Power System Protection and Control
ISSN:1674-3415

GSSB: A NOVEL METHOD FOR PROVIDING A HYBRID DEEP LEARNING MODEL FOR ANOMALY DETECTION IN INTERNET OF THINGS NETWORKS

Waleed Abdulzahra Jalil[1], Sima Emadi[2]*, Enas Fadhil Abdullah[3], Sanaz Asadinia[1]

Waleed.jalil@iau.ac.ir, Sima.emadi@iau.ac.ir
inasf.alturky@uokufa.edu.iq, sanaz.asadinia@iau.ac.ir
[1]Department of Computer Engineering, Isf.C., Islamic Azad University, Isfahan, Iran
[2]Department of Computer Engineering, Ya.C., Islamic Azad University, Yazd, Iran.
[3]Department of Computer Science, Collage Education for Girls, University of Kufa, Najaf, Iraq
*Corresponding author

Received:19 November 2025 **Revised**: 22 December 2025 **Accepted**: 4 January 2025

***ABSTRACT*:**

In recent years, extensive research has been published to detect anomaly in the Internet of Things, but this research has not been able to respond to challenges such as limited coverage of attack types, data imbalance, lack of effective optimization mechanisms, and inefficiency of sensor nodes for deep learning, etc. This paper presents an intelligent and decentralized intrusion detection system for IoT networks using a combination of deep learning techniques, artificial data generation, and optimization algorithms. The main objective is to design a model called GSSB (GAN_SCNN_SHO_BiLSTM) that is able to accurately detect various cyber-attacks, especially unknown and zero-day attacks. In this method, to slove the problem of imbalance data from Generative Adversarial Networks (GAN) are used, after generating augmented data by GAN, the Hurst profile is extracted as a complementary statistical feature by calculating the autocorrelation and stability of the data and added to the feature vector to increase the model's ability to recognize spatial patterns and temporal dynamics. Then complex location features of network traffic with the Stacked Convolutional Neural Network (SCNN) are extracted and the time data analysis was done by the BiLSTM network. Also used to select optimal features and reduce model complexity of the model, the binary version of the Seahorse Optimization (SHO) algorithm is used. The proposed structure is designed in a Fog Computing environment and uses federated learning to preserve data privacy and models trained in sub-nodes are aggregated together. To evaluate the model, experiments were conducted on the dataset Edge-IIoT 2023 and Edge-IIoT, and performance of the model was compared with criteria such as Accuracy, Recall, Detection Rate, False Positive Rate (FPR), and F1 Score. The results showed that the proposed model managed to achieve a F1 score of 99.3%, a recall of 98.6%, and an accuracy of 99.2%, and performed better compared to other reference methods such as CNN-BiLSTM and the federated method. These results demonstrate the ability of the GSSB model to identify advanced threats and improve the security of IoT networks in real-time situations.

INTRODUCTION

The term Internet of Things refers to a communication infrastructure in which a wide array of physical devices equipped with sensors, software, and electronic components, are used to record, store, analyze, and exchange of data are connected with each other. This technology is increasingly evolving physical environments into intelligent, connected systems. The wide variety of IoT devices, from household appliances and wearable equipment to industrial machinery and transportation systems, has led to its applications expanding in areas such as remote healthcare, smart agriculture, energy management, tourism, and digital banking [1]. It is estimates that by 2030, more than 29 trillion devices will be connected to the internet, which doubles the need to address the security aspects of this extensive infrastructure [2]. With the exponential growth of physical devices connected to the internet, security threats have developed simultaneously. Attacks such as network intrusions, device hijacking in the form of botnets, phishing attacks, and Distributed Denial of Service (DDoS) are only part of the risks that threaten IoT networks. Malware such as Mirai, Hajime and bashlite have put critical infrastructure under serious threat by exploiting security vulnerabilities of connected devices. Although traditional network security solutions such as firewalls, traffic filtering, and managed security services have been employed to counter these threats,

DOI: 10.46121/pspc.54.1.1

their rule-based and non-intelligent nature renders them incapable of dealing with emerging and complex attacks, often acting reactively [3]. On the other hand, with the increase in complex cyberattacks, tools relying on abnormal behavior detection, such as Intrusion Detection Systems (IDS), have gained attention. These systems can identify many threats before damage occurs by analyzing network traffic behavior and detecting abnormal patterns. Among these, machine learning methods, especially deep learning, have demonstrated significantly better performance than traditional methods, particularly in identifying zero-day attacks where no known signatures are available. However, these methods also face challenges such as the need for balanced data, high volumes of learnable parameters, the necessity for effective feature selection, and resource limitations of sensor nodes. Additionally, weaknesses in the security updates of IoT devices, lack of transparency in deployment methods, and high heterogeneity in IoT environments have resulted in a less than optimal security outlook for these networks. Particularly in critical infrastructures such as power plants, healthcare centers, and smart transportation systems, any vulnerability in the IoT layer can have widespread consequences for national, economic, and social security [4]. Consequently, there is an increasing need to develop comprehensive and innovative approaches that leverage artificial intelligence capabilities to ensure the security of IoT networks against known and unknown threats. Network intrusion detection systems, as one of the critical components of cybersecurity, are responsible for analyzing network traffic and classifying it into normal or malicious behaviors. These systems are generally divided into two categories: misuse detection and anomaly detection. In the former, patterns of known attacks are maintained in blacklists and matched against current traffic. Despite their simplicity, these methods lack the ability to counter new attacks. In contrast, anomaly detection methods, by statistically analyzing the behavior of authorized users, can identify zero-day attacks and have been largely developed based on machine learning and deep learning techniques. However, each of these methods faces challenges: blacklist methods require significant memory resources and lack the ability to detect emerging attacks; exploratory methods have limited accuracy and generalization capabilities; machine learning algorithms perform poorly in feature selection and are sensitive to data imbalance; and finally, deep learning models, despite their high accuracy, require complex tuning, lengthy training, and precise optimization to prevent detection errors [5]. Therefore, it is necessary to develop intelligent hybrid methods that cover these shortcomings. With the increasing expansion of smart technologies, the Internet of Things (IoT) plays a key role as one of the main pillars of the Fourth Industrial Revolution, transforming industrial, urban, agricultural, and healthcare processes. By connecting billions of physical devices to the global network, the technology has enabled real-time data exchange and analysis, creating an infrastructure for smart decision-making. However, the distributed and heterogeneous nature of IoT devices and their inherent limitations in processing and storage have made the environment vulnerable to cyber threats. Attacks such as Distributed Denial of Service (DDoS), device hijacking in the form of botnets, specific malware such as Mirai, and abuse of security vulnerabilities are among the common threats in this space. One of the most important tools for dealing with these threats is Intrusion Detection Systems (IDS), which detect suspicious or abnormal activities by analyzing the network traffic behavior [6]. Traditional IDS systems, mainly based on signatures or fixed rules, are incapable of dealing with new and unknown attacks and have limited accuracy. In contrast, machine learning and deep learning methods with the ability to learn complex behavioral patterns have a better ability to detect zero-day attacks and emerging threats. Convolutional models (CNN) are effective at extracting spatial features from data, and long-term memory networks LSTM and BiLSTM can analyze temporal dependencies in network traffic [7]. However, challenges such as an imbalance in traffic data, the complexity of learning models, high resource consumption, and lack of effective privacy protection mechanisms still remain.

Recent research has tried to ensure the confidentiality of data without the need to transfer raw information using federated learning. However, many of these approaches are limited only to identifying specific types of attacks, have not used feature selection algorithms, and have put high computational pressure on weak sensor nodes. On the other hand, in the Fog Layer, there are also numerous challenges such as resource heterogeneity, decision-making delays, non-optimized task assignment, and complex interactions with edge and cloud that can greatly affect the efficiency of intrusion detection systems. Therefore, the need to develop an intelligent, light, distributable, and reliable method for detecting influence in real architectures of IoT, taking into account the considerations of the fog layer, is quite tangible. In the present paper, a hybrid and optimized approach is proposed that utilizes SCNN [8] to extract spatial feature, BiLSTM for time modeling, GAN for data balancing, Seahorse optimization algorithm for effective feature selection, and federated learning to protect privacy, a new architecture is designed to detect intrusion in the fog layer of IoT networks. This hierarchical architecture, in addition to increasing the accuracy of diagnosis and reducing error, allows for deployment in real environments with limited resources. This study aims to fill the gaps present in previous methods, has been taken a step towards improving the security of IoT networks on a large scale.

DOI: 10.46121/pspc.54.1.1

In this research, the main focus is on identifying anomaly in the IoT Fog Layer, where we face challenges such as a lack of training data, limited processing resources, and the need for rapid and distributed decision-making. To overcome the problem of data scarcity, GAN networks have been used to generate artificial data. Then, using the architectures of SCNN and BiLSTM, the key features are extracted and categorized. The Seahorse optimization algorithm [9] is used to select effective feature and adjust hyperparameter to reduce complexity and increase accuracy. The final model is trained locally in fog nodes and aggregated with federated learning in the central node. The final version of the aggregated model is re-sent to sub-nodes so that the process of detecting attacks continues simultaneously and decentralized. The main challenge in the Fog Layer is the fast, light, and secure processing of data in a heterogeneous and distributed environment, which the proposed method tries to use the combination of GAN, SCNN, SOA, and BiLSTM respond to it. The main innovation of the proposed method in the simultaneous combination of three key components is to improve the accuracy and efficiency of the intrusion detection system in IoT environments: balancing imbalanced data using Generative Adversarial Networks (GAN), optimal feature extraction and selection utilizing Stacked Convolutional Neural Networks (SCNN) and the Seahorse Optimization (SHO) algorithm, and modeling temporal dependencies through BiLSTM. These components are implemented in the form of a federated learning framework to maintain data confidentiality and to distribute the final integrated model between Fog nodes. Additionally, a secure mechanism for exchanging blacklists using steganography in network traffic data enhances the system's resilience against attacks and information tampering.

The structure of this paper is briefly outlined as follows: In the first section elaborates on the importance of network security in the Internet of Things (IoT) and the Fog Layer, as well as the objectives of the research. In the second section previous studies and their limitations are examined in this section. In the third section of the proposed method, the GSSB framework is introduced, which includes modules for data balancing, feature extraction and selection, and temporal dependency modeling. The fourth section is dedicated to introducing the datasets and evaluation criteria, followed by the presentation of experimental results and comparisons with reference models, and finally, in the fifth section, there are future conclusions and proposals.

BACKGROUND RESEARCH

A review of previous studies showed that in the field of network intrusion detection systems, two main approaches, including centralized models and distributed models, are employed in the Fog or Edge Layer. In the distributed approach, data processing is carried out close to its source of production, which increases response speed, reduces latency, and reduces false alarm rates, although challenges such as coordination between processing nodes and maintaining the security of communications between them remain. Conversely, centralized models that perform processing at a central point have more advantages in terms of communication control and security but in scalability and high-volume management of data, especially in dynamic and wide-ranging environments, they show less efficiency. Chilmakurti et al. [10] in the year (2018) proposed a distributed deep learning-based network attack detection system in the Fog Layer of IoT, and investigated the effectiveness of deep models compared to shallow models. The advantages of this research include high accuracy of the deep model of the shallow model, better performance of the distributed models than the centralized model, real-time processing, and a low false alarm rate were mentioned, and the disadvantages of it was mentioned to investigate only four types of attacks, high temporal for training, the use of many resources for the training phase. Shafiei et al. [11] (2019) presented a framework for anomaly detection, comparing detection mechanisms on cloud and Fog nodes. In machine learning-based anomaly detection, Fog nodes showed better computational results (attack detection) compared to cloud infrastructures. Sinaipourfard et al. [12] (2019) introduced the Fog-to-cloudlet-to-Cloud ICT (F2c2C-ICT) architecture and the distributed D2C-ICT architecture in smart cities. These models identify attacks based on SDN at the network edge and controllers using machine learning techniques (Support Vector Machine). The D2C architecture includes four main layers: objects, Fog, micro-cloud, and cloud. The I2CM-IOT BOX in the cloud layer organizes all resources, data, software, and the D2C architecture's network. In the D2C-ICT architecture, communication exists between the leaders of distributed nodes in the cloudlet layer, allowing distributed leaders to manage their networks independently while communicating with other distributed and central leaders. This architecture has advantages over others due to the presence of the micro-cloud layer. Laval et al. [13] (2021) provided a framework for reducing DDoS attacks on IoT networks using fog computing. Using from distributed processing capabilities in the fog layer, this framework allows identifying and mitigating DDoS attacks close to the source of data production and preventing malicious traffic from being sent to the central layers. The proposed method involves analyzing and filtering traffic in the Fog Layer, using machine learning techniques to identify attack patterns, and making quick decisions to block malicious traffic.

DOI: 10.46121/pspc.54.1.1

Hybrid approaches designed by combining models such as CNN and LSTM or CNN and autoencoder (AE), by simultaneously exploiting from the ability to extract spatial features by CNN and identifying temporal dependencies by LSTM or AE, usually provide higher accuracy than individual models in intrusion and anomaly detection. However, due to the high complexity of architecture and the high volume of computing, these methods require powerful processing resources and more memory, which can make them difficult to implement in resource-restricted environments. Huang et al. [14] in the year (2020) proposed a hybrid model of their convolutional neural network and encoder neural network to detect network traffic anomalies early. The main goal of the research is to increase cybersecurity by identifying abnormal patterns in network traffic without the need for labeled data sets. This model uses the data self-management organization to extract features and calculate reconstruction errors, and these errors are actually used as an anomaly identification score. The model architecture includes coding and reopening, which uses precise approaches to revise features and identify abnormal traffic. Ravi et al. [15] in the year (2022) introduced an innovative recurrent deep learning-based method for intelligent network intrusion detection systems. This approach uses a combination of different features to enhance detection accuracy, so that the features are first extracted from the network input data and then a blending model is used to integrate them optimally. Recurrent deep learning algorithms have been particularly used to process sequenced data that is very common in networks. In addition to using basic models, this method has considered a meta-classifier to analyze and predict the final attacks. Empirical results show superior performance of this approach in accurately identifying attacks and reducing false alarm rates compared to traditional methods. Halbouni et al. [16] in the year (2022) introduced a hybrid deep neural network including Convolutional Neural Networks and long-term short-trerm memory networks for network intrusion detection systems. In this approach, CNN is used to extract spatial features from input data, especially complex patterns in network traffic, and LSTM is used to process and simulate temporal dependencies and long sequences in network data. This combination of two architectures has effectively achieved the advantages of both methods: the power of CNN to identify important features and the ability of LSTM to understand temporal dependencies. The proposed system has been remarkably successful in identifying complex attacks and reducing false alarm rates. The results of the experiments have shown that this method has higher accuracy and better performance in complex large network environments than traditional and single-model techniques. Ain et al. [17] published in the year (2025) evaluated and developed a hybrid deep learning model to identify DDoS attacks on Internet of Things (IoT) networks. In this study, various deep learning models such as hidden Autoencoders, LSTM Autoencoders, and convolutional neural networks (CNN) have been used to detect DDoS attacks in IoT environments. Also, a new hybrid model has been introduced that combines CNNs to extract features, LSTM networks to identify temporal patterns, and Autoencoders to reduce data dimensions. The advantages of this research include, high accuracy in identify complex DDoS attacks and the model's ability to identify complex network traffic patterns. Also, the hybrid model has been able to improve its performance in identifying attacks. Disadvantages of this research also include limitations in identifying rare types of attacks and problems related to data imbalances that require more attention in future research. Alshammari et al. [18] in the year (2024) suggested an intrusion detection model in cloud computing environments by using the Convolutional Neural Networks (CNN). Their main innovation was the design of a multi-stage architecture, including data preprocessing, feature selection using Pearson correlation matrix, balancing classes with SMOTE method and teaching CNN model. The main advantage of this method was high accuracy, efficiency in processing large data volumes, and ability to detect various types of attacks such as DDoS, SQL injection attacks, and botnets. However, the model's performance was evaluated in the face of infiltration attacks, be slightly weaker than other scenarios. Other strengths include the average complexity of architecture and the suitability for implementation in cloud environments. The MAS-LSTM method proposed by Qin et al. [19] in the year (2025) is a framework for anomaly detection in Industrial Internet of Things (IIoT) networks, which combining from Multi-Agent Systems (MAS) and Long Short-Term Memory (LSTM) networks. This method uses MAS to provide high scalability for processing sparse and large data ،While LSTM helps to simulate temporal dependencies and identifies long-term and complex anomalies. Its advantages include high scalability, accurate detection of temporal anomalies, and optimization using adaptive training techniques. However, the need for labeled data and problems in simulating complex attacks are one of the disadvantages of this method. Zhao et al. [20] in the year (2024) introduced a deep learning-based intrusion detection system to identify anomalous traffic in networks. This system uses deep learning models to analyze and is able to identify different types of network attacks and anomalies. Deep learning algorithms have been able to identify complex features and abnormal patterns in network traffic that may not be detected by traditional methods. This method uses different training data to automatically improve attack detection capabilities and provide higher accuracy in detecting malicious traffic. Conducted experiments have shown that this system has been able to have a high detection rate against various threats and reduced false alarm rates. Rouhani et al. [21] in the year (2018) conducted a comparison between different classifiers for identify attacks that included neural networks, random forests, k-nearest

DOI: 10.46121/pspc.54.1.1

neighbors, support vector machines, and decision trees that aim to detect and prevent DOS attacks. Linear SVM classification performs the worst performance. Decision tree classification was well done and k-nearest neighbor classification also achieved the same accuracy. The neural network performed surprisingly well despite having less than half a million Training samples. Also, Shafiei et al. [22] in the year (2018) provided a new architecture for detecting and preventing attacks at the network edge, and trained and tested four machine learning classifiers with the UNSW-NB15 dataset to detect attacks in an automated way. They used multi-layer perception, an alternative decision trees, and a neural network to build E3ML block. Finally, it can be said that 4 to the classifiers have reached the highest rates of TP.

Optimization and feature selection methods aim to reduce data dimensions and identify the most effective parameters, thereby increasing processing speed and reducing the computational complexity of the model. Although these approaches save resources and improve system efficiency, but in some caese, they may lead to the removal of important features and the loss of some vital information to accurately identify attacks or abnormalities. Dovindoran et al. [23] in the year (2024) has been introduced a deep learning-based network intrusion detection system and chaotic optimization strategy. This system, named Dugat-LSTM, combines Long Short-Term Memory (LSTM) networks and chaotic optimization techniques to improve the performance of intrusion detection systems. In this method, LSTM's capabilities are used to simulate temporal dependencies in network traffic, and chaos optimization algorithm is used to search for the most optimal parameters of the deep learning model. This technique is particularly effective in identifying complex attacks and network anomalies that may remain hidden from the perspective traditional systems. The authors have shown with extensive experiments that the Dugat-LSTM system has a higher accuracy of detecting threats and reducing false alarm rates than other similar methods. Turukman et al. [24] in the year (2024) introdued an efficient network intrusion detection system using feature selection and machine learning. This system, called M-MultiSVM, uses advanced feature selection techniques to improve the performance of machine learning algorithms in identifying network attacks. The authors by using multiple Support Vector Machine (SVM) algorithms extracted and selected important features from the network traffic datasets to increase model accuracy. Feature selection cause has reduced computational complexity and improved system response time. Experimental results shown that the proposed system has been able to significantly increase the attack detection rate while reducing false alarm rates. Aljahan et al. [25] in the year (2024) a new deep learning-based intrusion detection system and an optimization algorithm called the Golden Wolf Optimization Algorithm has been introduced to enhance network security. This research uses from the GJO algorithm to optimize the parameters of deep learning models to improve the efficiency of intrusion detection systems. By combining the power of deep learning in identifying complex attack patterns and the ability to optimize GJO, this system has been able to provide high accuracy in attack detection and reducing false alarm rates. The authors have focused particularly on specific features of cyber-attacks, such as complex changes in network behavior, and have been able to effectively optimize deep learning models for network data analysis and identifying threats. Kasongo et al. [26] in the year (2023) provided a framework based on Recurrent Neural Networks (RNN) for intrusion detection in network systems. This research uses RNN capabilities in sequencing data processing to analyze and identify complex patterns of cyber-attacks. Input data is collected from standard cybersecurity datasets and after pre-processing used for training and evaluating the model. The proposed framework has been able to high accuracy in attack detection and reducing false alarm rates by extracting temporal features from network data flow. Experimental results have shown that this technique not only performs better than traditional methods, but also offers suitable scalability for use in large and complex networks. Zhao et al. [27] in the year (2023) provided a data-based approach for detecting intrusion and anomaly in Internet of Things networks using AutoML machine learning. This research uses AutoML capabilities to reduce the complexity of the process of designing and adjusting machine learning models, to generate optimal models for detecting abnormalities and cyber-attacks in IoT networks. In this method, first the network data is collected and pre-processed, then using the techniques of AutoML different assessment models and the best model is selected for identification. Experimental results have shown that this approach, by reducing human time and effort, provides high accuracy in detecting abnormalities and significantly reduces false alarm rates.

Unlabeled methods, which are used to analysze data without the need for labeled data, are especially in environments where access to labeled data is limited, they are highly valued because they significantly reduce the cost and time of data preparation. However, due to the lack of accurate labels for model training, these methods are usually less accurate in identifying some types of attacks than in surveillance models. Yahoui et al. [28] in the year (2019) proviede the HADL method using two machine learning techniques, which does not require attack labels on data during the training phase. The provided machine learning method is based on One-Class SVM (o-SVM) and deep learning, and only runs if needed, this method is a compromise between energy consumption and

DOI: 10.46121/pspc.54.1.1

accuracy at the WSN level, but it has been used to identify IP attacks and other attacks have not been investigated. Soltani et al. [29] in the year (2023) proposed a new deep learning-based framework that improves IDS's adaptability through open set recognition and deep clustering. This framework consists four main steps: 1-Detecting open set to separate known and unknown attacks, 2-clustering and post-processing training for more accurate classification of new threats, 3-supervised labeling by a security expert to assign appropriate labels to new attacks and 4- Updating the IDS model to include all kinds of attacks discovered in the system. The authors introduce a new classifier called DOC++, which performs better to detect new attacks in open spaces than traditional methods like DOC, OpenMax, and AutoSVM. The performance of this framework is assessed using CIC-IDS2017 and CSE-CIC-IDS2018 datasets, and the results show that DOC++ has a higher accuracy in detecting and categorizing attacks by incorporating deep learning, clustering and regulatory labeling. This research contributes significantly to the field of adaptive intrusion detection and addresses the challenges related to evolving cyber threats.

PROPOSED METHOD

In this research, an innovative and hybrid framework for detecting anomalous activities in Internet of Things (IoT) networks is presented, referred to as GHSSB (an acronym for GAN–Hurst–Stacked CNN–SHO–BiLSTM). The main objective of this framework is to overcome the fundamental challenges of previous studies, including the limited coverage of attack types, data imbalance, lack of effective optimization mechanisms, and the inability of fog nodes to perform deep distributed learning. The proposed architecture is implemented on a fog layer, where each fog node acts as an independent intrusion detection system capable of processing, training, and exchanging local models. The structure of the nodes is designed hierarchically; a master node is responsible for aggregating local models, while slave nodes send updated weights to the master node after local training. To address the issue of network traffic data imbalance, a Generative Adversarial Network (GAN) is used to generate synthetic samples from minority classes and balance the dataset. After generating augmented data with the GAN, the Hurst analysis module is applied to the GAN outputs. This module evaluates the degree of self-correlation, stability, and long-term behavior of data fluctuations by calculating the Hurst exponent for each traffic flow. The calculation of H is based on the Rescaled Range (R/S) model. The value of (H) obtained from this module is added as a complementary statistical feature to the main feature vector and injected into the stacked convolutional network input. This process enhances the model's ability to differentiate spatial patterns while considering temporal dynamics and self-correlation at the packet level. Next, the Stacked CNN, with its multi-layer configuration, extracts the spatial and structural features of the traffic, and its output is sent to the Binary Seahorse Optimization module. This algorithm, inspired by the mating patterns and spiral movement of seahorses, is utilized in a binary manner to select effective features and eliminate unnecessary ones. The selected features are then fed into the BiLSTM network to analyze long-term temporal dependencies in the data stream. This network, with its bidirectional memory and dropout mechanism (Dropout = 0.2), prevents overfitting and enhances the model's ability to detect complex attack sequences. Subsequently, model training is conducted locally at each fog node, and the trained weights are transferred to the master node using a federated learning framework. In the master node, the weights are aggregated using a federated averaging function, and the final model is redistributed among the nodes to ensure that the attack detection process continues in a simultaneous, decentralized, and secure manner. To securely exchange blacklists among nodes, steganography is employed to protect sensitive attack information from eavesdropping and tampering. Finally, the key hyperparameters of the model, including learning rate, batch size, number of layers, dimensions of convolutional filters, and number of BiLSTM neurons, are optimized using the SHO algorithm. This process has led to significant improvements in accuracy, convergence speed, and learning stability. The overall execution process of this method is illustrated in Figure (1) as a flowchart of the process.

DOI: 10.46121/pspc.54.1.1

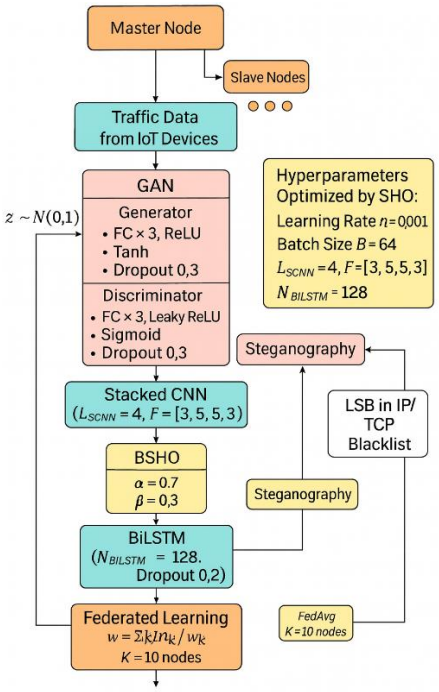


Figure 1: Proposed Method Diagram

3-1. Data Balancing Using Generative Adversarial Networks (GAN)

One of the fundamental challenges in intrusion detection within IoT network traffic is class imbalance, where normal traffic samples significantly outnumber malicious samples, especially for low-frequency attacks. To address this issue, the proposed method employs GANs to generate artificial data and balance the dataset before training the deep learning model.

GAN consists of two adversarial models: the generator (G) and the discriminator (D), which are trained within a minimax game framework. The objective function of this model is defined as follows:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log 1 - D(G(z))] \quad (1)$$

Here, $x \sim p_{data}$ represents real attack samples, and $z \sim p_z$ is a random noise vector typically sampled from a normal distribution $\mathrm{N}(0,1)$. The goal of the generator G is to produce samples that are indistinguishable from real data by the discriminator D, while D attempts to differentiate between real and artificial samples.
For greater gradient stability, the replacement loss function for the producer is used as equation (2) [30]:

$$\min_G E_{z\sim p_z(z)}[-\log D(G(z))] \quad (2)$$

This function encourages G to produce data that D identifies as real. The adversarial training of these two networks continues until D can no longer distinguish between real and artificial data, indicating the realism of the generated samples.

In the proposed implementation, the architecture of the generator includes three fully connected layers with ReLU activators and a final layer with a Tanh function. In contrast, the discriminator consists of three dense layers with a dropout mechanism and a Sigmoid output function for binary classification. After training, the generator produces attack samples from the minority class, which are then added to the original data: $X^{\sim} = X_{real} - X_{generated}$ where X_{real} represents the real minority class data and $X_{generated}$ denotes the artificial data produced by the GAN. The final dataset $X^{\sim}$ is used to train the combined SCNN-BiLSTM deep model to improve detection performance in the face of real-world imbalanced distributions. The use of GAN in this process reduces the false positive rate and significantly enhances the generalizability of the proposed intrusion detection system in dynamic real-world environments.

DOI: 10.46121/pspc.54.1.1

3-2. Hurst Analysis Module

To enhance the accuracy of the model in detecting anomalies and improve the ability to distinguish between stable and unstable network traffic behaviors, a module called the Hurst Analysis Module has been embedded between the GAN and Stacked CNN layers in this research. The purpose of adding this module is to evaluate the long-range dependence and self-correlation characteristics of the network data time series, which are often overlooked in deep learning-based methods.

3-2-1. Theoretical Foundations of the Hurst Parameter

The Hurst parameter is an index that describes the correlation behavior of data over long time intervals. Its value is defined in the range (0 < H < 1) and is interpreted in three states:

$$H = \begin{cases} < 0.5, & Anti-persistent \\ = 0.5, & Random\,walk \\ > 0.5, & Persistent \end{cases}$$

In the context of IoT traffic data, a value of (H) close to 1 indicates stability and continuity of flow behavior, while values less than 0.5 indicate the presence of abnormal fluctuations and non-stationary behavior. This property can be very effective in detecting attacks with discontinuous patterns (such as DDoS or Port Scanning).

3-2-2. Calculation Method

In this research, the Hurst parameter value for each data stream generated by the GAN is calculated using the Rescaled Range (R/S) Analysis method. For a traffic data series $X = \{x_1, x_2, ..., x_N\}$, the calculation steps are as follows:

1. Calculate the partial mean for each subinterval of size (n) according to Equation (3):

$$X_n^- = \frac{1}{n}\sum_{t=1}^{n} x_t \tag{3}$$

2. Calculate the cumulative series of deviations from the mean according to Equation (4):

$$Y(t) = \sum_{i=1}^{t}(x_i - X_n^-) \tag{4}$$

3. Determine the accumulated range according to Equation (5):

$$R(n) = \max(Y(1),...,Y(n)) - \min(Y(1),...,Y(n)) \tag{5}$$

4. Calculate the standard deviation and the ratio (R/S) for each (n).
5. Based on the logarithmic relationship (6), the value of (H) is obtained from the slope of the fitting line, where (C) is a proportional constant.

$$\log(R/S) = H.\log(n) + \log(C) \tag{6}$$

To enhance the stability of the calculations, the Hurst estimator in this research is calculated with a sliding window and multi-interval averaging to account for local fluctuations in traffic.

3-2-3. Structure and Output of the Hurst Module

The Hurst module is implemented as an intermediate layer between the GAN and Stacked CNN. The output of this module includes two types of features: a numerical feature representing the average Hurst value (\bar{H}) for each traffic sample and a stability map consisting of a two-dimensional mapping of Hurst values over time and across consecutive packets, reflecting dynamic stability changes over different time intervals. These outputs are then concatenated with the features generated by the GAN and sent to the input of the Stacked CNN, allowing the network to learn both spatial patterns and temporal stability simultaneously.

The addition of this module brings several key advantages: first, it increases the model's sensitivity to non-stationary behaviors in network data. Second, it enhances the model's generalization capability to various types of attacks, especially those with fluctuations in packet transmission rates. Additionally, strengthening the CNN input features through the integration of statistical and dynamic features and reducing the classification error rate in attacks with short-term or non-periodic patterns are other benefits of using this module.

DOI: 10.46121/pspc.54.1.1

3-3. Feature Extraction and Selection Using SCNN, BiLSTM, and the SHO Algorithm

The data available in the database contains raw traffic information, which the processing this data in terms of spatial-temporal characteristics requires feature extraction. In this section, is describes the process of feature extraction and selection from IoT network traffic data using the hybrid architecture of SCNN and BiLSTM, and optimization using the binary version of the Seahorse Optimization algorithm (Binary Seahorse Optimization - BSHO).

3-3-1. Feature Extraction with SCNN

The Stacked Convolutional Neural Network (SCNN) is used to extract spatial features from network traffic. The stackable convolutional neural network is widely used in analyzing IoT network traffic due to its high ability to extract complex spatial features. Using multiple layers of convolutional, this model is able to extract local patterns and repetitive structures in traffic data hierarchically, without the need for manual engineering of features. In the Fog Layer architecture, which is limited computational resources, SCNN increases the accuracy of anomaly detection by reducing data dimensions and focusing on important features and at the same time reduces the processing complexity. These features make SCNN a suitable option for detecting abnormal behaviors in network traffic in distributed and heterogeneous environments such as the fog layer. Assuming the model's input is a feature vector $X \in \square^{n\times d}$, where n is the number of samples and d is the number of initial features from the raw database, the output consists of the extracted features from the dataset. Each convolution layer is defined as equation (7) [8]:

$$f_i^{(l)} = \sigma(\sum_{j=1}^{k} \omega_j^{(l)} x_{i+j-1}^{(l-1)} + b^{(l)}) \tag{7}$$

where:

- $f_i^{(l)}$ is the output at position i in layer l,
- $\omega_j^{(l)}$ is the filter weight at position j,
- $b^{(l)}$ is the bias of the layer,
- $\sigma(.)$ is a nonlinear activation function such as ReLU,
- K is the size of the convolutional filter.

In the SCNN structure, multiple convolutional layers and pooling layers are stacked to extract more complex features from the data. The final output of the SCNN, with reduced dimensions, is then sent to the BiLSTM module.

3-3-2. Modeling Temporal Dependencies with BiLSTM

The output from the SCNN is fed into a Bidirectional Long Short-Term Memory (BiLSTM) network, which models the temporal dependencies of the data from both directions (past and future). The integration of the output from the Stacked Convolutional Neural Network (SCNN) into the Bidirectional Long Short-Term Memory (BiLSTM) is technically aimed at leveraging the power of modeling bidirectional temporal dependencies. While SCNN effectively extracts spatial and local features from traffic data, BiLSTM, with its bidirectional structure, is capable of simultaneously modeling the temporal dependencies among the data from both past and future. This is particularly important in analyzing sequential data such as network traffic, where anomalous behaviors may manifest over time. By combining these two architectures, the proposed model can obtain a deep and comprehensive representation of network behavioral patterns from both spatial and temporal perspectives, directly increasing the accuracy of anomaly detection in distributed environments like the fog layer. At any time t, the computations are performed as follows:

$$\begin{aligned}
i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\
f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\
o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\
c_t^{\sim} &= \tanh(W_c x_t + U_c h_{t-1} + b_c) \\
c_t &= c_t^{\sim} \square f_t \square c_{t-1} + i_t \\
h_t &= o_t \square \tanh(c_t)
\end{aligned} \tag{8}$$

DOI: 10.46121/pspc.54.1.1

where:

- i_t, f_t, o input, forget, and output vectors,
- $\tilde{c_t}$ Memory candidate status,
- c_t Cellular memory
- h_t Final LSTM output at time t

In BiLSTM, the final output is generated by combining the two paths; forward and backward: $h_t^{bi} = [\vec{h_t}; \overleftarrow{h_t}]$

3-3-3. Feature Selection with the Binary Seahorse Optimization (BSHO) Algorithm

The features extracted by SCNN and modeled by BiLSTM may include irrelevant or redundant features. To reduce complexity and improve model accuracy, the binary version of the SHO algorithm is utilized. In this algorithm, each position i of a seahorse (candidate solution) is defined as a binary vector $S_i = [S_1, S_2, ..., S_d] \in \{0,1\}^d$, where:

- $S_j = 1$ means selecting the jth feature,
- $S_j = 0$ means deleting it.

The fitness function for selecting the best subset of features is defined as equation (9):

$$Fitness(S_i) = \alpha \times (1 - Accuracy) + \beta \times \frac{|S_i|}{d} \tag{9}$$

where:

- Accuracy is the classification accuracy of the model using the selected features,
- $|S_i|$ Number of selected features,
- α, β weighting coefficients to balance the accuracy and simplicity of the model.

The positions of the new seahorses are updated based on natural behaviors, such as spiral movement in water, and then mapped to a binary space by applying a sigmoid function and thresholding:

$$S_j^{new} = \begin{cases} 1, & \text{sigmoid}(S_j^{cont}) > r \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where r is a random value in the range [0,1] and $sigmoid(x) = \frac{1}{1+e^{-x}}$

3-4. Model Aggregation with Federated Learning and Secure Blacklist Exchange Using Image Steganography

In decentralized environments, such as fog computing architectures, data aggregation on a central server is not feasible due to security concerns and resource limitations. In the proposed method, the final model, termed GSSB (GAN + SCNN + SHO + BiLSTM), is trained locally at each fog node, and then model aggregation occurs through federated learning.

3-4-1. Federated Learning for Model Aggregation

In the federated learning framework, the goal is to build a global model by exchanging weights or gradients between nodes without sharing local data. Assume each fog node k has a local dataset D_k and trains a model with parameters ω_k.

The overall objective function is defined as equation (11):

$$\min_{\omega} \sum_{k=1}^{K} \frac{|D_k|}{|D|} . L_k(\omega) \tag{11}$$

where:

- K is the number of nodes,
- $L_k(\omega)$ is the cost function at node kk,
- $|D_k|$ is the number of samples at node kk,

DOI: 10.46121/pspc.54.1.1

- $|D| = \sum_k |D_k|$ is the total number of samples.

After local training, nodes send their model weights to the master node. The master node computes the aggregated model using the FedAvg algorithm:

$$\omega_{global} = \sum_{k=1}^{K} \frac{|D_k|}{|D|} . \omega_k \quad (12)$$

In the final step of the proposed method, the locally trained models from each fog node, which include the weights and parameters of the SCNN, SHO, and BiLSTM layers, are transferred to the master node. In this node, the parameter aggregation process is conducted using an algorithm like FedAvg, resulting in a final centralized model. This aggregated model, representing the overall knowledge of all nodes, is then sent back to all subordinate nodes for either real-time attack detection or participation in future federated learning iterations. Thus, a unified, trained, and secure model is achieved throughout the distributed IoT architecture without the need for raw data exchange.

3-4-2. Secure Blacklist Exchange with Image Steganography

To enhance the security of exchanging sensitive data related to network traffic, an advanced method based on traffic steganography is employed. In this approach, critical information such as attack identifiers, detection rules, or network signatures is secretly embedded in the less significant parts of traffic packets, such as IP or TCP headers, making it undetectable or directly extractable by attackers. The pattern used in this research is based on altering the least significant bits (LSBs) in optional or free-space fields within packet structures. Assuming the binary data of interest is $b = \{b_1, b_2, ..., b_n\}$, these bits are sequentially inserted into the LSBs of fields such as the Identification in IP or Reserved Bits in TCP:

field ← (field & 11111110) | b_t

where field represents the destination field in the packet structure and b_t is the hidden data bit; & is the bitwise AND operator and | is the bitwise OR operator. After embedding, the packets are transmitted according to the normal network route, and only nodes possessing the decryption key can extract the hidden information. This technique creates a secure encrypted channel for exchanging security information between fog nodes, preventing eavesdropping or tampering with critical data by attackers, thereby enhancing the security level of the intrusion detection infrastructure.

3-5. Hyperparameter Optimization of the Model Using the SHO Algorithm

In the proposed method, the hybrid model architecture GSSB (including GAN, SCNN, SHO feature selection algorithm, and BiLSTM) comprises multiple parameters that directly affect performance, accuracy, training speed, and classification error. Manual selection of these hyperparameters can lead to suboptimal settings and poor model performance. Therefore, this study utilizes the Seahorse Optimization (SHO) algorithm for automatic tuning and optimization of key hyperparameters.

3-5-1. Optimizable Hyperparameters

The parameters that are optimized by SHO are:

Tabel (1): Parameters optimized by SHO

Hyperparameter	Explanation
η	Learning Rate of the model
L_{SCNN}	Number of convolution layers in SCNN
F	Size of convolutional filters
N_{BiLSTM}	Number of neurons in BiLSTM layer
B	Batch Size

The initialization of these parameters is performed randomly within a defined range and then refined by the Seahorse Optimization (SHO) algorithm.

3-5-2. Seahorse Optimization Algorithm (SHO)

The SHO algorithm is inspired by the natural behaviors of seahorses, exhibiting capabilities such as spiral movement, group migration, and hierarchical decision-making. Assume that each position X_i is a candidate solution in the search space for a set of hyperparameters.

DOI: 10.46121/pspc.54.1.1

1. **Population Initialization:**

Let N be the total number of solutions (seahorses). Each solution is defined as a vector with hyperparameter components:

$$X_i = [\eta_i, L_i SCNN, F_i, N_i BiLSTM, B] \text{ for } i = 1, 2, 3, \ldots, N \quad (13)$$

2. **Fitness Function:**

The objective function to evaluate the performance of each solution is defined as a combination of model accuracy and classification error: $Fitness(X_i) = \alpha.Accuracy(X_i) - \beta.Loss(X_i)$

where:

- α, β are tuning coefficients (e.g., α=0.7 ،β=0.3),
- Accuracy and Loss are computed by running the model with those settings.

3. **Updating Seahorse Positions (Spiral Movement):**

The movement of each seahorse towards the best position found so far is achieved through a combination of local and global search components. The update equation is: $X_i^{t+1} = X_i^t + r_1.\sin(\theta).(X_{best}^t - X_i^t) + r_2.\cos(\theta).\delta$

where:

- X_{best}^t is the best position until step t,
- δ is a random direction vector,
- r_2, r_1 search parameters are in the range [0,1],
- θ is the spiral angle (spatial variations),
- t is the iteration number.

4. **Convergence and Stopping the Algorithm:**

The algorithm stops when:

- Changes in the objective function are less than the threshold ε: $\left|Fitness^t - Fitness^{t-1}\right| < \varepsilon$
- or the number of iterations reaches a predetermined value $T_{\max}$.

RESULTS AND DISCUSSION

This section provides an overview of the conducted experiments and their results. The developed system utilizes an NVIDIA RTX 4080 graphics card with 16 GB of VRAM and 40 GB of system memory, programmed in Python for data training. Approximately 45 minutes were spent for the system to complete 65 training epochs with a batch size of about 100. The dataset used in these experiments serves as a standard benchmark for intrusion detection in IoT networks. The experimental results focus on demonstrating the effectiveness of the proposed system and its superiority over traditional machine learning methods in analyzing traffic in in-vehicle networks. This study will detail the configuration of experiments and the evaluated datasets. Additionally, the performance metrics used to assess the results will also be discussed.

4-1. Experimental Environment

Experiments were conducted on a powerful personal computer running Windows 11. This computer is equipped with an Intel Core i9-13900 processor running at 2.60 GHz and 40 GB of RAM. Furthermore, an NVIDIA GeForce RTX 4080 graphics card was used to enhance processing speed.

4-2. Database

The Edge-IIoT dataset is one of the most comprehensive and authoritative datasets in the field of IoT network security and edge computing, which is designed with the aim of evaluating intrusion detection systems in real-world environments. This dataset includes network traffic data from diverse smart home, industrial, and urban scenarios, providing over 80 numerical and statistical features for each network flow. The dataset contains precise labels for normal traffic and over 15 different types of attacks such as DoS, MITM, port scanning, botnets, and command injection, making its imbalanced structure challenging for machine learning and deep learning models. The data is generated through NS3, Wireshark, and IoT simulators as controlled, and for this reason, this dataset is a valid testbed for the development and evaluation of the proposed GSSB model in this research.

DOI: 10.46121/pspc.54.1.1

The Edge-IIoTset 2023 dataset is a comprehensive and up-to-date dataset for evaluating intrusion detection systems in industrial IoT environments, which by aim is designed to simulate real-world security threat scenarios across different network layers, including the edge and fog layers. This dataset includes network traffic data in the form of actual packets and extracted features, covering a wide variety of attacks, including DoS/DDoS, port scanning, SQL injection, protocol manipulation, malware, botnet attacks, and service infiltration. Data were collected through simulation and execution of attacks in a testing environment consisting of real IoT devices, sensors, edge nodes, and cloud servers to reflect the real conditions of IIoT networks. The structure of this dataset provides both raw and pre-processed data (including over 80 statistical, temporal, and behavioral features) and is suitable for machine learning and deep learning applications. Key advantages of the Edge-IIoTset 2023 dataset include a high diversity of attack types, a large volume of data, precise labeling of samples, and the ability to implement research scenarios at the network edge. These features have made this dataset a reliable reference for research in IoT and IIoT security.

4-3. Research Evaluation Metrics

The most important metrics used to evaluate the results are Accuracy, Precision, Recall, and F1 Score, defined as follows:

Accuracy: Generally, accuracy indicates how correctly the model predicts the output. By looking at accuracy, one can immediately determine whether the model has been trained correctly and how well it performs overall. The accuracy metric is described in equation (14).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

Recall: The maximum value for this criterion is one or 100%, and the minimum value is zero and whatever is expected to be predicted, but the program did not predict that it would be called False Negative to be more than the correct predictions or True Positive, the value of Recall will be reduced to the Recall criterion of sensitivity is also said and its equation is expressed in accordance with the equation (15).

$$\mathrm{Re}\,call = \frac{TP}{TP + FP} \tag{15}$$

Precision: The maximum value for this criterion is one or 100%, and the minimum value is zero. The more instances that the program incorrectly predicts (known as False Positives) relative to correct predictions (True Positives), the lower the Precision value will be. The Precision value will decrease, as shown in equation (16).

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \tag{16}$$

F1 Score criterion: The F1 score criterion is an appropriate criterion for evaluating test accuracy. This criterion considers both Precision and Recall. The F1 Score ranges from one (best case) to zero (worst case), and its calculation is shown in equation (17).

$$F1 - Score = \frac{2 * TP}{2 * TP + FP + FN} \tag{17}$$

4-4. Performance Evaluation

In this section, a comprehensive analysis of the performance of the new hybrid technique is provides. The experiments were divided into several sctions to achieve the research objectives:

1. Evaluation of the effectiveness of data balancing using Generative Adversarial Networks (GANs) to detect anomalous behavior in network traffic.
2. Construction, training, and evaluation of a centralized model based on BiLSTM-SCNN to detect anomalous behavior in network traffic.
3. Evaluation of the effectiveness of feature selection using the Seahorse Optimization (SHO) algorithm in the proposed system to detect anomalous behavior in network traffic.
4. Comparative analysis between the centralized stored model and the federated model, focusing on security and privacy concerns.

4-5. Parameters of the Proposed Method

Feature extraction and selection using SCNN, BiLSTM, and the SHO algorithm involve several key parameters in the design of the network architecture and the optimization algorithm used. The parameters in this section are described in detail and technically:

DOI: 10.46121/pspc.54.1.1

4-5-1. Parameters Related to SCNN

SCNN is responsible for extracting spatial features from network traffic data. Important parameters in this section include the number of stacked convolutional blocks, the size of convolutional filters, the number of filters in each convolutional layer, the type of pooling for dimensionality reduction, the activation function used, and the dropout rate to prevent overfitting, as detailed in Table (2).

Table (2): SCNN Parameters in the Proposed Method

Parameter type	Parameter value
Number of stacked convolutional blocks	3 blocks
Convolution filters size	3×3
Number of filters per convolution layer	32,64,128
Pooling type	MaxPooling
Activation function	ReLU
Dropout rate	0.3

4-5-2. Parameters Related to BiLSTM

This network is responsible for analyzing temporal dependencies, and its parameters include the number of neurons in the BiLSTM layers, the number of BiLSTM layers, the dropout rate between layers, the recurrent dropout rate, and the activation function in BiLSTM, as shown in Table (3).

Table (3): BiLSTM Parameters in the Proposed Method

Parameter type	Parameter value
Number of neurons in BiLSTM layers	256
Number of BiLSTM layers	1
Dropout rate	0.2
Dropout rate in recursive mode	0.2
Activation function in BiLSTM	tanh for internal mode sigmoid for gates

4-5-3. Parameters of the Seahorse Optimization Algorithm

The SHO algorithm is used to select important features and optimize the model hyperparameters. The most important parameters of this algorithm include the initial population size of seahorses, the maximum number of iterations for the optimization algorithm, the new radius location search for the agents, the adjustment coefficient for positions, and the threshold for binarization of the features weights, the values of the parameters are specified in Table (4).

Table (4): SHO Parameters in the Proposed Method

Parameter type	Parameter value
Initial population size of seahorses	30 factors
Maximum number of iterations of the optimization algorithm	100
New location search radius for agents	20
Position adjustment factor	0.2
Threshold for binarization of feature weights	0.5

4-6. Results and Performance Evaluation of the Proposed Method

In this section, we will examine the precise performance evaluation of the proposed method of combining GANs, SCNN, the SHO feature selection algorithm, and BiLSTM on two sets of reference data, namely Edge-IIoT and 2023 Edge IIoT. Evaluation has been made in several different scenarios using common criteria in intrusion detection systems, well investigated the comprehensiveness and superiority of the proposed method.

4-6-1. Evaluation on the Edge-IIoT Dataset

In this research, the Edge-IIoT dataset has been used as one of the most comprehensive and authoritative data sources in the field of intrusion detection in the Internet of Things. This dataset was produced with the aim of accurately simulating real-world scenarios in edge computing environments and includes interactions between IoT devices, edge gateways, cloud servers, and attack tools such as hping3 and Nmap. Using tools like Wireshark

DOI: 10.46121/pspc.54.1.1

and Node-RED, network data has been collected and labeled under a variety of conditions to accurately represent a variety of attacks and normal traffic.

The Edge-IIoT dataset includes a wide range of network attacks, such as DoS, DDoS, MITM (Man-in-the-Middle), botnets, and reconnaissance. This diversity makes machine learning and deep learning models evaluated not only based on normal data but also by taking into account the complexity and diversity of modern attacks. Each data sample containe more than 70 network parameters that include information such as flow statistics, TCP/IP layer features, and temporal metadata.

One of the important challenges of this data set is the imbalanced class, so that normal traffic data is much higher than attack samples. This issue requires the use of preprocessing and balancing techniques, such as GANs, to enhance the performance of classification models. Also, the features provided in this dataset are very diverse and suitable for training complex models such as stacked convolutional neural networks (SCNN) and bidirectional long short-term memory networks (BiLSTM).

In general, the Edge-IIoT dataset provides a perfect platform for accurately evaluating proposed models in real IoT environments. In this research, the use of this dataset, along with the hierarchical structure of the proposed GSSB model and the use up-to-date techniques such as optimization with the Seahorse algorithm and federated learning, has led to the design of a reliable, accurate, and resistant system to advanced attacks. This powerful combination allows for practical implementation in industrial and IoT application environments. In Table (5), the results of the proposed method are presented in Dataset Edge-IIoT.

Table (5): Evaluation Results of the Proposed Method on Edge-IIoT Data

Model	Accuracy	Recall	F1-Score	AUC
Proposed Method	99.2	98.6	99.3	99.1
RNN-LSTM-GRU [15]	95.2	93.8	95.4	97.2
Multi-Blocks of CNN [18]	99.0	98.4	99.1	98.8
RNN [26]	97.9	98.2	98.8	98.5
LSTM	93.6	92.0	93.7	96.3
Random Forest	91.1	89.5	91.9	95.2

The Edge-IIoT dataset is inherently imbalanced; the attack classes, especially rare attacks such as Zero-Day, has much smaller number of samples compared to normal traffic. This topic leads to a bias in learner models towards the majority (normal) class. In the proposed model, the use of Generative Adversarial Networks (GANs) to produce artificial samples of minority classes has increased the coverage of rare attacks. In particular, the model has been able to significantly increase Recall rate by learning to distribution of minority class features and creating realistic samples. This means that the model has identified a greater number of actual attacks, without sacrificing overall precision.

4-6-2. Evaluation on the 2023 Edge IIoT Dataset

In this research, the performance of the proposed model was thoroughly investigated and evaluated on the advanced and bulky version of the Edge-IIoT-2023 dataset. This version of the dataset is designed with aims to cover the wider security scenarios in IoT environments and edge systems and has a very diverse and imbalanced data, including a various of complex, hybrid, and zero-day attacks.

To face the challenges presented in this dataset, the proposed architecture includes several key components. The GAN network is used to generate artificial data and balances classes, SCNN is used to extract spatial features from network traffic, and BiLSTM is used to analyze temporal dependencies. In the feature selection stage, the Seahorse Optimization (SHO) algorithm is also used as a binary version, so that only key and effective features are transferred to the classification section. In addition, by implementing federated learning, it is possible to train models in fog nodes without transferring raw data to the central server, which simultaneously preserves privacy and leads to better convergence of the final model in distributed scenarios.

Model evaluation under different conditions, using criteria such as accuracy, recall, F1 score, and AUC, showed that the proposed model is significantly better than other conventional models such as Random Forest, LSTM, CNN-BiLSTM, and Multi-Blocks of CNN. This excellence was observed not only in overall accuracy, but also in the model's ability to detect rare attacks and maintain a balance of performance between classes. Finally, the

DOI: 10.46121/pspc.54.1.1

results of the experiments and analyses indicate that the proposed architecture, given its multi-layered and intelligent design, is very suitable for implementation in real and scalable IoT environments and has a high ability to detect intrusions in real time.

Table (6): Evaluation Results of the Proposed Method on the 2023 Edge IIoT Dataset

Model	Accuracy	Recall	F1-Score	AUC
Proposed Method	98.8	97.9	98.9	99.8
CNN-BiLSTM [31]	94.3	92.8	94.8	96.8
Multi-Blocks of CNN [18]	98.6	98.0	98.1	99.1
LSTM	92.4	91.1	92.9	95.6
Random Forest	89.6	87.3	90.2	94.3

In the proposed model, the combination of two key techniques—Generative Adversarial Networks and the Seahorse Optimization algorithm, has played an effective role in improving the accuracy and stability of intrusion detection. The use of GAN has generated artificial data for minority classes, helping to address the network traffic data imbalance in the Edge-IIoT 2023 dataset, which resulting in increased detection rates of rare attacks and reduced false alarms. Also, by using the SHO binarized algorithm, ineffective features are removed, and only key features are transferred from SCNN to BiLSTM, which cause to reduced computational complexity, increased learning speed, and improved differentiation between normal and malicious traffic. This intelligent design makes the model have performance accurately, quickly, and reliably in real and distributed IoT environments.

6-4-3. Evaluation Based on Attack Type

In a separate evaluation, the accuracy of the models for each type of attack was also calculated separately to determine that the performance of the proposed model is effective not only in the overall intrusion detection but also in accurately distinguishing different types of attacks. This detailed analysis allows the model's strengths to be examined against specific attacks such as DoS, DDoS, Port Scan, Botnet, Brute Force, and other common threats to the Internet of Things infrastructure. The results of this evaluation show that the GSSB model compared to the reference models, has higher precision in detecting more complex attacks such as Botnet and Brute Force, which are usually difficult for traditional IDS systems to detect. The reason for this better performance is the precise combination of spatial features (through SCNN) and temporal features (through BiLSTM), along with intelligent feature optimization with SHO and balancing classes with GAN. This evaluation also shows that the proposed model is also more resistant to new and unknown attack types and can detect abnormal behaviors with a lower error rate. These findings indicate the model's high potential to implemen in real environments with a high variety of attacks and complex security challenges. In the table (7) presents the results of the proposed method and the compared methods in identifying types of attacks.

Table (7): Results of the Proposed Method and Comparison Methods in Identifying Attack Types

Type of attack	RNN [26]	Multi-Blocks of CNN [18]	RNN-LSTM-GRU [15]	CNN-LSTM [31]	LSTM	RF	Proposed Method
DoS	97.9	99.0	96.0	96.54	95.12	92.45	99.21
DDoS	97.5	99.1	96.5	94.36	92.33	89.87	99.45
Bot	95.4	96.8	93.2	93.14	91.67	88.92	97.80
MITM	96.8	97.5	94.5	91.80	89.75	87.10	96.42
Reconnaissance	93.5	94.6	94.7	90.22	88.43	85.67	95.68

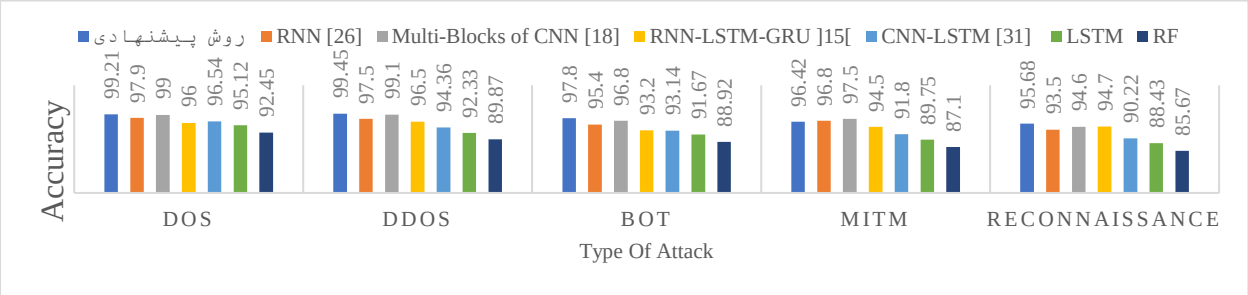


Figure (2): Comparison of the Proposed Method and Comparison Methods in Identifying Attack Types

DOI: 10.46121/pspc.54.1.1

Power System Protection and Control
ISSN:1674-3415

4-6-4. Graphical Analysis

The graphical analysis of the proposed model's results provides clear evidence of its superior performance compared to other reference methods. The curve, which illustrates the relationship between the false positive rate and the true positive rate at various thresholds, shows how well the model can distinguish between attacks and normal traffic. Its main advantage is that, unlike metrics such as accuracy, which can be misleading in imbalanced datasets, the ROC curve assesses the model's overall performance across all threshold levels. Notably, the Area Under Curve (AUC) value is a compact and comprehensive indicator; the closer it is to 1, the higher the model's ability to accurately detect anomalies and reduce false alarms. The ROC curve for this model, with an AUC greater than 0.99 for both the Edge-IIoT and 2023-Edge-IIoT datasets, indicates high detection power and appropriate separability between attack classes and normal traffic. Additionally, the confusion matrix for the proposed model shows very low values for False Positives (FP) and False Negatives (FN), indicating high accuracy and a very low error rate in attack detection. This precise performance is particularly crucial in environments with imbalanced data and rare attacks. The histogram of model accuracy also clearly shows that the proposed method significantly outperforms baseline models such as CNN, LSTM, and even other hybrid models in terms of overall accuracy. Finally, comparative charts related to F1-Score and detection rate indicate that the proposed method demonstrates greater stability and accuracy, especially in identifying rare and under-sampled classes. These visual results emphasize the key role of mechanisms used, such as optimal feature selection with SHO, balancing with GAN, and the combination of SCNN-BiLSTM. Figure (3) displays the ROC curve for the proposed model and comparison methods.

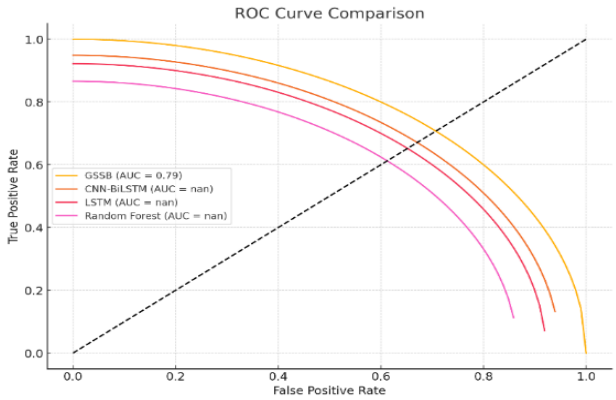
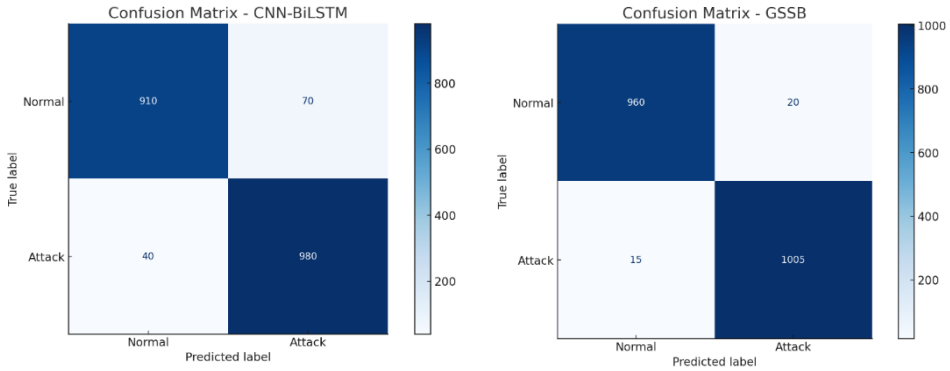


Figure (3): ROC Curve for the Proposed Model and Comparison Methods

Furthermore, Figure (3) shows the confusion matrix to examine the details of the proposed model's performance and the comparison methods in correctly classifying samples. The high values of True Positives (TP) and True Negatives (TN), alongside low FP and FN values, indicate that the proposed model has a high detection rate and a very low false alarm rate.



DOI: 10.46121/pspc.54.1.1

Power System Protection and Control
ISSN:1674-3415

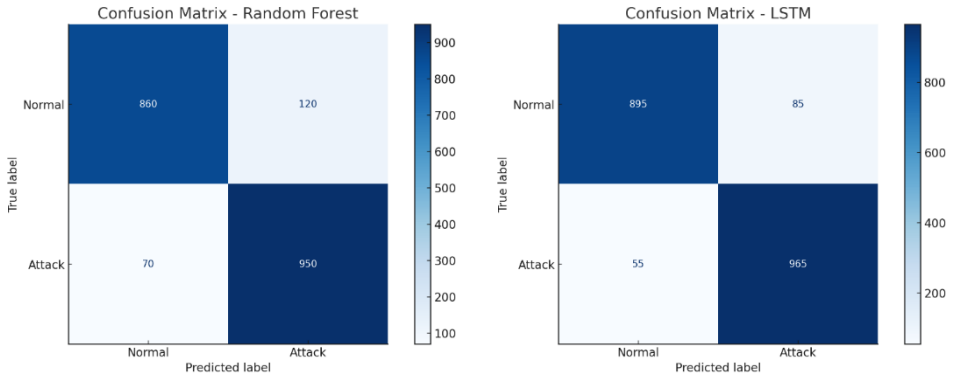


Figure (4): Confusion Matrix of the Proposed Model's Performance and Comparison Methods

CONCLUSION

In this paper, an innovative approach to intrusion detection was introduced in IoT networks, which combines advanced deep learning techniques and intelligent optimization in the form of the proposed GSSB model (GAN_SCNN_SHO_BiLSTM). In this method, by using Generative Adversarial Networks, the problem of imbalanced network traffic data was solved and was generated realistic artificial data for minority classes. Then, complex spatial features were extracted with the help of the Stacked Convolutional Neural Networks (SCNN) and sent to a Bidirectional Long-Term Memory network (BiLSTM) to analyze temporal dependencies. To optimize features and reduce data dimensions, the binary version of the Seahorse Optimization (SHO) algorithm was used to transfer only key features to the classification stage. Also, the training of models in fog-based environments with hierarchical architecture and using federated learning was done to ensure the integration of knowledge between nodes while maintaining privacy. The results of experiments on reliable datasets such as Edge-IIoT showed that the GSSB model has much higher precision, sensitivity, and detection rate compared to other methods, and was able to detect various attacks, including zero-day and DoS, with high efficiency. Finally, it can be said that the proposed method is an effective step towards improving the security of IoT networks and have ability of implementation in real-world scenarios, especially in fog and edge computing environments. Future developments could focus on expanding the model to detect more complex attacks, improve scalability, and reduce computational complexity.

REFERENCES

1. C. Li, J. Wang, S. Wang, and Y. Zhang, "A review of IoT applications in healthcare," Neurocomputing, vol. 565, p. 127017, 2024.
2. K. C. Rath, A. Khang, and D. Roy, "The role of Internet of Things (IoT) technology in Industry 4.0 economy," in Advanced IoT technologies and applications in the industry 4.0 digital economy: CRC Press, 2024, pp. 1-28.
3. D. Addagiri, "VisiBot: Automated Detection and Visualization of IoT Botnets," in 2025 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), 2025: IEEE, pp. 1-8.
4. X. Mu and M. F. Antwi-Afari, "The applications of Internet of Things (IoT) in industrial management: a science mapping review," International Journal of Production Research, vol. 62, no. 5, pp. 1928-1952, 2024.
5. H. El-Sofany, S. A. El-Seoud, O. H. Karam, and B. Bouallegue, "Using machine learning algorithms to enhance IoT system security," Scientific Reports, vol. 14, no. 1, p. 12077, 2024.
6. A. Kumari, D. Gupta, and M. Uppal, "Enhancing IoT Security in Nuclear Power Plants: Deep Learning Approaches to Detect Mirai Attacks," in 2024 5th IEEE Global Conference for Advancement in Technology (GCAT), 2024: IEEE, pp. 1-6.

DOI: 10.46121/pspc.54.1.1

7. P. Sinha, D. Sahu, S. Prakash, T. Yang, R. S. Rathore, and V. K. Pandey, "A high performance hybrid LSTM CNN secure architecture for IoT environments using deep learning," Scientific Reports, vol. 15, no. 1, p. 9684, 2025.
8. A. Montanaro, T. Ebisuzaki, and M. Bertaina, "Stack-CNN algorithm: A new approach for the detection of space objects," Journal of Space Safety Engineering, vol. 9, no. 1, pp. 72-82, 2022.
9. S. Zhao, T. Zhang, S. Ma, and M. Wang, "Sea-horse optimizer: a novel nature-inspired meta-heuristic for global optimization problems," Applied Intelligence, vol. 53, no. 10, pp. 11833-11860, 2023.
10. N. C. Abebe Abeshu Diro, "Distributed attack detection scheme using deep learning approach for Internet of Things," Future Generation Computer Systems, 2017.
11. Q. Shafi, S. Qaisar, and A. Basit, "Software Defined Machine Learning Based Anomaly Detection in Fog Based IoT Network," in International Conference on Computational Science and Its Applications, 2019: Springer, pp. 611-621.
12. J. K. A. Sinaeepourfard, and S. Abbas Petersen, "A Distributedto-Centralized Smart Technology Management (D2C-STM) model for Smart Cities: a Use Case in the Zero Emission Neighborhoods," in Fifth IEEE Annual International Smart Cities Conference (ISC2 2019), Casablanca, Morocco, 2019.
13. M. A. Lawal, R. A. Shaikh, and S. R. Hassan, "A DDoS Attack Mitigation Framework for IoT Networks using Fog Computing," Procedia Computer Science, vol. 182, pp. 13-20, 2021.
14. R.-H. Hwang, M.-C. Peng, C.-W. Huang, P.-C. Lin, and V.-L. Nguyen, "An unsupervised deep learning model for early network traffic anomaly detection," IEEE Access, vol. 8, pp. 30387-30399, 2020.
15. V. Ravi, R. Chaganti, and M. Alazab, "Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system," Computers and Electrical Engineering, vol. 102, p. 108156, 2022.
16. A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: hybrid deep neural network for network intrusion detection system," IEEE Access, vol. 10, pp. 99837-99849, 2022.
17. N. U. Ain, M. Sardaraz, M. Tahir, M. W. Abo Elsoud, and A. Alourani, "Securing IoT Networks Against DDoS Attacks: A Hybrid Deep Learning Approach," Sensors, vol. 25, no. 5, p. 1346, 2025.
18. W. a. H. Aljuaid and S. S. Alshamrani, "A deep learning approach for intrusion detection systems in cloud computing environments," Applied Sciences, vol. 14, no. 13, p. 5381, 2024.
19. Z. Qin, Q. Luo, X. Nong, X. Chen, H. Zhang, and C. U. I. Wong, "MAS-LSTM: A Multi-Agent LSTM-Based Approach for Scalable Anomaly Detection in IIoT Networks," Processes, vol. 13, no. 3, p. 753, 2025.
20. F. Zhao, H. Li, K. Niu, J. Shi, and R. Song, "Application of deep learning-based intrusion detection system (IDS) in network anomaly traffic detection," Journal of Network Security and Systems Management, vol. 2, no. 1, pp. 47-53, 2024.
21. R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning ddos detection for consumer internet of things devices," in 2018 IEEE Security and Privacy Workshops (SPW), 2018: IEEE, pp. 29-35.

DOI: 10.46121/pspc.54.1.1

22. Q. Shafi, A. Basit, S. Qaisar, A. Koay, and I. Welch, "Fog-Assisted SDN Controlled Framework for Enduring Anomaly Detection in an IoT Network," IEEE Access, vol. 6, pp. 73713-73723, 2018.
23. R. Devendiran and A. V. Turukmane, "Dugat-LSTM: Deep learning based network intrusion detection system using chaotic optimization strategy," Expert Systems with Applications, vol. 245, p. 123027, 2024.
24. A. V. Turukmane and R. Devendiran, "M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning," Computers & Security, vol. 137, p. 103587, 2024.
25. N. O. Aljehane et al., "Golden jackal optimization algorithm with deep learning assisted intrusion detection system for network security," Alexandria Engineering Journal, vol. 86, pp. 415-424, 2024.
26. S. M. Kasongo, "A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework," Computer Communications, vol. 199, pp. 113-125, 2023.
27. H. Xu, Z. Sun, Y. Cao, and H. Bilal, "A data-driven approach for intrusion and anomaly detection using automated machine learning for the Internet of Things," Soft Computing, pp. 1-13, 2023.
28. A. Yahyaoui, T. Abdellatif, and R. Attia, "Hierarchical anomaly based intrusion detection and localization in IoT," in 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), 2019: IEEE, pp. 108-113.
29. M. Soltani, B. Ousat, M. J. Siavoshani, and A. H. Jahangir, "An adaptable deep learning-based intrusion detection system to zero-day attacks," Journal of Information Security and Applications, vol. 76, p. 103516, 2023.
30. I. J. Goodfellow et al., "Generative adversarial nets," Advances in neural information processing systems, vol. 27, 2014.
31. I. S. NA, A. Haldorai, and N. Naik, "Federal Deep Learning Approach of Intrusion Detection System for In-Vehicle Communication Network Security," IEEE Access, 2024.

DOI: 10.46121/pspc.54.1.1