preprocessing steps included in this research:

3.1.1 Data Cleaning:

In this research, we used WSN-DS dataset which is publicly accessible. The information is kept in a CSV file format. In this stage, the details of each dataset were read using the Pandas package, and in order to prepare for the following steps, the datasets were cleaned by filtering out of redundant and null values.

3.1.2 Data Encoding:

Data encoding is an essential preprocessing step in preparing the WSN-DS dataset for input into the model. The dataset consists of both numerical and categorical features, and to ensure compatibility with the neural network, all categorical variables were converted into numerical representations. This was achieved using label encoding, where each unique category was assigned an integer value. Label encoding was applied particularly to categorical columns such as protocol types or attack labels, ensuring that the encoded values retained the necessary distinctions for classification without introducing artificial ordinal relationships.

3.1.3 Data Normalization:

Data Normalization of the WSN-DS dataset is performed to scale all numerical feature values to a fixed range, so that no single feature dominates the learning process due to its high value. For this purpose, the MinMaxScaler technique is used from the sklearn.preprocessing library. This technique transforms each feature individually by scaling it to a specified range, that is, between 0 and 1.

3.1.4 Sequential Feature Structuring

To capture temporal characteristics of WSN traffic, the dataset is organized into fixed-length sequences. Each sequence represents a time-ordered set of observations, preserving the chronological relationships among features such as packet rate, delay, and energy consumption. This representation enables effective temporal learning using LSTM.

3.1.5 Train-Test Split:

The original WSN-DS dataset used in our model is divided into training, validation, and testing sets to ensure proper model training and performance testing. Firstly, the dataset is split into 80-20 ratio as the training and testing dataset using the train_test_split() technique from the sklearn.model_selection library. For further fine-tuning the model and preventing overfitting the model, the training set was subdivided into training and validation subsets. This process allowed us to monitor and adjust the model's hyperparameters during training. For robust performance evaluation, we further employed the Stratified K-Fold Cross-Validation technique.

4. HYBRID CNN–LSTM WITH ATTENTION ARCHITECTURE

The proposed architecture integrates CNN, LSTM, and Attention layers to effectively learn spatial and temporal patterns in WSN data.

4.1 CONVOLUTIONAL NEURAL NETWORK (CNN)

The CNN component performs spatial feature extraction through convolution and pooling operations.

4.1.1 Convolution Layer

In the convolution operation, a set of filters slides over the input matrix and computes feature maps

as:

$$Z = h(\sum w \cdot v + b)$$

where w represents filter weights, v denotes input values, b is the bias, and h is the activation function. The Rectified Linear Unit (ReLU) activation is used:

$\text{ReLU}(z) = \max(0, z)$

4.1.2 Pooling Layer

Max pooling is applied to reduce the dimensionality of feature maps while retaining the most significant features, thereby improving computational efficiency and reducing overfitting.

4.1.3 Batch Normalization

Batch normalization is used to stabilize training by normalizing the output of each layer:

$$X - \mu B$$

$$Xo =$$

$$\sqrt{\sigma 2 + \epsilon}$$

$$Yo = \gamma Xo + \beta$$

Where:

- X: Input to the batch normalization layer
- μB: Mean of the mini-batch
- $\sigma 2$: Variance of the mini-batch
- ϵ: Small constant for numerical stability
- Xo: Normalized output
- γ: Learnable scaling parameter
- β: Learnable shifting parameter
- Yo: Final output after scaling and shifting

Batch normalization first standardizes the input using the batch mean and variance. Then, it applies learnable parameters, γ and β to allow the network to restore representational power if needed. This process reduces internal covariate shift, improves gradient flow, and accelerates convergence during training.

4.2 LONG SHORT-TERM MEMORY (LSTM)

The LSTM network captures temporal dependencies in sequential data. At each time step t, the LSTM processes the input $X(t)$ and the previous hidden state $h(t - 1)$, updating its internal memory using gating mechanisms.

$ht = \text{LSTM}(Xt, ht-1)$

The LSTM consists of forget, input, and output gates, which regulate the flow of information and enable the model to retain long-term dependencies in WSN traffic patterns.

4.3 ATTENTION MECHANISM

To improve the model's ability to focus on critical temporal features, an attention mechanism is applied over the LSTM outputs. Instead of relying solely on the final hidden state, attention assigns importance weights to all time steps.

The attention mechanism is defined as follows:

$et = vT\tanh(Waht + ba)$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{t=1}^{T} \exp(e_t)}$$

$$c = \sum_{t=1}^{T} \alpha_t h_t$$

where etdenotes the attention score, αtrepresents the attention weight, and $c$is the context vector obtained as a weighted sum of LSTM outputs. The parameters Wa, v, and $ba$are learnable.

This mechanism enables the model to emphasize important time steps, improving detection accuracy and interpretability.

4.4 DROPOUT LAYER

To mitigate overfitting, dropout is applied after LSTM layers. During training, a fraction of neurons (e.g., 20%) is randomly deactivated, forcing the network to learn more generalized representations.

4.5 FULLY CONNECTED LAYER AND CLASSIFICATION

The context vector obtained from the attention mechanism is passed to a fully connected layer:

$zi = Wfc + bf$

The Softmax function is then applied to obtain class probabilities:

$\hat{y}i$ = Softmax(zi)

The class with the highest probability is selected as the final prediction.

5. RESULTS AND DISCUSSION

The hybrid CNN-LSTM model's performance evaluation on the WSN-DS dataset shows excellent performance with the detection of different network attacks in Wireless Sensor Network (WSN). The detailed results of the performance metrics were carefully considered using different classification metrics, including accuracy, precision, recall, and F1-score.

The classification performance metrics of the hybrid CNN-LSTM model in the classification of different attacks and normal traffic is shown in Table 1. The overall performance of the hybrid CNN-LSTM model is summarized in Table 2 to highlight its good classification capability across all attack types. The overall accuracy of the hybrid CNN-LSTM model was 98.75%, indicating the thye model could reliably perform intrusion detection. The weighted average precision, recall, and F1-score outputs came out to be approximately 0.99, indicating the robustness of the model in handling various attack types. Fig. 3, shows the variations in classification success across the classes.

Table 1: Class-wise performance of the hybrid CNN-LSTM Model on the WSN-DS Dataset

Class	Accuracy	Precision	Recall	F1-score
Blackhole	99.73	77.93	99.75	87.75
Flooding	99.75	90.93	99.93	94.75
Grayhole	97.75	99.95	75.93	86.95
Normal	99.95	99.95	99.95	99.95
TDMA	96.63	99.95	93.73	96.75

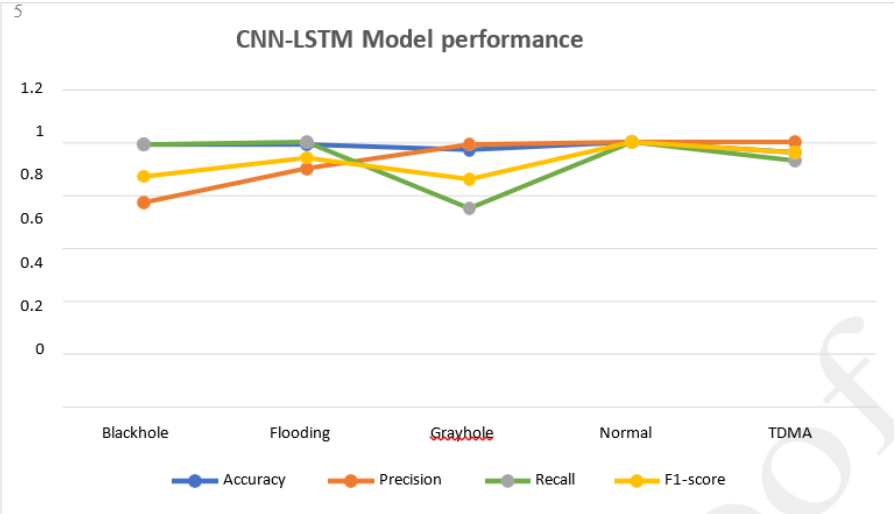

Fig. 3. Graphical representation of the Class-wise performance of the hybrid CNN-LSTM Model.

Table 2: Overall Performance of the hybrid CNN-LSTM Model on the WSN-DS Dataset.

Metric	Value (%)
Accuracy	98.75
Precision	98.93
Recall	98.75
F1-score	98.73

The high precision and recall scores indicate that there are few false positives and false negatives, allowing the model to be used reliably for intrusion detection. The overall classification level of performance is shown in its entirety in Fig. 4.

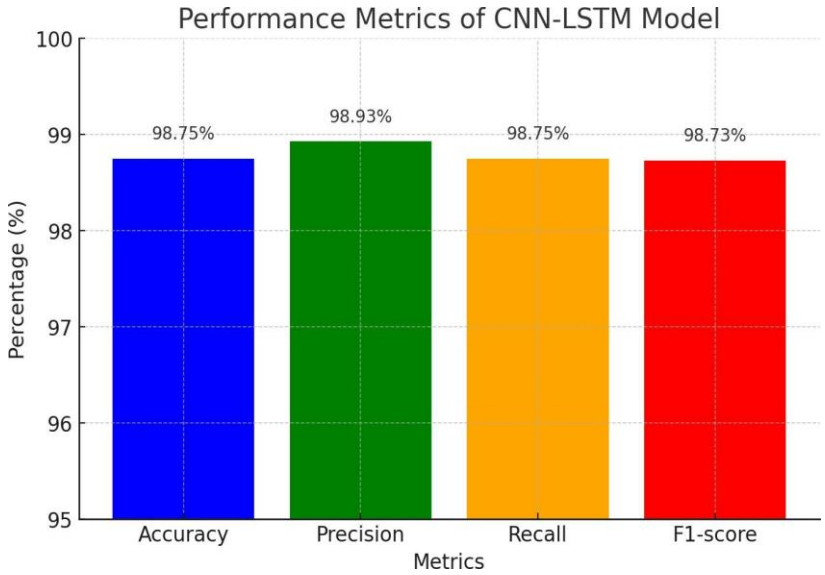

Fig. 4. Overall Performance Evaluation Metrics of the hybrid CNN-LSTM Model.

The confusion matrix (fig. 5) was then examined to further assess the model's detection capability. The confusion matrix provides details of the model's strong classification ability in detecting the various attack types with the model performing best in distinguishing between Flooding and TDMA attacks.

Fig. 5. Confusion Matrix for the hybrid CNN-LSTM Model

The hybrid CNN-LSTM model achieved good detection abilities to different attack types, including recall of 0.99 for Blackhole attacks, a perfect recall detection of 1.00 with 0.90 precision for Flooding attacks (with no false negatives); Grayhole attacks were accurately detected with a precision of 0.99. The TDMA attack was also accurately detected with an F1-score of 0.96 support to recognize the model's reliable intrusion detection capabilities.

To better understand the model's ability to differentiate between attack and normal traffic, the ROC-AUC curve in Fig. 6 was evaluated as AUC scores close to 1 represent good results. Further analysis of the Area Under the Curve (AUC) values demonstrated excellent model performance in detection with an AUC score was close to 1

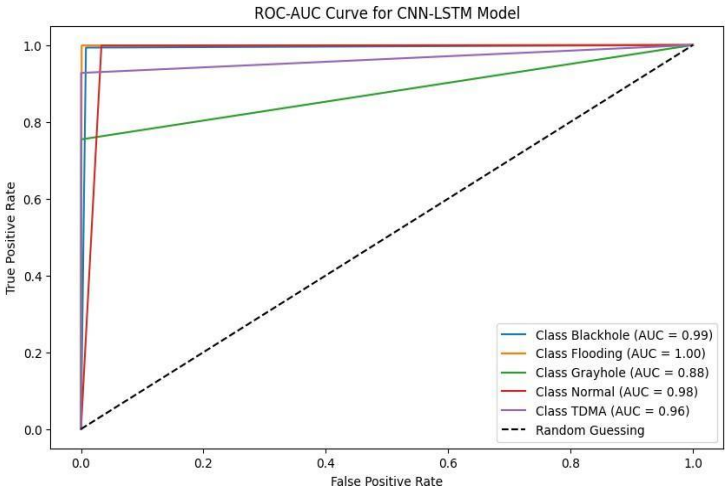

Fig. 6. ROC-AUC Curve for the hybrid CNN-LSTM Model

The ROC curve illustrates the model's ability to maintain a high detection rate across different attack types while minimizing false positives and false negatives. This further supports model's robustness and effectiveness in intrusion detection for WSNs.

Table 3: Comparative Performance Analysis of Existing and

Model	**Accuracy**	**Precision**	**Recall**	**F-Score**
SVM (Gowdhaman & Dhanapal, 2022)	67.91	69.60	68.75	58.39
DT (Gowdhaman & Dhanapal, 2022)	79.98	77.39	74.59	72.00
RF (Gowdhaman & Dhanapal, 2022)	83.73	82.25	76.66	69.60
Autoencoder (Zhang Chongzhen et al., 2019)	79.74	82.22	79.74	76.47
DNN (Shakya et al., 2025)	96.23	95.75	92.82	94.53
DNN (Gowdhaman & Dhanapal, 2022)	95.53	94.65	91.92	92.43
Proposed CNN-LSTM Methods	98.75	98.93	98.75	98.73

As reflected in Table 1, the suggested hybrid CNN-LSTM method shows a significant improvement over all baseline models on most of the important performance measures, with a highest accuracy (98.75%), precision (98.93%), recall (98.75%), and F-score (98.73%). Traditional ML models like SVM, Decision Tree, and Random Forest achieved relatively lower accuracies between 67.91% and 83.73%, which indicates their less capability to comprehend the intricate, nonlinear relationships involved in large-scale network data. Even though deep learning frameworks like the DNN models of Shakya et al. (2025) and Gowdhaman & Dhanapal (2022) posted high scores (95–96% accuracy), their performance stagnated owing to the lack of mechanisms that could capture sequential temporal relationships in dynamic network traffic patterns.

From the presented analysis, it is evident that the CNN-LSTM based hybrid model performs exceptionally well in classifying both normal and attack traffic in WSNs. The classification metrics, confusion matrix, and ROC curve collectively establish its high accuracy, precision, recall, and robustness in detecting various types of network intrusions. These results confirm the model's effectiveness as a promising approach for securing WSNs against malicious attacks.

6. CONCLUSION

This research study demonstrates that the hybrid CNN-LSTM is a viable and effective model to use for detecting network security threats in WSN. This model effectively brought together the strengths of the CNN, LSTM, and the attention mechanism. It achieved an overall accuracy of 98.76%. This shows the overall effectiveness and efficiency of the hybrid CNN-LSTM model in detecting the presence of specific network attacks in WSN. The model also has strong performance on other metrics.Comparative analysis with some existing research works confirmed that the proposed model outperforms conventional machine learning and standalone deep learning approaches for WSN intrusion detection.The results indicate that attention-assisted hybrid deep learning architectures can provide reliable and efficient intrusion detection for resource-constrained WSN environments

References:

1. Akpinar, M., Adak, M. F., & Guvenc, G. (2021). SVM-based anomaly detection in remote working: Intelligent software SmartRadar. Applied Soft Computing, 109, 107457. https://doi.org/10.1016/J.ASOC.2021.107457
2. Alars, E. S. A., & Kurnaz, S. (2024). Enhancing network intrusion detection systems with combined network and host traffic features using deep learning: deep learning and IoT perspective. Discover Computing, 27(1), 1–19.
3. https://doi.org/10.1007/S10791-024-09480-3/FIGURES/10
4. Aldhaheri, A., Alwahedi, F., Ferrag, M. A., & Battah, A. (2024). Deep learning for cyber threat detection in IoT networks: A review. Internet of Things and Cyber-Physical Systems, 4, 110–128. https://doi.org/10.1016/J.IOTCPS.2023.09.003
5. Al-Selwi, S. M., Hassan, M. F., Abdulkadir, S. J., Muneer, A., Sumiea, E. H., Alqushaibi, A., & Ragab, M. G. (2024). RNN-LSTM: From applications to modeling techniques and beyond—Systematic review. Journal of King Saud University - Computer and Information Sciences, 36(5), 102068. https://doi.org/10.1016/J.JKSUCI.2024.102068
6. Asad, H., Adhikari, S., & Gashi, I. (2024). A perspective–retrospective analysis of diversity in signature-based open-source network intrusion detection systems. International Journal of Information Security, 23(2), 1331–1346. https://doi.org/10.1007/S10207-023-00794-9/FIGURES/22
7. Bamber, S. S., Katkuri, A. V. R., Sharma, S., & Angurala, M. (2025). A hybrid CNN-LSTM approach for intelligent cyber intrusion detection system. Computers & Security, 148, 104146. https://doi.org/10.1016/J.COSE.2024.104146
8. Bhavsar, M., Roy, K., Kelly, J., & Olusola, O. (2023). Anomaly-based intrusion detection system for IoT application. Discover Internet of Things, 3(1), 1–23. https://doi.org/10.1007/S43926-023-00034-5/FIGURES/7
9. Ghosal, A., & Halder, S. (2017). A survey on energy efficient intrusion detection in wireless sensor networks. Journal of Ambient Intelligence and Smart Environments, 9(2), 239–261. https://doi.org/10.3233/AIS-170426
10. Gowdhaman, V., & Dhanapal, R. (2022). An intrusion detection system for wireless sensor networks using deep neural network. Soft Computing, 26(23), 13059–13067. https://doi.org/10.1007/S00500-021-06473-Y
11. Halder, R. K., Uddin, M. N., Uddin, M. A., Aryal, S., & Khraisat, A. (2024). Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications. Journal of Big Data 2024 11:1, 11(1), 1–55. https://doi.org/10.1186/S40537-024-00973-Y
12. Holdbrook, R., Odeyomi, O., Yi, S., & Roy, K. (2024). Network-Based Intrusion Detection for Industrial and Robotics Systems: A Comprehensive Survey. Electronics 2024, Vol. 13, Page 4440, 13(22), 4440. https://doi.org/10.3390/ELECTRONICS13224440
13. Khraisat, A., & Alazab, A. (2021). A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. Cybersecurity, 4(1), 1–27. https://doi.org/10.1186/S42400-021-00077-7/TABLES/10
14. Kushal, S., Shanmugam, B., Sundaram, J., & Thennadil, S. (2024). Self-healing hybrid intrusion detection system: an ensemble machine learning approach. Discover Artificial Intelligence, 4(1), 1–20. https://doi.org/10.1007/S44163-024-00120-9/TABLES/10
15. Nazir, A., He, J., Zhu, N., Qureshi, S. S., Qureshi, S. U., Ullah, F., Wajahat, A., & Pathan, M. S. (2024). A deep learning-based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem. Ain
16. Shams Engineering Journal, 15(7), 102777. https://doi.org/10.1016/J.ASEJ.2024.102777
17. Pekar, A., & Jozsa, R. (2024). Evaluating ML-based anomaly detection across datasets of varied integrity: A case study. Computer Networks, 251, 110617.
18. https://doi.org/10.1016/J.COMNET.2024.110617
19. Pourrahmani, H., Yavarinasab, A., Monazzah, A. M. H., & Van herle, J. (2023). A review of the security vulnerabilities and countermeasures in the Internet of Things solutions: A bright future for the Blockchain. Internet of Things, 23, 100888. https://doi.org/10.1016/J.IOT.2023.100888
20. Pundir, M., & Sandhu, J. K. (2021). A Systematic Review of Quality of Service in Wireless Sensor Networks using Machine Learning: Recent Trend and Future Vision. Journal of Network and Computer Applications, 188. https://doi.org/10.1016/j.jnca.2021.103084
21. Salmi, S., & Oughdir, L. (2022). CNN-LSTM Based Approach for Dos Attacks Detection in Wireless Sensor Networks. IJACSA) International Journal of Advanced Computer Science and Applications, 13(4), 835–842. www.ijacsa.thesai.org

22. Sekhar, V. C., & Sarvabhatla, M. (2012). Security in wireless sensor networks with public key techniques. 2012 International Conference on Computer Communication and Informatics, ICCCI 2012. https://doi.org/10.1109/ICCCI.2012.6158861
23. Shakya, V., Choudhary, J., & Singh, D. P. (2025). Deep Learning based Intrusion Detection System for WSN. Procedia ComputerScience, 258, 2101–2106. https://doi.org/10.1016/J.PROCS.2025.04.460
24. Sivagaminathan, V., Sharma, M., & Henge, S. K. (2023). Intrusion detection systems for wireless sensor networks using computational intelligence techniques. Cybersecurity, 6(1), 1–15. https://doi.org/10.1186/S42400-023-00161-0/TABLES/6
25. Wang, Z. Q., & Zhang, D. K. (2012). HIDS and NIDS Hybrid Intrusion Detection System Model Design. Advanced Engineering Forum, 6–7, 991–994. https://doi.org/10.4028/WWW.SCIENTIFIC.NET/AEF.6-7.991
26. Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. Insights into Imaging, 9(4), 611–629. https://doi.org/10.1007/S13244-018-0639-9/FIGURES/15
27. Zhang Chongzhen, Ruan Fangming, Yin Lan, Chen Xi, Zhai Lidong, & Liu Feng. (2019). A deep learning approach for network intrusion detection based on NSL-KDD dataset. 2019 IEEE 13th International Conference on Anti-Counterfeiting, Security, and Identification (ASID), 41–45.