Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.022259
Article

Tech Science Press

A Novel Hybrid Deep Learning Framework for Intrusion Detection Systems in WSN-IoT Networks

M. Maheswari[1,2,*] and R. A. Karthika[1]

[1]Vels Institute of Science, Technology and Advanced Studies, Chennai, 600117, India
[2]SRM Institute of Science and Technology, Chennai, 603203, India
*Corresponding Author: M. Maheswari. Email: maheswam@srmist.edu.in

Received: 02 August 2021; Accepted: 07 November 2021

Abstract: With the advent of wireless communication and digital technology, low power, Internet-enabled, and reconfigurable wireless devices have been developed, which revolutionized day-to-day human life and the economy across the globe. These devices are realized by leveraging the features of sensing, processing the data and nodes communications. The scale of Internet-enabled wireless devices has increased daily, and these devices are exposed to various cyber-attacks. Since the complexity and dynamics of the attacks on the devices are computationally high, intelligent, scalable and high-speed intrusion detection systems (IDS) are required. Moreover, the wireless devices are battery-driven; implementing them would consume more energy, weakening the accuracy of detecting the attacks. Hence the design of the IDS is required, which has to establish the good trade-offs between Energy and accuracy. This research includes the Multi-tiered Intrusion Detection (MDIT) with hybrid deep learning models for improved detection accuracy in wireless networks; spotted hyena optimization (SHO) and Long short-term memory (LSTM) have been studied to design IDS effectively. Extensive experimentation has been carried out in real-time scenarios using the Node MCU Embedded boards and standard benchmarks such as CIDDS-001, UNSWNB15 and KDD++ datasets compared with the other traditional and existing learning models. The average prediction accuracy of 99.89% for all datasets has been achieved. The results show that the proposed system guarantees a high detection accuracy and reduces the prediction time, making this system suitable for resource-constrained IP-enabled wireless devices.

Keywords: Internet-enabled; reconfigurable wireless devices; spotted hyena optimizer; long short term memory; node MCU; multi-tier architecture

1 Introduction

Internet of Things (IoT) enabled Wireless Sensor Networks (WSN) are finding their various applications such as health care [1], consumer electronics [2] and even image transmission [3]. The increased connectivity between the networks, more extensive deployment of these devices, and the broadcasting nature of communication make the devices vulnerable to different attacks, resulting in casualties in the network [4].

Nowadays, these intrusion detection systems have changed their face of dimension with an application of artificial intelligence (AI), especially of machine learning (ML) and deep learning (DL) applied to both attack and defence measures in IP enabled wireless networks. These algorithms provide defence strategies and resistance against security threats to prevent and minimize the impacts or casualties adaptively. Many machine learning and deep learning models have applied intrusion detection [5–7], malware detection [8–11], cyber-physical attacks [12–14] and data privacy protection [14].

In machine learning models, neural networks offer many advantages such as self-learning, proper classification, scalability, which have prompted researchers to investigate the intrusion detection systems based on neural networks and have achieved more excellent detection performance. As of late, Extreme Learning Machines (ELM), considered single feedforward neural networks (SLFNN), have gained popularity in designing the IDS for efficient detection of various attacks in the wireless networks [15]. Also, Wenjie et al. [16] proposed the new Kernelized Extreme Learning Machine (KELM) methodology, which has overcome the drawbacks of traditional extreme learning machines and has shown more compelling results in time and accuracy. These algorithms fail to detect different categories of attacks in networks and have not been tested in real-time. The handling of more attacks leads to complex data formation, which leads to the misclassification of attacks. Many deep learning models such as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) have been proposed to detect intrusion systems, giving better accuracy in detection and reducing false alarm rates to overcome this problem [17–22]. Researchers have suggested and tested high performance deep learning algorithms such as Channel Boosted Residual Convolutional Neural networks (CBR-CNN) [23], Improved Genetic algorithm optimized Deep learning models [24], and ensemble deep learning models [25,26]. Some are CICIS2017, UNSW NB15, NSL-KDD++ datasets found suitable for accurate detection of attacks.

The deep learning methods were applied in the Internet of Things to identify the malicious nodes. The deep learning model was applied to improve the efficiency of the detection. However, the most existing deep learning-based IDS have difficulty in applying intrusion detection for real-world environments. The implementation in the real world scenario has resulted in a higher miscalculation rate. Hence the intelligent system is required for solving the real-world attacks in the WSN-IoT environment. This research article proposes the novel SHOLEN-IDS (Spotted Hyena Optimized Long Efficient Networks), multi-tier architecture.

2 Related Works

Christopher et al. developed two intelligent investigation techniques for WSN intrusion detection. The model was developed based on the backpropagation of the Support Vector Machine (SVM) supervised classifier module. The method is deployed in the investigation of six types of cyber threats. The dataset used is NSL-KDD data. The advantage of the model is that it holds good for low sample sizes and provides highly accurate favourable detection rates. The drawback is that it suits less sample size linked with the FPR rate of less than 1% [27].

Almomani et al. proposed and formulated a benchmark dataset for detecting four significant cyber-attacks. The attacks include Blackhole, Grayhole, Flooding and Scheduling attacks. The LEACH protocol is employed as a routing protocol in the WSN network. The dataset developed when analyzed by the NS-2 simulator illustrates about 23 features. The dataset is named WSN-DS. The Artificial Neural Network (ANN) model helps in the detection and classification of attacks. WEKA toolbox helps in the validation of the dataset. It excludes the wormhole or Sybil attacks [28].

Anthi et al. introduced a threefold intrusion detection model. The module aids in the supervised detection of weak links in IoT networks. Initially, the type and the behaviour of devices connected to the IoT link of interest were classified. Malicious packs are identified, which can cause attacks on the network in the second stage. In the third stage, the attacks were categorized. This scheme is deployed to identify 12 attacks in 4 IoT network links [29].

Chaabouni et al. performed a detailed survey classifying the IoT attacking threats and the challenges. The survey mainly attempted studying the machine learning approaches for tracking the attacks with their accuracy rates. The survey shows that drastic improvement is accomplished in tracking intrusion by machine learning than traditional techniques [30].

Yin et al. proposed a learning model for intrusion detection systems based on the Recurrent Neural Network (RNN) model. The model tracks the attack based on binary classification and multilevel classification. The accuracy is higher than most machine learning algorithms such as ANN, Random Forest Model, and SVM [31].

3 Proposed Framework

3.1 System Overview

Fig. 1 shows an overview of the proposed multi-tier SHOLEN IDS (Spotted Hyena Optimized Long Efficient Networks-speed intrusion detection systems) system suitable for the real-time environment. The first tier of the proposed framework consists of real-time data collection and is classified into two phases. In the first phase, real-time implementation of the wireless nodes for data gathering in an energy-saving hierarchal clustering environment. In the second phase, injections of the different attacks in the network were carried out. At the second tier, different characteristics were retrieved from the pre-processed data utilized to train the suggested model. SHOLE-networks were built at the third tier to anticipate various types of assaults. The present research sets forth the suggested methodology for predicting the malicious node and attack type.

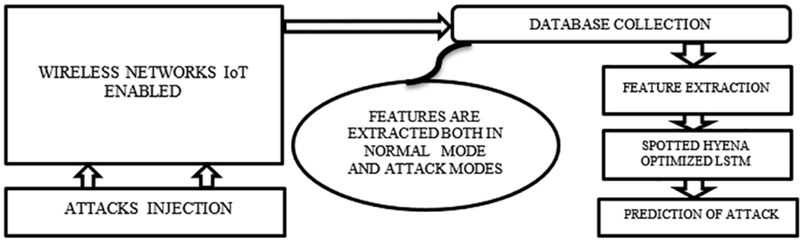
WIRELESS NETWORKS IoT ENABLED
ATTACKS INJECTION
DATABASE COLLECTION
FEATURES ARE EXTRACTED BOTH IN NORMAL MODE AND ATTACK MODES
FEATURE EXTRACTION
SPOTTED HYENA OPTIMIZED LSTM
PREDICTION OF ATTACK

Figure 1: Overall framework for the proposed architecture–SHOLEN IDS

3.1.1 Real-Time Data Collection

Wireless Sensor-IoT Assisted Model

Typical wireless sensor-IoT assisted model consists of sensor nodes, cluster heads, a base station (BS) and aggregation nodes [32]. For the stable operation of the network, the WSN-IoT devices are clustered and cluster heads transmit the collected information to the base station (BS) that reaches the central monitoring station (CMS) through the Internet. The user can remotely monitor the nodes' activities and control the nodes by issuing different tasks.

In this research, intelligent health care data monitoring and collection system have been considered. Fig. 2 shows the wireless model used in the research.

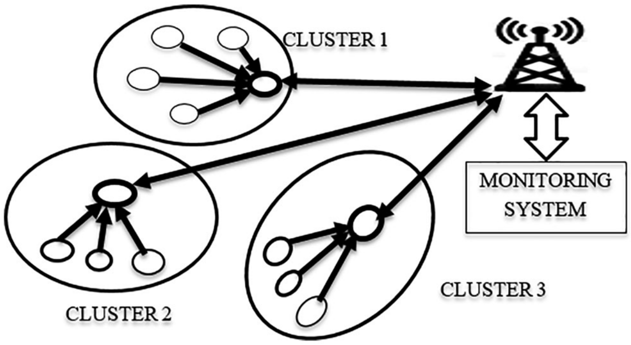

Figure 2: IP enabled wireless sensor model

The proposed IP-enabled wireless sensor nodes are deployed as clusters, cluster heads in the real-time environment, base station, and monitoring systems. The MICOTT (Node MCU and MCP3008) interfaced with four health care sensors used to design the sensor nodes; gateways are used as the base stations, and cloud (firebase) is used as the monitoring station. The complete specification of the real-time setup is shown in Tab. 1.

Table 1: Specification of wireless sensor nodes used for experimentation

Sl. no	Hardware used	Specifications
01	Heart beat sensor	MAX30100
02	Pulse rate	The standard male header connectors with three holes around the outside edge
03	Temperature sensor	High sensitive, LM35 sensor
04	Blood pressure sensor	
05	Sensor node CPU	MICOTT (ABEmake)-operating voltage is 3.3 V with 9 channel analog to digital convertors with 200 KHz sampling and inbuilt WIFI ESP8266 transceivers.
06	Monitoring station	Firebase cloud

The above specifications are used to incorporate real-time wireless testbeds and are primarily used for data collection under different scenarios such as normal conditions and attack conditions.

The assumptions of the IP enabled wireless sensor environment are quad folded and are given as follows

- This research uses clustered networks for data transmission, and nodes can communicate directly to the cluster heads. The cluster nodes can transmit the data to the monitoring station through internet services.
- Each Node is static and has a unique IP address, and belongs to one cluster. Each cluster is identified by another unique ID which consists of an IP address
- This research uses the hybrid data transmission model, including the event-based transmission and sustainability models.
- The nodes state includes a sleep state, idle state and active state.

This experimental setup is now used for the dataset collection unit. Nearly 15 nodes are used for experimentation in which the five nodes are randomly chosen for inducing the various attacks in the networks. The various types of attack models which are used for emulating the networks are listed in Tab. 2.

Table 2: List of attacks used in the experimentation

SI. no	Types of attacks	Description of attacks
01	DoS attacks	Creates unwanted traffic in the network and sends more malicious nodes [33,34]
02	DDoS attacks	
03	Wormhole attacks	Observation/copying the data from one node to another [35]
04	Sybil attacks	The introduction of duplicate cluster heads in the networks [36] affects the routing and increases packet drops.
05	Probe attacks	Introduces the numerous malicious data packets after obtaining the request of access in the networks [37].
06	R2L	
07	U2R	

Feature Extraction

Feature extraction plays an important role to differentiate the normal nodes and attack nodes. These features are calculated using real-time testbeds by using various mathematical expressions listed in Tab. 3.

Table 3: Types of features and their extraction mechanism

Sl. no.	Features used	Descriptions	Mathematical expression
01	Node ID	The IP address allocated for each sensor nodes	Static IP address for each nodes(192.168.1.23)
02	Cluster-ID	The IP address of cluster heads	The static IP address for cluster heads nodes (192.168.1.40)
03	Initial energy of nodes	Initial Energy is calculated at the beginning of the experimentation rounds. This Energy is used to transmit the sensors data to the cluster heads.	E(i) = [E(n)]*n Where E(i) is the Initial Energy, E(n) is the energy of wireless sensor nodes and n–no of bytes of transmitted data
04	Energy consumption (Ec).	The consumed energy is calculated at every iteration of data transfer between the nodes and sink	$Ec \ = [Erx \times Nbs] + [Etx \times \{Nts \times d\}] \quad (1)$ Where, Ec = Energy Consumed for every iteration, Erx = received Energy for the data from cluster heads to sensor nodes, Etx = Transmission energy of the nodes/distance which is then elaborated as Etx = [E(n)]*n] × d2, Nch = No of data bytes transmitted from nodes to cluster head, d = Distance between the nodes and the base station

(Continued)

Table 3 (continued)

Sl. no.	Features used	Descriptions	Mathematical expression
05	Residual energy (Er)	The residual energy is the remaining energy after each iteration	$E(i) - Ec = Er$ (2) Where E (i) is the initial energy, and Ec is energy consumption.
06	Cluster head distance (Dch)	The distance between the Cluster head and other nodes	As been kept constant and calculated manually after the selection of cluster heads
07	Throughput (T)	The throughput measures as the ratio between the received bytes to the transmitted bytes (at Cluster head Side)	$\mathrm{T} = D(CH,\ R)/(D(N(i),\ T) \times 100$ (3) Where, D(CH, R)-Data received at Cluster head, $D(N(i),\ T)-$Data Transmitted From each and every Nodes
08	Throughput-2 (T)	The throughput measures as the ratio between the received bytes to the transmitted bytes (at Base station)	$\mathrm{T} = D(BS,\ R)/(D(CH(i),\ T) \times 100$ (4) Where, D(CH, R)-Data received at base stations $D(CH\ (i),\ T)-$Data Transmitted From each and every cluster heads
09	Delay (ms)	Time calculated for data to reach from nodes to cluster heads	$Delay = T(R) - T(T)$ (5) T(R) is the time taken to reach the destination T (T) is the time to transmit from the source nodes.
10	Received signal strength indicators (RSSI)	This parameter is predominantly used to detect signal strength which is in turn used to calculate the distance.	RSSI is calculated using the mathematical expression as mentioned in [34]

The above features are stored separately in CSV file formats and pre-processed for further prediction/detection of the attacks.

3.2 Proposed Learning Models

In this section, the new hybrid integration of Spotted Hyena Optimizer (SHO) [37] and LSTM networks [38] has been discussed. Even though LSTM is considered more advantageous than the existing RNN, it suffers from overfitting problems when large datasets also lead to computational complexity [38]. It leads to the LSTM failure in achieving the higher accuracy detection. The paper proposes the computationally efficient hybrid modelling of LSTM with Spotted Hyena Optimizers, which can overcome the problem mentioned above and accurately predict the different categories of attacks in WSN-IoT networks to overcome this drawback.

In the proposed model, training LSTM cells can find the best weights and bias values. Two steps should be carried out to formulate the problem of training the LSTM cells using Spotted Hyena Optimizers (SHO).

- A suitable way of representing the bias weights in LSTM cells
- Formulating the Fitness Function

The weights of the LSTM cells are given by the mathematical expression below

$$\mathrm{V} = \{w1,\ w2,\ w3,\ w4,\ w5,\ \ldots wn,\quad \theta 1,\ \theta 2,\ \theta 3,\ \theta 4 \ldots \theta n\} \tag{6}$$

where wn are the weights of each LSTM cell and θn is the desired threshold values for obtaining the best threshold weights. The fixing of the objective function is the essential mechanism. The primary objective of training the feedforward deep learning models is to reach the highest classification in terms of prediction/detection accuracy for both training and testing samples. In this proposed model, the fitness function for training the LSTM cells is given by

$$fitness\ function = minimum\ (MSE) \tag{7}$$

where

$$MSE = \sum_{k=1}^{t}\sum_{i=1}^{m}(O_i^t - E_i^t)\frac{1}{N} \tag{8}$$

O_i^t is the threshold outputs, E_i^t is the obtained outputs from each LSTM cell, and N is the number of training samples. Usually, the performance of the learning models is measured by Mean Square Error, whose outputs are calculated by the mathematical expression (6). The fitness function is fixed to the lowest MSE so that the highest accuracy of prediction can be obtained for scalable real-time datasets. The new SHOLEN model, which has been formulated, are given as follows:

Sl. No. **Algorithm 1** // Pseudo Code for the Proposed SHOLEN Models

```
01  Input = {f1, f2 , f3, f4, …f10} Let f be the Input Features
02  Output: Categorization of the attacks
03  Number of Epochs: 50
04  Assign the Input bias and weights randomly
05  While (true)
06  Calculate the output value from the LSTM cell using Eq. (6)
07  Calculate the MSE using the Eq. (8)
08  If (MSE> threshold)
09  For t = 1 to Max_iteration Max_iteration ;50
10  Assign the bias weights and input layers by the position of the vector equation $\vec{P}(x+1) = \frac{\vec{C}_f}{N}$
11  Calculate the output value from the LSTM cell using Eq. (6)
12  If (MSE== threshold)
13  Go to Step 18
14  Else
15  Go to Step 9
16  End
```

(Continued)

Algorithm 1 (continued)

```
17  End
18  If (output value <=1)
19  / Normal Data traces / // No traces found
20  Else if(output value <=2&& output value > 1)
21  / DDos attack is detected
22  Else if Else if(output value <=3&& output value > 2)
23  Wormhole attack is detected
24  Else if Else if(output value <=4&& output value > 3)
25  Sybil attack is detected
26  Else
27  Go to step 18
28  End
29  End
30  End
```

The complete working of the proposed deep learning model has been presented in the pseudo-code, as shown in Fig. 3.

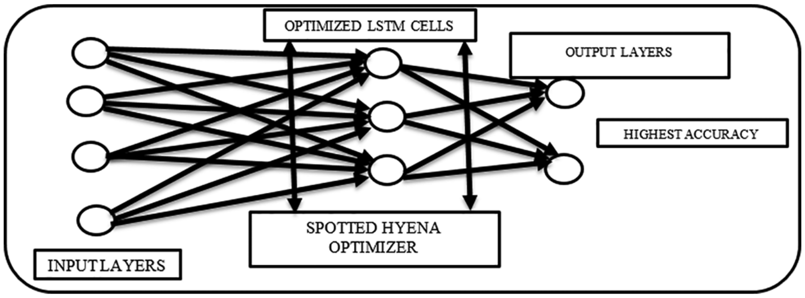

Figure 3: SHOLEN framework for the 3-layered LSTM cells

4 Experimental Setup

The real-time experimentations are carried out as discussed in Section. Nearly two months of different traces of data were accumulated and used for further analysis. The details of the data traces which are collected during the experimentation are given in Tab. 4.

32,273 records of data were collected for 2 months, and it has been used to train and test the proposed model. The analysis was developed using Python-Tensorflow 1.3. The version runs on an i7 CPU, 2.4 GHz, 2TB HDD, 16GB RAM, and 2GB NVIDIA Geoforce.

Table 4: Total number of data traces recorded in the experimentation

Sl. no.	Details of the data	No traces were recorded per day	Total traces
01	Number of normal data	19,600	
02	Number of DoS/DDoS data	17,225	
03	Number of Sybil data	15,400	62,273
04	Number of wormhole data	5225	
05	Number of probe data	1223	
06	Number of RPL	1200	
07	Number of	2400	

The other universal benchmarks such as CIDDS-001 [35], UNSW-NB15 [32] and NSLKDD [33] are used to evaluate the model under the various scenario to validate the performance of the proposed model. These data sets contain live traffic with different attacks: 80,000 on average and 20,000 DoS records in a total area are covered by the CIDDS-001 features. UNSW-NB15 benchmarks have 49 characteristics with one class label and roughly 1,75,000 cases, comprising standard and attack data. For NSL-KDD, each instance of regular traffic has 37,000 label attributes, while each instance of malicious traffic has 45,332.

4.1 Performance Metrics

Accuracy of prediction is used as the total number of correct decisions whether the incident of an attack happened to evaluate the performance of the proposed model. The metrics such as precision, recall, specificity and f-score were used for the analysis, representing the number of identified attacks in the networks, false identification of attacks by the model and the number of cases identified as usual, respectively.

$$Accuracy\text{:}\ \frac{TP + TN * (100)}{Total\ Testing\ Samples} \tag{9}$$

$$Precision = \frac{TP}{TP + TN} \tag{10}$$

$$Recall = \frac{FP}{TP + TN} \tag{11}$$

$$Specificity = \frac{FN}{FN + TP} \tag{12}$$

TP are True positive values, TN-True Negative values, FP-False positive and FN-False-negative values.

4.2 Results and Discussion

This section highlights various performance analyses of the proposed deep learning models juxtaposed with the existing deep learning models specific to the real-time datasets and other benchmarks. In this validation, we have used the different ratios of data splitting that has been used for training and testing to prove the efficiency of the proposed deep learning model.

4.2.1 SHOLEN-IDS Experimental Results

We have implemented SHOLE-spotted hyena optimized LSTM models for an optimal deep learning model selection and performed 10 iterations using the real-time datasets as shown below. The test datasets were set to a ratio of 80% training and 20% testing. Moreover, the SHO process tuned the model's hyperparameters to obtain the best accuracy results. The epochs are optimized to 100 with an output batch size of 50 and a learning rate of 0.0001.

Figs. 4–9 shows the detection accuracy (%) for the proposed learning models using the real-time testbeds. In all the cases, the average detection accuracy is 99.89% for predicting the different categories of attacks in the network. Moreover, optimized 50 epochs demonstrated more accuracy, almost equal to 99.89%, with the stability of the learning network. Further, we have also calculated the other performance metrics, which are presented in Tab. 5 below:

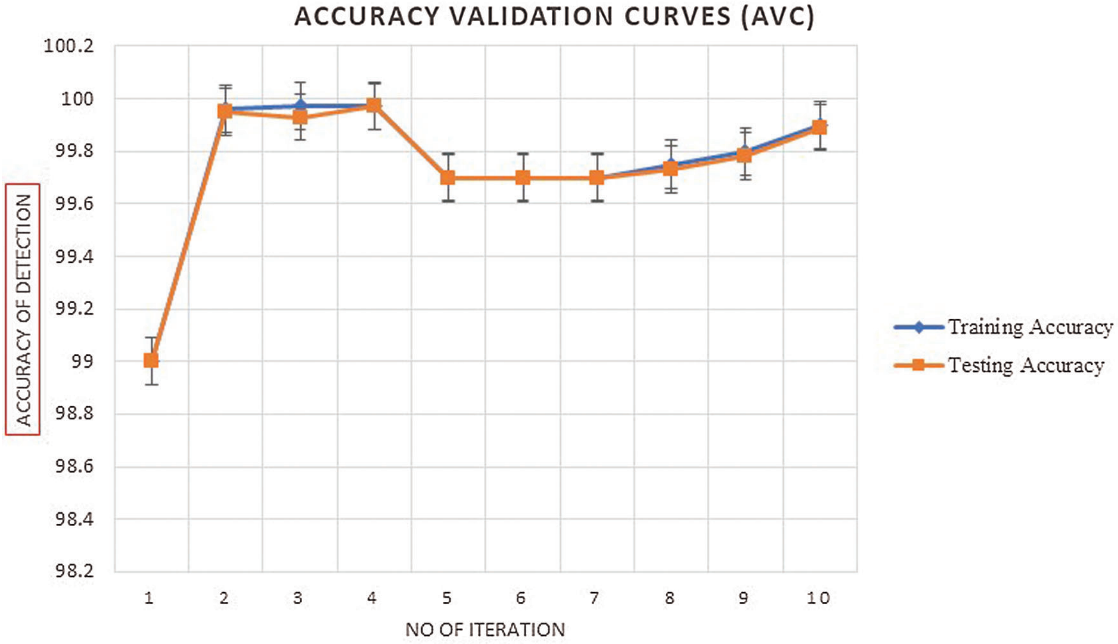

Figure 4: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets at 50 epochs

Tab. 5 depicts the performance of the proposed model in detecting the various attacks. The proposed model exhibits good precision, recall, sensitivity, and f-score to detect the attacks. The optimized learning model has proved its efficiency in detecting attacks using the real-time scenario. In the following scenario, we have trained the proposed model with different benchmarks as mentioned in Section 4.1 and calculated performance metrics under various scenarios.

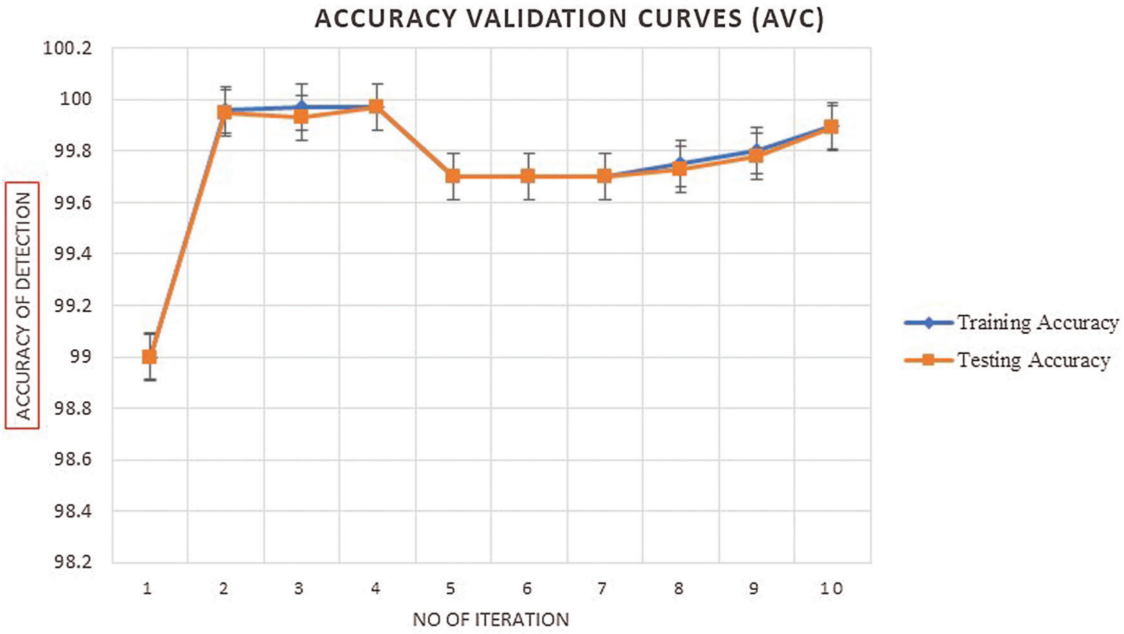

Figure 5: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets in detecting the dos attacks at 50 epochs

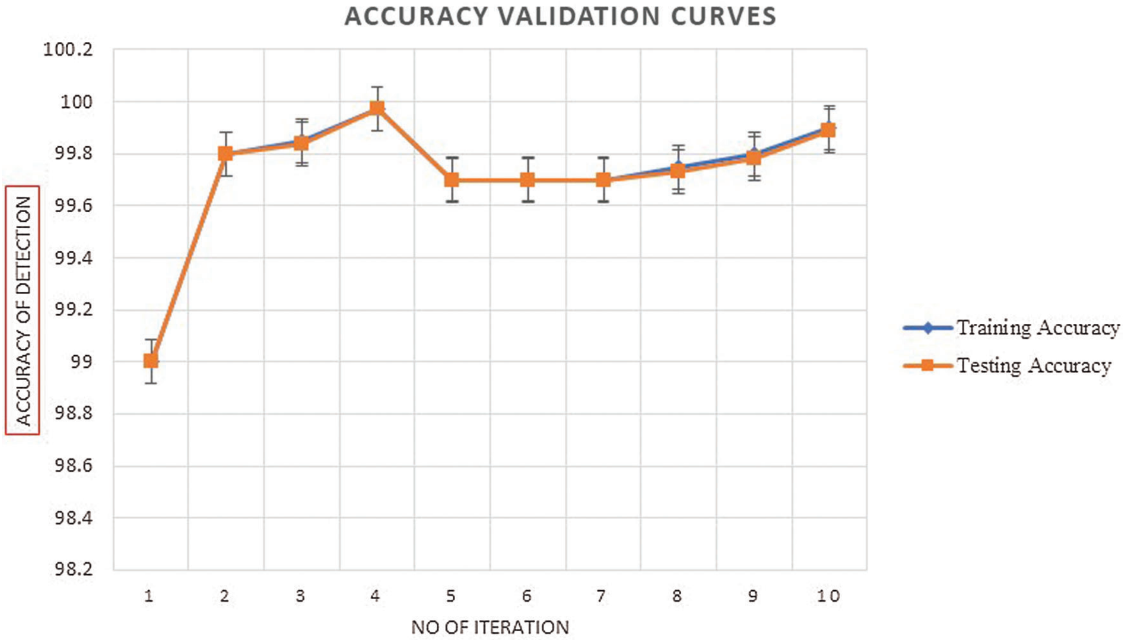

Figure 6: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets in detecting the DDoS attacks at 50 epochs

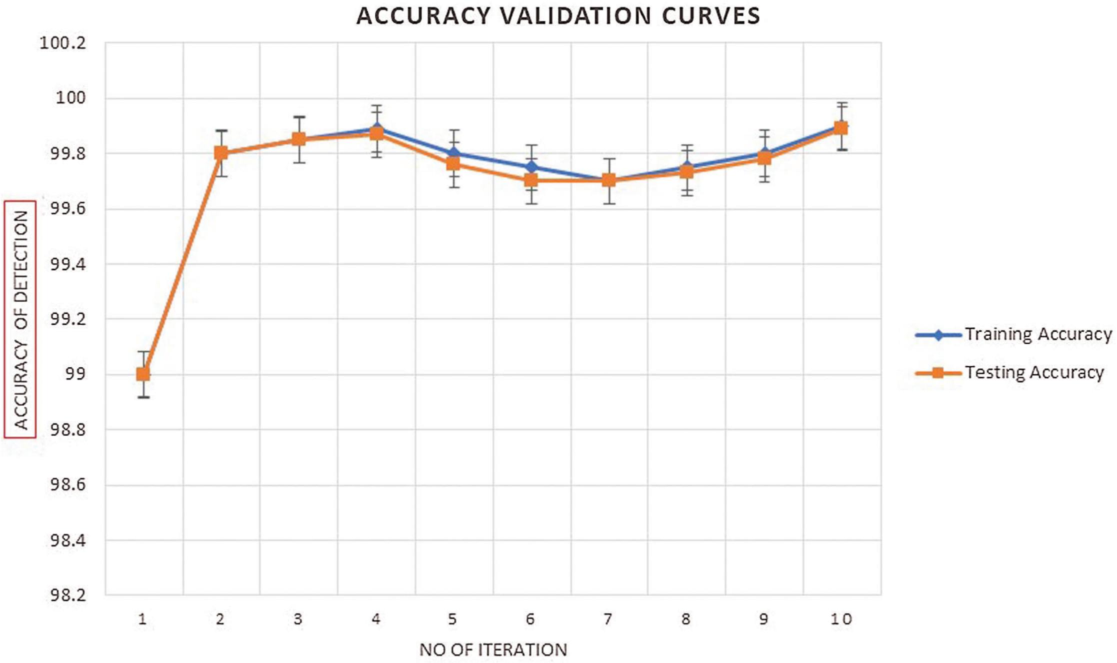

Figure 7: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets in detecting the Sybil attacks at 50 epochs

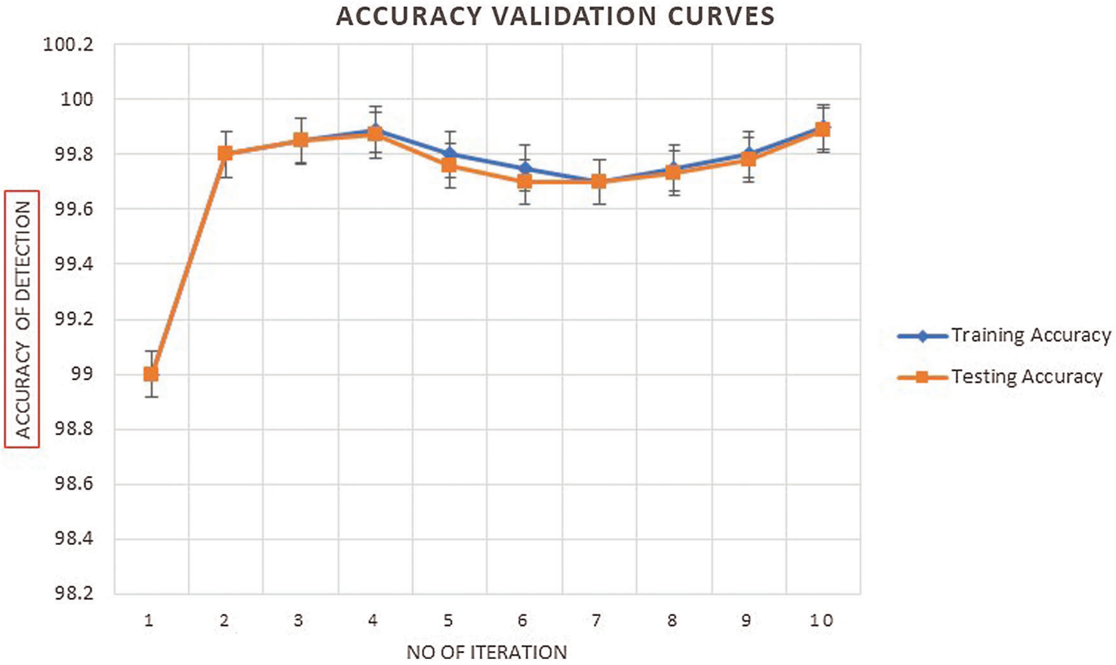

Figure 8: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets in detecting the wormhole attacks at 50 epochs

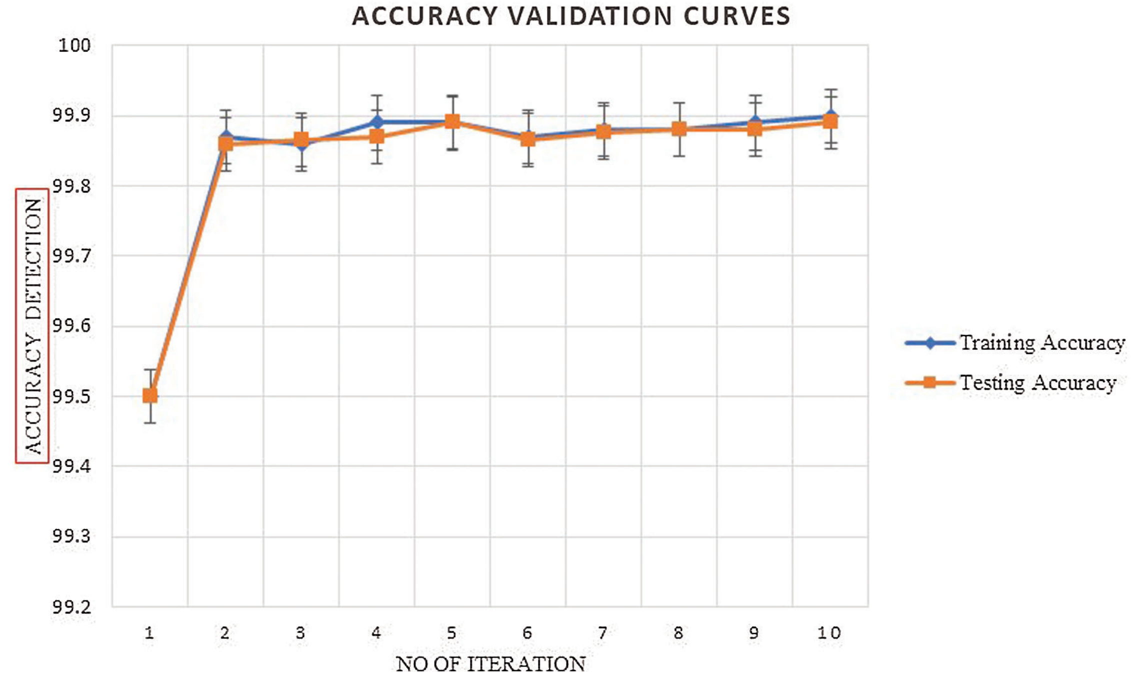

Figure 9: Accuracy validation mechanism for the proposed SHOLEN models using the real-time datasets in detecting the RPL and URP attacks at 50 epochs

Table 5: Performance metrics for the proposed model in the detection of categories of attacks

Sl. no	Types of Attacks	Precision	Recall	Sensitivity	F-Score
01	DoS attacks	0.9986	0.9978	1.00	0.9815
02	DDoS attacks	0.998	0.998	0.994	0.9814
03	Sybil attacks	0.9987	0.9988	0.978	0.9786
04	Wormhole attacks	0.9986	0.9992	0.989	0.956
05	Probe attacks	0.9987	0.978	0.982	0.966
06	RPL	0.988	0.976	0.978	0.9672
07	URP	0.986	0.967	0.977	0.9745

The performance metrics of the proposed algorithm is calculated with the benchmark datasets and presented in Tabs. 6–8. Tab. 6 presents the performance metrics of the proposed algorithms using UNSW-NB15 Benchmarks; performance metrics such as accuracy, precision, recall, Sensitivity, and F-score has demonstrated a very stable performance, ranging from 99% to 100% in detecting the various categories of attacks. Similarly, Tab. 7 depicts the performance metrics of the proposed algorithm in detecting the attacks using NSL-KDD datasets 99 in which the accuracy ranges from 99.87% to 99.64%, 99.5% to 98.67% precision, 98.50% to 97.50% recall, 99% to 98.67% sensitivity and 98.23% to 97.02% F-score. Finally, Tab. 8 shows the performance of the proposed model using the CIDCC datasets 2017, exhibiting a very high performance in detecting the various attacks. The experimental result shows that the integration of the spotted Hyena optimizer in LSTM has exhibited a high performance in which it has reduced the high false alarm rates.

Table 6: Performance evaluation of proposed model using the UNSW-NB15 benchmarks

Sl. no	Benchmarks used	Attack types	Performance metrics				
			Acc (%)	Precision	Recall	Sensitivity	F-score
01	UNSW-NB15	Dos	99.89	0.9997	0.9989	0.9934	0.9815
		DDos	99.90	0.9984	0.9991	0.9943	0.9765
		RPL	99.95	0.9882	0.992	0.9950	0.9723
		URL	99.89	0.9875	0.974	0.9873	0.9726

Table 7: Performance evaluation of proposed model using the NSL-KDD datasets benchmarks

Sl. no	Benchmarks used	Attack types	Performance metrics				
			Acc (%)	Precision	Recall	Sensitivity	F-score
01	NSL-KDD DATASETS 99	Dos	99.87	0.9995	0.9974	0.9945	0.9823
		DDos	99.88	0.9945	0.9932	0.9934	0.9745
		RPL	99.76	0.9856	0.993	0.9945	0.9719
		URL	99.64	0.9867	0.985	0.9867	0.9702

Table 8: Performance evaluation of proposed model using the CIDCC 001 benchmarks

Sl. no	Benchmarks used	Attack types	Performance metrics				
			Acc (%)	Precision	Recall	Sensitivity	F-score
01	CIDCC datasets	Dos	99.87	0.9815	0.9924	0.9945	0.9823
		DDos	99.86	0.9812	0.9930	0.9934	0.9745
		RPL	99.75	0.9678	0.9912	0.9945	0.9719
		URL	99.55	0.9745	0.9715	0.9867	0.9702

4.2.2 Comparative Analysis

In this section, we have compared the existing deep learning algorithms such as RNN [31], ASCH-IDS [30], Improved GA based DNN [38] and DBN [22] algorithms for the detection of various attacks using the collection of the real-time dataset.

The performance metrics of different hybrid deep learning models have been compared for real-time datasets, as shown in Fig. 10. It is evident from Fig. 10 that the proposed algorithm has outperformed the other learning models with an accuracy of 99.89%. The AHCS-IDS has reasonable accuracy of 95%, whereas other algorithms have even lesser performance than AHCS-IDS. The integration of the SHO in LSTM has shown stable performance in the real-time test data, which makes it suitable for the real-time scenario. The other benchmark datasets are taken to compare deep learning models' performance in Figs. 11–13. Experimental results showed the accuracy of all hybrid models ranges from 99.989% to 99.65%; precision ranges from 95.0% to 98%, recall ranges from 96% to 97%, and F-score ranges from

97%–98%. Hence, the performance of the proposed model has a decisive edge over other deep learning models. It was found that the proposed SHOLEN model has shown stable performances with the real-time data and also with the other benchmarks. However, the performance of other hybrid deep learning models has shown darker performance in real-time datasets and brighter under benchmarks. Our proposed SHOLEN model has demonstrated its suitability for a real-time dataset applied in smart health care monitoring systems considering the above results.

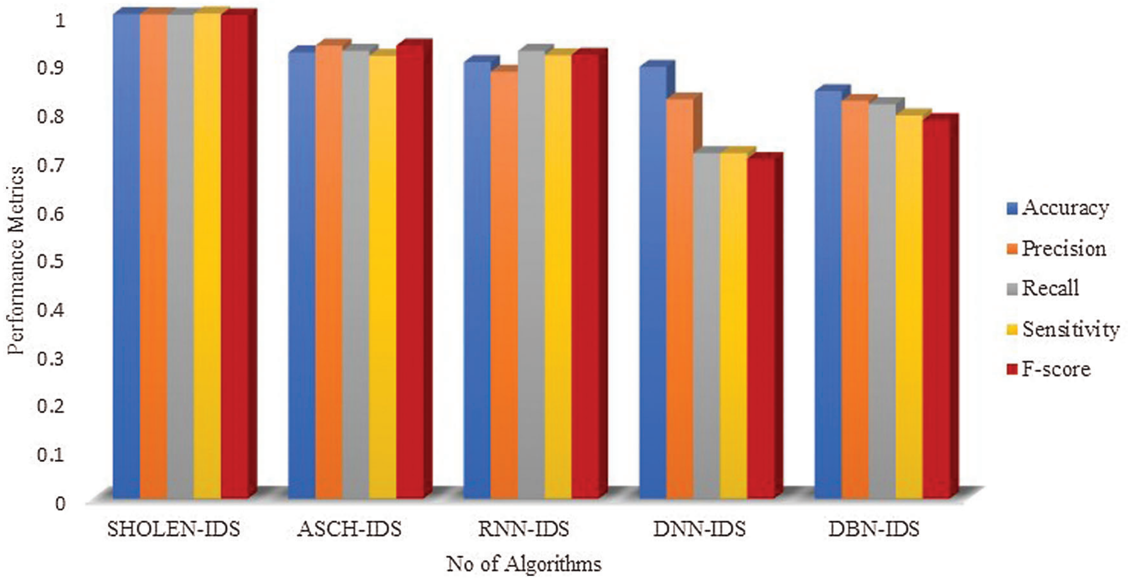

Figure 10: Comparative analysis between the different deep learning modes using the real-time datasets

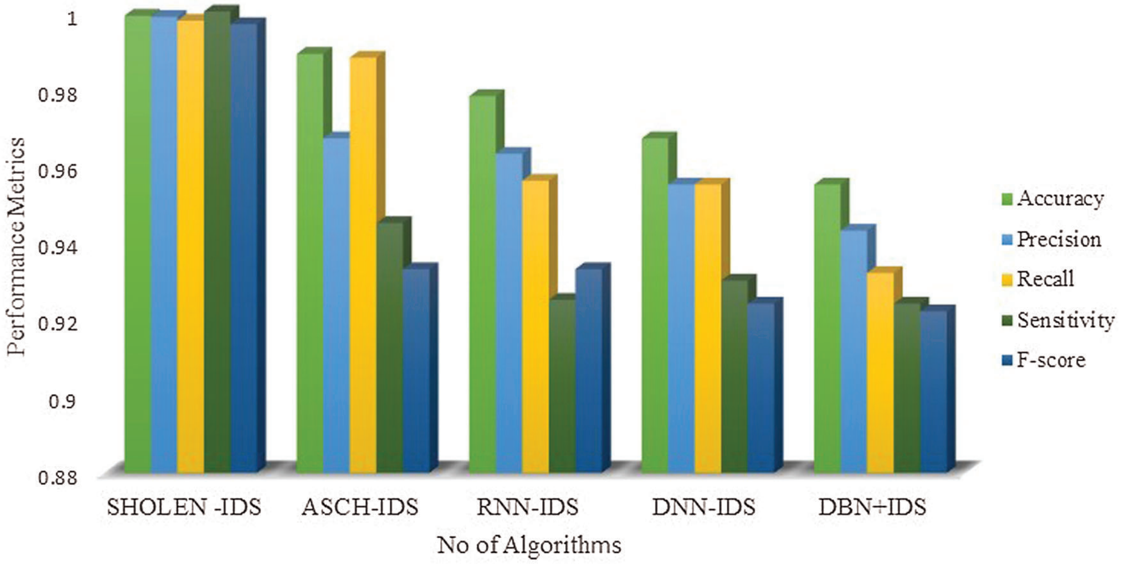

Figure 11: Comparative analysis between the different deep learning modes using the NSL-KDD++ datasets

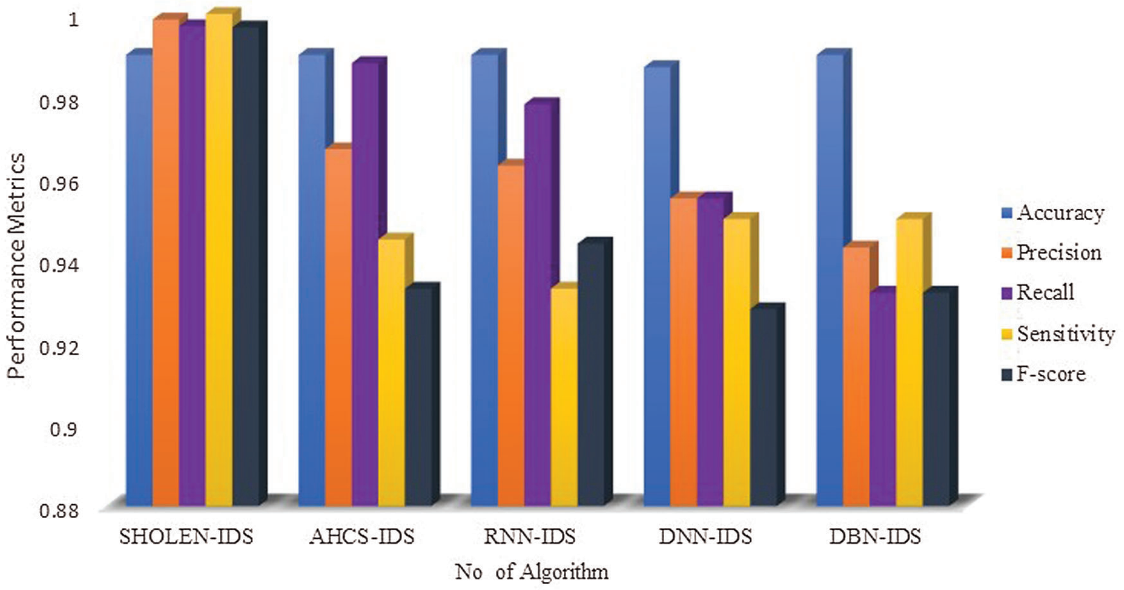

Figure 12: Comparative analysis between the different deep learning modes using the UNSW-NB15 benchmark dataset

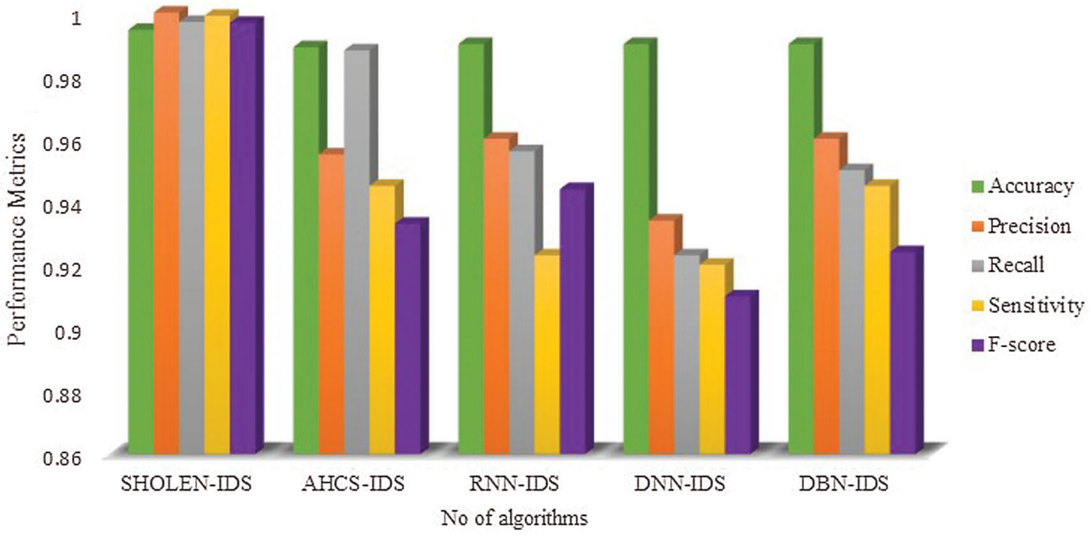

Figure 13: Comparative analysis between the different deep learning modes using the CIDCC-dataset 2017

5 Conclusion

The Spotted Hyena Optimizer integrated LSTM networks have been proposed in the paper. The proposed algorithm is applied to the clustered WSN-IoT environments, architecting the multi-tier WSN-IoT intrusion detection system model. The real-time datasets were collected using the embedded CPU interfaced with esp8266 transceivers. Nearly 60,000 data were collected and used for training and testing. Further, the proposed algorithm has been compared with the other hybrid deep learning models under two scenarios: real-time environments and benchmark datasets. The results show that the proposed algorithm has shown consistent performance, such as accuracy of 99.89%, precision of 98%, 97.5% recall, and 99%

f-score in predicting the various categories of attacks in the WSN-IoT environment using scalable datasets. The other hybrid deep learning models have exhibited exemplary performance using benchmarks but showed a marked dip in performance while using the real-time datasets. The experimentation shows that the proposed deep learning model has shown better performance for different datasets and can gain immense traction in innovative health care monitoring applications. Further, the proposed models need to focus on identifying the increased attacks without compromising the reduction of energy consumption.

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. Oliveira and J. J. 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, R. Chen and M. J. Chang, "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, 2016.

[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, L. A. Guido and M. Marchetti "On the effectiveness of machine and deep learning for cyber security," in *Proc. of Int. Conf. on Cyber Conflict (CyCon)*, 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] R. Mohammed Harun Babu, R. Vinayakumar and K. P. Soman, "A short review on applications of deep learning for cyber security," *Cryptography and Security*, pp. 1–15, 2018, ArXiv preprint arXiv: 1812.06292.

[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. 122, pp. 1–35, 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," *Neuro Computing*, 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: how do IoT devices use AI to enhance security?," in *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 41–49, 2018. https://dx.doi.org/10.1109/MSP.2018.2825478.

[15] C. Cheng, W. P. Tay and G. B. Huang, "Extreme learning machines for intrusion detection," in *Proc. of Int. Joint Conf. on Neural Networks*, Brisbane, QLD, Australia, pp. 1–8, 2018.

[16] Z. Wenjie Z, H. Dezhi, K. C. Li and F. I. Massett, "Wireless sensor network intrusion detection system based on MK-eLM," *Soft Computing*, vol. 24, pp. 12361–12374, 2018.

[17] N. Shone, T. Nguyen Ngoc, V. Dinh 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.

[18] K. Wu, Z. Chen and W. Li, "A novel intrusion detection model for a massive network using convolutional neural networks," *IEEE Access*, vol. 6, pp. 50850–50859, 2018.

[19] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han *et al.,* "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018.

[20] Y. Ding and Y. Zhai, "Intrusion detection system for NSL-kDD dataset using convolutional neural networks," in *Proc. of Int. Conf. on Computer Science and Artificial Intelligence*, Shenzhen, China, pp. 81–85, 2018.

[21] S. Otoum, B. Kantarci and H. T. Mouftah, "On the feasibility of deep learning in sensor network intrusion detection," *IEEE Networking Letters*, vol. 1, no. 2, pp. 68–71, 2019.

[22] N. Chouhan, A. Khan and H. U. R. Khan, "Network anomaly detection using channel boosted and residual learning based deep convolutional neural network," *Applied Soft Computing*, vol. 83, no. 105612, pp. 1–34, 2019.

[23] 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.

[24] J. Tian and M. Gao, "Intelligent community intrusion detection system based on wireless sensor network and fuzzy neural network," in *Proc. of Int. Colloquium on Computing, Communication, Control, and Management*, Sanya, China, pp. 102–105, 2009.

[25] H. Zhang, B. Zhao, H. Yuan, J. Zhao, X. Yan *et al.,* "SQL injection detection based on deep belief network," in *Proc. of Int. Conf. on Computer Science and Application Engineering*, New York, NY, USA, pp. 1–6, 2019.

[26] O. Faker and E. Dogdu, "Intrusion detection using big data and deep learning techniques," in *Proc. of ACM Southeast Conf.*, Kennesaw, pp. 86–93, 2019.

[27] D. Christopher and P. Andrei, "Investigation of computational intelligence techniques for intrusion detection in wireless sensor networks," *International Journal of Computer Networks & Communications*, vol. 9, no. 4, pp. 45–56, 2017.

[28] 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, 2016.

[29] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos and P. Burnap, "A supervised intrusion detection system for smart home IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042–9053, 2019.

[30] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac and P. Faruki, "Network intrusion detection for iot security based on learning techniques," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019.

[31] C. Yin, Y. Zhu, J. Fei and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[32] 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.

[33] I. Andrea, C. Chrysostomou and G. Hadjichristofi, "Internet of things: Security vulnerabilities and challenges," in *Proc. of IEEE Symp. on Computers and Communication*, Larnaca, Cyprus, pp. 180–187, 2015.

[34] C. Kolias, G. Kambourakis, A. Stavrou and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[35] M. A. Patel and M. M. Patel, "Wormhole attack detection in wireless sensor network," in *Proc. of Int. Conf. on Inventive Research in Computing Applications*, Coimbatore, India, pp. 269–274, 2018.

[36] S. Sridevi and R. Anandan, "RUDRA-A novel re-concurrent unified classifier for the detection of different attacks in wireless sensor networks," *Intelligent Computing in Engineering*, vol. 1125, pp. 251–259, 2020.

[37] Q. Lio, J. Li, Y. Zhou and L. Liao "Using spotted hyena optimizer for training feedforward neural networks," *Cognitive Systems Research*, vol. 65, pp. 1–16, 2021.

[38] Z. Chiba, N. Abghour, N. Moussaid, A. El Omri and M. Rida, "Intelligent approach to build a deep neural network based IDS for cloud environment using combination of machine learning algorithms," *Computers and Security*, vol. 86, pp. 291–317, 2019.