10.48047/jocaaa.2024.33.06.99

Enhanced Classification For Intrusion Detection In WSNS Using Bi-Directional RNNS

D.Priyadarshini, Ph.D. Research scholar, PG and Research Department of Computer Science, Chikkanna Government Arts College, Tirupur.
Dr.K.Sarojini*, MCA, M.Phil., Ph.D., Assistant Professor, PG and Research, Department of Computer Science, Chikkanna, Government Arts College, Tirupur.
Gmail.id: saromaran@gmail.com

Received: 11.06.2024 Revised : 16.07.2024 Accepted: 22.08.2024

ABSTRACT
This paper presents a robust Intrusion Detection System (IDS) for Wireless Sensor Networks (WSNs) based on a Bi-directional Recurrent Neural Network (Bi-RNN) model that takes advantage of the temporal dependencies inherent to network traffic to enhance anomaly detection. Although other models can be used to examine time series data of traffic, The Bi-RNN architecture examines input sequences in both forward and backward fashions. This negates potential subsumption and allows the system to recognize more unique patterns and contextual backgrounds that may indicate a malicious event. This can be especially useful in WSNs where an attack is coordinated or very gradual (e.g sinkhole or selective forwarding), and typically goes unnoticed. Therefore the Bi-RNN system and finally the IDS and was trained on time-series traffic data that was minimally and logically pre-processed, meaning that potential feature engineering was extremely limited, in addition to being more contextually adaptable in bases of noisy and imbalanced datasets. Using recurrent neural structure and in tandem based on mathematical fundamentals or formulations, our Bi-RNN model updates its hidden states on both knowledge and look ahead of past historical input vectors and utilize stochastic and/or mini-batch cross-entropy loss for classification. Because the Bi-RNN IDS is implemented at the edge with limited resources at the sensor nodes, it can operate, real time, in the live network environment. Through the usability of empirical study using benchmark metrics, despite the slight better performance in raw accuracy for MLSTL-WSN and SVM + BGWO, the Bi-RNN model outperformed both of them overall in better precision (98.80%), recall (98.90%), and F-measure (98.85%), and with the lowest false positive rate of (3.9%) of all models tested, with a specificity of 99.04%, demonstrating high reliability to separate benign from malicious traffic. These results corroborate the Bi-RNN can work as a scalable and intelligent IDS structure even with these dynamically changing conditions as an outcome of the resource and energy limitations of next generation IoT-WSNs.

Keywords: Wireless Sensor Networks, Intrusion Detection System, Bi-directional Recurrent Neural Network, Anomaly Detection, Time-Series Traffic Analysis, Edge Computing, Deep Learning, Cyber security, Real-Time Detection.

1. INTRODUCTION

The high velocity of Wireless Sensor Networks (WSNs) coupled with the Internet of Things (IoT) transformed fields like environmental monitoring, health, and industrial automation. Nevertheless, since there is increased connectivity, WSNs are vulnerable to a vast array of attacks, and thus, there is a need for strong adaptive intrusion detection. As [1] have already noted, existing security frameworks tend to fail when applied to addressing the WSNs' unique requirements like limited processing power, energy, and decentralized structures.

Developments of recent times have increasingly centered on Machine Learning (ML) and deep learning methods for improving Intrusion Detection Systems (IDSs). [2] utilized an advanced IDS built on an optimized k-Nearest Neighbors (kNN) algorithm that is more effective in terms of resource constraint handling. [3] Also proposed the MLSTL-WSN framework that utilizes the SMOTETomek resampling

10.48047/jocaaa.2024.33.06.99

technique to handle class imbalance in training sets. Context-aware mechanisms, as envisioned by [4], also facilitate greater IDS responsiveness against adaptive threat levels.

With a quest for scalable and smart solutions, hybrid approaches involving the union of optimization techniques and ensemble learning are on the rise. [5] Proposed a GSWO-CatBoost strategy that involves applying global search strategies to optimize features, and it significantly enhanced the detection capability. [6] Handled the issue of data privacy by using federated learning with SCNN-Bi-LSTM models to support collaborative intrusion detection without central data sharing. [7] Used an evolutionary computation-inspired model that learns low-complexity detection for constrained WSN nodes.

To solve detection accuracy and generalizability, [8] proposed a generalized intrusion detection system for powerful intruders with protocol manipulation ability. [9] Used Gaussian process regression to forecast k-barrier coverage probabilities, which is crucial in deciding the physical coverage-based defense strategies in WSNs. [10] proposed a multi-level intrusion detection method for node-level drop attacks, providing more granularity in attack detection.

Other strategies have been suggested for managing algorithmic efficiency and spatial modeling. [11] Described a more sophisticated negative selection algorithm with spatial partitioning to enhance rates of anomaly detection. [12] Utilized a machine learning-based Firefly algorithm for enhancing IDS performance and reducing energy consumption in WSN-IoT networks. In parallel, [13] showed experimentally the performance of deep learning models in intrusion detection using extensive real-world datasets.

The comprehensive analysis discussed in these surveys, coupled with recent research conducted by researchers such as [14], who used a wide range of ML classifiers to intrusion detection in WSNs, highlight the multi-dimensional challenges still left in this field. [15] Investigated the practicality of deploying deep learning models in sensor networks, noting both the benefits of increased detection accuracy and the potential issues involved with computational overhead.

Despite all these advancements, considerable gaps are still present most notably on model interpretability, real-time detection, robustness against adversarial attacks, and efficiency on constrained computational resources. As responses to these unresolved issues, this work introduces a new hybrid intrusion detection system suitable for WSN scenarios. The new model incorporates efficient feature selection, balance techniques, and simple classifiers with the purpose of achieving high detection accuracy at the cost of saving energy and computing resources.

Contribution: The main contribution of this paper is the development of a high performance IDS designed for WSN using a Bi-RNN. The model takes advantage of both forward and backward temporal dependencies in network traffic to allow for the detection of sophisticated intrusions before they can be hidden by unidirectional models. Ultimately, the system accuracy is high, with low false positives, is adaptable to new and evolving threats and is implementable in resource-limited edge environments.

Organization: This paper is organized as follows. Section 1 describes the motivation and challenges of intrusion detection in WSNs. Section 2 describe related work and current IDS methods. Section 3 introduces the materials and methods, including Bi-RNN architecture and the algorithmic framework. In Section 4, experimental design, evaluation metrics, and experiments results are discussed, followed by conclusions and future works in Section 5.

2. Background Study

The Adaptive Ensemble Learning Framework proposed by Veeramachaneni (2025) [16] aims to improve intrusion detection system robustness in WSNs. The study uses an adaptive ensemble learning framework that enables the combination of multiple classifiers that were dynamically selected based on environmental and network conditions to achieve improved adaptability and detection capability. The approach to intrusion detection and prediction can comprehensively address concept drift and variability in attack patterns, showing a specific need for derived intrusion detection approaches from real-world deployments. The ensemble approach also will offer resilience from attacks across both known and unknown threats.

Han et al. (2019) [17] introduces a new intrusion detection framework that implements game theory and autoregressive models to predict and detect intrusion attempts in WSNs. The game-theoretic model aims to accurately model player interactions (the attackers and defenders) while the autoregressive model captures temporal dependencies in the network traffic. The dual contributions integrate predictive analytics with strategic defense mechanisms to enhance detection performance in adaptive and intelligent threat environments.

10.48047/jocaaa.2024.33.06.99

The protocol for detecting and preventing intrusions in IoT-enabled WSNs developed by Krishnan et al. (2022) [18] describes the development of a single protocol composed of two ends of the intrusion model spectrum (proactive prevention, and reactive detection) that supports multi-layered defence mechanisms. The model is developed with multi-layer context in mind and provides nuanced considerations for interoperability, energy consumption and resilience - all heavily weighted design considerations for resource-constrained IoT devices. An important consideration of the model is developing an end-to-end intrusion detection and prevention protocol that extends from the sensor nodes all the way to functionalities at the cloud interface.

Reddy et al. (2024) [19] optimized the placement of barriers in WSNs for intrusion detection and prevention. Their work is presented as a solution to physical-layer security because it provides an efficient way of placing sensor nodes. This maximizes coverage for the detection of an intrusion. Reddy et al. utilize computational geometry with heuristic algorithms, depending on the application's needs, to maximize coverage reliability and minimize gaps in security. This is a foundational study when developing intrusion resistant WSN topologies especially for critical infrastructure.

Safaldin et al. (2021) [20] introduced an improved Binary Gray Wolf Optimizer (BGWO) in conjunction with Support Vector Machines (SVM) for intrusion detection in WSNs. They combine these two techniques to improve feature selection and classification performance, which increases accuracy of intrusion detection and minimization of false positives. The BGWO algorithm explores the feature space, while the SVM provides reliable decisions. This hybrid process is a superior option for lightweight real-time intrusion detection in resource constrained sensor systems.

Table 1: Comparison table on Intrusion Detection in WSN

Authors & Year	Approach	Techniques	Strengths	Limitations
Gowdhaman & Dhanapal (2022) [21]	Deep Learning-based IDS	Deep Neural Network (DNN)	High detection accuracy and ability to learn complex patterns	Computationally intensive; may not suit highly constrained nodes

10.48047/jocaaa.2024.33.06.99

Zhang et al. (2020) [22]	Lightweight IDS	MK-ELM (Modified Kernel Extreme Learning Machine)	Fast training and classification; suitable for real-time detection	May face limitations in handling diverse attacks
Alruhaily & Ibrahim (2021) [23]	Multi-layer ML IDS	Layered architecture with ML classifiers	Improves detection granularity and flexibility	Increased design complexity; tuning required for each layer
Ghazal (2022) [24]	Data Fusion ML Architecture	Sensor data fusion + ML	Enhances context-awareness and detection accuracy	Data fusion can introduce latency or synchronization issues
Kaur & Rattan (2021) [25]	Review & Analysis	Critical literature review	Summarizes current challenges and outlines future research paths	Does not propose a novel model or conduct experiments

Pan et al. (2021) [26] developed a lightweight intelligent intrusion detection model, considering the constraints on energy and processing capabilities of WSNs. They utilized reduced machine learning algorithms that are able to detect malicious activity without compromising the node resources. Their approach was very efficient computationally and still allowed for high accuracy of detection in real time and living WSNs that are remote or critical applications.

Zhang et al. (2025) [27] presented SC-MLIDS, a machine learning based framework for intrusion detection in WSN that uses a fusion-based approach. They situated their model so that multiple classifiers through ensemble learning and decision-level fusion could account for generalization and robustness. They found that their model could reduce false positives and improve detection rates over heterogeneous attack types. This was especially useful in varied network environments with continuous data variability.

Talukder et al. (2025) [28] also presented a hybrid intrusion detection model that combined data balancing methods and dimensionality reduction to improve ML performance in WSNs. Their method addressed other including challenges the limitations of class imbalance and the dimensionality of features, both of which often resulted in reduced accuracy and/or overfitting. The authors demonstrated that their model was both highly effective and precise leading them to recommend this model for real-time and energy-constrained intrusion detection systems.

Ramadan (2020) [29] researched optimized intrusion detection algorithms for smart cities that utilize wireless sensing technology. This research develops modify detection models that deal with the multifaceted security threats of urban sensor networks with a particular focus on scalability, speed, and

10.48047/jocaaa.2024.33.06.99

low energy consumption. The research successfully reconciles theoretical IDS models with real-world applications in smart city infrastructures such as transportation, utility, and surveillance infrastructures.
Zhukabayeva et al. (2024) [30] introduced a framework for intrusion detection in WSNs with machine learning components to fulfill the requirements for smart grid systems. The authors used analysis of traffic (and awareness of the type of node) in their model to help to optimize feature selection and improve detection accuracy. By using context to identify normal behavior and differentiate the roles of nodes, their scheme improved cybersecurity for the heterogeneous model of smart grids. Notably, the authors highlighted the importance of utilizing real-time traffic characteristics to improve the intrusion detection and prevention in such an environment that inherently has an incredibly high level of dynamics.

2.1 Problem Identification

While there has been substantial progress on IDS in WSN, there are still several barriers to enabling effective deployment of practical systems. Many of the proposed models offer good performance in controlled scenarios, but their practicality in real-time situations involving powerful side-channel attacks is constrained by their high complexity, high energy consumption, and sensitivity to both data imbalance and dimensionality. Furthermore, real-time adaptability in a WSN intrusions detection system is required as they can represent a blend of ever-changing attack avenues, node resource constraints, and changeable topologies. Current solutions in WSN intrusions detection represent efficiency and accuracy tradeoffs, but they can also struggle to generalize in heterogeneous, large-scale deployments. Thus, there is still a need for robust and scalable IDS architecture that is energy aware, and that can effectively balance detection performance with operational feasibility within these fluid and resource-constricted environments.

3. Materials and Methods

This section explains how to design and implement an Intrusion Detection System (IDS) for Wireless Sensor Networks (WSNs) using a Bi-directional Recurrent Neural Network (Bi-RNN). In this section, we introduce the architectural principles, mathematical descriptions, and stepwise training process procedure with time-series network traffic data. The approach focuses on the overall forward and backward capturing of temporal interdependencies to improve anomaly detection. Algorithm 1 outlines a structured process of data preprocessing, training and evaluation, and ultimately deployment on edge-based environments.

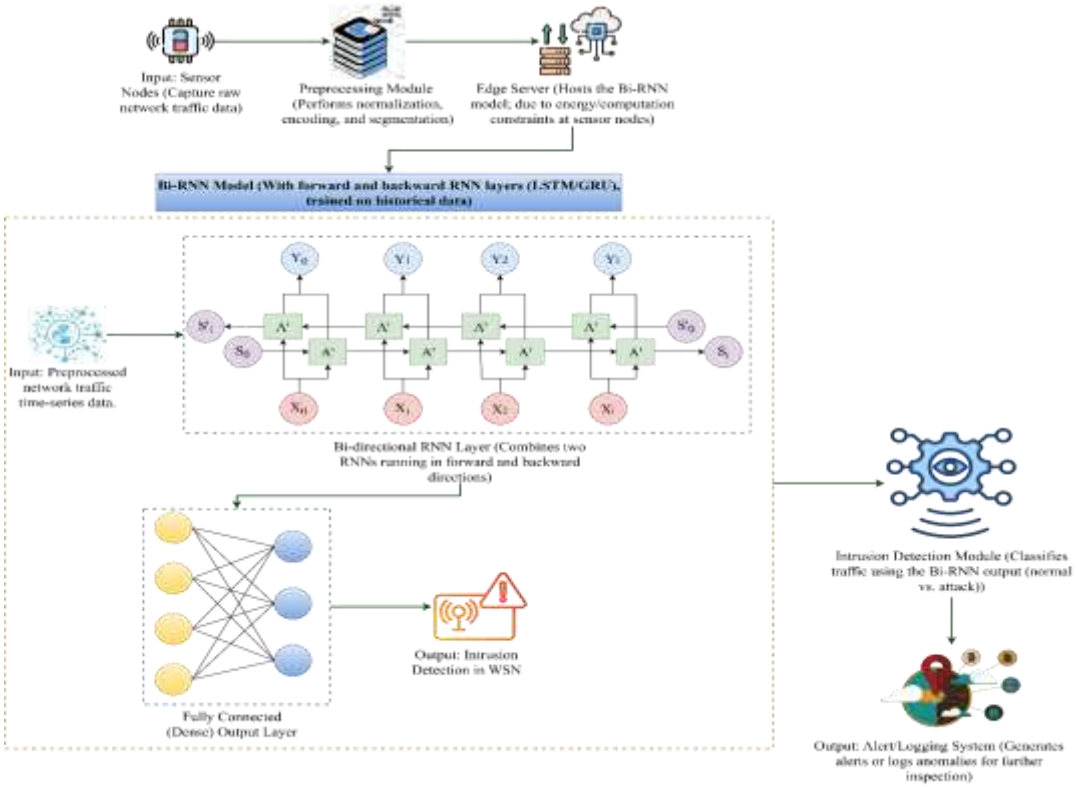


Figure 1: Overall Work Flow Architecture

10.48047/jocaaa.2024.33.06.99

The figure 1 depicts the framework of Bi-RNN based IDS for WSN. The sensor nodes track network traffic and send it to an edge server after the data is preprocessed (e.g., normalized, segmented, etc.). The Bi-RNN model, made up of forward and backward RNN layers (e.g., LSTM/GRU), is trained in a supervised manner over historical data. When the Bi-RNN is trained it can identify patterns that account for temporal dependencies in both time steps (forward and backward). The processed input stream is sent through the forward and backward RNN layers, passed into a fully connected output layer for classification of the input sequence as either normal operation or intrusion. The classification results are passed into the IDS decision module which provides alerts or other actions as well as logs the suspicious activity for forensic analysis. This model allows for real-time threat detection in WSNs as well as rapid responsiveness of the entire system.

3.1 Bi-directional Recurrent Neural Networks (Bi-RNN)

Detecting Intrusions in WSN with Bi-RNN offers a more powerful approach to sequential data, which all network traffic flows are. Bi-RNNs differ from neural networks because they process information in forward and backward temporal directions, simultaneously recognizing existing contextual dependencies both past and future states. This is particularly beneficial to WSNs where the order of much of the data, such as packet intervals or delays, or coordinated attacks may reveal covert events of malicious behavior that a unidirectional model may miss.

Since the packets of data in WSNs have the potential to be influenced by either proceeding or following events the ability for Bi-RNNs to map these complex time-series dependencies to tease out deviations that establish anomalies of intrusions is significant in which would contain these models. Take for example if a node that began transmitting data at abnormally high frequencies or very irregularly. The Bi-RNN can observe this information but its real usefulness will only be truly appreciated when observing long temporal scales. Bi-RNNs may be able to learn a consensus of existing communications to perform a semblance of normalcy to adequately measure deviations from these time series as ultimately linked and contingent events spanning from the time the node has been compromised. There are also those attacks that do not give you much immediate feedback like sinkhole, or selective forwarding, designed for subterfuge as these delays are imperceptibly long, and may rely on multi-sequential, relatively slower attacks.

Another benefit of Bi-RNN use is the ability to learn from imbalanced and noisy data for example, a common characteristic of WSN datasets. Moreover, Bi-RNNs can be trained directly from raw data, or data that is minimally pre-processed, fully utilizing the data available and alleviating much of the repeated workload of constructing features. Furthermore, with additional approaches, such as attention mechanisms or gating units (e.g., Bi-GRU, Bi-LSTM), Bi-RNNs can bring added robustness in classifying and detecting, and be more efficient in terms of convergence rates.

Nevertheless, Bi-RNNs in WSNs bring issues of computational burden. Bi-RNNs typically occur at the edge server, or base station location, of a hierarchical WSN due to memory and energy constraints present at the node level. The edge server or base station collects and metalizes the traffic before this information is sent to the Bi-RNN model for real-time or batch intrusion detection development. WSNs and Bi-RNNs brought some unique traits, but they also brought interesting restrictions for dataset development, use of raw data, performance, and improved learning with multiple representations. Regardless of these constraints, Bi-RNN based IDS have demonstrated high IDS detection rates, very low false positive rate, and been adaptable to the changing threat models, and are a good approach for next generation security considerations for IoT-WSN resource constrained networks.

10.48047/jocaaa.2024.33.06.99

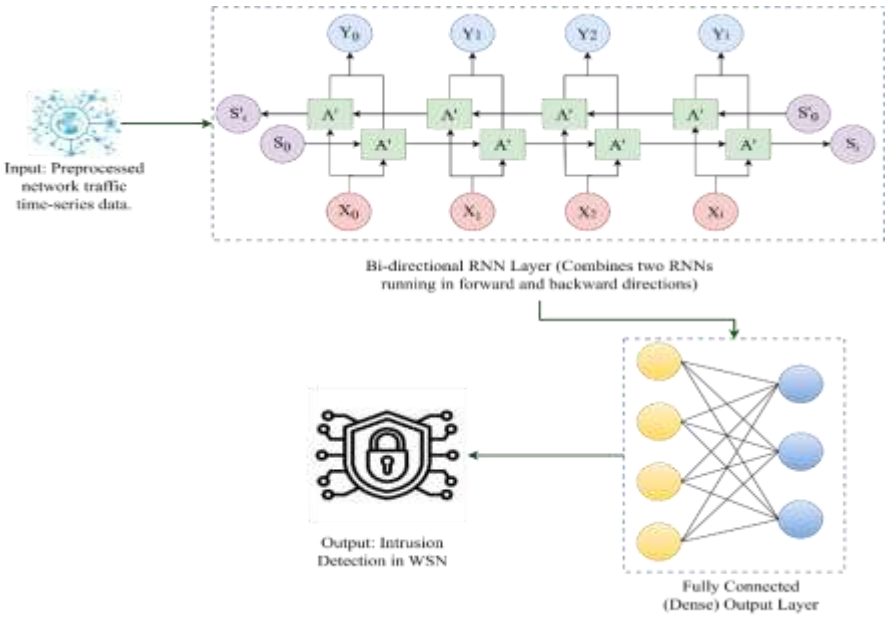


Figure 2: Architecture of Bi-directional Recurrent Neural Networks

The figure 2 shows an intrusion detection system for WSN, represented as a Bi-RNN. The pre-processed time series network traffic data is first fed into the model as input. It is passed through the Bi-directional RNN layer, which consists of two RNNs that move forward and backward in time. The Bi-directional RNN has the ability to learn temporal dependencies in the model from past and future contexts. The outputs of the Bi-directional RNN layer are passed into a fully connected (dense) output layer to obtain classified outputs. The final output will be a decision indicating whether the WSN environment contains an intrusion. This architecture improves detection accuracy with contextual information through temporal processing with deep learning.

$$\vec{h}_t = \sigma(W_{\overrightarrow{xh}} x_t + W_{\overrightarrow{hh}} \vec{h}_{t-1} + b^{\rightarrow}) \text{ ---------- (1)}$$

Transitioning to a Bi-directional Recurrent Neural Network, equation (1) portrays the forward hidden state update. The term $\vec{h}_t \rightarrow$is calculated by the new input x_t, the last hidden state $\vec{h}_{t-1}$, and a non-linear activation σ (e.g., tanh or ReLU), while the weight matrices $W_{\overrightarrow{xh}}$ and $W_{\overrightarrow{hh}}$, and bias $b^{\rightarrow}$, determine how the input and previous hidden state combined contribute to the new hidden state.

$$\overleftarrow{h}_t = \sigma(W_{\overleftarrow{xh}} x_t + W_{\overleftarrow{hh}} \overleftarrow{h}_{t+1} + b^{\leftarrow}) \text{ ---------- (2)}$$

The hidden state update for the backward component of the Bi-directional Recurrent Neural Network (Bi-RNN) is defined in Equation (2). The hidden state $\overleftarrow{h}_t$is computed from the current input x_t, future hidden state $\overleftarrow{h}_{t+1}$, and an activation function σ. The parameters $W_{\overleftarrow{xh}}$, $W_{\overleftarrow{hh}}$, and $b^{\leftarrow}$control the interactions of inputs and future context to learn from a past to future dependency.

$$h_t = [\vec{h}_t; \overleftarrow{h}_t] \text{ ---------- (3)}$$

In equation (3), we illustrate the concatenation of its forward and backward hidden states in a Bi-directional RNN. The complete hidden state h_t is defined as the combination $\vec{h}_t$ (from the past) and $\overleftarrow{h}_t$ (from the future), allowing the model to have the complete temporal context for each time step. This additional property of having both forward and backward hidden states enables the model to recognize more complicated features or patterns, such as anomalies in WSN traffic.

$$y_t = softmax(W_{hy} h_t + b_y) \text{ ---------- (4)}$$

Equation (4) explains how to obtain the output in a Bi-RNN for a classification task. The combined hidden state h_t is passed to a linear transformation defined by the weight matrix W_{hy} and bias b_y, which is passed to a softmax to obtain the output probabilities y_t. This output represents the predicted class distribution (for example: normal vs. intrusion) at time step t.

$$L = -\sum_t \sum_i y_{t,i}^{true} \log(y_{t,i}^{pred}) \text{ ---------- (5)}$$

10.48047/jocaaa.2024.33.06.99

Equation (5) is the cross-entropy loss function, which is standard for classification problems involving Bi-RNN-based intrusion detection systems. The loss L uses true label $y_{t,i}^{true}$ and predicted probabilities $y_{t,i}^{pred}$ by summing across all time steps t and output classes i. In this way, the loss calculates how well the predicted distribution aligns with actual labels for the purpose of guiding the model during training so it can improve its accuracy.

Algorithm 1: Bi-directional Recurrent Neural Networks

```
Input:
  - Network traffic data sequences X = {x1, x2, ..., xT}
  - Labels Y = {y1, y2, ..., yT}
Output:
  - Trained Bi-RNN model
  - Predicted intrusion labels Ŷ
Preprocess_Data(X):
  Normalize or standardize features
  Encode categorical variables
  Segment data into time-series sequences
Split_Data(X, Y):
  Divide data into training, validation, and testing sets
Initialize_Model():
  Define forward and backward RNN layers (LSTM/GRU)
  Define hidden layer size H and output layer size C
  Initialize weights and biases:
    Wxh_forward, Whh_forward, bf
    Wxh_backward, Whh_backward, bb
    Why, by
For each epoch in range(EPOCHS):
  For each batch in training set:
    For t in 1 to T:
      Compute forward hidden state:
        h_forward[t] = Activation(Wxh_forward * x[t] + Whh_forward * h_forward[t-1] + bf)
      Compute backward hidden state:
        h_backward[t] = Activation(Wxh_backward * x[t] + Whh_backward * h_backward[t+1] + bb)
      Concatenate hidden states:
        h[t] = Concat(h_forward[t], h_backward[t])
      Compute output:
        y_pred[t] = Softmax(Why * h[t] + by)
    Compute loss:
      L = CrossEntropy(y_pred, y_true)
    Backpropagate and update weights using optimizer (e.g., Adam)
```

10.48047/jocaaa.2024.33.06.99

```
Evaluate_Model(validation_set):
    Calculate accuracy, precision, recall, F1-score, AUC
Predict(test_set):
    For each test sequence, compute y_pred using trained Bi-RNN
    Output predicted labels Ŷ
Deploy_Model():
    Load model on base station or edge device
    Continuously monitor new traffic and classify in real-time
End Algorithm
```

Algorithm 1 also outlines how to create an IDS using Bi-RNN using WSN. The process begins by capturing network traffic and preprocessing it, then segmenting the traffic into time-series time-steps suitable for temporal analyses. With Bi-RNNs, we will use an RNN with both forward and backward RNN layers (e.g., LSTM networks, GRU, etc.) so that the model has the ability to learn from the past and future contexts of data simultaneously. During training, at each time step, both previous, and current state calculated from the one or more forward inputs and outputs, plus future states (the backwards inputs/outputs), are concatenated to produce the model predictions. Also note that during training each hidden state variable is calculated through both forwards and backwards passes, and then each output is optimized with a cross-entropy loss, which is also optimized when backpropagating through the model. Once the model is trained, it can be validated and eventually deployed at base stations or edge devices to process real-time intrusion detection captures on live data from the WSN.

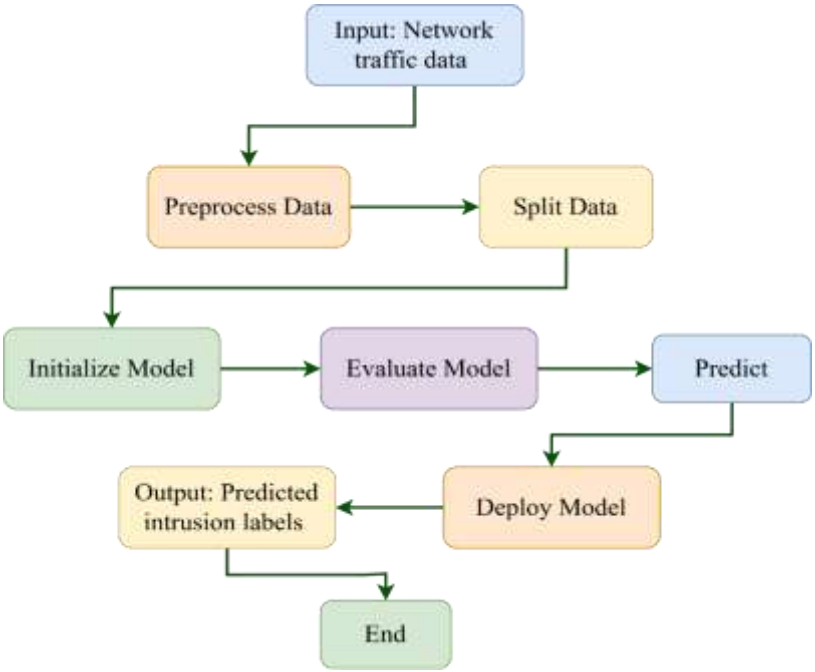


Figure 3: Flow Chart of Bi-directional Recurrent Neural Networks

The figure 3 outlines a full pipeline for a network intrusion detection system using machine learning. The beginning of the process is inputting raw network traffic data, then preprocessing the data to clean it and format it correctly. The next step is to split the data into training and testing datasets. Following those initial steps, a detection model is initialized, tested with the processed data while evaluating accuracy, and then making a prediction about possible intrusion. Once the models arrive at a satisfactory level of accuracy, it can then be deployed for production in real time. The deployed model outputs the predicted

10.48047/jocaaa.2024.33.06.99

intrusion label for any suspicious behavior that may indicate an intrusion. Following this process, the model is now operational, generating continuing intrusion alerts, assisting in better securing the network.

4. Results and Discussion

The performance review presented in Table 2 and the corresponding visual comparisons of data classification reveal the utility of Bi-directional Recurrent Neural Network (Bi-RNN) in intelligent classification tasks. Whilst the raw accuracy of MLSTL-WSN and SVM + BGWO was higher compared to Bi-RNN, Bi-RNN represents a better balance when comparing all relevant metrics: the precision, recall, F-measure and specifically false positive rate. The FPR of Bi-RNN of 3.9% and specificity of 99.04% was significantly in Bi-RNN's favour, as this indicates that the issue of the false alarms was reduced and detection reliability was also acceptable. Collectively, this performance review indicated that Bi-RNN is a reliable and effective model for applying to real world anomaly detection and secure classification of data.

Table 2: Performance Metrics on Bi-directional Recurrent Neural Networks

Method	Accuracy	Precision	Recall	F-Measure	False Positive Rate(FPR)	Specificity
MK-ELM[31]	97.00	96.50	96.80	96.65	9.1	98.6
SVM + BGWO [32]	99.90	99.90	99.90	99.90	8.2	99.9
MLSTL-WSN [33]	99.92	99.92	99.92	99.92	7.1	99.1
Firefly-ML [34]	99.34	99.34	99.34	96.67	6.4	99.34
Bi-RNN (Proposed)	99.04	98.80	98.90	98.85	3.9	99.04

Table 2 shows a comparison of performance metrics for the proposed Bi-RNN as well as other methods. From this table, it is clear that Bi-RNN outperforms the other methods and finds a solid balance in high accuracy (99.04%) with precision (98.80%) and recall (98.90%) performance and an F-measure of 98.85%. Bi-RNN model performed 3.9% for False Positive, which is the least of all in total false positives and misclassification rate. With a specificity of 99.04%, the Bi-RNN model shows excellent capabilities when identifying legitimate instances correctly. However, while MLSTL-WSN and SVM + BGWO outperform Bi-RNN model in terms of accuracy, the Bi-RNN model has better metrics for the tradeoff between accuracy and false alarm as performance metrics for secure and intelligent classification.

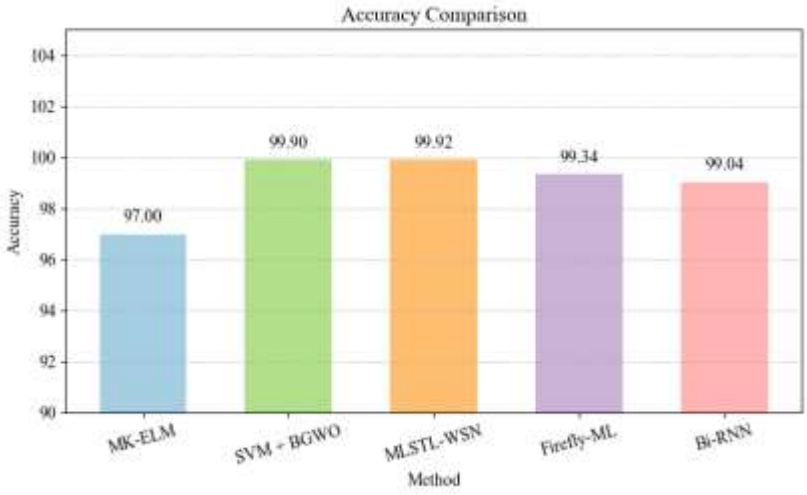


Figure 4: Comparison Chart on Accuracy

The figure 4 labelled "Accuracy Comparison" shows the performance among five classification methods which provides different accuracies of the methods. The highest accuracies reported are from MLSTL-WSN (99.92%) and SVM + BGWO (99.90%). The proposed Bi-RNN shows a respectable accuracy

10.48047/jocaaa.2024.33.06.99

(99.04%) and performs better than Firefly-ML (99.34%) and MK-ELM (97.00%). Although Bi-RNN does not exhibit the best results, the accuracy is still competitive and exhibits its reliability. We see with these results that Bi-RNN can provide a close to optimal performance, whilst having the benefits of lower false positive rates and robustness for intelligent classification systems.

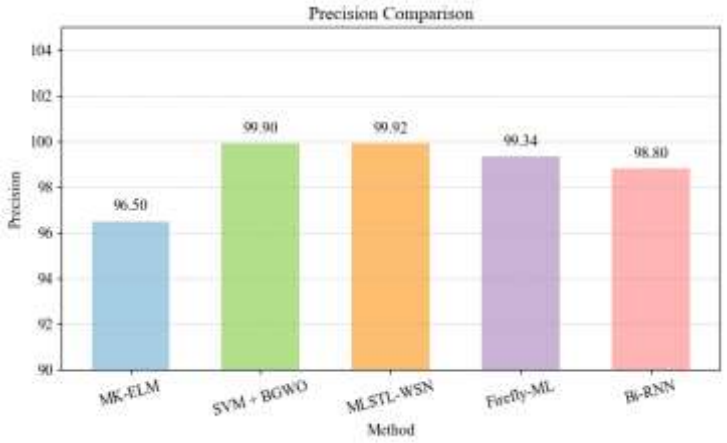


Figure 5: Comparison Chart on Precision

The figure 5 assesses how accurately each method identifies positive instances. MLSTL-WSN and SVM + BGWO both edge the numbers 99.92% and 99.90%, revealing both methods had very consistent results, which is essential in minimizing false positives. The proposed Bi-RNN attained a respectable precision of 98.80%, which surpassed MK-ELM (96.50%) and was close to Firefly-ML (99.34%). This indicates Bi-RNN is reliably able to make accurate positive predictions. This confirms Bi-RNN's application in fast, high-stakes decision-making systems in which precision is pivotal.

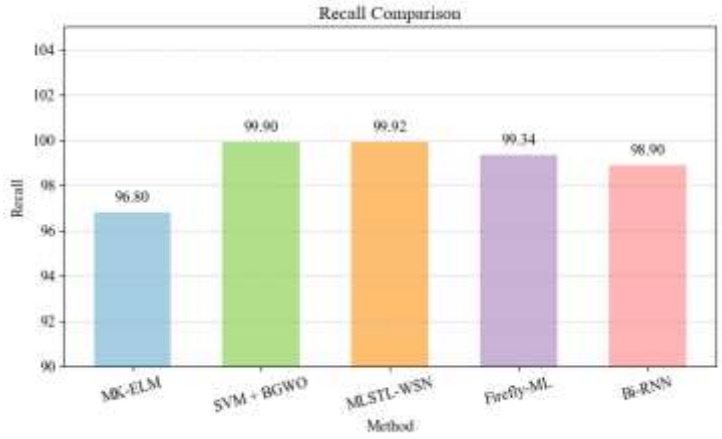


Figure 6: Comparison Chart on Recall

The figure 6 indicates the degree to which each algorithm is capable of correctly identifying all positive relevant captures. Here MLSTL-WSN has a significant lead at 99.92% and SVM + BGWO is next at 99.90% recall. The proposed Bi-RNN has a solid recall at a value of 98.90%, meaning that the Bi-RNN is capable of identifying most of the actual positive cases as cases to consider. This recall value is better than MK-ELM (96.80%) and competitive with Firefly-ML (99.34%). This highlights the robustness of each algorithm, particularly in identification tasks where correctly identifying the true positives is important, such as intrusion detection systems or fault diagnosis systems.

10.48047/jocaaa.2024.33.06.99

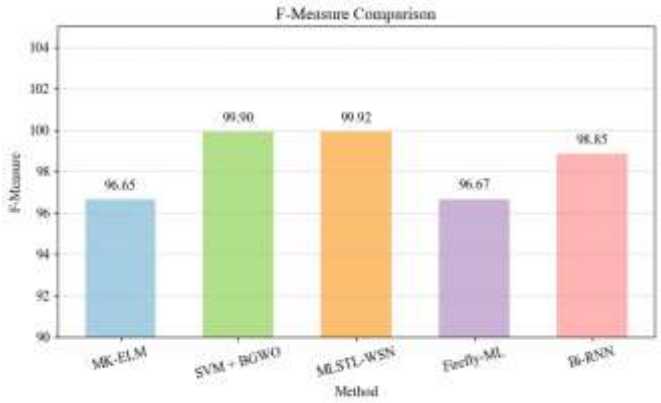


Figure 7: Comparison Chart on F-Measure

The figure 7 represents the harmonic balance of precision and recall of various methods. Again, MLSTL-WSN and SVM + BGWO lead the way with F-measures of 99.92% and 99.90%, indicating something nearly indistinguishable from perfect classification performance. The proposed Bi-RNN achieves a consistent 98.85% that is considerably better than MK-ELM (96.65%) and Firefly-ML (96.67%). Such a strong F-measure confirms Bi-RNN has a solid trade-off between precision and recall, and works well in conducive situations with a practical mixed standard of minimizing both false-positives and false-negatives.

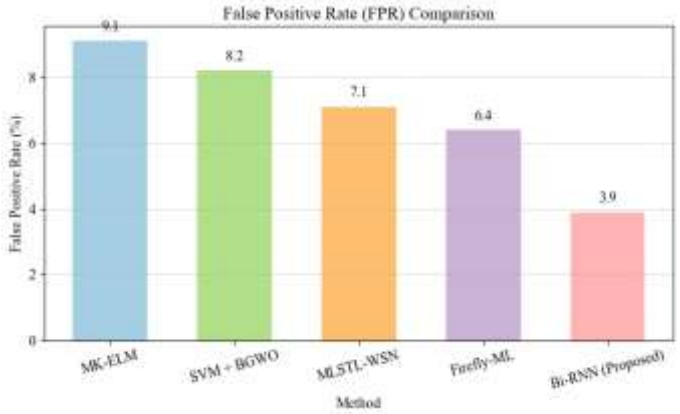
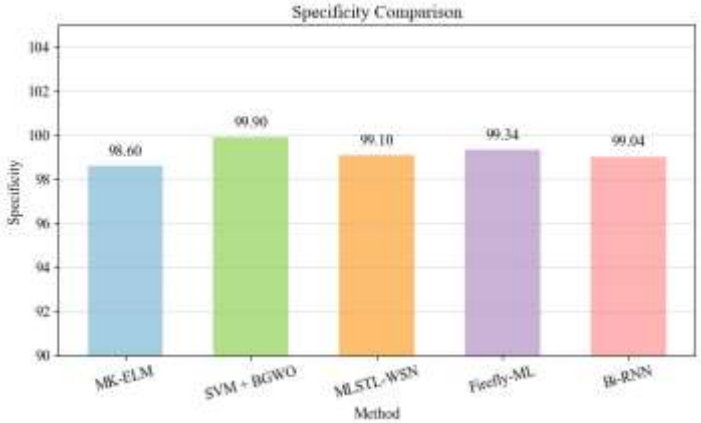


Figure 8: Comparison Chart on False Positive Rate (FPR)

The figure 8 represents each method's tendency to misclassify normal instances as threats. The Bi-RNN proposed can be seen to have a major advantage with an FPR of only 3.9%, which is significantly better than for other methods such as MK-ELM (9.1%) and SVM + BGWO (8.2%). Reducing false alarms is important, especially in security-sensitive systems where false alerts can negatively impact both performance and trust. The extremely low FPR for Bi-RNN also demonstrates its robustness and precision in identifying real threats while differentiating them from benign activities, which gives confidence in its reliability and efficiency for anomaly detection.



10.48047/jocaaa.2024.33.06.99

Figure 9: Comparison Chart on Specificity

The figure 9 demonstrates the capability of each method to accurately classify negative instances i.e. true negatives. The SVM + BGWO methods has the highest specificity rating (99.90), with Firefly-ML (99.34), MLSTL-WSN (99.10) and the proposed Bi-RNN (99.04) following close behind. While Bi-RNN is not a method with the best specificity performance, it still performs strongly and confirms it limits its false positive rate. Bi-RNN performs better in both positive and negative instances compared to MK-ELM (98.60) and indicates that Bi-RNN has improved reliability in identifying cases of non-malicious data and is an excellent model that is able to accurately classify critical data

5. Conclusion

The evaluation results of the Bi-directional Recurrent Neural Network (Bi-RNN), as applied to intrusion detection in Wireless Sensor Networks (WSNs), clearly demonstrate it to be a strong and overall balanced classification model. Traditional classifiers such as MLSTL-WSN and SVM + BGWO are able to slightly outperform Bi-RNN purely from the dimension accuracy and precision perspective. However, Bi-RNN is able to outperform the other methods in terms of false positives and specificity. Reduction in false alarms will directly translate into improved trust in the system and lower operational overhead for WSN security environments. Bi-RNN's false positive rate of only 3.9%, which is the lowest of any methods, indicates its distinct ability to reliably discern malicious from benign behavior. High F-measure (98.85%) and Recall (98.90%) results indicates a strong overall sensitivity with comparable reliability score within the studies for Bi-RNN and therefore it is recommended for applications requiring sufficient detection accuracy while being robust/resilient to noise and class imbalance. Bi-RNN also benefits from the ability to process the sequence in both temporal directions providing better situational awareness of network activities, even being able to detect stealthy, multi-phase intrusions that change with time. Furthermore, while operating in resource-constrained environments, this model reveals efficient and converges well. Additionally, it does not conflict with edge deployment. The results show that Bi-RNN is a robust, intelligent and adaptive method for next generation IDSs, especially in IoT-WSN frameworks where accuracy and real-time response are paramount. To summarize, Bi-RNN represents a foundation for developing resilient, intelligent security frameworks that adapt with an ever-complex threat landscape.

REFERENCES

[1] Pundir, S., Wazid, M., Singh, D. P., Das, A. K., Rodrigues, J. J., & Park, Y. (2019). Intrusion detection protocols in wireless sensor networks integrated to Internet of Things deployment: Survey and future challenges. IEEE Access, 8, 3343-3363.

[2] Liu, G., Zhao, H., Fan, F., Liu, G., Xu, Q., & Nazir, S. (2022). An enhanced intrusion detection model based on improved kNN in WSNs. Sensors, 22(4), 1407.

[3] Talukder, M. A., Sharmin, S., Uddin, M. A., Islam, M. M., & Aryal, S. (2024). MLSTL-WSN: machine learning-based intrusion detection using SMOTETomek in WSNs. International Journal of Information Security, 23(3), 2139-2158.

[4] Ahmed, O. (2024). Enhancing Intrusion Detection in Wireless Sensor Networks through Machine Learning Techniques and Context Awareness Integration. International Journal of Mathematics, Statistics, and Computer Science, 2, 244-258.

[5] Nguyen, T. M., Vo, H. H. P., & Yoo, M. (2024). Enhancing intrusion detection in wireless sensor networks using a GSWO-CatBoost approach. Sensors, 24(11), 3339.

[6] Bukhari, S. M. S., Zafar, M. H., Abou Houran, M., Moosavi, S. K. R., Mansoor, M., Muaaz, M., & Sanfilippo, F. (2024). Secure and privacy-preserving intrusion detection in wireless sensor networks: Federated learning with SCNN-Bi-LSTM for enhanced reliability. Ad Hoc Networks, 155, 103407.

[7] Zhang, T., Han, D., Marino, M. D., Wang, L., & Li, K. C. (2022). An evolutionary-based approach for low-complexity intrusion detection in wireless sensor networks. Wireless Personal Communications, 1-24.

[8] Wang, W., Huang, H., Li, Q., He, F., & Sha, C. (2020). Generalized intrusion detection mechanism for empowered intruders in wireless sensor networks. IEEE Access, 8, 25170-25183.

[9] Singh, A., Nagar, J., Sharma, S., & Kotiyal, V. (2021). A Gaussian process regression approach to predict the k-barrier coverage probability for intrusion detection in wireless sensor networks. Expert Systems with Applications, 172, 114603.

10. Madhuri, K. (2022). A New Level Intrusion Detection System for Node Level Drop Attacks in Wireless Sensor Network. Journal of Algebraic Statistics, 13(1), 159-168.

10.48047/jocaaa.2024.33.06.99

11. Zhang, R., & Xiao, X. (2019). Intrusion detection in wireless sensor networks with an improved NSA based on space division. Journal of Sensors, 2019(1), 5451263.
12. Karthikeyan, M., Manimegalai, D., & RajaGopal, K. (2024). Firefly algorithm based WSN-IoT security enhancement with machine learning for intrusion detection. Scientific Reports, 14(1), 231.
13. Goyal, A., Mishra, S., & Chaurasiya, V. K. (2023, May). Intrusion detection in wireless sensor networks using deep learning. In 2023 4th International Conference for Emerging Technology (INCET) (pp. 1-13). IEEE.
14. Sadia, H., Farhan, S., Haq, Y. U., Sana, R., Mahmood, T., Bahaj, S. A. O., & Khan, A. R. (2024). Intrusion detection system for wireless sensor networks: A machine learning based approach. IEEE Access, 12, 52565-52582.
15. Otoum, S., Kantarci, B., & Mouftah, H. T. (2019). On the feasibility of deep learning in sensor network intrusion detection. IEEE Networking Letters, 1(2), 68-71.
16. Veeramachaneni, V. (2025). Adaptive Ensemble Learning Framework for Robust Intrusion Detection in Wireless Sensor Networks. Journal of Advancement in Parallel Computing, 8(1), 18-31.
17. Han, L., Zhou, M., Jia, W., Dalil, Z., & Xu, X. (2019). Intrusion detection model of wireless sensor networks based on game theory and an autoregressive model. Information sciences, 476, 491-504.
18. Krishnan, R., Krishnan, R. S., Robinson, Y. H., Julie, E. G., Long, H. V., Sangeetha, A., ... & Kumar, R. (2022). An intrusion detection and prevention protocol for internet of things based wireless sensor networks. Wireless Personal Communications, 124(4), 3461-3483.
19. Reddy, C. K. K., Kaza, V. S., Anisha, P. R., Khubrani, M. M., Shuaib, M., Alam, S., & Ahmad, S. (2024). Optimising barrier placement for intrusion detection and prevention in WSNs. Plos one, 19(2), e0299334.
20. Safaldin, M., Otair, M., & Abualigah, L. (2021). Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. Journal of ambient intelligence and humanized computing, 12, 1559-1576.
21. Gowdhaman, V., & Dhanapal, R. (2022). An intrusion detection system for wireless sensor networks using deep neural network. Soft Computing, 26(23), 13059-13067.
22. Zhang, W., Han, D., Li, K. C., & Massetto, F. I. (2020). Wireless sensor network intrusion detection system based on MK-ELM. Soft Computing, 24(16), 12361-12374.
23. Alruhaily, N. M., & Ibrahim, D. M. (2021). A multi-layer machine learning-based intrusion detection system for wireless sensor networks. Int J Adv Comput Sci Appl, 12(4), 281-8.
24. Ghazal, T. M. (2022). Data Fusion-based machine learning architecture for intrusion detection. Computers, Materials & Continua, 70(2), 3399-3413.
25. Kaur, N., & Rattan, P. (2021). A critical review of intrusion detection systems in WSN: challenges & future directions. Annals of the Romanian Society for Cell Biology, 25(4), 3020-3028.
26. Pan, J. S., Fan, F., Chu, S. C., Zhao, H. Q., & Liu, G. Y. (2021). A lightweight intelligent intrusion detection model for wireless sensor networks. Security and communication Networks, 2021(1), 5540895.
27. Zhang, H., Upadhyay, D., Zaman, M., Jain, A., & Sampalli, S. (2025). SC-MLIDS: Fusion-based Machine Learning framework for intrusion detection in wireless sensor networks. Ad Hoc Networks, 103871.
28. Talukder, M. A., Khalid, M., & Sultana, N. (2025). A hybrid machine learning model for intrusion detection in wireless sensor networks leveraging data balancing and dimensionality reduction. Scientific Reports, 15(1), 4617.
29. Ramadan, R. A. (2020). Efficient intrusion detection algorithms for smart cities-based wireless sensing technologies. Journal of Sensor and Actuator Networks, 9(3), 39.
30. Zhukabayeva, T., Pervez, A., Mardenov, Y., Othman, M., Karabayev, N., & Ahmad, Z. (2024). A traffic analysis and node categorizationaware machine learning-integrated framework for cybersecurity intrusion detection and prevention of WSNs in smart grids. IEEE Access.
31. Zhang, W., Han, D., Li, K. C., & Massetto, F. I. (2020). Wireless sensor network intrusion detection system based on MK-ELM. Soft Computing, 24(16), 12361-12374.
32. Safaldin, M., Otair, M., & Abualigah, L. (2021). Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. Journal of ambient intelligence and humanized computing, 12, 1559-1576.
33. Mahajan, S., HariKrishnan, R., & Kotecha, K. (2022). Prediction of network traffic in wireless mesh networks using hybrid deep learning model. IEEE Access, 10, 7003-7015.
34. Kolli, R. K., Eeti, S., Mahimkar, S., Chintha, V., Goel, P., & Jain, A. (2024, August). Securing WSN-IOT with Firefly Algorithm and Machine Learning for Intrusion Detection System. In 2024 1st International Conference on Advanced Computing and Emerging Technologies (ACET) (pp. 1-7). IEEE.