SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I11P108

Volume 13 Issue 11, 139-155, November 2026
© 2026 Seventh Sense Research Group®

Original Article

HGBTransNet: A Hybrid Deep Learning and Transformer based Framework for Intrusion Detection in WSN

Veggalam Shivaraj Kumar[1], K Renu[2], Ch Venkat Rao[3]

[1,2]*Department of EECE, GITAM Deemed to be University, Visakhapatnam, Andhra Pradesh, India.*
[3]*Department of ECE, Sanketika Vidya Parishad Engineering College, Andhra Pradesh, India.*

[1]*Corresponding Author: shiva.raj437@gmail.com*

Received: 23 July 2026 Revised: 16 October 2026 Accepted: 31 October 2026 Published: 27 November 2026

Abstract *- Wireless Sensor Networks (WSNs) are composed of various sensor nodes which receive and send environmental data. There are various applications of WSNs to real-time application in the real world, such as environment, healthcare, and surveillance in the military. They are, however, very vulnerable to various attacks due to their distributed and resource constrained nature. As a solution to these points, in this work, Hybrid GRU–BiLSTM–Transformer Network (HGBTransNet) model an intelligent Deep-Learning (DL) model, has been proposed to detect intrusions in WSNs with high efficiency using Convolutional-Neural Networks (CNN), Transformer Encoder, Gated Recurrent Unit (GRU) and Bi-Long Short-Term Memory (Bi-LSTM) networks. The CNN part derives spatial correlations of network traffic data, and the Bi-LSTM layer derives temporal dependencies to increase the accuracy of classification. The proposed method is tested on five publicly available datasets, comprising normal and attack traffic logs. The experimental observations show that HGBTransNet model has a higher detection accuracy of 99.8%, 98.20%, 99.99%,97.20% and 99.70% on WSN-DS, WSN-BFSF, CICIDS-2017, UNSWNB15 and NSL-KDD datasets, respectively and proves that proposed model is efficient in detecting various attacks compared with state-of-the-art methods.*

Keywords *- Bi-Long Short Term-Memory (Bi-LSTM), Gated-Recurrent Unit (GRU), Intrusion Detection System (IDS), Transformer encoder, WSNs.*

1. Introduction

WSNs consist of sensor nodes which are spatially distributed and that are used to sense physical environmental conditions and coordinate the sharing of data using wireless networks. The networks are being more widely applied in areas like military surveillance, medical systems, and environmental monitoring, and in most cases remote sensing features of infrastructure that is critically important.

Nonetheless, the resource-constrained essence of WSNs, the limited memory, power supply and computation speed, as well as the fact that they are usually unattended and deployed as ad-hoc, pre-condition the vulnerability to a wide range of cyber-attacks and intrusions.

Provision of security and reliability of WSN is thus one of the central research problems. WSN-specific Intrusion Detection Systems (IDS) may offer a crucial second line of defense: they observe network operations, detect when things are not behaving as expected, and generate alarms when malicious movement is being exhibited.

CNNs have been utilized to extract spatial information of raw data, whereas GRUs and, specifically, LSTM networks are the most skilled at extracting time-related information in sequential data. The studies that use CNN-BiLSTM hybrids in intrusion detection have indicated that they have a higher detection rate due to the ability to capture spatial and temporal features of patterns [1].

Continuing on these developments, the current work presents HGBTransNet model, a smart DL system that combines CNN, GRU, Transformer and Bi-LSTM networks in recognizing various cyberattacks in WSNs. The proposed architecture is specifically designed to make use of the spatial and temporal nature of network traffic, which allows making more precise discrimination between normal and malicious behavior. The performance of the model is tested using the WSN-DS dataset [2]. The achieved results verify that HGBTransNet model has high detection accuracy and low False Positive Rate (FPR) in wireless sensor space and has high adaptability and high robustness to different types of attacks.

 This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

The framework of this paper has been as follows: Section 2 provides Literature Survey and overview of the current research; Section 3 outlines the methodology and proposed HGBTransNet model architecture; Section 4 provides the results of the experiments and discussion; finally, Section 5 provides the Conclusion of the work and gives recommendations on what further research should do.

2. Background and Related Work

WSN intrusion detection has been treated in numerous ways, starting with traditional anomaly rule-based systems and Machine Learning and Deep Learning (ML-DL) systems. According to recent studies, ML-DL applications lead to higher detection rates than conventional based methods, although they also have more real-world limitations, including computing limitations, data bias, energy on sensor nodes, and minor assessments in practical applications. These challenges motivate research into novel architectures that combine strong representational power with efficiency and robustness.

Pinto et al. [3] developed ML-DL IDS for critical infrastructures and identifies realistic issues such as the need for benchmark datasets, management concept drift, and deployment constraints on edge devices. Altunay et al. [4] propose multiple deep models, including a hybrid CNN+LSTM that first extracts local spatial features via 1-D/2-D convolutional layers and then models temporal dependencies using LSTM. The paper reports high detection rates on industrial IoT benchmark datasets and shows that the hybrid design improves recall and F1 relative to standalone CNN or LSTM models.

However, the work is evaluated primarily on IIoT datasets and does not address WSN-specific resource constraints, limiting direct applicability to sensor networks with severe energy limits. Singh et al. [5] developed a CNN–LSTM IDS that combines filter-based and wrapper feature selection with SMOTE oversampling and a CNN–LSTM classifier.

Reported results on benchmark network datasets show substantial improvements in F1 and recall over single-model baselines. The study focuses on network traffic rather than WSN-specific constraints, and the architecture is not evaluated for on-device feasibility.

Talukder et al. [6] MLSTL-WSN pipeline, a WSN-tailored ML/DL solution that uses SMOTE-Tomlink for imbalance handling and evaluates multiple classifiers on the WSN-DS dataset. The authors report multiclass accuracies above 99% in offline experiments and highlight feature-selection benefits.

However, the reported numbers likely reflect strong dataset fit. Hussain et al [7]. design a CNN–LSTM IDS that integrates statistical pre-filters and deep feature extractors; experimental evaluation on standard IDS benchmarks shows competitive Area Under the Curve (AUC) and F1 values versus classical ML baselines. However, the evaluation primarily uses well-known datasets rather than WSN-specific traffic, and the model size may hinder direct deployment on sensor nodes.

Sasi et al. [8] introduced a CNN–LSTM network for IoT attack detection that embeds an attention module after CNN-LSTM blocks to weight informative features. On multiple IoT datasets, the model attains very high reported accuracies. However, the extreme accuracy is obtained in offline, preprocessed conditions; attention modules increase parameter count, which can be problematic for constrained WSN devices unless model compression is applied.

Multiple recent studies incorporate attention into CNN–LSTM hybrids and report consistent AUC/F1 Score improvements. These studies demonstrate that attention can both boost detection and facilitate interpretability, but they also emphasize the need for quantization for edge deployment. Recent works include attention-enhanced hybrids that achieve near-perfect AUC on IoT corpora but leave deployment and energy-cost analysis for future work [9].

Anwar et al. [10] showed that a federated LSTM approach can maintain high detection accuracy across heterogeneous IoT clients, while Albanbay et al. [11] evaluate balancing between local model complexity and communication cost. The major limitation of this model is that federated aggregation introduces communication overhead, staleness and non-IID data problems that can lower convergence in resource-constrained WSNs.

Pandey et al. [12] proposed that combine metaheuristic optimization with Random Forest algorithms improve hyper parameter tuning and feature selection for IDS tasks, often yielding measurable gains in accuracy and robustness. However, optimization steps add offline computational cost and do not directly reduce inference cost on sensor nodes. Several studies apply ensemble learning to improve detection, particularly for minority attack classes.

The results show better recall on hard-to-detect classes, though it may demand centralized data collection and computation on collaborative voting. Federated ensemble techniques strive to decentralize training but increase synchronization and communication costs. These trade-offs emphasize the importance of hybrid models that are accurate and compressible [13].

Zhenyue Long et al. [14] proposed a Network-IDS Transformer model and tested it with CICIDS2018 dataset for cloud computing environments. The model successfully classified the complex pattern of network traffic with an accuracy of 93.00%, showing the efficiency of the Transformer model in capturing complex network traffic

patterns. This framework, however, was tailored towards cloud infrastructures and demands substantial computing resources. Therefore, it is not suitable for the application of resource-limited WSNs. Ajayan, A. et al. [15] presented DistilXIDS, a lightweight DistilBERT-based Transformer model for real-time network IDS. The model was validated on different datasets with an accuracy of around 99.00% with a lesser computational complexity than traditional Transformer models. The lightweight design does not mean that the framework is not demanding in terms of computational resources. However, this prevents its use on resource-constrained WSN nodes.

Kamal and Mashaly [16] suggested a Hybrid Transformer-CNN model to detect intrusions in networks and tested the projected approach on a CICIDS2017 dataset. The efficiency of the model was 99.13% correct, which demonstrates how the Transformer model can capture global dependencies and how the CNN can extract local features. Although it has been demonstrated to have strong detection capability, it is not appropriate for WSNs applications, which have limited resources, due to the need for large training datasets and training time.

Talukder et al. [17] discussed a hybrid IDS that uses KMeans-SMOTE and PCA to enhance the detection of attacks against WSN. The model was tested with WSN-DS and TON-IoT datasets, with an accuracy of 99.66% that managed to overcome the class imbalance and feature dimensionality. In fact, the framework has several pre-processing stages, leading to a higher execution time and computational overhead. This could restrict its usage in real time and resource-constrained WSNs.

Fares H et al. [18] developed an adaptive ML-IDS in WSNs and tested the efficiency of the projected system on the WSNDS data-set with accuracy of 99.00%, which indicated that it could be used to detect attacks in the system well and has a strong classification ability. The framework, however, is limited to traditional ML methods and does not leverage attention-based or Transformer models to learn complex feature relationships and changing attack patterns.

The studies available show significant advancement of IDS of WSNs using DL and optimization methods. However, persistent remaining challenges are associated with generalization, resource efficiency, and interpretability and continue to limit real-world deployment. In order to solve these issues, the proposed HGBTransNet model introduces a CNN-BiLSTM-GRU-Transformer model to be used for efficient and precise attack detection in real-time conditions.

Research on IDS in WSNs has been crucial, with several methods developed to improve network security. These techniques can be mainly divided into three, which are hybrid, anomaly and signature-based methods. The conventional techniques have better results; however, they have drawbacks to which DL techniques seek remedies. Table 1 illustrates the different existing models along with their key limitations, highlighting the gaps addressed in that work.

The existing methods exhibit several limitations, such as high FPR, low detection rate, generalization to a variety of attacks and higher computational time. Moreover, most existing architecture validate their models on a limited number of datasets.

To address these challenges, the proposed HGBTransNet architecture improves discriminative illustration learning and classification robustness, thereby achieving enhanced accuracy and lower FPR across diverse datasets. The final aim is to develop DL-based IDS that can identify different types of attacks.

3. Materials and Methods

The proposed HGBTransNet architecture has a systematic approach that is done by first preparing and refining data, after which feature normalization and encoding are performed so that it can be compatible with DL operations. HGBTransNet architecture is developed and trained to capture both spatial and time traffic peculiarities of wireless data.

The aim of the HGBTransNet model is to design and test HGBTransNet for achieving robust intrusion detection in various WSN environments using five datasets with 26 attack types. The multi-dataset evaluation aims to evaluate the model's generalization performance on a variety of attack scenarios and network conditions to overcome the limited number of datasets and attack categories covered in state-of-the-art methods.

The performance of HGBTransNet was tested with the help of five datasets, i.e. WSN-DS, CICIDS-2017 [34], WSN-BFSF [35], UNSWNB15 [36] and NSL-KDD [37], used in the research of IDS in WSN.

Most of the current works on WSN intrusion detection focus on testing models over restricted sets of attacks and datasets, thus limiting the ability to assess their generalization to the various traffic distributions. HGBTransNet fills this gap by testing it on five benchmark datasets from various attack categories.

Existing architectures may be insufficient to model the local traffic patterns, sequential dependencies and long-range feature relationships all at once; thus, HGBTransNet features 1D-CNN, GRU, BiLSTM, and Transformer encoder layers.

Comparative evaluation against other approaches can give insights into the relative detection performance of the model over other datasets. In addition, the analysis reveals the categories of attacks for which classification is still low.

Table 1. Shows the various existing methods and their limitations

S.No	Author(s)	Method / Technique	Dataset Used	Performance metric	Limitations
1	Kasongo and Sun [19]	ANN-XGBoost	UNSWNB15	Accuracy:90.85%	Restricted to only ONE dataset
2	Halbouni et al. [20]	CNN-LSTM	UNSWNB15	Accuracy:93.78%	Low detection rate and high FAR
			WSN-DS	Accuracy:99.58%	
			CICIDS-2017	Accuracy:99.56%	
3	Salmi and Oughdir [21]	CNN + RNN	WSN-DS	Accuracy:97.54%	Low Accuracy Limited to only ONE Dataset
4	H.-Y. Kwon et al. [22]	Hybrid- IDS	NSL-KDD	Accuracy: 96.0%, Recall: 95.3%, Precision: 95.0%, FAP: 3.3%, F1: 95.1%	Complexity & Low FPR
5	Gowdhaman and Dhanapal [23]	DNN	WSN-DS	Accuracy: 95.53	High computational complexity. Low classification accuracy
6	M.A. Elsadig et al. [24]	RF, XGBoost, KNN	WSN-DS	Accuracy: 97.5%; Precision: 93.8; FAR: 0.05	Low overhead; lighter than ensemble models; real-world deployment not tested
7	Reza et al. [25]	DL	WSN-BFSF	Accuracy: 98%	Limited to only WSN-BFSF Dataset
8	Deshmukh, A. [26]	CNN	UNSWNB15	Accuracy:95.47%	Intrusion detection performance
9	Behiry and Aly [27]	KMC-IG	UNSWNB15	Accuracy: 98%	Binary Class classification
			CICIDS-2017	Accuracy:99.98%	
10	Devi et al. [28]	DCGAN	WSN-BFSF	Accuracy: 94%	Low Accuracy and AUC
11	Al-Khatib et al. [29]	Improved BiLSTM	WSN-DS	Accuracy: 99%	The detection needs to be robust against various types of attacks,
12	Baniata et al. [30]	CNN + GCN + BiLSTM	WSN-DS	Accuracy:98.00%	In future adaptive federated implementations for decentralized WSN nodes,
13	Talukder et al. [6]	MLSMOTETomek	WSN-DS	Accuracy:99.78%	No Transformer-based feature extraction.
14	Ramana et al. [31]	Whale Optimization + GRU (WOGRU)	WSN-DS	Accuracy: 99.8%	High computational overhead and limited validation on zero-day attacks.
15	Stein et al. [32]	Transformer-based Payload IDS	UNSWNB15	Accuracy: 79% -Binary 72% -Multi-class	Lower multiclass accuracy;
16	C. Xi et al. [33]	IDS-MTran (Multi-Transformer)	NSL-KDD, CICDDoS2019, UNSWNB15	Accuracy: >99%	High computational complexity; not validated on WSN datasets.

3.1. Data Preprocessing

Data preprocessing enhances the quality of data and facilitates the effective model learning process. The preprocessing pipeline involves various steps such as feature-extraction, data-cleaning, feature-normalization, label encoding, and one-hot encoding, which process the raw network traffic data into a format fit for DL.

The data set is first split into the feature matrix to be used as input and the labels themselves. Assume N number of samples with d input features exist in the dataset. The data set (D) is represented as in equation 1.

$$D = \{(x_i, y_i)\}_{i=1}^{N} \quad (1)$$

Where $\in R^d$, $y \in \{1,2,3,\ldots. C\}$
X: Feature matrix and y: Class labels.

Real-world network traffic data may include invalid values or infinite values, so a few of the invalid values are replaced with missing values NaN, and the data that contain infinite values are eliminated, so that the model training process will not have numerical instability. The process can be denoted by the following equation 2.

$$x_{ij} = \begin{cases} 0, if\ x_{ij} = NaN \\ 0, if\ x_{ij} = \pm\infty \\ x_{ij}, Otherwise \end{cases} \quad (2)$$

Once cleaned, the numerical attributes are Z-score normalized so that they are not on different scales for different network attributes. Standardization makes all features have a mean of zero and a variance of one, enhancing optimization efficiency and convergence speed of the model. Each numerical feature is standardized using equation 3.

$$z = \frac{x\ -\mu}{\sigma} \quad (3)$$

Then the categorical attack labels are converted into numerical values through Label Encoding, to be efficiently processed by the DL model. The encoding function is defined as it is in equation 4.

$$f: y \to \{0,1,2,3\ldots, C-1\} \quad (4)$$

C denotes the number of attack classes.

Lastly, one-hot vectors for multi-class classification are attained by equation 5 to the encoded labels.

$$y_{ik} = \begin{cases} 1, k = y_i^{enc} \\ 0, Otherwise \end{cases} \quad (5)$$

Where $y_i^{enc} = f(y_i)$

This pre-processing approach is an effective method for discarding invalid and noisy data, normalizing data distribution and transforming the labels to machine-readable format. Thus, it can boost numerical stability, speed up training convergence and boost the overall detection performance of the HGBTransNet model for various datasets.

3.2. HGBTransNet Architecture

The proposed IDS adopts several benchmark datasets as the main set of network traffic data. First, the dataset is acquired and then data preprocessing takes place. After preprocessing, the dataset is split into 80:20, with 80% being used for training and 20% for testing. This data-splitting method guarantees adequate learning samples and an unbiased evaluation set. The further process of feature selection, model training and attack classification follows on to the data after it has been processed and partitioned.

The proposed HGBTransNet is a combination of convolutional, recurrent, and attention-based DL models to effectively extract hierarchical and temporal features from network traffic data for intrusion detection. While the Convolution Layer-64 extracts initial features at local level by learning discriminative spatial patterns from the input data, Batch Normalization normalizes feature distributions to accelerate convergence and enhance training stability.

These normalized features are then passed to a GRU-64 Layer to effectively capture short-term sequential dependencies with minimal computational complexity and a Bi-LSTM-32 Layer to learn contextual information from both forward and backward directions, thus modeling long-range temporal dependencies. Then, the sequential representations are followed by the enriched sequential representations to a Transformer Block with head size 32, which introduces a self-attention mechanism that enables it to capture the global relationships between features and enrich feature learning in context.

A Global-Average Pooling (GAP) Layer is used thereafter to combine the learned feature maps into smaller representations and decrease the number of trainable parameters. The outputs of this are transferred to a Dense Layer-256 for higher level nonlinear feature transformations and final decision-making.

A Dropout Layer randomly turns off neurons during the training process to make the model more robust and to avoid overfitting. Through the incorporation of CNN, GRU, Bi-LSTM, and Transformer modules, the proposed architecture can effectively leverage the information extracted from both spatial and sequential contexts as well as global contexts, thereby improving the accuracy of intrusion detection and performance in different network attack scenarios.

Table 2. Layers used in the proposed HGBTransNet architecture

Layer No.	Layer Name	Purpose
1	Input Layer	Accepts normalized network traffic features
2	CNN layer-32,64	Local feature extraction
3	GRU-64 Layer	Learns short-term temporal dependencies
4	Bi-LSTM-32 Layer	Learns bidirectional sequential features
5	Self-Attention: Transformer Layer	Captures global contextual relationships
6	Batch Normalization	Residual connection and feature normalization
7	Pooling Layer	Converts feature maps into a feature vector
8	Dense Layer-256,128,64	Learns high-level discriminative features
9	Dropout Layer	Prevents overfitting
10	Output -SoftMax Layer	Performs multi-class intrusion classification

Once the models are trained, the HGBTransNet is preserved and then tested on the unseen test set to evaluate model performance. The test phase determines model accuracy in the real environment at which to classify the normal traffic and attack traffic instances. A thorough Result and Evaluation phase is performed to analyze the prediction obtained, in which various performance metrics are calculated. These metrics include Accuracy, Precision, Recall, and F1-Score. Table 2 presents the layers used in the HGBTransNet architecture.

Furthermore, the Area Under the Receiver Operating Characteristic Curves (AUC-ROC) are plotted to investigate the model's discrimination power at varying classification levels,/ and the Training and Testing Curves are analyzed to track the learning behavior, convergence properties and overfitting issues of the model. These evaluation measures allow for a thorough examination of the proposed IDS effectiveness, robustness and generalization capabilities. The overall workflow of intrusion detection in the proposed DL model will be explained in the algorithm, where the stages of data processing, model development, and evaluation are represented in sequence.

Algorithm: HGBTransNet-based WSN-IDS
Input:
WSN Dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$
Output:
Trained classification model(M), Performance evaluation metrics, Confusion Matrix, predicted labels ỹ.

Step 1: ***Data Loading and Preprocessing***
Load dataset (D) from CSV file.
Separate features (X) and labels (y).
Handle missing and infinite values in (X) by replacing/removing them.
Encode categorical labels into integers if necessary.
Normalize features using Z-score standardization.

Step 2: ***Dataset Partitioning***
Split (X) and (y) into training and testing sets: (X_{train}, X_{test}), (y_{train}, y_{test}). Train & Test Split (X, Y, ratio = 0.8) with stratified sampling.
Convert labels to one-hot encoding for multi-class classification.
Compute class weights (w_c) to handle class imbalance.

Step 3: ***Transformer Encoder Definition***
For input tensor X= [x_1, x_2, x_3…x_t] € R^{t*d}
Apply Multi-Head Self-Attention (MHSA).
Add residual connection and layer normalization.
Apply Feed-Forward Network (FFN) with activation and dropout.
Add second residual connection and layer normalization.
Output: encoded representation of input sequence

Step 4: ***Model Architecture***
Input layer: x € R^{m*1}.
Convolutional feature extraction:
Apply 1D convolution-64 + Batch Normalization + Activation (ReLU) and repeat for CNN layers 32.
Apply GRU-64 and Bi-LSTM Layer-32
Transformer blocks: Apply one or more transformer encoder layers with head size -64 on CNN output.
Global Average Pooling to reduce sequence dimension.
Dense layers: 256, 128, and 64 with Batch Normalization and Dropout-0.5.
Output layer: Dense with Softmax activation for (C) classes.

Step 5: ***Model Compilation***
Optimizer: AdamW (learning rate = 10^{-3}), *weight-decay*=1e-5
Loss function: Categorical Cross-Entropy.
Performance metric: Accuracy.

Step 6: ***Model Training***
Train model on (X_{train}, y_{train}) with a validation split.
Early stopping based on Validation loss.
Reduce learning rate on plateau.
Class weights: Epoch 100, batch size: 32.

Step 7: ***Model Evaluation***
Predict probabilities on (Xtest).
Convert predicted probabilities to class labels.
Compute evaluation metrics.

Step 8: ***Output***

Trained model (M).
Classification metrics.
AUC-ROC plot.
Confusion Matrix plot.

3.3. Training Methodology

The proposed HGBTransNet training was performed with the support of the TensorFlow-Keras DL environment to promote the efficiency of the computation and reproducibility. The model was assembling using AdamW optimizer, as it was chosen due to its adaptive learning rate capability that makes it converge more quickly.

The learning rate was established at 0.001, which was an experimental parameter to ensure that there is a trade-off between convergence rates and stability of the model. The cross-entropy loss function was also used as categorical loss function since it is suitable for a multi-class classification problem.

The training was carried out on 100 epochs, with a batch size was 32, and this was discovered to be a reasonable trade-off between training time and generalization. At each epoch of training, validation accuracy and loss were tracked to guarantee consistent convergence as well as to determine whether any overfitting is occurring. The introduction of Batch Normalization layers in all the model layers allowed stabilizing the learning process by minimizing the internal covariate shift, whereas the introduction of Dropout discouraged co-adaptation among neurons and enhanced the overall performance of generalization.

In order to further ascertain reproducibility and strength, an artificial seed was applied when splitting and initializing data, which means that the experimental setup did not vary across runs. The forward propagation was used in the training process to learn features, backpropagation was used to update weights, and the AdamW algorithm was used to optimize the gradient. The computing experiments were implemented in a GPU-enabled environment on Google Colaboratory (Google COLAB).

The proposed HGBTransNet was coded on the cloud environment of Google COLAB using Python 3. They ran the tests on a virtual machine with an NVIDIA Tesla-T4 GPU with 16 GB GDDR6 VRAM and CUDA acceleration, Intel Xeon CPU at 2.2 GHz and around 12.7 GB of RAM. The model was built with the TensorFlow/Keras DL framework. The COLAB environment with GPU support greatly shortened the training time and facilitated the effective implementation of the HGBTransNet DL architecture on various datasets. Figure 1 below demonstrates the architecture of the HGBTransNet.

During the training period, validation and training accuracy increased gradually, demonstrating the successful acquisition of the model of complicated spatiotemporal correlations in WSN traffic. The trained model that was obtained was found to be highly accurate and had low FPR in all the types of attacks. These results confirm the effectiveness of the HGBTransNet and its applicability as a practical architecture to the intrusion detection of WSNs. Table 3 shows the training parameters of the HGBTransNet Architecture.

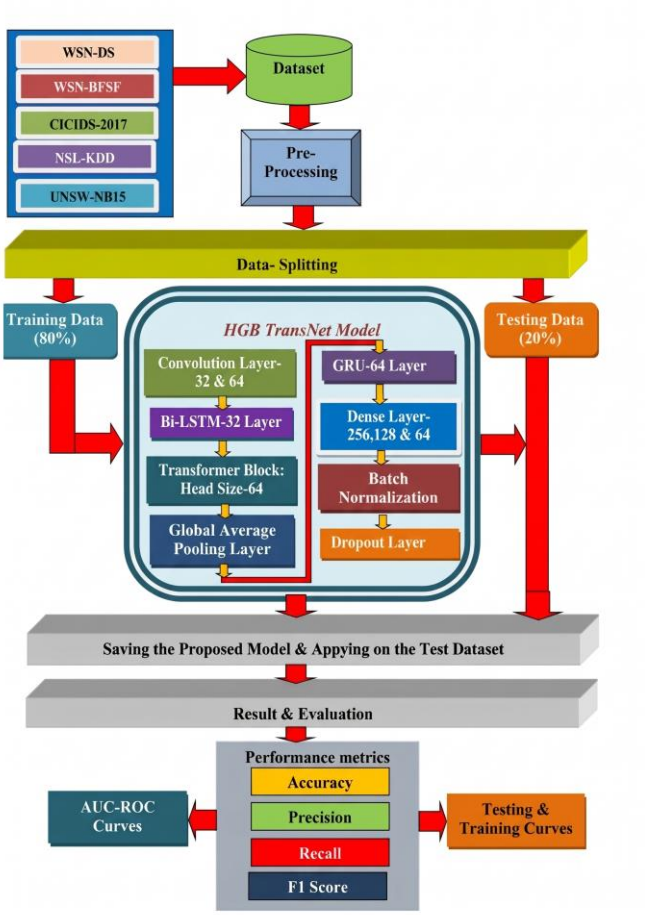


Fig. 1 Architecture of HGBTransNet model

Table 3. Model training parameters of HGBTransNet model

Parameter	Value / Setting
Framework	Tensor Flow–Keras
Optimizer	AdamW Optimizer
Learning-Rate	0.001
Loss-Function	Categorical Cross-Entropy
Batch-Size	32
Epochs	100
Dropout Rate	0.5
Batch Normalization	Applied after each Conv1D, Bi-LSTM & Dense
Activation Functions	ReLU, SoftMax

Layers	GRU, Transformer and Bi-LSTM
Training–Testing Split	80% Training, 20% Testing
Random State	42
Software Platform	Google COLAB

The HGBTransNet is a combination of 1D-CNN, GRU with 64 units, BiLSTM with 32 units, Transformer Encoder, and three Dense layers with 256, 128 and 64 neurons. The GRU model captures sequential dependencies with lesser computation, and the 1D-CNN model captures local and discriminative patterns from the input features. The BiLSTM also captures bidirectional dependencies and complementary temporal representations. The Transformer Encoder uses attention mechanisms to identify the important relationships between the learned features. Lastly, the extracted representations are gradually modified by the three Dense layers to classify normal and attack traffic effectively.

4. Results and Discussion

HGBTransNet analysis was done on Accuracy of classifications, Precision, Recall, F1-score, FPR, and AUC-ROC. All these performance metrics confirm that the model is able to effectively differentiate between normal and malicious WSNs behaviour. A testing accuracy was attained using the trained HGBTransNet model on datasets, which means that model is very reliable in its classification. The performance measure is evaluated using the equations below.

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

$$FPR = \frac{FP}{(FP + TN)}$$

4.1. Performance Evaluation Metrics

The proposed IDS model had shown high robust classification performance on the WSN-DS dataset, which was tested with each of the attack classes. The model achieved excellent classification reliability, with precision oscillating from 0.90 to 1.00, and recall from 0.93 to 1.00, while F1 score ranged from 0.95 to 1.00, showing good predictive performance. Furthermore, the classification accuracy was between 99.59% and 99.91%, and the FPR was extremely low, which resulted in a very low number of false alarms. The AUC-ROC values of 1.00 for Blackhole, Flooding, Gray-hole and Normal classes, and 0.98 for the TDMA class, also validate the model is better discriminative ability.

HGBTransNet achieved very good performance in multiclass classification on NSL-KDD dataset. High precision, recall and F1-scores spanning from 0.808 to 0.998, 0.769 to 0.999 and 0.824 to 0.998, respectively, for classification capability of the model. The classification accuracy ranged from 99.01% to 99.94%, and the FPR was extremely low in the range of 0.0003 – 0.0143, meaning that the number of false alerts generated was very low. Moreover, the AUC-ROC curves for all classes have an excellent value of 1.00, which shows the excellent discriminative power and robustness of the model in correctly distinguishing legitimate network traffic from the different kinds of intrusion, even though the relatively low value of recall for R2L class suggests that this class is the most difficult to detect because of the rarity and subtlety of such an attack pattern.

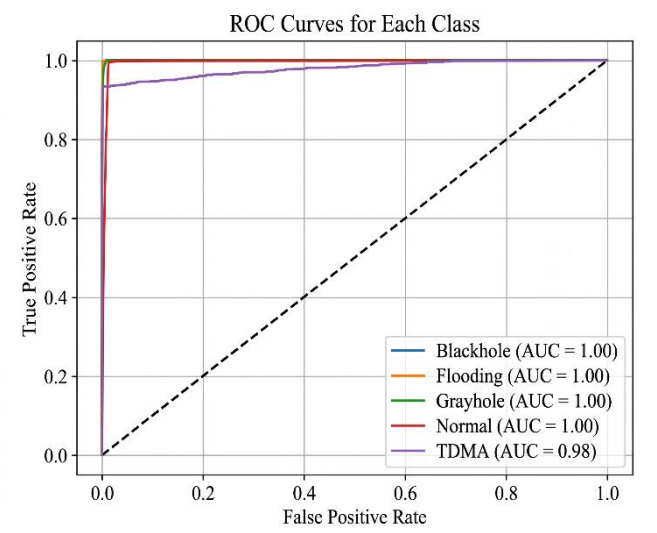
The HGBTransNet model showed excellent multiclass classification results for the WSN-BFSF dataset with respect to Blackhole, Flooding, Forwarding, and Normal traffic classes. The maximum precision value is 0.97, the Recall values range from 0.50 to 0.99, and the F1-scores range from 0.60 to 0.98, showing good performance across most of the attack categories, and the recognition was found to be more challenging for Blackhole and Forwarding. The classification accuracy ranges from 96.26% to 99.52%, with an FPR between 0.0039 and 0.1550, indicating generally low rates of false alarms except for the Normal class.

Furthermore, AUC-ROC scores within the range of 0.96–1.00 indicate that the model has a high discriminative power. The proposed IDS has different classification accuracy in the multiclass setting for the UNSWNB15 dataset. The classification accuracy fell between 89.21% and 99.93%, and the FPR was between 0.0000 and 0.1084, indicating a low number of false alarms for most classes, while the misclassification rate was relatively high for the Exploits class. Moreover, the AUC-ROC values between 0.94 and 1.00 show that the model has good discriminative power.

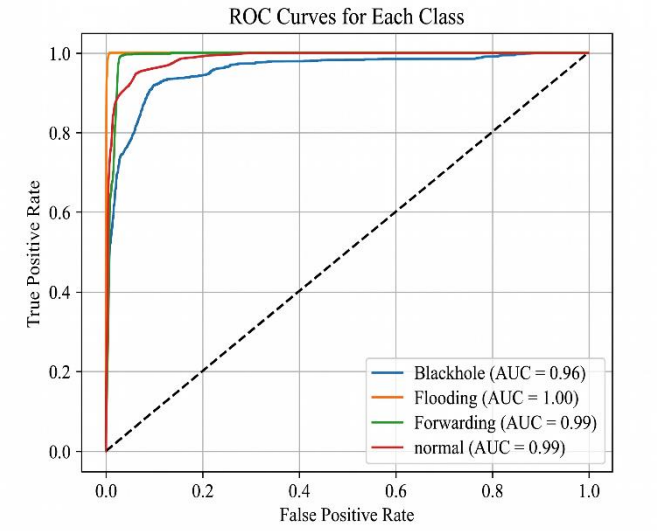
It is detected that the HGBTransNet model successfully outperforms the near-perfect binary classification performance on the CICIDS2017 dataset in the case of BENIGN and DDoS traffic class. The precision, recall and F1-Scores of the model reached 1.00 for both classes, showing exceptional performance for detection accuracy and balanced classification capability. The following Table 4 is a summary of the performance of each category of attacks.

While HGBTransNet has a high AUC-ROC on some attack classes, this does not necessarily mean it is able to detect the minority class effectively or that there is no overfitting. Hence, precision, recall and F1-score are also taken as measures to assess the detection performance for each class, especially for minority attack classes.

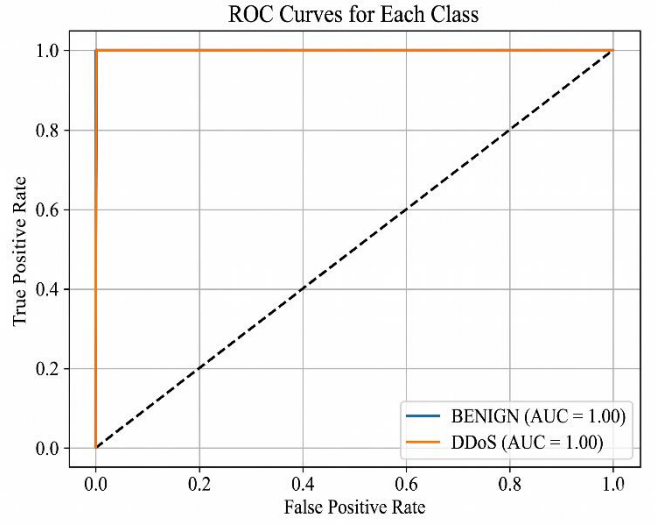
Low recall or F1-scores suggest that there are problems in recognizing certain attacks, even if the AUC-ROC value is good. To check for overfitting, training and validation curves are explored to gain insight into convergence and generalization. Moreover, the normal class one-versus-rest FPR refers to the percentage of attack samples misclassified as Normal.



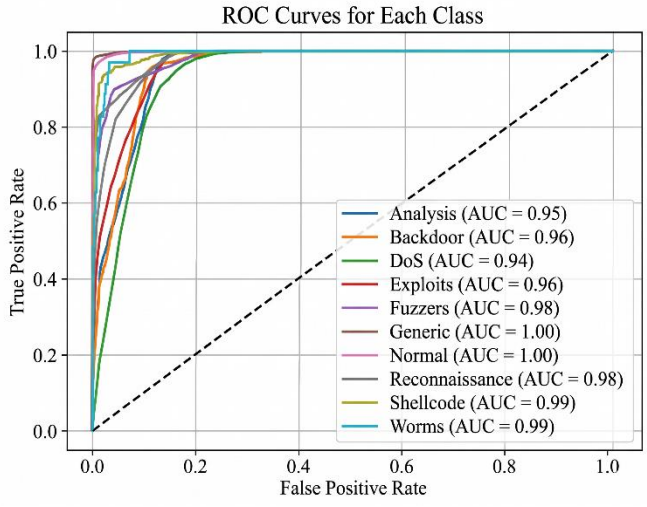
(a)



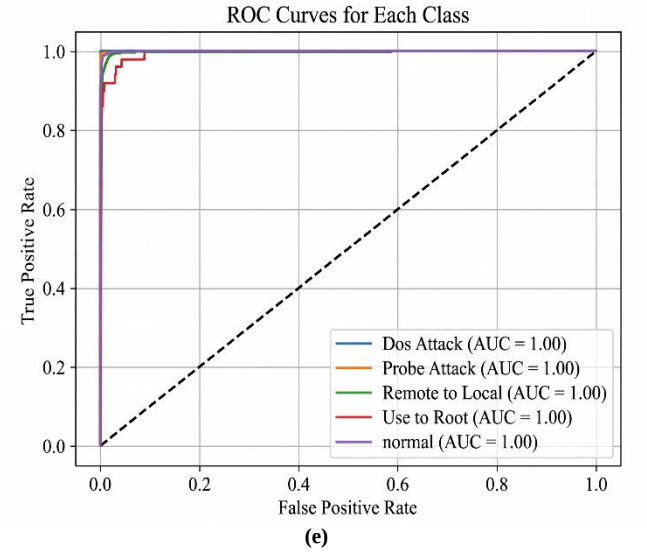
(b)



(c)



(d)



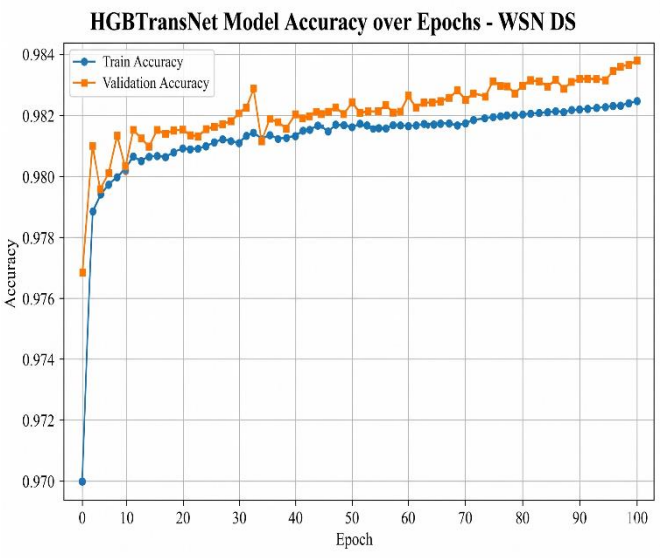
(e)

Fig. 2 ROC curves for attack detection across different datasets: (a) WSN-DS, (b) WSN-BFSF, (c) CICIDS-2017, (d) UNSWNB15, and (e) NSL-KDD datasets.

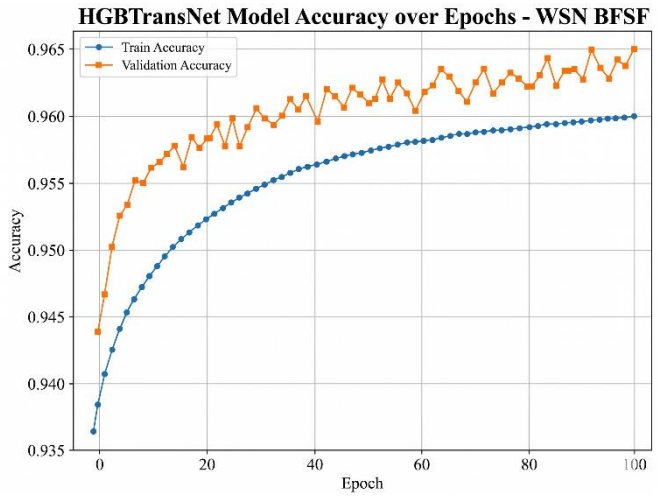
Also, accuracy of the classification was 99.97%, and FPR of the BENIGN class was 0.0002, and the DDoS class was 0.0004, meaning that the number of false alarms generated were extremely low.

Moreover, the values of 1.00 for both classes on the AUC-ROC confirm that the model works at a very high discriminative level and can make an excellent distinction between legitimate network traffic and DDoS attacks with near-perfect confidence.

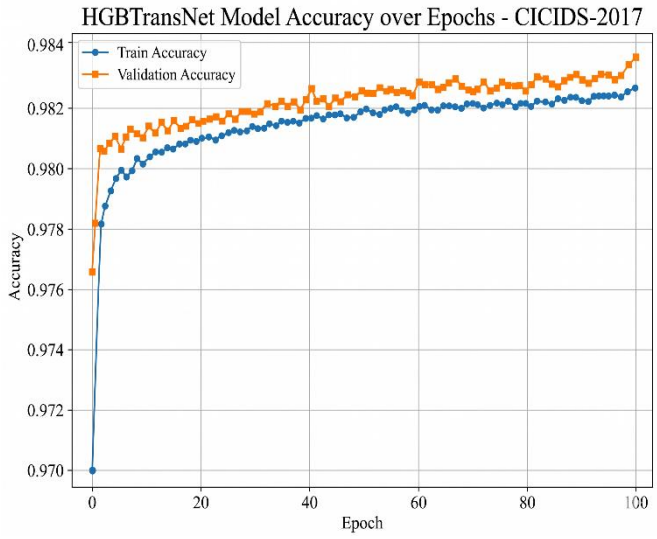
The ROC curve shown in Figure 2 analysis of the HGBTransNet on all the datasets, the highest AUC achieved score is 1.00, demonstrating near-perfect parting between normal and intrusion traffic. These results show that the proposed architecture generalizes well across diverse types of datasets without suffering from overfitting.



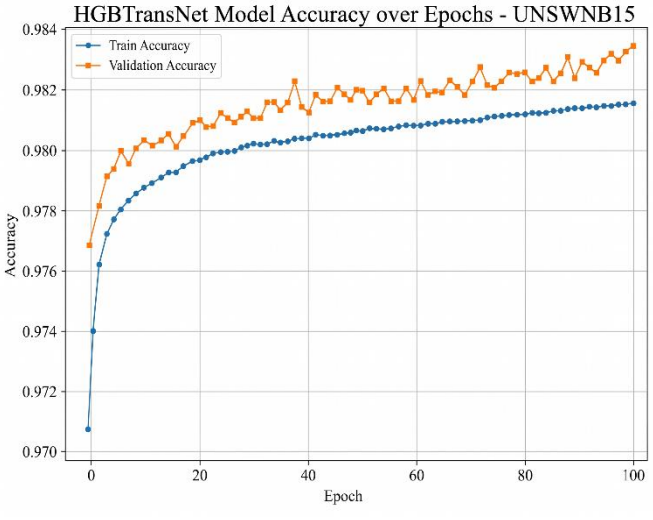
(a)



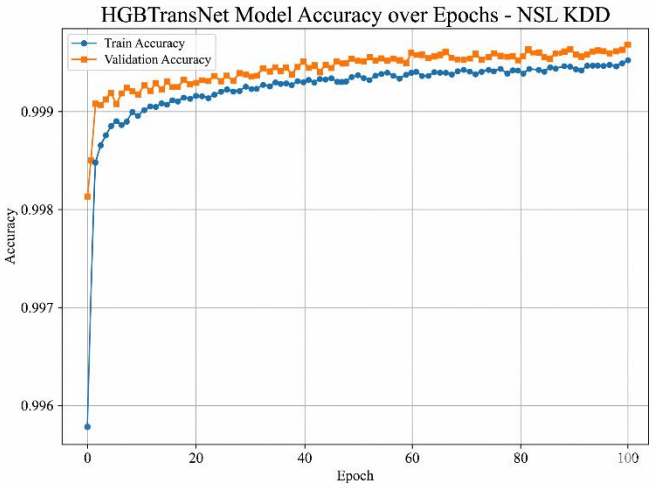
(b)



(c)



(d)



(e)

Fig. 3 Train and validation accuracy (Epochs vs accuracy) curves for attack detection across different datasets (a) WSN-DS, (b) WSN-BFSF, (c) CICIDS-2017, (d) UNSWNB15, and (e) NSL-KDD datasets.

Figure 3 shows the graph between accuracy versus epochs of HGBTransNet model on all the datasets. Overall, HGBTransNet model attained higher performance in identifying multiple intrusion types, offering a promising key for real-time WSN security monitoring. Its hybrid architecture, combined with rigorous preprocessing and optimized hyperparameters, ensures that the model can adapt efficiently to varying network conditions and attack behaviours.

4.2. Comparative Analysis

In Table 5, the proposed HGBTransNet is compared with the latest existing IDS methods on the different datasets. Two other architectures with high accuracy were developed by Saleh et al. [38] and Baniata et al. [30], both of which used GNB+SGD and CNN+GCN+BiLSTM, respectively, and yielded a 98.00% accuracy. Jabber et al. [39] used an LSTM

model, and Al-Khatib et al. [29] an Improved BiLSTM, both with an accuracy of 99.00%. Moreover, Halbouni et al. [20], Nguyen et al. [42], Talukder et al. [17], and Khraisat et al. [13] suggested the CNN-LSTM model, which gained accuracy of 99.58%, GSWO-CatBoost model with accuracy 99.65%, GAN model with accuracy 99.66%, and RF-FedAvg model with accuracy 99.67%, respectively. On the other hand, the HGBTransNet architecture combines GRU, BiLSTM and Transformer networks to learn local, temporal and global contextual features that result in the highest accuracy of 99.80%, which proves the superior presentation of IDS compared to other methods on the WSN-DS dataset.

Table 5. Comparison of HGBTransNet model with existing models

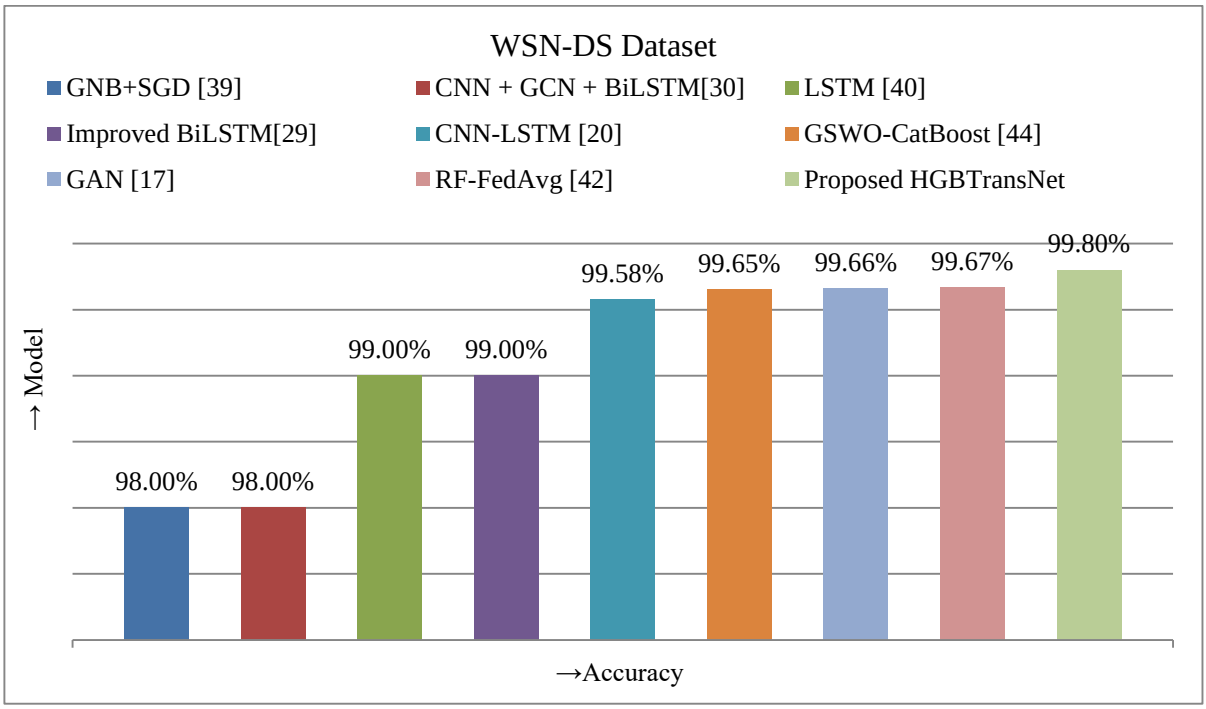
Dataset	Model	Accuracy
WSN-DS	GNB+SGD [38]	98.00%
	CNN + GCN + BiLSTM [30]	98.00%
	LSTM [39]	99.00%
	Improved BiLSTM [29]	99.00%
	CNN-LSTM [20]	99.58%
	GSWO-CatBoost [42]	99.65%
	GAN [17]	99.66%
	RF-FedAvg [13]	99.67%
	Proposed HGBTransNet	**99.80%**
WSN-BFSF	DCGAN [28]	94.00%
	Enhanced DL [25]	98.12%
	Proposed HGBTransNet	**98.20%**
CICIDS2017	TRBMA [41]	98.66%
	Transformer-CNN [16]	99.13%
	CNN-LSTM [20]	99.56%
	GSWO-CatBoost [42]	99.74%
	Proposed HGBTransNet	**99.99%**
UNSWNB15	Transformer-based IDS [33]	79.00%
	ANN [19]	90.85%
	NN [43]	92.00%
	CNN-LSTM [20]	93.78%
	CNN +RNN [40]	94.00%
	Enhanced CNN [26]	95.47%
	Proposed HGBTransNet	**97.20%**
NSL-KDD	OS-RVFL [44]	89.43%
	CNN +RNN [40]	97.40%
	CNN+BiLSTM [45]	98.42%
	Multi-Scale Transformer [33]	99.00%
	Proposed HGBTransNet	**99.70%**

Devi et al. [28] presented a DCGAN for IDS aiming for a detection rate of 94.00% which proves the efficiency of Adversarial learning in feature generation. A model named Enhanced DL was developed by Reza et al. [25] that automatically learned features to achieve 98.12% detection accuracy. The HGBTransNet model attaining the peak accuracy of 98.20% on the WSN-BFSF dataset, thereby outperforming the existing works in the field. Figure 4 diagram compares multiple existing models with the proposed HGBTransNet model.

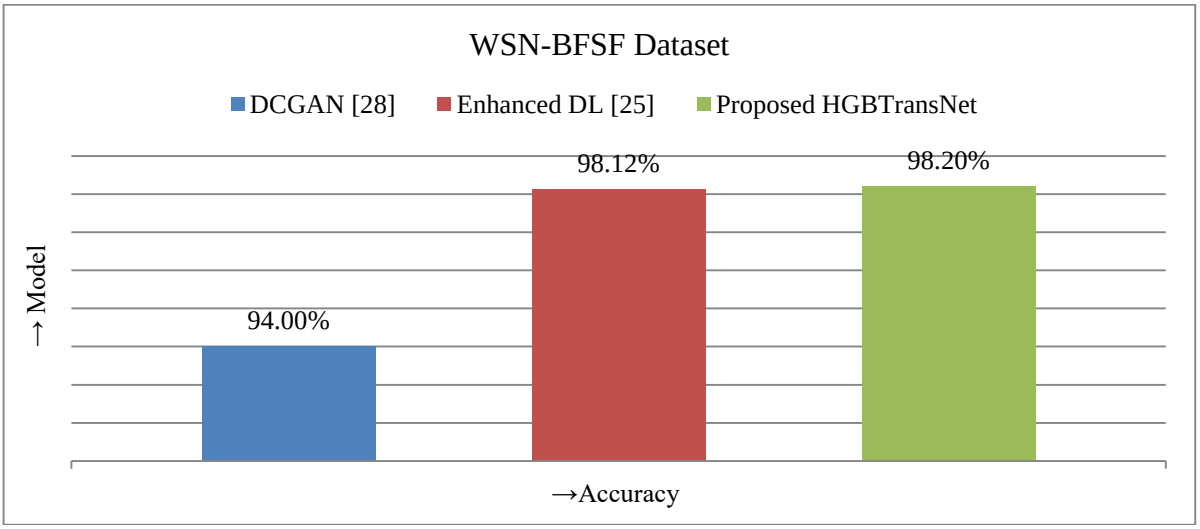
The TRBMA model was introced by Guo et al. [41], which was optimized based on feature learning and achieved 98.66% accuracy in detecting intrusion. To further enhance the classification performance, Kamal and Mashaly [16] developed a Transformer-CNN model that achieved 99.13% test accuracy using self-attention and CNN feature extraction. Halbouni et al. [20] introduced a CNN-LSTM model that achieved an accuracy of 99.56%, while Nguyen et. al. [42] introduced a GSWO-CatBoost model, achieving an accuracy of 99.74%.

The proposed HGBTransNet model can reach the best accuracy rate of 99.99% on the CICIDS2017 dataset, which shows that it has excellent intrusion detection performance than the current state-of-the-art methods.

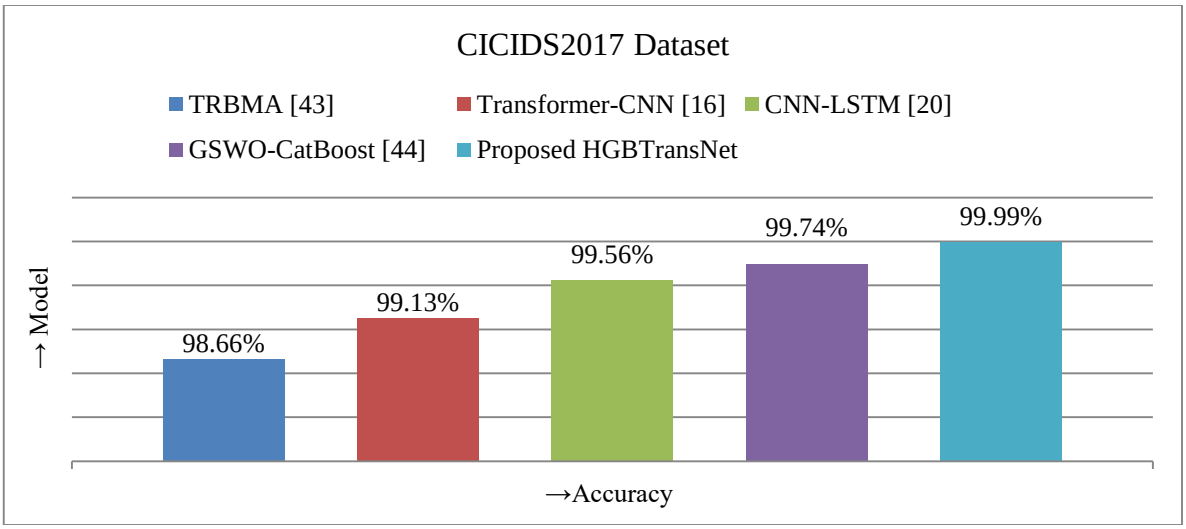
Stein et al. [32] suggested a Transformer based IDS with accuracy of 79.00%, and proved that the self-attention architecture can be applied for detection, however with low detection performance on UNSWNB15 dataset. Kasongo and Sun [19] used an Artificial Neural Network (ANN) with an accuracy of 90.85%, while Putra et al. [43] used a Neural Network (NN) with an accuracy of 92.00%. Moreover, CNN-LSTM, CNN+RNN and Enhanced CNN models were projected by Halbouni et al. [20], Sinha et al. [40], and Deshmukh et al. [26], respectively, with the accuracy of 93.78%, 94.00%, and 95.47%. Compared to this, the proposed HGBTransNet model is capable of effectively capturing the local, sequential and global contextual features from the network traffic by combining the GRU, BiLSTM, and Transformer models. This feature learning approach is comprehensive and helps the proposed model to attain maximum accuracy of 97.20% with superior intrusion detection performance compared to other existing IDS in the UNSWNB15 dataset.



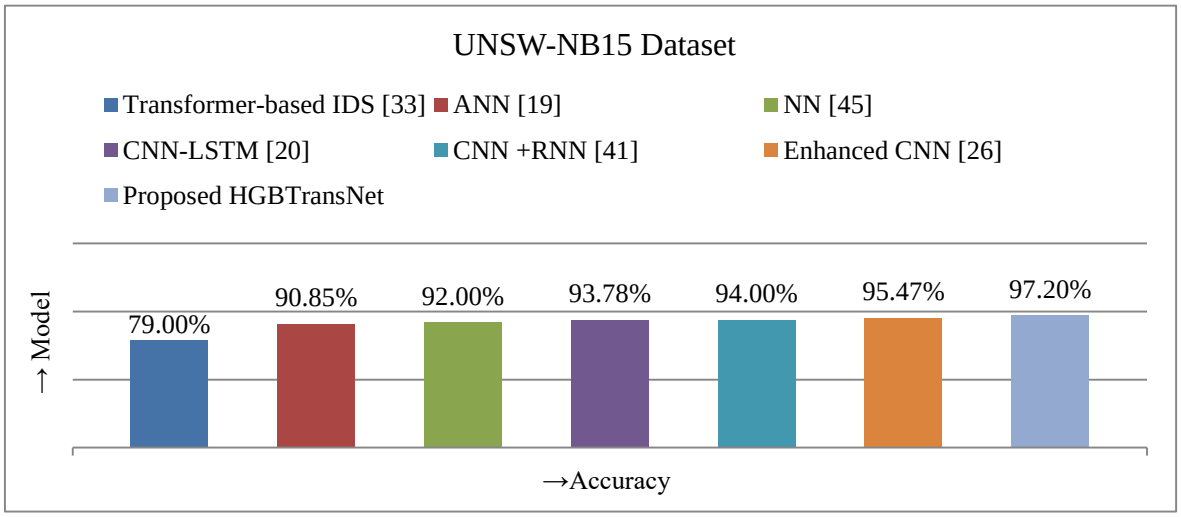
(a)



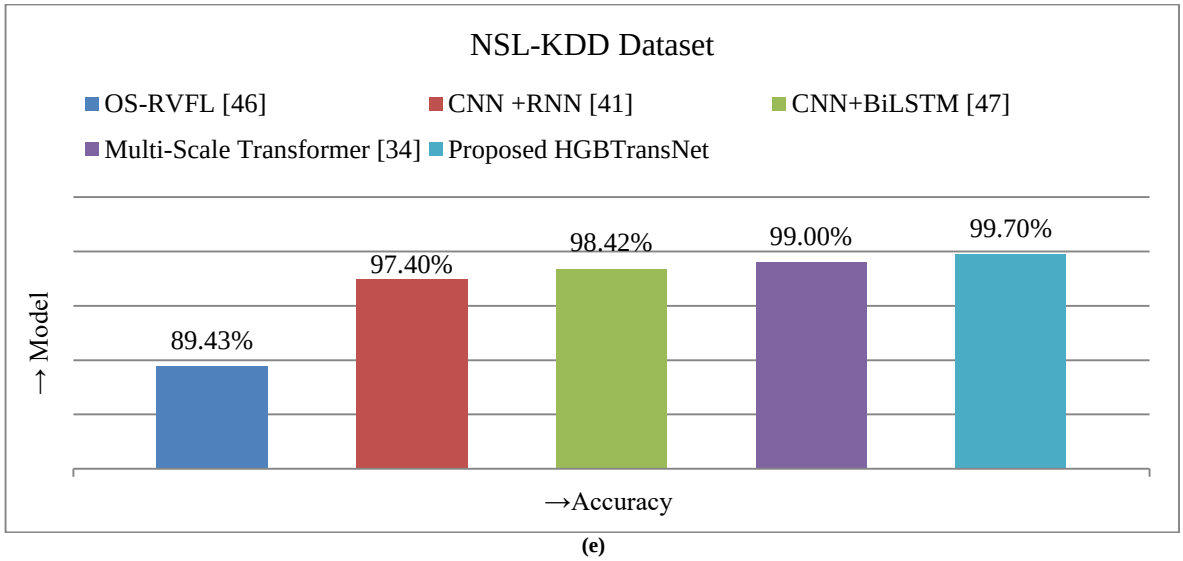
(b)



(c)



(d)



(e)

Fig. 4 Comparison of HGBTransNet with existing IDS models (a) WSN-DS, (b) WSN-BFSF, (c) CICIDS-201,7 (d) UNSWNB15, and (e) NSL-KDD datasets.

Sahoo et al. [44] presented the OS-RVFL model and obtained 89.43% accuracy, which was very efficient in online learning for intrusion detection. Sinha et al. [40] presented a CNN+RNN model, and Said et al. [45] presented a CNN+BiLSTM model, with the former model attaining an accuracy of 97.40% and the latter model, 98.42%, respectively, by successfully learning spatial and temporal features. To further boost the accuracy of the detection, Xi C et al. [33] introduced a Multi Transformer model that employs self-attention to enable it to learn global features. The HGBTransNet model, on the other hand, which integrates GRU, BiLSTM and Transformer architectures, achieves the highest accuracy of 99.70% on the NSLKDD and surpasses the current state-of-the-art approaches.

The combination of hierarchical feature extraction and Transformer-based attention mechanisms is responsible for the improved performance of HGBTransNet. The hierarchical architecture allows the model to learn discriminative features at various representation levels, and the Transformer component is able to represent important relationships and dependencies between features in the network. This is a combination that is richer in feature representation than approaches with a single learning mechanism. Also, suitable feature standardization and preprocessing further ensure stable training and effective feature representation. HGBTransNet is tested using five datasets covering 26 attack categories, to offer a wider assessment in various network contexts and attack patterns. Several existing studies, however, have been assessed on a limited number of datasets and attack categories, possibly constraining the assessment of the generalization. Thus, the proposed architecture, preprocessing strategy and multi-dataset evaluation lead to the observed performance improvements.

5. Conclusion

This work proposes a HGBTransNet model to detect intrusions in WSNs, combining GRU, Bi-LSTM and Transformer networks to capture local, temporal and global context information from network traffic. The proposed model has been experimentally evaluated on various benchmark datasets, and the experimental results showed that the proposed architecture always outperformed the existing approaches. In particular, it managed to outperform the classic ML, hybrid DL and the Transformer-based IDS, with average accuracies of 99.80%, 98.20%, 99.70%, 97.20% and 99.99%, respectively. The superior performance validates the efficacy of concatenating these two learning mechanisms, namely recurrent and attention-based, for powerful feature learning and attack classification. Thus, the HGBTransNet framework is a reliable and efficient approach to secure next-generation WSNs from various cyber threats. The architecture of HGBTransNet is relatively compact to minimize the computation overhead, but it needs to be evaluated by conducting experiments on resource-constrained WSN nodes using hardware-level measurements of model size, computational complexity, energy requirement, memory requirement, and inference latency. Further research is needed to design a lightweight version of the HGBTransNet for deployment on the resource-limited WSN sensor nodes. Furthermore, federated learning, online/incremental learning, and Explainable Artificial Intelligence (XAI) can be implemented to improve privacy preservation, adaptability to new attack patterns, and interpretability of the model for real-time WSN-IDS systems. Future work involves the study of imbalance handling techniques like focal loss and class-weighted learning to increase the minority-class detection performance.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Funding Statement

No Funding.

References

[1] T. Gopala, V. Raviram, and N.L. Udaya Kumar, "Detecting Security Threats in Wireless Sensor Networks using Hybrid Network of CNNs and Long Short-Term Memory," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 1s pp. 704-722, 2023. [Google Scholar] [Publisher Link]

[2] Iman Almomani, Bassam Al-Kasasbeh, and Mousa AL-Akhras, "WSN-DS: A Dataset for Intrusion Detection Systems in Wireless Sensor Networks," *Journal of Sensors*, vol. 2016, pp. 1-16, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[3] Andrea Pinto et al., "Survey on Intrusion Detection Systems based on Machine Learning Techniques for the Protection of Critical Infrastructure," *Sensors*, vol. 23, no. 5, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hakan Can Altunay, and Zafer Albayrak, "A Hybrid CNN+LSTM-based Intrusion Detection System for Industrial IoT Networks," *Engineering Science and Technology, an International Journal*, vol. 38, pp. 1-23, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Kuljeet Singh, Amit Mahajan, and Vibhakar Mansotra, "Hybrid CNN-LSTM Model Combined with Feature Selection and SMOTE for Detection of Network Attacks," *International Journal of Sensor Networks*, vol. 43, no. 4, pp. 208-222, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Md. Alamin Talukder et al., "MLSTL-WSN: Machine Learning-based Intrusion Detection using SMOTETomek in WSNs," *International Journal of Information Security*, vol. 23, no. 3, pp. 2139-2158, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ovi Abul Hussain, Zigang Chen, and Haihua Zhu, "sSecure Net: A Hybrid CNN-LSTM-based Intrusion Detection System for Securing IoT Networks," *Proceedings of the 4th International Conference on Computer, Artificial Intelligence and Control Engineering*, pp. 537-544, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[8] Tinshu Sasi et al., "An Efficient Self Attention-based 1D-CNN-LSTM Network for IoT Attack Detection and Identification using Network Traffic," *Journal of Information and Intelligence*, vol. 3, no. 5, pp. 375-400, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Pendukeni Phalaagae et al., "A Hybrid CNN-LSTM Model with Attention Mechanism for Improved Intrusion Detection in Wireless IoT Sensor Networks," *IEEE Access*, vol. 13, pp. 57322-57341, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[10] Raja Waseem Anwar et al., "Federated Learning with LSTM for Intrusion Detection in IoT-based Wireless Sensor Networks: A Multi-Dataset Analysis," *PeerJ Computer Science*, vol. 11, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] Nurtay Albanbay et al., "Federated Learning-based Intrusion Detection in IoT Networks: Performance Evaluation and Data Scaling Study," *Journal of Sensor and Actuator Networks*, vol. 14, no. 4, pp. 1-24, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] Vivek Kumar Pandey et al., "Enhancing Intrusion Detection in Wireless Sensor Networks using a Tabu Search based Optimized Random Forest," *Scientific Reports*, vol. 15, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[13] Ansam Khraisat et al., "RF-FedAvg: Federated Learning-based Random Forest Model for Intrusion Detection in Wireless Sensor Networks," *Cluster Computing*, vol. 28, no. 13, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[14] Zhenyue Long et al., "A Transformer-based Network Intrusion Detection Approach for Cloud Security," *Journal of Cloud Computing*, vol. 13, no. 1, pp. 1-11, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Amal Ajayan, G. Kirubavathi, and Iqbal H. Sarker, "DistilXIDS: Efficient, Lightweight and Explainable Transformer-based Language Model for Real-Time Network Intrusion Detection," *Neurocomputing*, vol. 668, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[16] Hesham Kamal, and Maggie Mashaly, "Advanced Hybrid Transformer-CNN Deep Learning Model for Effective Intrusion Detection Systems with Class Imbalance Mitigation Using Resampling Techniques," *Future Internet*, vol. 16, no. 12, pp. 1-74, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Md. Alamin Talukder, Majdi Khalid, and Nasrin Sultana, "A Hybrid Machine Learning Model for Intrusion Detection in Wireless Sensor Networks Leveraging Data Balancing and Dimensionality Reduction," *Scientific Reports*, vol. 15, pp. 1-23, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] H. Fares, N. Aknin, and Z. Oughannou, "An Adaptive Framework for Real-Time Intrusion Detection in WSNs and IoMT: Comparative Study using CICDDoS-2019, WSN-DS and CICIoMT2024 Datasets," *Mathematical Modeling and Computing*, vol. 13, no. 1, pp. 240-251, 2026. [CrossRef] [Publisher Link]

[19] Sydney M. Kasongo, and Yanxia Sun, "Performance Analysis of Intrusion Detection Systems using a Feature Selection Method on the UNSW-NB15 Dataset," *Journal of Big Data*, vol. 7, no. 1, pp. 1-20, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Asmaa Halbouni et al., "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," *IEEE Access*, vol. 10, pp. 99837-99849, 2022. [Google Scholar] [Publisher Link]

[21] Salim Salmi, and Lahcen Oughdir, "Performance Evaluation of Deep Learning Techniques for DoS Attacks Detection in Wireless Sensor Network," *Journal of Big Data*, vol. 10, no. 1, pp. 1-25, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Hee-Yong Kwon, Taesic Kim, and Mun-Kyu Lee, "Advanced Intrusion Detection Combining Signature-based and Behavior-based Detection Methods," *Electronics*, vol. 11, no. 6, pp. 1-19, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] V. Gowdhaman, and R. Dhanapal, "An Intrusion Detection System for Wireless Sensor Networks using Deep Neural Network," *Soft Computing*, vol. 26, no. 23, pp. 13059-13067, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Muawia A. Elsadig et al., "Detection of Denial-of-Service Attack in Wireless Sensor Networks: A Lightweight Machine Learning Approach," *IEEE Access*, vol. 11, pp. 83537-83552, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[25] Farhana Reza et al., "DDoS-Net: Classifying DDoS Attacks in Wireless Sensor Networks with Hybrid Deep Learning," *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, Dhaka, Bangladesh, pp. 487-492, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[26] Amogh Deshmukh, and Kiran Ravulakollu, "An Efficient CNN-based Intrusion Detection System for IoT: Use Case towards Cybersecurity," *Technologies*, vol. 12, no. 10, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[27] Mohamed H. Behiry, and Mohammed Aly, "Cyberattack Detection in Wireless Sensor Networks using a Hybrid Feature Reduction Technique with AI and Machine Learning Methods," *Journal of Big Data*, vol. 11, pp. 1-39, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[28] Manu Devi, Priyanka Nandal, and Harkesh Sehrawat, "A Lightweight Approach for Intrusion Detection in WSNs based on DCGAN," *International Journal of Information Technology*, vol. 17, no. 2, pp. 951-957, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[29] Ra'ed M. Al-Khatib et al., "A Novel Improved Deep Learning Model based on Bi-LSTM Algorithm for Intrusion Detection in WSN, vol. 20, no. 2, pp. 532-565, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[30] Laith H. Baniata et al., "A Dual-Attention CNN–GCN–BiLSTM Framework for Intelligent Intrusion Detection in Wireless Sensor Networks," *Future Internet*, vol. 18, no. 1, pp. 1-21, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[31] Kadiyala Ramana et al., "WOGRU-IDS — An Intelligent Intrusion Detection System for IoT Assisted Wireless Sensor Networks," *Computer Communications*, vol. 196, pp. 195-206, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[32] Kyle Stein et al., "A Transformer-based Framework for Payload Malware Detection and Classification," *2024 IEEE World AI IoT Congress (AIIoT)*, Seattle, WA, USA, pp. 105-111. 2024. [CrossRef] [Google Scholar] [Publisher Link]

[33] Chiming Xi, Hui Wang, and Xubin Wang, "A Novel Multi-Scale Network Intrusion Detection Model with Transformer," *Scientific Reports*, vol. 14, pp. 1-17, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[34] Shailesh Singh Panwar, Y. P. Raiwani, and Lokesh Singh Panwar, "An Intrusion Detection Model for CICIDS-2017 Dataset Using Machine Learning Algorithms," *2022 International Conference on Advances in Computing, Communication and Materials (ICACCM)*, Dehradun, India, pp. 1-10, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[35] Murat Dener et al., "WSN-BFSF: A New Data Set for Attacks Detection in Wireless Sensor Networks," *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 2109-2125, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[36] Nour Moustafa, and Jill Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set)," *2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, ACT, Australia, pp. 1-6, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[37] L. Dhanabal, and S.P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System based on Classification Algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 6, pp. 446-452, 2015. [Google Scholar] [Publisher Link]

[38] Hadeel M. Saleh, Hend Marouane, and Ahmed Fakhfakh, "Stochastic Gradient Descent Intrusions Detection for Wireless Sensor Network Attack Detection System using Machine Learning," *IEEE Access*, vol. 12, pp. 3825-3836, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[39] Sanaa Ali Jabber, Soukaena H.H Ashem, and Shatha H. Jafer, "A Novel Approach to Intrusion-Detection System: Combining Lstm and the Snake Algorithm," *Jordanian Journal of Computers and Information Technology*, vol. 9, no. 4, pp. 360-376, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[40] Priyanshu Sinha et al., "An Efficient Data Driven Framework for Intrusion Detection in Wireless Sensor Networks using Deep Learning," *Scientific Reports*, vol. 15, pp. 1-25, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[41] Derui Guo, and Yufei Xie, "Research on Network Intrusion Detection Model based on Hybrid Sampling and Deep Learning," *Sensors*, vol. 25, no. 5, pp. 1-32, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[42] Thuan Minh Nguyen, Hanh Hong-Phuc Vo, and Myungsik Yoo, "Enhancing Intrusion Detection in Wireless Sensor Networks Using a GSWO-CatBoost Approach," *Sensors*, vol. 24, no. 11, pp. 1-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[43] Zico Pratama Putra et al., "Evaluating the Performance of Classification Algorithms on the UNSW-NB15 Dataset for Network Intrusion Detection," *FIFO Scientific Journal*, vol. 16, no. 1, pp. 84-95, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[44] Shreeya Swagatika Sahoo, and Dibyasundar Das, "Design of Key Agreement Authentication Scheme with IDS Using OS-RVFL Neural Network for Secure Fog Computing," *IEEE Internet of Things Journal*, vol. 12, no. 14, pp. 28521-28530, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[45] Rachid Ben Said, Zakaria Sabir, and Iman Askerzade, "CNN-BiLSTM: A Hybrid Deep Learning Approach for Network Intrusion Detection System in Software-Defined Networking with Hybrid Feature Selection," *IEEE Access*, vol. 11, pp. 138732-138747, 2023. [CrossRef] [Google Scholar] [Publisher Link]

Appendix

Table 4. Shows the performance on diverse dataset of HGBTransNet model

Dataset	Total Samples	Training (80%)	Testing (20%)	Class	T.P.	T.N.	F.P	F.N.	Precision	Recall	F1-Score	Accuracy	FPR	AUC-ROC
WSN-DS	374661	299728	74933	Blackhole	1,972	72,836	87	38	0.96	0.98	0.97	0.9983	0.0012	1.00
				Flooding	661	74,201	70	1	0.9	1	0.95	0.9991	0.0009	1.00
				Gray-hole	2,807	71,863	151	112	0.95	0.96	0.96	0.9965	0.0021	1.00
				Normal	67,829	6,800	119	185	1	1	1	0.9959	0.0172	1.00
				TDMA	1,233	73,601	4	95	1	0.93	0.96	0.9987	0.0001	0.98
NSL-KDD	148517	118813	29704	DoS Attack	10,669	19,002	25	8	0.998	0.999	0.998	0.9989	0.0013	1.00
				Probe Attack	2,790	26,853	35	26	0.988	0.991	0.989	0.9979	0.0013	1.00
				Remote to Local	577	28,923	31	173	0.949	0.769	0.85	0.9931	0.0011	1.00
				User to Root	42	29,644	10	8	0.808	0.84	0.824	0.9994	0.0003	1.00
				Normal	15,321	14,089	204	90	0.987	0.994	0.99	0.9901	0.0143	1.00
WSN-BFSF	312106	249684	62422	Blackhole	1,180	59,643	426	1173	0.73	0.5	0.6	0.9744	0.0071	0.96
				Flooding	5,893	56,232	221	76	0.96	0.99	0.98	0.9952	0.0039	1.00
				Forwarding	954	60,437	456	575	0.68	0.62	0.65	0.9835	0.0075	0.99
				Normal	51,765	8,324	1,527	806	0.97	0.98	0.98	0.9626	0.155	0.99
UNSWNB15	257673	206138	51535	Analysis	98	50,904	96	437	0.51	0.18	0.27	0.9897	0.0019	0.95
				Backdoor	72	51,034	35	394	0.67	0.15	0.25	0.9917	0.0007	0.96
				DoS	639	47,497	767	2632	0.45	0.2	0.27	0.934	0.0159	0.94
				Exploits	7,963	38,010	4,620	942	0.63	0.89	0.74	0.8921	0.1084	0.96
				Fuzzers	3,537	46,136	550	1312	0.87	0.73	0.79	0.9639	0.0118	0.98
				Generic	11,500	39,711	50	274	1	0.98	0.99	0.9937	0.0013	1.00
				Normal	18,218	32,228	707	382	0.96	0.98	0.97	0.9789	0.0215	1.00
				Reconnaissance	2,116	48,571	166	682	0.93	0.76	0.83	0.9835	0.0034	0.98
				Shellcode	214	51,046	187	88	0.53	0.71	0.61	0.9947	0.0036	0.99
				Worms	0	51,500	0	35	0	0	0	0.9993	0.0000	0.99
CICIDS2017	225745	180596	45149	BENIGN	19,532	25,603	4	10	1	1	1	0.9997	0.0002	1.00
				DDoS	25,603	19,532	10	4	1	1	1	0.9997	0.0005	1.00