Vol.8 Issue 4 Oct 2025

Multimedia Research

Received 8 September, Revised 11 September, Accepted 10 October

# ResNeXT_TEI: A Deep Learning Framework with Integrated Loss Function for Attack Detection

**Vishal Sharad Hingmire**
*Department of E & TC Engineering,*
*Arvind Gavali College of Engineering,*
*Satara, Maharashtra, India.*
*vs.hingmire@gmail.com*

**Abstract:** Detecting attacks in Wireless Sensor Networks (WSNs) is vital to protect data integrity, prevent unauthorized access, ensure reliable communication, and protect resource-constrained nodes from threats like spoofing and routing attacks in hostile environments. Since WSNs are typically deployed in unattended and resource-limited settings, they are highly susceptible to both external adversaries and internal compromised nodes. Effective identification of malicious activities in WSNs safeguards data accuracy, enhances operational reliability, and supports extended system lifespan. This is particularly important in mission-critical applications such as military surveillance, disaster response, and industrial monitoring, where undetected attacks could cause catastrophic failures. However, prevailing methods often fail to attain high detection efficiency due to noisy inputs, dimensional complexity, and high false positives. High-dimensional sensor data often contains redundant or irrelevant features, which makes traditional machine learning models prone to overfitting and misclassification. Thus, an effective model, termed ResNeXT_Tri-Error Index (ResNeXT_TEI), is established for attack detection in WSNs. Initially, WSN is simulated, and then, the input network log data is applied to data normalization. The data normalization is performed by using median normalization for transforming the data into a consistent format. After that, feature selection is done by utilizing the Fisher score. Lastly, attack detection is accomplished by the introduced ResNeXT_TEI. In addition, TEI is developed by the combination of Mean Squared Logarithmic Error (MSLE), Mean Bias Error (MBE), and Mean Absolute Error (MAE). Furthermore, the developed ResNeXT_TEI achieved a superior accuracy, True Positive Rate (TPR), and True Negative Rate (TNR) of 96.690%, 97.368%, and 96.118%, respectively.

**Keywords:** Wireless Sensor Networks, Resnext, Median Normalization, Fisher Score, Attack Detection

***Nomenclature***

| Abbreviation | Description |
|---|---|
| WSN | Wireless Sensor Network |
| IoT | Internet of Things |
| M2M | Machine-to-Machine |
| ML | Machine Learning |
| DL | Deep Learning |
| CNN | Convolutional Neural Networks |
| Bi-LSTM | Bidirectional Long Short-Term Memory |
| SCO-LSTM | Single Candidate Optimizer-Long Short-Term Memory |
| MDCNN-Bi-LSTM-AM | Modified Deep Convolutional Neural Network with Bi-LSTM and Attention Mechanism |
| DoS | Denial of Service |
| DDoS | Distributed DoS |
| GAN | Generative Adversarial Network |
| WOA | Whale Optimization Algorithm |
| ECS-WSN-SAPVAGAN-HBA | Enhancing Cyber Security in WSN using SAPVAGAN optimized with Honey Badger Algorithm |
| IDS | Intrusion Detection Systems |
| SG-IDS | Stochastic Gradient Descent-based IDS |
| GAN | Generative Adversarial Network |
| SGD | Stochastic Gradient Descent |
| SVM | Support Vector Machines |
| KNN | K-Nearest Neighbours |
| GB | Gradient Boosting |
| DT | Decision Trees |
| LR | Logistic Regression |
| ReLU | Rectified Linear Unit |



https://doi.org/10.46253/j.mr.v8i4.a2

# 1. Introduction

WSNs, characterized by the interaction of autonomous and spatially dispersed nodes, form a sophisticated framework for data collection. These networks operate in diverse and often hostile environments, making them highly valuable for continuous sensing and monitoring tasks. In recent years, this technology has gained increased interest, mainly due to its wide applicability and cost-effectiveness. WSNs typically consist of numerous compact sensing devices that are affordable, energy-efficient, and capable of short-range wireless communication, despite having limited memory and processing resources [1]. The compactness and affordability of these nodes enable large-scale deployment, but the inherent resource constraints create trade-offs between performance and security. This limitation hinders the adoption of advanced security frameworks, leaving many WSN deployments vulnerable to sophisticated cyberattacks. Modern systems depend on WSNs for efficient, real-time collection and transmission of data across sectors like health, automation, and environmental monitoring [2]. A key limitation of WSNs is the restricted processing power of their sensors, which can affect system performance and security. This challenge makes it difficult to implement security-enhancing algorithms within the network [3]. Attackers may deliberately exploit vulnerabilities to launch various attacks, potentially leading to unauthorized access and exposure of confidential data [4]. Such breaches may not only compromise sensitive information but also disrupt mission-critical operations. Therefore, detecting and preventing attacks is crucial for maintaining the integrity and reliability of WSNs [5]. Malicious devices can impersonate legitimate nodes, allowing attackers to infiltrate networks, eavesdrop on data transmissions, and carry out disruptive actions such as DoS attacks [6] [7]. These threats exploit the collaborative nature of WSNs, where trust among nodes is assumed but not always verified. In WSNs, attacks are deliberate efforts by adversaries to impair network functionality, compromise data integrity, or deplete the computational and energy resources of sensor nodes. Such attacks can severely compromise the reliability and performance of critical applications. Further, they can disrupt systems and services by making them temporarily or permanently inaccessible to users [2].

ML has recently introduced innovative strategies for improving the deployment and functionality of WSNs. ML-based models can learn from patterns in sensor data, enabling them to distinguish between legitimate and malicious activity with minimal human intervention. ML offers a foundational framework for integrating various technological domains, enabling seamless interaction among M2M systems and cyber-physical infrastructures, the IoT. While different supervised ML techniques, such as LR, DT, KNN, GB, SVM, and ensemble methods like the Voting classifier [2], have been used to improve WSN security, many current studies do not fully address key aspects like scalability, adaptability, and real-time responsiveness in dynamic threat environments. Despite notable progress in applying WSNs across various fields, a significant research gap persists at the intersection of WSN security and ML, especially regarding cyber-attack detection [8]. Security provisioning is an essential requirement for both wired and wireless communication infrastructures. However, implementing robust security measures in wireless systems, particularly those with limited resources, presents greater challenges than in conventional wired networks [9] [10]. Deep Learning models have been increasingly exploited to classify events in WSNs as either attack-related or benign, supporting automated threat detection. CNNs are employed to extract meaningful features from input data, while Bi-LSTM networks improve the performance of deep CNNs by capturing temporal dependencies in both forward and backward directions [11] [12]. These deep learning methods provide enhanced accuracy but often come with computational costs, creating a demand for models that balance precision with efficiency.

In this paper, a novel technique, named ResNeXT_TEI, is devised for detecting attacks in WSNs. Here, the WSN simulation is conducted first, and network log data is subjected to normalization using the median normalization technique. Feature selection is formerly accomplished by exploiting the Fisher score method. Finally, the proposed ResNeXT_TEI model is used for detecting attacks. Here, the ResNeXT is integrated with the TEI loss function, which incorporates MSLE, MAE, and MBE.

The primary contribution of this paper is shown below,

- **Established ResNeXT_TEI for attack detection in WSNs:** An efficient model, called ResNeXT_TEI, is developed for detecting attacks in WSNs in this research. Here, attack detection is accomplished using ResNeXT, which is incorporated with TEI designed by merging MSLE, MBE, and MAE.

This paper proceeds as follows: Section 2 surveys several works on detecting attacks. Section 3 introduces the ResNeXT_TEI model. Section 4 covers results and analysis, and Section 5 outlines the conclusion and future directions.

# 2. Motivation

The literature review of various frameworks related to attack detection in WSNs is detailed in this section. Existing studies have demonstrated that while traditional IDS can identify certain threats, they often fail to adapt to the dynamic and resource-constrained nature of WSNs. Attack detection in WSNs is vital due to their vulnerability to threats, limited resources, and deployment in critical environments. Ensuring timely identification of malicious activity safeguards data integrity, network reliability, and operational continuity in sensitive applications. Several models have been proposed for detecting attacks in WSN effectively; however, they suffer from high false positives. High false positive rates not only reduce detection efficiency but also increase unnecessary energy consumption due to false alarms, further draining already resource-constrained nodes. This research establishes a novel technique for detecting attacks, thereby enhancing WSN security.

## 2.1 Literature Survey

Dhanalakshmi, N., [12] proposed a MDCNN- Bi-LSTM- AM for attack detection and classification in WSNs. This technique reduced delay, extended network lifetime, and ensured energy-efficient and reliable data transmission. By combining deep CNNs with Bi-LSTM and attention mechanisms, the framework effectively captured both spatial and temporal dependencies. However, this model failed to focus on employing several over-sampling approaches to overcome the class imbalance factor and apply federated learning for better epoch training. ECS-WSN-SAPVAGAN HBA was established by Meenakshi, B., and Karunkuzhali, D. [13] for enhancing cyber security in WSN. This model had a high detection accuracy, reduced computational time, high robustness, scalability, and efficiency. Its integration of adversarial learning and heuristic optimization made it effective against diverse threats. Nonetheless, this technique was not suitable for real-time applications. Ramesh, R.B., *et al.* [10] designed a GAN with WOA for detection and prevention in WSN security against black hole and wormhole attacks. This technique robust security, faster convergence, reduced training time, and required a smaller number of parameters. The use of GANs also provided the benefit of generating synthetic data, which improved training efficiency. Despite its merits, the approach lacked implementation in a real-world testbed and omitted energy-efficient routing considerations. A SGD based IDS was introduced by Saleh, H.M., *et al.* [14] for WSN attack detection. This method was lightweight, reduced energy consumption, demonstrated adaptability, and improved real-time performance. Meanwhile, this approach failed to mitigated overfitting by integrating advanced detection mechanisms and delivering reliable real-time performance. The reliance on stochastic gradient descent allowed the model to converge quickly with minimal computational overhead. Nandhini, S., *et al.* [15] developed a SCO-LSTM for cyber-attack detection in WSNs. This technique consumed less energy, demonstrated better applicability, and had high detection accuracy; however, this approach failed to address attacks in the data link layer in IoT-WSN. This omission limits its effectiveness in multi-layered intrusion scenarios, where adversaries exploit vulnerabilities across multiple protocol levels.

## 2.2 Limitations

The following issues met in existing attack detection methods in WSNs:

- MDCNN-Bi-LSTM-AM in [12] effectively enhanced security over insecure channels by optimizing key distribution and eliminating irrelevant data. However, this model struggled with identifying in-network anomalies, such as subtle or evolving attack patterns. As a result, its robustness against adaptive adversaries remained limited.
- In [13], ECS-WSN-SAPVAGAN-HBA improved learning capabilities and generalization across different types of attacks and potentially leading to higher precision and recall in attack detection. Meanwhile, maintaining performance and accuracy may become difficult without further optimization or hierarchical modelling. The absence of hierarchical frameworks restricted scalability in complex, real-time WSN environments.
- The GAN+WOA in [10] optimized node selection to minimize unnecessary transmissions and reduced training overhead by generating realistic synthetic attack data effectively. Nevertheless, this method suffered from high false negatives. This drawback is critical because undetected attacks can silently degrade the performance of WSNs without triggering alerts.
- Prevailing models for attack detection in WSNs struggle with high data dimensionality, redundant features, and poor accuracy. Noise and irrelevant attributes hinder detection, while resource constraints demand lightweight solutions that traditional methods often fail to provide. Consequently, there is a pressing need for models that can combine dimensionality reduction, efficient feature selection, and robust learning to handle real-world attack scenarios.

# 3. Proposed ResNeXT_TEI for detecting attacks in WSNs

Detecting attacks in WSNs is crucial to ensure data integrity, confidentiality, and network availability. Without proper detection mechanisms, even a single compromised node can cascade failures across the network, threatening mission-critical applications. WSNs are often deployed in critical environments and making them vulnerable to attacks that can disrupt operations, steal data, or damage infrastructure. Examples include healthcare monitoring systems where tampered readings may endanger patients, or industrial control systems where disruptions can halt production lines. Besides, challenges like limited energy, low processing power, and hamper effective detection. Thus, an efficient framework, termed ResNeXT_TEI, is devised for detecting attacks in WSNs. Firstly, the WSN is simulated. Afterward, the network log data is collected from the dataset and subjected to normalization using median normalization [16]. Median normalization ensures robustness against outliers, which are common in WSN data due to noisy signals and environmental variations. Then, feature selection is performed exploiting Fisher score [17]. Thereafter, the proposed ResNeXT_TEI model is used for attack detection. Furthermore, ResNeXT [18] is used for detection and is used with the TEI loss function, which integrates MAE [19], MSLE [19], and MBE [19]. This integration allows the training process to penalize bias, scale-sensitive deviations, and average errors simultaneously, thereby improving generalization and reducing false positives. In Figure 1, structural representation of established ResNeXT_TEI for detecting attacks in WSNs. The Figure 1 illustrates how the preprocessing, feature selection, and deep learning components interact within the framework to provide a holistic attack detection pipeline.

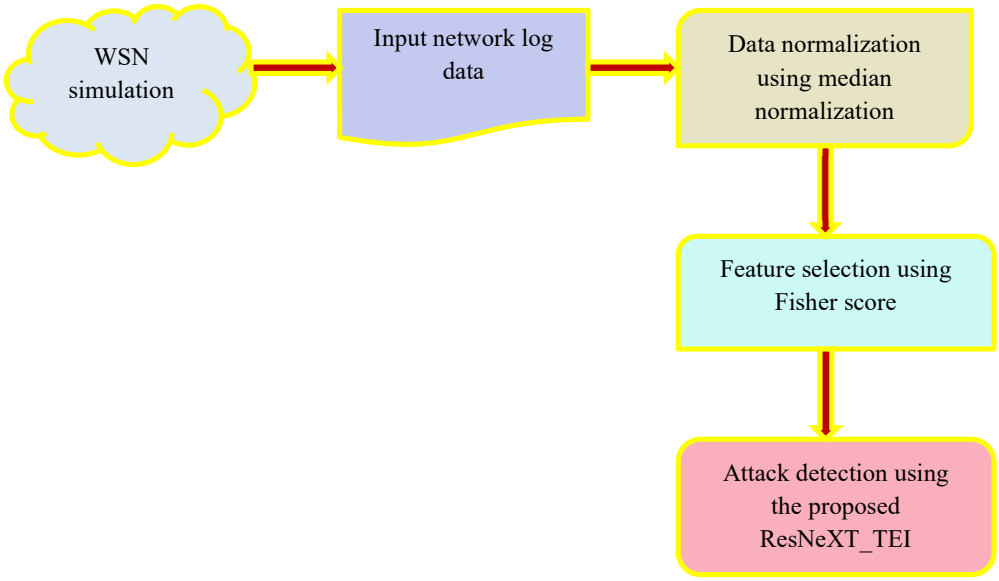


***Fig. 1.*** *Diagrammatic layout of proposed ResNeXT_TEI for detecting attacks in WSNs*

## 3.1 Acquisition of Network Log Data

Let us assume network log data considered as input for attack detection collected from database $M$ and is given by,

$$M = \left\{ m_1, m_2, ....., m_e, ....., m_f \right\} \tag{1}$$

where, $m_e$ symbolizes $e^{th}$ network log data and $f$ signifies the number of data records in the dataset. These logs typically contain crucial information such as timestamps, source and destination addresses, packet types, and protocol behaviors, which form the foundation for identifying suspicious or malicious patterns.

## 3.2 Normalization of data using Median Normalization

Data normalization refers to the systematic process of converting diverse and unstructured cybersecurity data into a unified format suitable for analysis. In this context, network log data $m_e$ is subjected to data normalization using median normalization [16]. Unlike min-max normalization, which is sensitive to extreme values, median normalization offers better resilience to noisy outliers commonly present in WSN traffic data. Median normalization is robust, preserves data distribution, simplifies scaling, and improves

model stability. In median normalization, the median of all values in a given attribute is used to normalize each input. The normalized value $N$ of $p^{th}$ attribute $A$ in the input is formulated by,

$$N = \frac{p}{median(A)} \tag{2}$$

The data normalized output is represented as $N$ .This transformation ensures that all features are comparable in scale, thereby preventing bias during the training of the ResNeXT_TEI model.

## 3.3 Feature Selection using Fisher Score

Feature selection is the process of isolating key features from a dataset that significantly influence the performance of models for detection of attacks. Reducing irrelevant or redundant attributes not only improves classification accuracy but also lowers computation costs, which is critical for WSNs with constrained resources. Here, the normalized data $N$ is provided as input for feature selection, and the most salient features are selected by utilizing Fisher score [17]. The Fisher score is a simple, fast, and effective method for selecting features from high-dimensional data. It operates by measuring the ratio of inter-class separation to intra-class variability, ensuring that selected features maximize class discrimination. The Fisher Score is employed to rank the features, and the highest scoring feature chosen for further analysis and expressed as,

$$Score_r = \frac{\sum_{h=1}^{B} d_r \left( \bar{k}_r^h - \bar{k}_r \right)^2}{\sum_{h=1}^{B} d_r \left( \sigma_r^h \right)^2} \tag{3}$$

where, $B$ indicates several classes, $d_r$ denotes overall samples in class $r$ , $\bar{k}_r$ stands for the mean, $Score_r$ denotes Fisher score, $\sigma_r^h$ and $\bar{k}_r^h$ represents the variance and mean of the class $r$ corresponding to the feature $F_r$ . The selected feature is signified as $D$ . By applying this step, the dimensionality of the input dataset is reduced, leading to faster convergence and improved generalization for the detection model.

## 3.4 Attack detection using ResNeXT_TEI

WSNs are often deployed in sensitive environments like healthcare systems, military zones, or industrial monitoring, making them prime targets for cyberattacks. The consequences of undetected intrusions in such contexts may include loss of human life, exposure of classified information, or large-scale operational disruptions. However, prevailing attack detection approaches face issues, including limited scalability, energy constraints, and high false positives, reducing overall reliability and efficiency. Here, the selected feature $D$ is provided as input to detecting attacks, which is performed by using ResNeXT_TEI. ResNeXT, with its aggregated transformations and parallelized architecture, is particularly suitable for extracting rich hierarchical representations from structured log data. Here, ResNeXT, applied for attack detection, is used with TEI, which is a novel loss function designed by combining MSLE [19], MAE [19], and MBE [19].

### 3.4.1 Architecture of ResNeXT

Here, the selected feature $D$ is fed as input to ResNeXt [18]. The choice of ResNeXt is motivated by its proven capability to balance computational efficiency and high representational power, making it well-suited for resource-constrained WSN environments. ResNeXt enhances attack detection with high accuracy, efficient feature extraction, scalability, reduced false alarms, and robust performance in complex WSN environments. The ResNeXt architecture, derived from the residual network, comprises ReLU activation, post-batch normalization, residual connections to mitigate vanishing gradients, and grouped convolutions to reduce computational complexity while maintaining depth. Residual connections ensure that critical gradient information is preserved during backpropagation, enabling deeper networks to be trained without performance degradation. The final classification is performed via a dense layer using a softmax function. Grouped convolutions reduce computational load by extracting feature maps through partitioned kernel filters. Leaky ReLU is used after batch normalization to improve non-linearity and reduce vanishing gradient and saturation problems during training. Unlike traditional ReLU, the Leaky ReLU avoids the “dying neuron” problem by allowing a small, non-zero gradient when the input is negative, improving model stability. In a simple neuron, the inner product is applied on the input vector and the weight vector, and the generated outputs are aggregated and the aggregated transformation is given by,

$$\sum_{a=1}^{I} b_a D_a \tag{4}$$

where, $D=\{D_1,D_2,D_3,.....,D_I\}$ represents the input vector, $I$ indicates dimensionality, $b_a$ denotes filter weight of $a^{th}$ neuron. In ResNeXT, the elementary transformation is swapped with generic function and Aggregated transformation is expressed by,

$$E(D)=\sum_{b=1}^{g}G_b(D) \tag{5}$$

where, $E(D)$ indicates an aggregated transformation function applied to the input $D$, $G_b(D)$ refers to an arbitrary transformation function, and $g$ stands for cardinality. The concept of cardinality, which refers to the number of independent transformation paths, has been shown to be a more effective dimension of improvement compared to merely increasing network depth or width. The residual block of ResNeXt is formulated as,

$$q=D+\sum_{b=1}^{g}G_b(D) \tag{6}$$

where, $q$ represents the output of ResNeXt. By stacking these residual blocks with aggregated transformations, ResNeXt achieves a strong balance between accuracy and efficiency, essential for real-time WSN attack detection. The final dense layer employs a softmax activation function to transform raw logits into class probabilities. Hence, the output reveals if the input feature is linked to any attack. The ResNeXt structure is displayed in Figure 2.

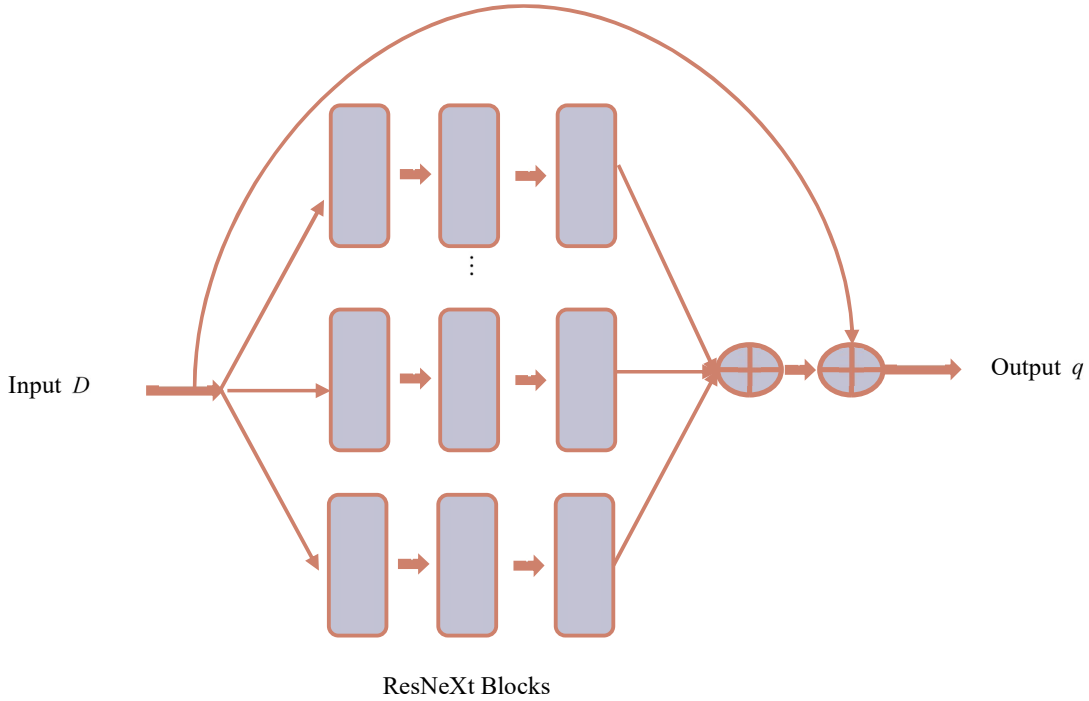


***Fig. 2**. ResNeXt structure*

### 3.4.2 Proposed Tri-Error Index

Here, TEI and it is formulated by fusing MSLE [19], MBE [19], and MAE [19]. The intuition behind this design is that no single error metric can fully capture the complex behavior of prediction errors in noisy WSN datasets. By combining complementary measures, TEI provides a holistic evaluation framework. In addition, TEI evaluates the gap between expected and actual outputs to improve model reliability. This integration ensures that both magnitude-based deviations and directional biases are taken into account during training. Moreover, TEI boosts attack detection by ensuring balanced error handling and more accurate threat identification in WSNs.

**a) MSLE**

MSLE [19] is a regression loss metric utilized to evaluate the accuracy of predicted values compared to actual values, especially when the data spans several orders of magnitude. It is particularly beneficial in

WSN attack detection, where abnormal network traffic may vary drastically in scale compared to normal traffic patterns. The MSLE is formulated as,

$$MSLE = \frac{1}{C}\sum_{n=1}^{C}\left[\log\left(1+q_n\right)-\log\left(1+s_n\right)\right]^2 \tag{7}$$

where, $s_n$ indicates predicted value, $q_n$ specifies observed value, $C$ represents the sample size. By penalizing underestimations more strongly than overestimations, MSLE helps in detecting subtle anomalies that might otherwise go unnoticed.

**b) MAE**

MAE [19] is a commonly used metric in regression analysis to compute the accuracy of predictions and is mathematically modelled as,

$$MAE = \frac{1}{C}\sum_{n=1}^{C}\left|s_n - q_n\right| \tag{8}$$

Unlike MSLE, which emphasizes logarithmic differences, MAE provides a straightforward average of absolute deviations, making it robust to outliers and easy to interpret.

**c) MBE**

MBE [19] is a statistical metric exploited to evaluate the accuracy of predictive models. Moreover, it quantifies the tendency of a model to consistently overestimate or underestimate actual values. A positive MBE indicates systematic overestimation, whereas a negative MBE reveals underestimation. The MBE is defined as,

$$MBE = \frac{1}{C}\sum_{n=1}^{C}\left(s_n - q_n\right) \tag{9}$$

The TEI is computed by combining MAE, MBE, and MSLE and is given by,

$$L_{TEI} = \alpha * MSLE + \beta * MAE + \gamma * MBE \tag{10}$$

where, $\alpha$, $\beta$, and $\gamma$ characterizes weights. These weights can be fine-tuned to prioritize different error types depending on the deployment context, such as mission-critical defense networks where false negatives are costlier than false positives. The TEI improves the ability of ResNeXT in effectively handling imbalanced data and minimizes false alarms, while improving detection accuracy. This makes the ResNeXT_TEI framework more adaptable and reliable in real-world WSN scenarios where attack types and frequencies vary dynamically.

# 4. Result and Discussion

Here, estimation of ResNeXT_TEI efficacy for detecting attacks in WSN regarding various performance metrics is illustrated. The discussion not only highlights the numerical performance of the model but also emphasizes its practical relevance in addressing the limitations of existing detection techniques.

## 4.1 Experimental Setup

The proposed ResNeXT_TEI for detecting attacks in WSN is employed using PYTHON tool. The experiments were executed on a high-performance computing environment to ensure reproducibility and fairness in comparisons with baseline methods.

## 4.2 Dataset Description

The BoT-IoT dataset [20] serves as a standardized resource for investigating anomaly detection in smart home IoT systems. It comprises both benign and adversarial network traffic, simulated through a realistic testbed that replicates actual IoT conditions. This dataset is widely accepted in cybersecurity research because of its diversity and scale, making it suitable for benchmarking advanced intrusion detection models. The dataset encompasses diverse cyberattack types, including DDoS, DoS, data theft, reconnaissance, and Exfiltration.

## 4.3 Performance Measures

The efficacy of the established ResNeXT_TEI is assessed for attack detection by using evaluation metrics, including TNR, accuracy, and TPR. These metrics were chosen because they collectively reflect the balance between detecting true threats, avoiding false alarms, and ensuring overall prediction reliability.

**a) Accuracy**

Accuracy measures the proportion of correctly identified instances, such as both attacks and normal traffic, out of overall instances analyzed and is defined as,

$$Accuracy = \frac{W + X}{W + X + Y + Z} \tag{11}$$

wherein, $W$ represents true positive, $X$ signifies true negative, $Y$ designates false positive, and $Z$ symbolizes false negative. Although accuracy is a widely used metric, it alone may not be sufficient in imbalanced datasets like BoT-IoT, hence the inclusion of TPR and TNR to provide a more complete picture.

**b) TPR**

TPR is a key metric used in attack detection systems to evaluate how effectively a model identifies actual threats and is expressed as,

$$TPR = \frac{W}{W + Z} \tag{12}$$

A higher TPR implies that the system can reliably capture malicious activities, minimizing the risk of undetected intrusions in sensitive WSN deployments.

**c) TNR**

TNR is probability of actual negative instances that are accurately predicted as negative by ResNeXT_TEI and is formulated as,

$$TNR = \frac{X}{X + Y} \tag{13}$$

Maintaining a high TNR is equally critical, as excessive false positives can burden resource-constrained WSN nodes with unnecessary alerts, leading to wasted energy and reduced efficiency.

## 4.4 Experimental Parameters

The experimental parameters for the ResNeXt technique for attack detection is presented in Table 1. These parameters include the learning rate, batch size, optimizer, and number of epochs, which were tuned carefully to achieve a trade-off between detection accuracy and computational efficiency. Such tuning ensures that the model remains suitable for real-world WSN environments, where energy and processing limitations are major constraints.

***Table 1.*** *Experimental parameters of ResNeXt model*

| Parameters | Value |
|---|---|
| Loss | TEI |
| Activation | Sigmoid |
| Strides | 2 |
| Kernel size | (3,3) |
| Dropout | 0.3 |
| Learning rate | 0.01 |
| Epsilon | 0.0001 |
| Optimizer | Adam |
| Beta_1 | 0.9 |
| Batch size | 32 |
| Beta_2 | 0.99 |
| Epoch | 50 |

## 4.5 Ablation Assessment

The efficacy of ResNeXT_TEI is evaluated by using various scenarios, namely ResNeXT+MSLE (without MBE and MAE) + ResNeXT+MAE (without MBE and MSLE) + ResNeXT+MBE (without MSLE and MAE). This ablation design helps in isolating the contribution of each error metric to the overall performance of TEI, thereby validating the effectiveness of the combined loss formulation. The ablation valuation of ResNeXT_TEI for attack detection in WSNs is portrayed in Figure 3. The ablation analysis of ResNeXT_TEI based on accuracy is presented in Figure 3a). At a learning data of 60%, the accuracy measured by the developed ResNeXT_TEI is 90.999%, ResNeXT+MSLE is 86.989%, ResNeXT+MAE is 87.999%, and ResNeXT+MBE is 88.099%. These results confirm that no single error function alone provides optimal performance, and that TEI achieves superior stability by integrating all three.The accuracy of ResNeXT_TEI is boosted by 3.19% higher than ResNeXT+MSLE. In Figure 3b), the ablation valuation of ResNeXT_TEI regarding TPR is described. The TPR gained by providing a learning data of 70% by ResNeXT+MSLE is 90.999%, ResNeXT+MAE is 91.999%, ResNeXT+MBE is 92.999%, and the introduced ResNeXT_TEI is 94.000%.Here, the incremental improvement highlights that TEI not only improves overall accuracy but also strengthens the model's sensitivity in detecting true attacks. The TPR obtained by ResNeXT_TEI is better by 2.25% than ResNeXT+MBE. Figure 3c) reveals the ablation valuation of ResNeXT_TEI concerning TNR. The TNR recorded by the proposed ResNeXT_TEI is 94.099%

ResNeXT+MSLE is 88.999%, ResNeXT+MAE is 89.100%, and ResNeXT+MBE is 91.999% when considering learning data of 80%.This demonstrates that TEI significantly reduces false alarms, which is crucial for energy-limited WSN nodes that cannot afford unnecessary computations. Here, when compared to ResNeXT+MAE, the TNR achieved by ResNeXT_TEI is improved by 4.29%.

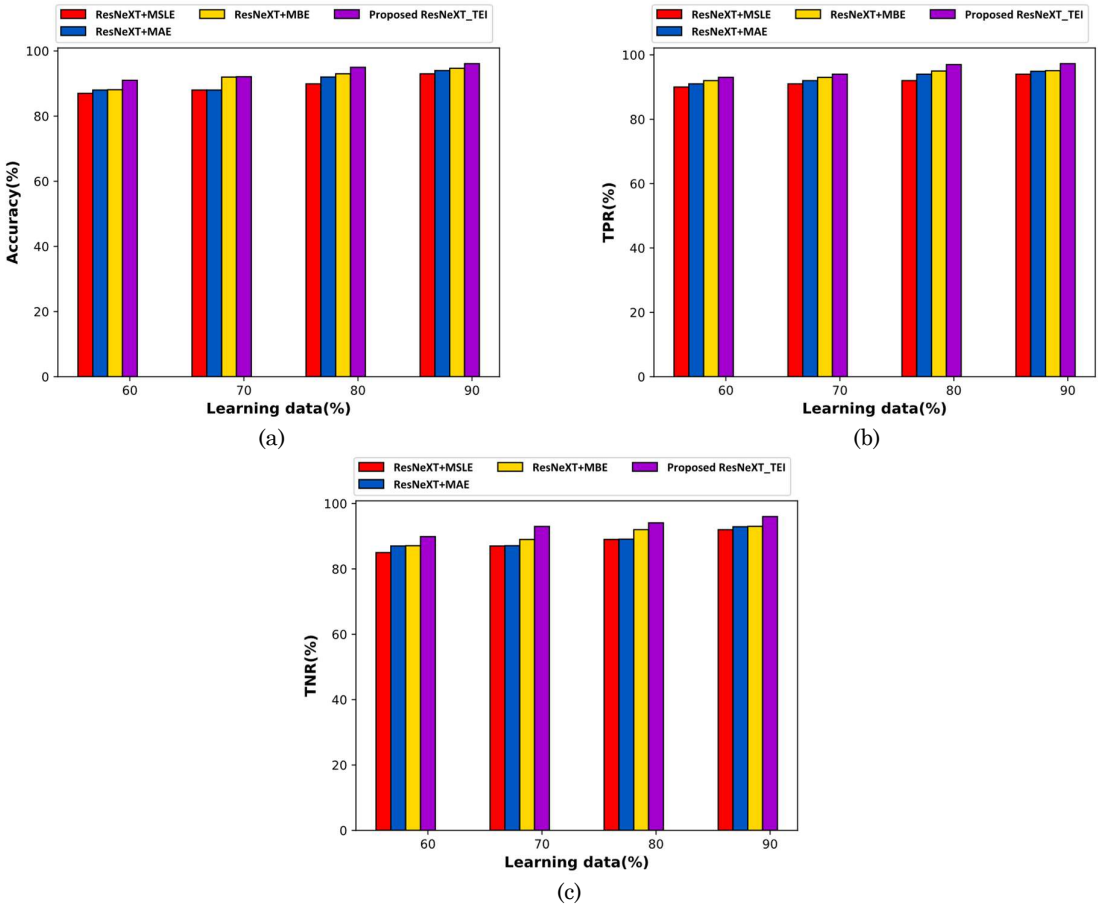


***Fig. 3.*** *Ablation valuation of ResNeXT_TEI for attack detection in WSN with (a) accuracy, (b) TPR, and (c) TNR*

## 4.6 Comparative Techniques

The efficiency of ResNeXT_TEI for attack detection in WSNs is assessed by comparing it with methods, such as MDCNN-Bi-LSTM-AM [12], ECS-WSN-SAPVAGAN-HBA [13], GAN+WOA [10], and SG-IDS [14], and is detailed in this section. These baseline techniques represent a mixture of deep learning, adversarial, and optimization-based strategies, making them suitable benchmarks for validating the superiority of the proposed framework.

## 4.7 K-Value Assessment

Figure 4 displays analysis of proposed ResNeXT_TEI for attack detection in WSNs by altering the k-value. The k-value, often associated with nearest-neighbor-based validation, provides insight into how the model adapts under different classification neighborhood settings. In Figure 4a), the evaluation of ResNeXT_TEI concerning accuracy. Employing a k-value of 5, accuracy achieved by prevailing techniques, such as MDCNN-Bi-LSTM-AM is 82.999%, ECS-WSN-SAPVAGAN-HBA is 84.988%, GAN+WOA is 87.999%, and SG-IDS is 88.888% and established ResNeXT_TEI is 91.888%.This shows that ResNeXT_TEI generalizes better across varying neighborhood sizes, sustaining higher accuracy where traditional models plateau. In comparison, ResNeXT_TEI's accuracy is improved by 4.27% than ECS-WSN-SAPVAGAN-HBA. Figure 4b) shows the assessment of ResNeXT_TEI according to TPR. The TPR attained by the developed ResNeXT_TEI is 94.998% and the existing models, including MDCNN-Bi-LSTM-AM is 88.998%, ECS-WSN-SAPVAGAN-HBA is 89.998%, GAN+WOA is 90.009%, and SG-IDS is 91.888%, when a k-value of 6. This demonstrates that ResNeXT_TEI consistently improves sensitivity, detecting more true attacks while minimizing missed threats. Here, ResNeXT_TEI recorded a 3.27% improvement in TPR compared to GAN+WOA. The ResNeXT_TEI's valuation as per TNR is presented in Figure 4c). On assuming a k-value of 7, the TNR attained by ResNeXT_TEI is 94.988% and the existing models, like MDCNN-Bi-LSTM-AM

is 88.998%, ECS-WSN-SAPVAGAN-HBA is 91.000%, GAN+WOA is 91.999%, and SG-IDS is 92.999%.By outperforming its counterparts in TNR, ResNeXT_TEI shows superior capability in minimizing false positives, a vital requirement for resource-constrained WSN deployments. The ResNeXT_TEI outperforms MDCNN-Bi-LSTM-AM with a 3.22% increase in TNR.

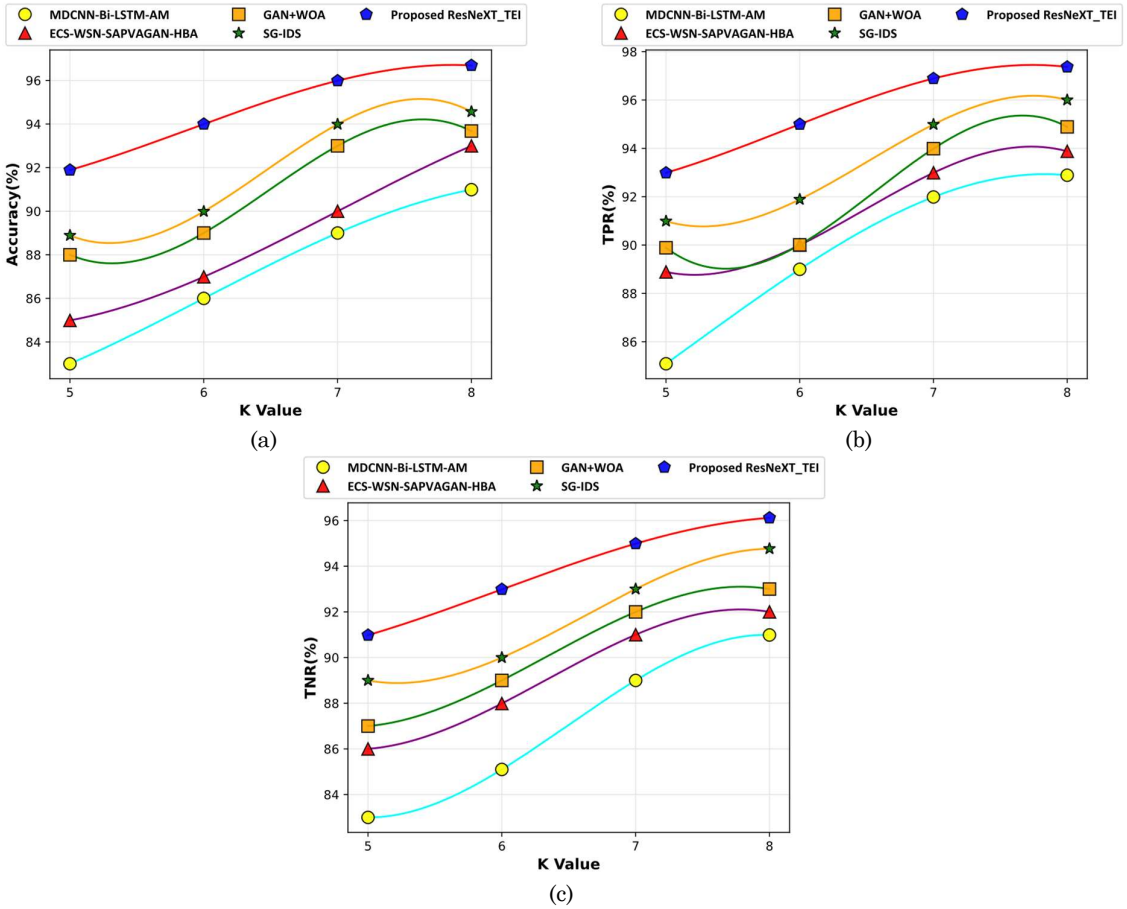


***Fig. 4.*** *Assessment of k-value of ResNeXT_TEI for detecting attacks in WSN with (a) accuracy, (b) TPR, and (c) TNR*

## 4.8 Comparative Discussion

In Table 2, the comparative discussion of ResNeXT_TEI for detecting attacks in WSN based on prevailing models with a k-value of 8 is portrayed. This comparison provides a holistic evaluation of how the proposed framework performs under identical parameter settings, ensuring fairness in benchmarking. The accuracy gained by prevailing models, like MDCNN-Bi-LSTM-AM is 90.988%, ECS-WSN-SAPVAGAN-HBA is 92.998%, GAN+WOA is 93.678%, SG-IDS is 94.566%, and the designed ResNeXT_TEI is 96.690. Further, the TPR measured by ResNeXT_TEI is 97.368%, and existing methods, such as MDCNN-Bi-LSTM-AM is 92.888%, ECS-WSN-SAPVAGAN-HBA is 93.879, GAN+WOA is 94.888%, and SG-IDS is 96.000%.A higher TPR indicates that ResNeXT_TEI can reliably detect more actual attacks compared to baseline methods, minimizing undetected intrusions in critical WSN applications. Also, the TNR recorded by the existing techniques, including MDCNN-Bi-LSTM-AM is 90.988%, ECS-WSN-SAPVAGAN-HBA is 91.999%, GAN+WOA is 92.999%, SG-IDS is 94.767%, and the developed ResNeXT_TEI is 96.118%. Maintaining such a high TNR reflects the robustness of the model in avoiding false alarms, which is essential for conserving energy and preventing alert fatigue in real-world WSN environments. Here, ResNeXt enhances attack detection by leveraging multi-path representation learning, which improves model accuracy, scalability, and feature reuse efficiency. Unlike traditional CNN or RNN-based solutions, ResNeXt's grouped convolutions and cardinality-based transformations enable the model to capture richer patterns in network traffic data. The proposed ResNeXT_TEI attains robustness, enhances classification accuracy, and reduces false positives.

***Table 2.*** *Comparative discussion*

| Metrics | Techniques | | | | |
|---|---|---|---|---|---|
| | MDCNN-Bi-LSTM-AM | ECS-WSN-SAPVAGAN-HBA | GAN+WOA | SG-IDS | Proposed ResNeXT_TEI |
| **Accuracy (%)** | 90.988 | 92.998 | 93.678 | 94.566 | **96.690** |
| **TPR (%)** | 92.888 | 93.879 | 94.888 | 96.000 | **97.368** |
| **TNR (%)** | 90.988 | 91.999 | 92.999 | 94.767 | **96.118** |

## 5. Conclusion

WSNs are composed of distributed, low-power sensing units that facilitate data acquisition and communication, serving critical roles in applications ranging from ecological observation to military and healthcare systems. Because these networks are frequently deployed in hostile or inaccessible environments, ensuring secure communication becomes paramount for mission success. The limited resources, like battery, memory, and processing power, of the nodes make them vulnerable to attacks. Such limitations also restrict the adoption of conventional heavy-weight intrusion detection solutions, underscoring the necessity of lightweight yet highly accurate detection frameworks. Nevertheless, WSN faced difficulties, like unreliable communication, complex patterns, evolving threats, and energy constraints. Therefore, in this work, an effective model, named ResNeXT_TEI, is proposed for attack detection in WSNs. Initially, the WSN is simulated, and the network log data is subjected to median normalization. This preprocessing step ensures consistent scaling across features and minimizes the influence of outliers. Then, feature selection is performed utilizing the Fisher score method. Attack detection is finally achieved through the ResNeXT_TEI model, where TEI is designed by incorporating MSLE, MAE, and MBE. Moreover, the established ResNeXT_TEI achieved a better accuracy, TPR, and TNR of 96.690%,97.368%, and 96.118%. In the future, ResNeXT_TEI develops a lightweight DL model tailored for WSN nodes, enabling efficient real-time intrusion detection directly on-device. Such optimization would allow the framework to operate in edge environments without constant reliance on centralized servers, thereby improving scalability. Further, the efficiency of the detection process can be enhanced by including advanced DL models based on adversarial training and including sophisticated hyperparameter tuning methods.

## Compliance With Ethical Standards

**Conflicts of interest:** Authors declared that they have no conflict of interest.

**Human participants:** The conducted research follows ethical standards and the authors ensured that they have not conducted any studies with human participants or animals.

## References

[1] M. R. Ahmed, S. M. Tahsien, M. Aseeri and M. S. Kaiser, "Malicious attack detection in underwater wireless sensor network," in IEEE International Conference on Telecommunications and Photonics (ICTP), 2015.
[2] G. A. Sukkar and S. Al-Sharaeh, "Enhancing Security in Wireless Sensor Networks: A Machine Learning-based DoS Attack Detection," Engineering, Technology & Applied Science Research, vol. 15, no. 1, pp. 19712-19719, 2025.
[3] K. Ávila, P. Sanmartin, D. Jabba and J. Gómez, "An analytical survey of attack scenario parameters on the techniques of attack mitigation in WSN," Wireless Personal Communications, vol. 122, no. 4, pp. 3687-3718, 2022.
[4] A. John, I. F. B. Isnin, S. H. H. Madni and M. Faheem, "Cluster-based wireless sensor network framework for denial-of-service attack detection based on variable selection ensemble machine learning algorithms," Intelligent Systems with Applications, vol. 22, p. 200381, 2024.
[5] L. S. R. Boddu, S. M. M. A. A. Haldorai, P. S. Reddy, S. Feng and J. Qin, "Node replication attack detection in distributed wireless sensor networks," Wireless Communications and Mobile Computing, no. 1, p. 7252791, 2022.
[6] A. R. Affane M. and H. Satori, "Machine learning attack detection based-on stochastic classifier methods for enhancing of routing security in wireless sensor networks," Ad Hoc Networks, vol. 163, p. 103581, 2024.
[7] X. Huan, K. S. Kim and J. Zhang, "NISA: Node identification and spoofing attack detection based on clock features and radio information for wireless sensor networks," IEEE Transactions on Communications, vol. 69, no. 7, pp. 4691-4703, 2021.
[8] T. S. Delwar, U. Aras, S. Mukhopadhyay, A. Kumar, U. Kshirsagar, Y. Lee, M. Singh and J.-Y. Ryu, "The intersection of machine learning and wireless sensor network security for cyber-attack detection: a detailed analysis," Sensors, vol. 24, no. 19, p. 6377, 2024.

[9] M. R. Ahmed, M. Aseeri, M. S. Kaiser, N. Z. Zenia and Z. I. Chowdhury, "A novel algorithm for malicious attack detection in uwsn," in Electrical Engineering and Information Communication Technology (ICEEICT), 2015.
[10] R. B. Ramesh, S. J. J. Thangaraj, G. M. A. Sagayee and K. Saravanan, "Detection and prevention in WSN security framework using deep learning against black hole and wormhole attacks," Ain Shams Engineering Journal, vol. 16, no. 10, p. 103624, 2025.
[11] M. A. Elsadig, "Detection of denial-of-service attack in wireless sensor networks: A lightweight machine learning approach," IEEE Access, vol. 11, pp. 83537-83552, 2023.
[12] N. Dhanalakshmi, "Unmasking encryption effects and modified Deep learning approaches for attack classification in WSN," Expert Systems with Applications, vol. 266, p. 126163, 2025.
[13] B. Meenakshi, D. Karunkuzhali "Enhancing cyber security in WSN using optimized self-attention-based provisional variational auto-encoder generative adversarial network," Computer Standards & Interfaces, vol. 88, p. 103802, 2024.
[14] H.M. Saleh, H. Marouane A. Fakhfakh, "Stochastic gradient descent intrusions detection for wireless sensor network attack detection system using machine learning," IEEe Access, vol. 12, pp. 3825-3836, 2024.
[15] S. Nandhini, A. Rajeswari, N.R. Shanker. "Cyber attack detection in IOT-WSN devices with threat intelligence using hidden and connected layer based architectures," Journal of Cloud Computing, vol. 13, no. 1, p. 159, 2024.
[16] F. T. Lima and V. M. A. Souza, "A large comparison of normalization methods on time series," Big Data Research, vol. 34, p. 100407, 2023.
[17] M. Rostami, K. Berahmand and S. Forouzandeh, "A novel community detection based genetic algorithm for feature selection," Journal of Big Data, vol. 8, no. 1, p. 2, 2021.
[18] D.P. Yadav, A.S. Jalal, D. Garlapati, K. Hossain, A. Goyal, G. Pant. "Deep learning-based ResNeXt model in phycological studies for future," Algal research, vol. 50, p. 102018, 2020.
[19] A.B. Adomako, E.J. Jamshidi, Y. Yusup, E. Elsebakhi, M.H. Jaafar, M.I. Ishak, H. San Lim, M.I. Ahmad. "Deep learning approaches for bias correction in WRF model outputs for enhanced solar and wind energy estimation: A case study in East and West Malaysia," Ecological Informatics, vol. 84, p. 102898, 2024.
[20] The Bot-IoT dataset," [Online]. Available: https://ieee-dataport.org/documents/bot-iot-dataset. [Accessed August 2025].