Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.026571
Article

Tech Science Press

A Novel MegaBAT Optimized Intelligent Intrusion Detection System in Wireless Sensor Networks

G. Nagalalli* and G. Ravi

Department of Electronics and Comunication Engineering, Sona College of Technology, Salem, 636005, India
*Corresponding Author: G. Nagalalli. Email: nagalalli87@gmail.com
Received: 30 December 2021; Accepted: 01 March 2022

Abstract: Wireless Sensor Network (WSN), which finds as one of the major components of modern electronic and wireless systems. A WSN consists of numerous sensor nodes for the discovery of sensor networks to leverage features like data sensing, data processing, and communication. In the field of medical health care, these network plays a very vital role in transmitting highly sensitive data from different geographic regions and collecting this information by the respective network. But the fear of different attacks on health care data typically increases day by day. In a very short period, these attacks may cause adversarial effects to the WSN nodes. Furthermore, the existing Intrusion Detection System (IDS) suffers from the drawbacks of limited resources, low detection rate, and high computational overhead and also increases the false alarm rates in detecting the different attacks. Given the above-mentioned problems, this paper proposes the novel MegaBAT optimized Long Short Term Memory (MBOLT)-IDS for WSNs for the effective detection of different attacks. In the proposed framework, hyperparameters of deep Long Short-Term Memory (LSTM) were optimized by the meta-heuristic megabat algorithm to obtain a low computational overhead and high performance. The experimentations have been carried out using (Wireless Sensor NetworkDetection System)WSN-DS datasets and performance metrics such as accuracy, recall, precision, specificity, and F1-score are calculated and compared with the other existing intelligent IDS. The proposed framework provides outstanding results in detecting the black hole, gray hole, scheduling, flooding attacks and significantly reduces the time complexity, which makes this system suitable for resource-constraint WSNs.

Keywords: Wireless sensor network; intrusion detection systems; long short term memory; megabat optimization

1 Introduction

The wireless sensor networks (WSNs) and Internet of Things (IoT) are a group of networks that employ reasonable low resources nodes that can perceive copious applications such as health care [1], consumer electronics [2], and even image transmission [3]. Due to the massive development of wireless networks and deployment of wireless devices connected to these heterogeneous networks are

vulnerable to miscellaneous malicious attacks. The enactment of IDS can acclimate with those challenges is of uttermost significance. The system collects data from this wireless heterogeneous network and performs the analysis for the detection of anomalous behavior in the wireless network [4].

In recent days, IDS evolved with the combination of various applications such as Artificial Intelligence (AI) especially Machine Learning (ML) and Deep Learning (DL). The IDS inherit the application can be used for both attacking and safeguarding the IP-enabled wireless network. However, these specialized algorithms are maneuvered for defense mechanisms and endure against the security menaces for the sake of minimizing the impacts or adversity of accidents occurred in the WSN. The numerous ML and DL-based IDS are developed for intrusion detection [5–7], malware detection [8–11], cyber-physical attacks [12–14], and data privacy protection [14].

ML algorithms are predominantly utilized for building error-free models that are specially designed and developed for clustering, classification, and prediction [15]. ML plays an important aspect in the intrusion detection in WSN, such as "Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), and Gaussian Naïve Bayes (GNB)" which are utilized to prognosis any type of malicious attacks. AI is used by ML at the different levels of abstraction to perform the task to exclude the human intervention.

ML is broadly classified based on the design and composition of the model. It is categorized as "Supervised Learning (SL), Reinforcement Learning (RL), and Unsupervised Learning". Kumar et al. [16] suggested the ML techniques for detecting the malicious attack in WSN. The investigation gives the merits, parameters with the statistical analysis. Supervised learning is used by many researchers for detecting the link between input and output pairs. At the process end, the premier output is estimated with its associated input [17].

Furthermore, many high-performance ML algorithms are employed for many issues in WSN, such as Routing, Localization, Target tracking, Data aggregation, Mobile sink, detection of Events, detection of an anomaly, Energy harvesting, and Synchronization. These issues are classified as functional and non-functional challenges. The functional challenge is the routing and its enhancements utilize SL and RL [18]. The clustering method resolves the data aggregation, connectivity, event detection using Decision Trees (DT) & Neural Networks (NN). The non-functional challenges such as security and anomalous ID are overcome by "Bayesian Networks (BN), K Nearest Neighbour (KNN), and forwarding inherits SVM" [18]. Moreover, multiple classification attacks with low detection time in existing WSN-IDS are considered to be the most daunting challenge among researchers. In this view of the challenge, this paper proposes the novel hybrid learning model MBLOT-Megabat Optimization on Long Term Short Memory (LSTM) training networks to achieve the higher classification rate of multiple attacks and less detection time.

2 Contribution of the Research

- A Novel and Hybrid Learning Model-based WSN–IDS (MBOLT-WSN-IDS) has been proposed in which the hyperparameters of LSTM cells are optimized by the MegaBAT heuristic algorithm which solves the classification problems and high-speed detection mechanism. It has higher accuracy, low complexity, and low detection time.
- This study uses the WSN-DS datasets to conduct the experimentations and comprehensive comparative analysis has been carried out using the existing learning model-based IDS system.
- A scalable, high accurate, high-speed MBOLT IDS System is introduced to handle the larger WSN datasets.

The organization of the paper is as follows: Related work is introduced in Section 2. Section 3 discusses the proposed framework, with background views on MegaBAT Optimization and LSTM. The

experimentation details, dataset description, result analysis along comparisons are presented in Section 4. Finally, the paper is concluded in Section 5.

3 Related Works

Swarnkar et al. [19] propounded the Singular Valued Decomposition (SVD) method to reduce the feature dimension. Eigenvectors are utilized to further reduce the features. The experimentation uses the KDD cup'99 dataset analyzed on various Intrusion Detections (IDs) methods. Here, multiple classifiers were utilized and this method achieved 97.90% accuracy when it is contrasted with other methods with the same dataset.

Li et al. [20] named a new method called Density Peak Nearest Neighbour (DPNN) which integrated the Density Peaks (DN) and KNN methods to achieve improved performance in terms of accuracy. In this method, the DN is adopted for training the data and KNN is for classification. For the validation, the KDD cup'99 dataset is utilized and the DPNN outperformed other methods such as K Nearest Neighbour (KNN) and Support Vector Machines (SVM) in regards to probe attack and average accuracy. The training time is greatly reduced and increased efficiency to 20.688%.

Moustafa et al. [21] proposed an ensemble IDS to prevent the botnet attacks against the Message Queuing Telemetry Transport (MQTT), Hypertext Transfer Protocol (HTTP), and Domain Name System (DNS) protocols. The proposed Adaboost Ensemble Learning (EL) method is designed based on three ML methods such as Decision Tree (DT), Naïve Bayes (NB), and Artificial Neural Networks (ANN). The specialized EL method calibrates the features and prognosis thof e malicious events accurately. The extensive work uses the UNSW-NB15 and NIMS botnet datasets to produce the best accuracy was correlated with the various classification technique. The results outperformed in regards to detection rate and false-positive rate.

Ifzarne et al. [22] focus on the detection of anomalous attacks to provide security to the network. The proposed model is designed by information gain ratio and the online Passive-aggressive classifier selects the appropriate features. It can able to detect the Daniel Service (DoS) attacks. The experimentation was regulated on the Wireless Sensor Network-Detection System (WSN-DS) dataset produces the best accuracy on detecting grey hole, flooding, and blackhole attacks. Thereby, the online learning-based Machine Learning (ML) method produces better accuracy results in detecting the attacks in WSN. Liu et al. [23] proposed the IDS for the notion of the game theory. It incorporates the feature of the SVM algorithm. It produces the best learning ability and generalization performance. The empirical studies conducted on the DARPA dataset produces feasible detection accuracy and its most efficient algorithm for detecting DDoS attacks.

Chang et al. [24] emerged a network IDS-based Random Forest (RF) and SVM algorithm. The RF is used for feature selection, SVM is utilized for intrusion classification. SVM classifiers inherit 14 appropriate features to incline the performance of the detection. The KDD 99 dataset was utilized by the experiment and correlated with the 41 features on popular classifiers. The results reveal that the proposed network IDS based on RF and SVM attains a better detection rate.

Kanjanawattana [25] evolved a novel method based on a clustering model with initial centers selection relayed on data density. The proposed method uses the two datasets. The first dataset is inclusive of outliers and the second is excludes the outliers by duplicating the first dataset. The dataset consists of 6 numeric attributes and 7200 instances included outliers. The experimentation uses the first dataset evaluation was compared among the adaptive K-means with the traditional k means algorithm precision (0.4), recall (0.38), accuracy (0.84), and F-measures (0.38). The second dataset experimentation, reaches good accuracy, thereby proving that the proposed algorithm outperforms the traditional algorithm obtained significant results.

Zhang et al. [26] introduced a hierarchical IDS in which grouping is done based on the operations. The main focus of this framework is to down the false alarm rate to increase the performance of the framework in regards to accuracy. Multi Kernel Extreme Machine Learning (ELM) is implemented for attack detection. This framework utilized the NSL-KDD dataset and UNSW-NB–15 datasets for validation. The results demonstrate better performance in terms of accuracy as well as time-consuming.

4 Overview of DL Models

4.1 Recurrent Neural Network

Generally, RNN models are specifically designed for time series of data and big data analytics because of their fast remembrance activity. Here the direct graphs are generated by the nodes with their respective sequences. With this statement, this method demonstrates dynamic synchronizations of sequences. For the input sequences process, internal memory is utilized. RNN structure uses the past data for the prediction of future values. But in real-time applications, if the transition time between past and future data is relatively large, then this method struggles to remember the past values and creates a disappearing gradient issue [27], so it requires updates to support real-time applications. To countermeasure these issues, the RNN is updated as an LSTM structure.

4.2 LSTM–Long Short Term Memory

LSTM is an updated version of RNN and which is represented in Fig. 1

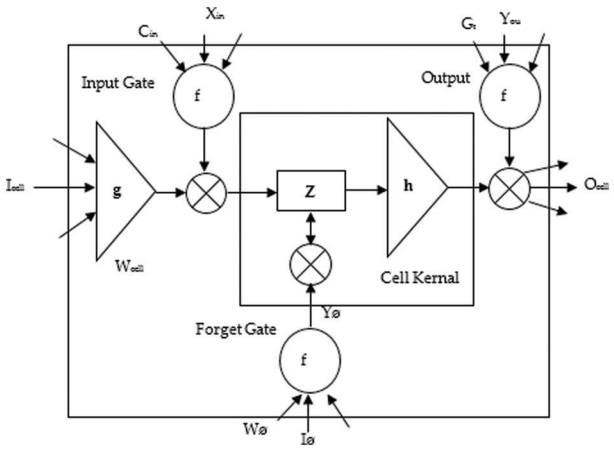

Figure 1: LSTM structrure

The proposed framework has LSTM and Mega BAT optimizer. The LSTM unit comprises of 3 units called “Input Gate (IG), Forget Gate (FG), Output Gate (OG), and Cell Input (CI)”. It is a neural network structure and it remembers the values of each iteration. The main advantage of the LSTM framework is it can remember the past output and finely merges with the current input. The FG plays a major role in memory updates. The three states of gates are denoted as j_t, T_f and T_0.

Let $Ct \rightarrow$ cell input state, G_t, $\rightarrow$ cell output state, and G_{t-1}, $ht \rightarrow$ hidden layer output, $ht-1 \rightarrow$ hidden layer former output. The LSTM structure relies on both G_t andh_t. The following Eqs. (1)–(4), are the mathematical expressions used to calculate G_t and h_t,

$$I.G:\ j_t = \theta(G_l^i.\ O_t + G_h^i.e_{t-1} + s_i) \tag{1}$$

$$F.G:\ T_f = \theta(G_l^f.O_t + G_h^f.e_{t-1} + s_f) \tag{2}$$

$$O.G:\ T_o = \theta\ (G_l^0.O_t + G_h^o.e_{t-1} + s_o) \tag{3}$$

$$C.I:\ \widetilde{T_C} = \tanh(G_l^C.O_t + G_h^C.e_{t-1} + s_C) \tag{4}$$

where

$G_h^i,\ G_h^f,\ G_h^o,\ G_h^C \rightarrow$ The weight conditions are hidden and input layers.

$G_l^0,\ G_l^f,\ G_l^i,\ G_l^C \rightarrow$ weight matrices IG and output layers

$s_i, s_f, s_o, s_C \rightarrow$ Bias vectors

$tanh \rightarrow$ hyperbolic function.

Cell output state expressed as follows

$$T_C = k_t * \ \widetilde{T_C} + T_f * T_{t-1} \tag{5}$$

$$e_t = T_o * \tanh(T_C) \tag{6}$$

The equation is used to get the Final output score.

4.3 LSTM Drawbacks

The training of LSTM networks requires a high computational complexity which impacts detection time and network overhead. Additionally, the overfitting problem also creates a bottleneck in the LSTM network when handling the larger dataset. Motivated by this problem, this paper proposes the new hybrid MEGABAT optimized LSTM training networks for better detection and less complexity. The prey searching process of megabats is used to optimize the hyperparameters of LSTM networks.

5 System Overview

The proposed Intrusion Detection Systems (IDS) are mainly divided into 3 phases: dataset collection unit, data preprocessing model, and intelligent detection model and response module. Initially, the data are collected by the data collection unit from different sensor locations then it is passed to the data preprocessing unit. Once the preprocessing is gets over then the proposed MegaBAT optimized Long Short Term Memory (MBOLT) module can detect the intrusion if it happens in the network. If the module detects an attack, it immediately updates the users regarding the attack occurrence. Fig. 2 depicts the proposed BOLT module which comprises Member Node (MN), Cluster Head (CH), and Sinks. Here the MN represents sensor node, CH represents cluster node and Sink represents Base Station (BS). To reduce computation overhead and to reduce energy consumption, sink nodes are chosen for the IDS implementation.

5.1 Proposed Framework

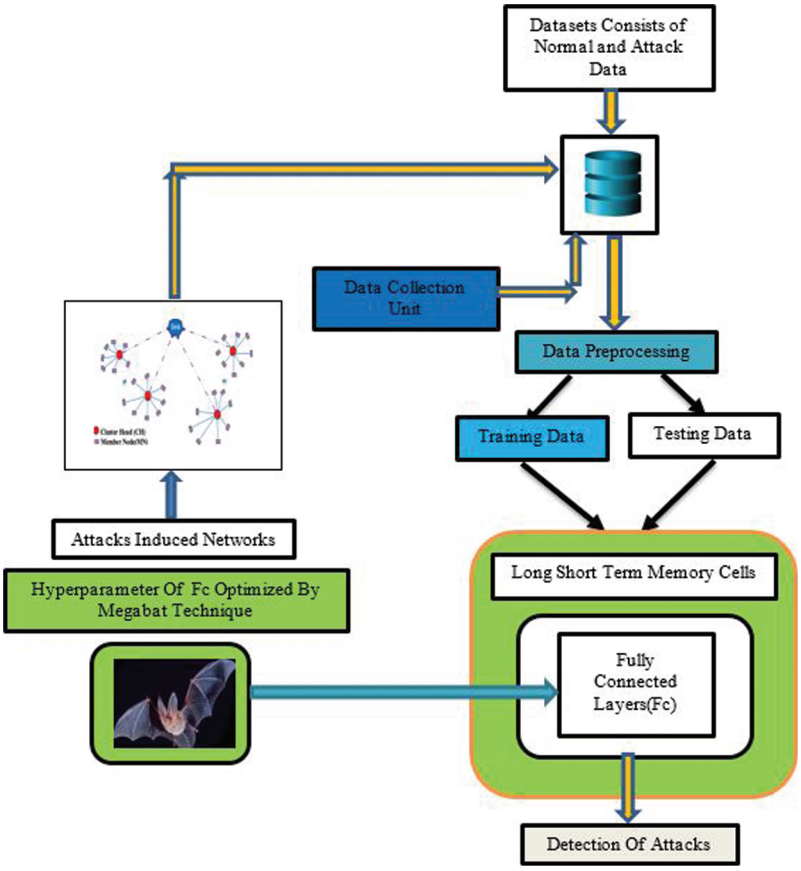

Figure 2: Architectural diagram for the proposed framework with BOLT module

5.2 Data Collection Unit

Wireless Sensor Network (WSN) is a wireless network composed of a large number of self-organizing sensor nodes to sense, collect, process and transmit the network information to the different users. These wireless sensor networks are simulated in different tools such as Network Simulators (NS-2) and Objective Modular Network (OMNET) software [28]. The proposed learning model uses the WSN datasets which are simulated by NS-2. The collected datasets use the Low Energy Adaptive Clustering Hierarchy (LEACH) routing protocol which circularly selects the CH and distributes the energy of the network to individual nodes for the better reduction of network energy consumption. The collected data are processed under attack scenarios and non-attack scenarios to detect the attack node's entire network. The data attributes of the proposed framework are depicted in Tab. 2.

5.3 Data Preprocessing

The dataset collected needs preprocessing mechanism since it consists of both letters and numerical. In preprocessing, letters are converted into numeric values. For the better identification of attacks, multi-class labeling is adopted in this paper. Tab. 1 presents the multi-class labeling for each attack.

Table 1: Data labeling for different attacks

Sl. No	Type of attack	Labeling
01	Normal	1
02	Blackhole	2
03	GrayHole	3
04	Scheduling	4
05	Flooding	5

Table 2: Types of attributes used for experimentation

Sl. No	Features used	Description
01	Node ID	The IP address allocated for every sensor nodes
02	Cluster-ID	The IP address of cluster heads
03	Initial energy of nodes	Initial energy is calculated at beginning of the experimentation rounds. It is used to transmit the sensor's data to the CHS.
04	Energy Consumption (EC)	The consumed energy is calculated at every iteration of data transfer between the nodes and sink
05	Residual energy	The residual energy is the remaining energy after every iteration
06	Cluster head distance	The distance between the cluster head and other nodes
07	Throughput	The throughput measures as the ratio between the received bytes to the transmitted bytes (at cluster head side)
08	Throughput-2 (T)	The throughput measures as the ratio between the received bytes to the transmitted bytes (at the sink)
09	Delay (ms)	Time calculated for data to reach from nodes to cluster heads
10	Received Signal Strength Indicators (RSSI)	It is used for the detection of signal strength which in turn is used to calculate the distance.

To have a better classification mechanism, the normalization technique [29] is adopted. In the data set, there are minimum values (<1) and maximum values (>1000) which may give a great influence on the classification performance. So, here data normalization plays a vital role in the proposed framework. We need to normalize the continuous data. The attributes along with the descriptions are presented in Tab. 2.

5.4 Out-Classification Model

By considering the drawback of the LSTM, the BOLT classification model used the MegaBAT algorithm and which is explained as follows,

5.5 MEGA BAT Algorithm-an Overview

The standard mega-bat algorithm generally relies on "Echolocation or bio-sonar attributes of microbats". Based on the echolocation, Yang et al. [30] built up the algorithm with the accompanying 3 glorified rulesets:

- Bats always use echolocation to detect prey mystically.
- Bats always have an initial frequency ($f_{\min}$), speed (v_i), position, and loudness A_0. These parameters are consequently modified in regards to the nearness of prey.

■ Here we can take that loudness is shifts from,

Every bat motion has an initial distance, the initial velocity of an respected in the search space. Among all bats, the best bat is chosen based on the 3 rules just mentioned above [30]. The updated velocity v_i^t & initial distance x_i^t using the three rules are given below

$$f_i = f_{min} + (f_{max} - f_{min})\beta \tag{7}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{8}$$

where

$\beta \in (0,\ 1)$,

$f_{\min} \rightarrow$ minimum frequency (0)

$f_{\max} \rightarrow$ maximum frequency.

Initially, every bat has a frequency between $f_{\min}$ and $f_{\max}$

To provide promising solutions, emission rates and loudness are calculated consequently. Once the prey is detected by the bat the loudness made y diminishes but as the pulse emission rate expands, the loudness can be picked as any estimation of accommodation among $A_{\min}$ and $A_{\max}$.

5.6 BOLT-IDS Working Mechanism

The simple bat algorithms are used to optimize the weights of LSTM networks which is discussed in Section 3.4.2. In this case, the bat's prey searching mechanism is used as the main term to optimize the hyperparameters of LSTM networks. The "input weights, hidden layers, epochs, and learning rates" are considered to be hyperparameters of any training network. Initially, these hyperparameters are selected randomly and passed to the LSTM training network. The fitness function of the proposed network is given by Eq. (9). For each iteration, hyperparameters are calculated by using Eqs. (7) and (8). The iteration stops when the fitness function matches the Eq. (9).

$$FitnessFunction\ (FF) = Max(Accuracy,\ \ Precision,\ \ REcall\&F1 - Score) \tag{9}$$

Sl. No.	Algorithm1 // Pseudo Code for the Proposed MBOLT-IDS
01	Input = Hyperparameters of the LSTM training network
02	Output: Categorization of the attacks
03	Randomly assigned hyperparameters
04	Initialize the Loudness, Frequency, Distance, No of bats and Velocity
05	While (true)
06	Calculate the output value from the LSTM cell using Eq. (6)
07	Calculate the FF using the Eq. (9)
08	For t = 1 to Max_iteration
09	Assign the bias weights and input layers by Eqs. (7) and (8)
10	Calculate the FF from the LSTM cell using Eq. (9)
11	If (FF = = Maximum Accuracy)
12	Go to Step 16

(Continued)

Algorithm 1 (continued)

```
13                                  Else
14                                      Go to Step 10
15          End
16          End
17            If (output_value <= 1)
18            /Normal Data traces//* No traces found
19           Else if(output_value <= 2&&output_value > 1)
20          / Blackhole attack is detected
21          Else if Else if(output_value <= 3&&output_value > 2)
22          Gray hole attack is detected
23          Else if Else if(output_value <= 4&&output_value > 3)
                Scheduling attack is detected
24          Else if Else if(output_value <= 4&&output_value > 3)
25          A flooding attack is detected
26            Go to step 1
27          End
28          End
29          End
```

6 Simulationn Experiment

The proposed algorithm was implemented using Tensorflow v1.18 with Keras Application Programming Interface (API) Environment and performed on a computer configured with "Intel Quad Core i5-10th generation CPU, 8 GB RAM, and 8 GB NVIDIA titan GPU enabled Windows 10 Operation Systems".

6.1 Dataset Description

The proposed architecture was evaluated by the WSN-DS public datasets [31]. This is an intrusion detection dataset developed using LEACH-based WSNs in an NS-2 environment. WSN-DS contains 4 types of Denial Service attacks (DoS), black hole, gray hole, flooding, and scheduling. Tab. 3 depicts the complete statistical information of WSN-DS public datasets used for the performance evaluation of the proposed framework.

Table 3: Number of traces used for training and testing the proposed network

Sl. No.	Attack details	No of traces
01	Normal	340006
02	Black hole attack	14596
03	Gray hole attack	10049
04	Scheduling attack	3312
05	Flooding attack	6638

Above Tab. 3 depicts the statistical details of the training data used for the network. Nearly 2,62,177 (70%) samples are used for training and 1,12,384 (30%) samples are used for testing. Nearly 22 attributes are present in the datasets and all parameters are used for the training and testing.

6.2 Performance Validation

To evaluate the proposed methodology on the datasets, performance metrics such as Accuracy, Precision, Recall, Specificity, and F1-score were used to identify the different attacks.

7 Results and Findings

Experimented with the proposed framework in the following aspects: 1) Analyzing the performance of the proposed framework against four attacks, to meet the networks' safety criteria against attacks and integrate into the networks' IDS. 2) Using the state-of-the-art ML and other existing learning classification models to compare with proposed architecture 3) Performing the time consumption experimentations to prove the performance of the proposed system.

The receiver operating curve (ROC) helps to understand the classifier performance by analyzing the actual positive rate *vs.* the false positive rate. The area under the ROC gives infers the best performance of the classifier. A higher ROC area infers higher performance of the classifier. Fig. 3 shows the ROC characteristics of the proposed model under different attack scenarios. Fig. 3 shows, the proposed framework covers a maximum area (0.95) and produced maximum accuracy of 99% in the attack detection category. Tab. 4 gives the proposed framework confusion matrix under the multiple attack detection category.

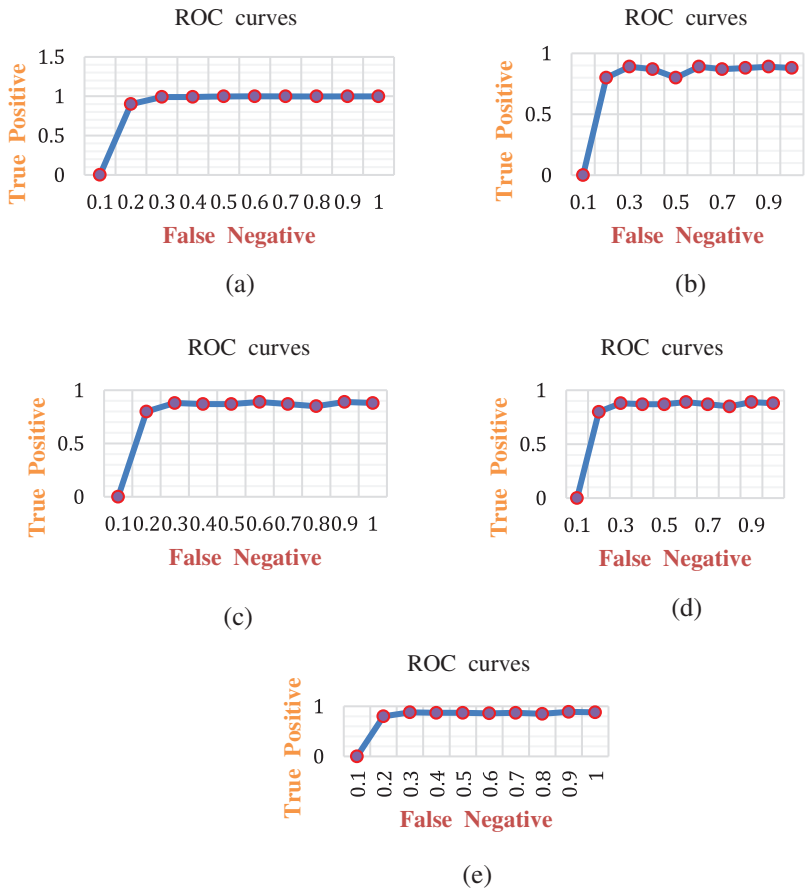

Figure 3: ROC curves for the proposed architecture: a) blackhole attacks, b) gray hole, c) flooding, d) scheduling, e) normal

Table 4: Confusion matrix for the proposed architecture for detection of different attacks

Class	Normal	Blackhole	Gray hole	Flooding	Scheduling
Normal	56%	2%	3%	1%	1%
Blackhole	1%	20%	0.5%	1.5%	1%
Gray hole	0.5%	1%	19%	1%	0.5%
Flooding	1%	0.5%	0.5%	5%	0.5%
Scheduling	0.25%	0.5%	0.5%	0.5%	5%

From Tab. 5, it is clear that performance metrics such as accuracy, precision, recall, specificity, and F1-score have exhibited stable performance which ranges from 99% to 100% in detecting the various categories of attacks. The inclusion of optimized hyperparameters in the deep learning model has produced constant performance in detecting the attacks such as blackhole, Grayhole, flooding, scheduling, and even the normal conditions.

Table 5: The performance metrics of the proposed model usingWSN-DS datasets

	Types of attacks	Accuracy	Recall	Specificity	Precision	F1-score
Proposed algorithm	Blackhole	99.8%	99.85%	99.84%	99.87%	99.87%
	Grey hole	99.79%	99.82%	99.82%	99.86%	99.86%
	Flooding	99.81%	99.82%	99.83%	99.85%	99.88%
	Scheduling	99.81%	99.83%	99.82%	99.88%	99.86%
	Normal	99.81%	99.83%	99.82%	99.88%	99.86%

There are different dropout models are incorporated to improve the generalization performance of the proposed model which is given in Tab. 6. The performance of the proposed model in detecting different attacks in accordance with the different dropout mechanisms is shown in Fig. 4. From the tables, we found that the selecting the different dropout values for experiments, the performance of the model is optimal for all dropouts due to the optimized hyperparameters of the proposed model.

Table 6: Dropout performances of the proposed algorithm in detecting various attacks

	Blackhole				
Dropout	Accuracy	Recall	Specificity	Precision	F1-score
0.2	99.88%	99.85%	99.84%	99.87%	99.88%
0.4	99.87%	99.82%	99.83%	99.86%	99.87%
0.6	99.87%	99.82%	99.83%	99.86%	99.87%
0.8	99.85%	99.82%	99.82%	99.84%	99.85%
	Grey hole				
0.2	99.84%	99.82%	99.82%	99.86%	99.86%
0.4	99.82%	99.81%	99.81%	99.85%	99.86%
0.6	99.82%	99.81%	99.81%	99.85%	99.86%
0.8	99.82%	99.80%	99.79%	99.80%	99.82%

(Continued)

Table 6 (continued)

Dropout	Blackhole				
	Accuracy	Recall	Specificity	Precision	F1-score
	Flooding				
0.2	99.81%	99.82%	99.83%	99.85%	99.87%
0.4	99.80%	99.81%	99.82%	99.84%	99.86%
0.6	99.79%	99.80%	99.82%	99.84%	99.86%
0.8	99.80%	99.80%	99.82%	99.84%	99.86%
	Scheduling				
0.2	99.81%	99.83%	99.82%	99.88%	99.86%
0.4	99.80%	99.82%	99.82%	99.86%	99.85%
0.6	99.80%	99.81%	99.81%	99.86%	99.84%
0.8	99.79%	99.81%	99.81%	99.85%	99.84%
	Normal				
0.2	99.86%	99.83%	99.82%	99.88%	99.86%
0.4	99.85%	99.82%	99.81%	99.86%	99.85%
0.6	99.85%	99.81%	99.81%	99.85%	99.85%
0.8	99.84%	99.81%	99.81%	99.85%	99.85%

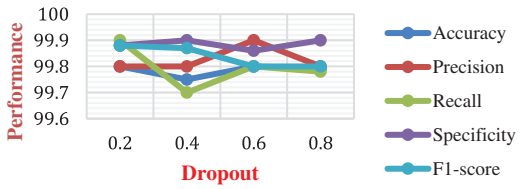

Figure 4: Comparative analysis between the average performance of the proposed algorithms at different epochs

Comparative Analysis

To prove the proposed model excellence it is compared with the other state of art learning models show in Fig. 5 such as SVM [32], ANN [33], KNN [34], RF [35], Light + GB [36], IGLGBM [37], SLGBM [38], Multi-layer-ELM (ML-ELM) [39] and LSTM [40].

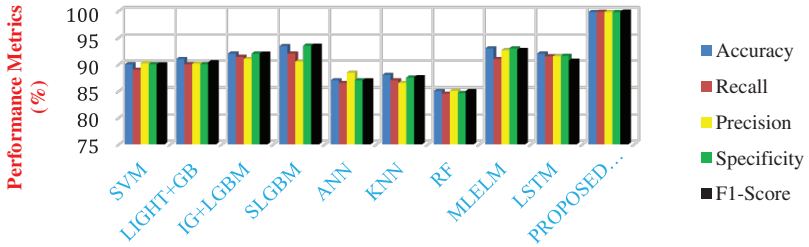

Figure 5: Comparative analysis of performances of different learning models in detecting the normal scenario

The performance metrics of different hybrid learning models have been compared with WSN-DS datasets and analysis are shown from Figs. 6 to 10. In Fig. 6, performance metrics of different models has been compared for detecting the normal scenario. In this case, the proposed model has produced 99.8% accuracy, 99.8% precision, 99% recall, and even a high F1-score of 99.8%. The LSTM and MLELM have produced the good detection performance but lesser than the proposed model. Moreover, the proposed model has produced better detection than ANN, KNN, SVM, LIGHT + GB, IGLGBM, and SLGBM respectively. This is because of the integration of bat optimization over the LSTM training network in the proposed framework. This improves the classification model for different attack detection. It is found that the proposed model has exhibited superior performances to the other existing algorithms in the detection of other attacks such as "Grayhole, Blackhole, scheduling and flooding attacks".

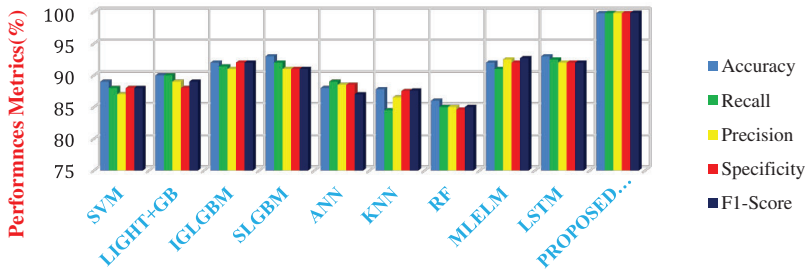

Figure 6: Comparative analysis of performances of different learning models in detecting the blackhole attacks

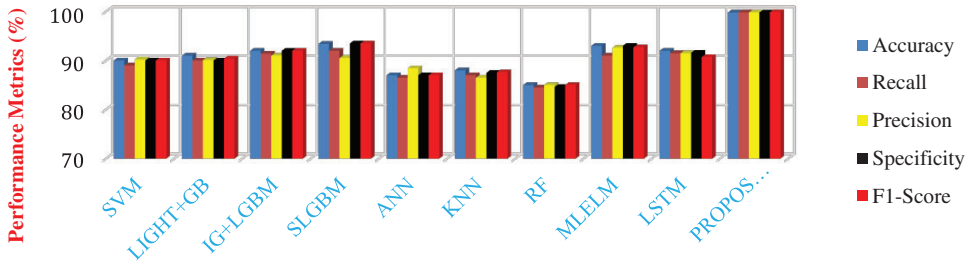

Figure 7: Comparative analysis of performances of different learning models in detecting the gray hole attacks

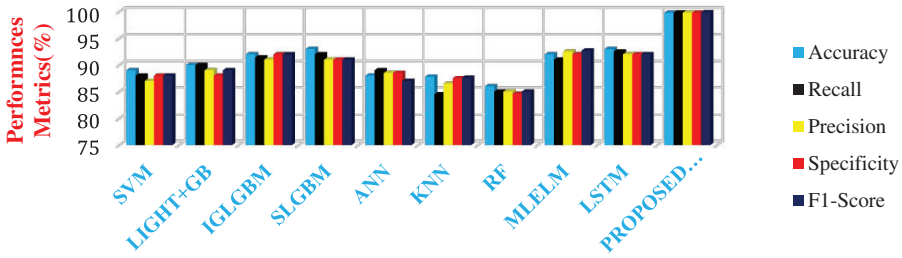

Figure 8: Comparative analysis of performances of different learning models in detecting the scheduling attack

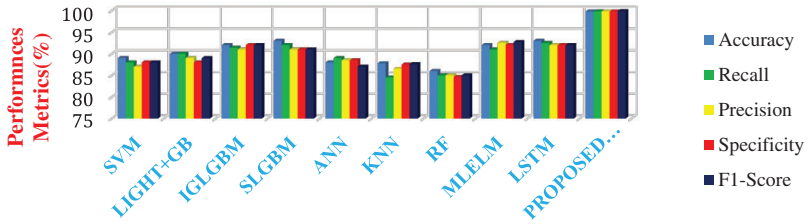

Figure 9: Comparative analysis of performances of different learning models in detecting the flooding attacks

In the next phase of experimentation, we calculated the detection time of every learning model for the data mentioned in the section. From the results depicted in Fig. 10, we can observe the proposed model consumes less time in handling the larger datasets and produces a lesser detection time than the other existing models. Moreover, the proposed model has produced high-speed detection and demonstrates that the proposed BOLT model has shown better scalability than the other existing algorithms in handling the larger multi-class data.

Figure 10: Time detection analysis for the different algorithm used for detecting different attacks

8 Conclusion and Future Enhancement

Based on the BAT optimization and principles of the LSTM training network, we architected the novel and intelligent intrusion detection model-BOLT IDS targeted for the clustered WSN environments. The proposed models have been compared with the other existing learning models. The extensive experimentation is carried out using WSN-DS intrusion datasets and contrasted with learning models. The outcome strongly proves that the proposed framework has shown better performances than the other existing learning models in terms of classification and detection time. The proposed model provides promising results in intrusion detection systems in the WSN environment. Even though this model solved the multiple classification attacks with a high classification rate and high-speed detection, energy, needs to make the thrust on Quality of Service (QoS) parameters of WSN environments. In future work, QoS parameters such as Packet Delivery Ratio (PDR), energy consumptions, and throughputs need brighter light of research to improve the overall performances of WSN.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] C. W. Tsai, C. F. Lai and A. V. Vasilakos, "Future internet of things: Open issues and challenges," *Wireless Networks*, vol. 20, no. 8, pp. 2201–2217, 2014.

[2] L. M. L. Oliveira and J. J. P. C. Rodrigues, "Wireless sensor networks: A survey on environmental monitoring," *Journal of Communications*, vol. 6, no. 2, pp. 143–151, 2011.

[3] K. Guleria and A. K. Verma, "Comprehensive review for energy efficient hierarchical routing protocols on wireless sensor networks," *Wireless Networks*, vol. 25, no. 3, pp. 1159–1183, 2019.

[4] F. Bao, I. R. Chen, M. Chang and J. H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection," *IEEE Transactions on Network and Service Management*, vol. 9, no. 2, pp. 169–183, 2012.

[5] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[6] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *Proc. 10th Int. Conf. on Cyber Conflict (CyCon X)*, Tallinn, Estonia, pp. 371–390, 2018.

[7] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li *et al.,* "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.

[8] N. Milosevic, A. Dehghantanha and K. K. R. Choo, "Machine learning aided android malware classification," *Computers and Electrical Engineering*, vol. 61, pp. 266–274, 2017.

[9] H. B. R. Mohammed, R. Vinayakumar and K. P. Soman, "A short review on applications of deep learning for cyber security," arXiv:1812.06292v2 [cs.CR], 2019.

[10] D. S. Berman, A. L. Buczak, J. S. Chavis and C. L. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, pp. 122, 2019.

[11] S. Paul, Z. Ni and C. Mu, "A learning-based solution for an adversarial repeated game in cyber-physical power systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 11, pp. 4512–4523, 2020.

[12] D. Ding, Q. L. Han, Y. Xiang, X. Ge and X. M. Zhang, "A survey on security control and attack detection for industrial cyber physical systems," *Neurocomputing*, vol. 275, pp. 1674–1683, 2018.

[13] M. Wu, Z. Song and Y. B. Moon, "Detecting cyber-physical attacks in cyber manufacturing systems with machine learning methods," *Journal of Intelligent Manufacturing*, vol. 30, no. 3, pp. 1111–1123, 2019.

[14] L. Xiao, X. Wan, X. Lu, Y. Zhang and D. Wu, "IoT security techniques based on machine learning," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 41–49, 2018.

[15] H. Jadidoleslamy, "A high-level architecture for intrusion detection on heterogeneous wireless sensor networks: Hierarchical, scalable and dynamic reconfigurable," *Wireless Sensor Network*, vol. 3, no. 7, pp. 241–261, 2011.

[16] D. P. Kumar, T. Amgoth and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Information Fusion*, vol. 49, pp. 1–25, 2019.

[17] I. Onat and A. Miri, "A real-time node-based traffic anomaly detection algorithm for wireless sensor networks," in *Proc. 2005 Systems Communications*, Montreal, Quebec, Canada, pp. 422–427, 2005.

[18] D. E. Baraneetharan, "Role of machine learning algorithms intrusion detection in WSNs: A survey," *Journal of Information Technology and Digital World*, vol. 2, no. 3, pp. 161–173, 2020.

[19] M. Swarnkar and N. Hubballi, "OCPAD: One class naive Bayes classifier for payload based anomaly detection," *Expert Systems with Applications*, vol. 64, pp. 330–339, 2016.

[20] L. Li, H. Zhang, H. Peng and Y. Yang, "Nearest neighbors based density peaks approach to intrusion detection," *Chaos, Solitons & Fractals*, vol. 110, pp. 33–40, 2018.

[21] N. Moustafa, B. Turnbull and K. K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, 2019.

[22] S. Ifzarne, H. Tabbaa, I. Hafidi and N. Lamghari, "Anomaly detection using machine learning techniques in wireless sensor networks," *Journal of Physics: Conference Series*, vol. 1743, pp. 012021, 2021.

[23] Y. Liu and D. Pi, "A novel kernel svm algorithm with game theory for network intrusion detection," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 8, pp. 4043–4060, 2017.

[24] Y. Chang, W. Li and Z. Yang, "Network intrusion detection based on random forest and support vector machine," in *Proc. 2017 IEEE Int. Conf. on Computational Science and Engineering (CSE) and IEEE Int. Conf. on Embedded and Ubiquitous Computing (EUC)*, Guangzhou, China, pp. 635–638, 2017.

[25] S. Kanjanawattana, "A novel outlier detection applied to an adaptive k-means," *International Journal of Machine Learning and Computing*, vol. 9, no. 5, pp. 569–574, 2019.

[26] W. Zhang, D. Han, K. C. Li and F. I. Massetto, "Wireless sensor network intrusion detection system based on MK-ELM," *Soft Computing*, vol. 24, no. 16, pp. 12361–12374, 2020.

[27] I. Butun, S. D. Morgera and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 266–282, 2014.

[28] K. Khan and A. Sahai, "A comparison of BA, GA, PSO, BP and LM for training feed forward neural networks in e-learning context," *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 7, pp. 23–29, 2012.

[29] Z. Wang, G. Yu, Y. Kang, Y. Zhao and Q. Qu, "Breast tumor detection in digital mammography based on extreme learning machine," *Neurocomputing*, vol. 128, pp. 175–184, 2014.

[30] X. S. Yang and X. He, "Bat algorithm: Literature review and applications," *International Journal of Bio-Inspired Computation*, vol. 5, no. 3, pp. 141–149, 2013.

[31] I. Almomani, B. Al-Kasasbeh and M. Al-Akhras, "WSN-DS: A dataset for intrusion detection systems in wireless sensor networks," *Journal of Sensors*, vol. 2016, pp. 4731953, 2016.

[32] G. M. Borkar, L. H. Patil, D. Dalgade and A. Hutke, "A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN: A data mining concept," *Sustainable Computing: Informatics and Systems*, vol. 23, pp. 120–135, 2019.

[33] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat *et al.,* "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[34] C. C. Su, K. M. Chang, Y. H. Kuo and M. F. Horng, "The new intrusion prevention and detection approaches for clustering-based sensor networks [wireless sensor networks]," in *Proc. IEEE Wireless Commununications and Networking Conf.*, New Orleans, LA, USA, pp. 1927–1932, 2005.

[35] J. Su, Z. Sheng, A. X. Liu, Y. Han and Y. Chen, "A group-based binary splitting algorithm for UHF RFID anti-collision systems," *IEEE Transactions on Communications*, vol. 68, no. 2, pp. 998–1012, 2020.

[36] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen *et al.,* "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. 31st Conf. on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, pp. 3146–3154, 2017.

[37] T. P. Rani and C. Jayakumar, "Unique identity and localization based replica node detection in hierarchical wireless sensor networks," *Computers & Electrical Engineering*, vol. 64, pp. 148–162, 2017.

[38] S. Jiang, J. Zhao and X. Xu, "SLGBM: An intrusion detection mechanism for wireless sensor networks in smart environments," *IEEE Access*, vol. 8, pp. 169548–169558, 2020.

[39] J. Sharma, C. Giri, O. C. Granmo and M. Goodwin, "Multi-layer intrusion detection system with extra trees feature selection, extreme learning machine ensemble, and softmax aggregation," *Eurasip Journal on Information Security*, vol. 15, no. 1, pp. 1–16, 2019.

[40] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 41–50, 2018.