ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20230712000798
Original scientific paper

Blockchain Assisted Fireworks Optimization with Machine Learning based Intrusion Detection System (IDS)

Sudhakar THIRUVENKATASAMY*, Rajappan SIVARAJ, Murugasamy VIJAYAKUMAR

Abstract: In order to cope with the growing complexity of cyber attacks, it is imperative to have efficient intrusion detection systems (IDSs) that can monitor computer resources and produce data on abnormal or suspicious activities. The security of IoT networks is increasingly becoming a crucial concern as the Internet of Things (IoT) technology receives widespread use. Protecting the IoT framework with a conventional Intrusion Detection System (IDS) might be challenging due to the vast quantity and diversity of IoT devices. Traditional Intrusion Detection Systems (IDSs) face limitations when deployed in IoT networks due to resource limitations and the inherent complexity of these networks.This research proposed the Blockchain Assisted Fireworks Optimization with Machine Learning based Intrusion Detection System (BAFWO-MLIDS) technique in the healthcare platform. The major purpose of the BAFWO-MLIDS system is to apply BC technology (BCT) with IDS for enhanced security in the healthcare sector. The BCT enables to achieve secure data transmission in the healthcare platform. The BCT enables to achieve secure data broadcast in the healthcare environment. The BAFWO-MLIDS technique involves a three-stage procedure: FWO based FS process, ENN-based detection, and BO-based parameter optimization. In the proposed BAFWO-MLIDS technique, the FWO-based feature selection process is involved to select optimal features. For intrusion detection, the BAFWO-MLIDS technique uses Elman Neural Network (ENN) model. Finally, the Bayesian optimization (BO) technique is applied to modify the parameters compared with the ENN model and thereby it accomplishes enhanced detection performance. The simulation results of the BAFWO-MLIDS system can be inspected in a series of experiments and the obtained results ensured greater efficiency of the BAFWO-MLIDS methodology with other recent algorithms.

Keywords: bayesian optimization; blockchain; healthcare; intrusion detection; machine learning; security

1 INTRODUCTION

The Internet of Things (IoT) is mainly dependent upon interrelated smart devices, and various services can be employed to incorporate them as a single network [1]. It enables smart devices to collect confidential data and perform main functions, and these devices connect and interact with one another at maximum speed and make decisions based on indicator data [2]. The IoT platform exploits cloud service as a backend for maintaining remote control and processing information. Client user uses web services or mobile applications for data accessibility and controlling the devices [3]. IoT framework uses millions of sensors for extracting relevant data, and this data can be analyzed by artificial intelligence (AI) technology. Intrusion detection systems (IDSs) are the administrative, technical, and regulatory systems used to avoid abuse, unauthorized access, and recovery of computer data and communication methods and the data by comprising [4], used for enhancing the privacy, protection, and confidentiality of personal information by taking all measures and ensuring the continuity and availability of the information system [5].

Network intrusion occurs as a result of potential packets in the network model, to execute performances like Denial of Service (DoS) attacks. DoS attack attempt to create PC resources distant from its planned clients, for instance, flood attacks, land attacks, and ping of death (POD) [6]. Indications of intrusion integrating abnormal outcomes while implementing client charges including moderate system implementations, and unexpected system crashes and modifications in parts of a data structure are, bizarrely, moderate system executions (for example, accessing sites or opening records). AI is a type of data-driven approach where the first step is to understand the data. The advancements in deep learning (DL) techniques assist in detecting these behaviors of the network [7]. Extensive research has presented the development of security of network models. AI has played a very important role in the field of cybersecurity and it depends on IoTs to design an intelligent method for security in the IoT infrastructure [8]. Combining blockchain (BC) technology with DL approaches for the detection and prevention of intrusion can offer many possible advantages to enhance the network security system. BC technology (BCT) provides an immutable and decentralized ledger that can be used for storing records and logs related to security events and network activities [9]. The information becomes tamper-proof and transparent, which ensures the integrity of the logs by storing intrusion detection and prevention data on the BC [10]. This can be especially helpful during auditing and forensic analysis.

2 RELATED WORKS

The authors [11] introduce a novel approach in the IoT platform namely BCAided with IDS Differential Flower Pollination Using DL (BAIDS-DFPDL) method. The proposed BAIDS-DFPDL system mostly considers the classification of intrusions and the detection in the IoT platform. To accomplish this, the existing method has followed the BC technique for efficiency and to protect the information transmission between the representatives. Moreover, the existing BAIDSDFPDL system designed Differential Flower Pollination based FS (DFPFS) method to select the features. Lastly, sailfish optimizer (SFO) with RBM technique is utilized for effective identification of intrusions. In [12], the authors suggested BC deep-rooted Bi-level ID and graph-based mitigation model called Hybrid-Chain-IDS. Primarily, time-based authentication to validate the legal users utilizing the NIK512 hashing procedure is performed. Afterwards, user planning which applied Cheetah Optimizer Algorithm (COA) is achieved which decreases the difficulty, and access control is offered to authorize users. Furthermore, bi-level ID utilizing ResCapsNet is achieved which can extract the adequate features and classify efficiently. Lastly, the risk of attacks

can be estimated, and the attack graphs are produced by using an improved KNN procedure. The authors [13] suggest a deep BC framework (DBF) created to offer the privacy-based BC with smart contracts (SCs) in IoTs networks and security-based distributed ID. The ID technique is used by the BiLSTM-DL algorithm to handle consecutive network information and is measured by utilizing the information sets of BoT-IoT and UNSW-NB15. The developments of the SC technique and privacy-based BC are applying the Ethereum library to offer privacy for distributing ID engines. The DBF model is the comparison with peer privacy maintaining ID methods.

Ashraf et al. [14] propose a method called FIDChain IDS that utilizes weightless artificial neural networks (ANNs) in a federated learning (FL) setting to protect the privacy of medical information. This technique combines the use of blockchain (BC) to provide shared records, which are then combined with weighted averaging to prevent poisoning attacks and ensure transparency and stability in the distributed model. The eXtreme Gradient Boosting (XGBoost) and The ANNs system were estimated by applying the BoT-IoT database. In [15], the authors presented distributed ML-based IDS in IoT utilizing BCT. Specifically, spectral separating is suggested to separate the IoT network into autonomous systems (AS) permitting the traffic supervision for ID to be achieved by the designated AS-limited region nodes in a shared manner. The IDS is based on ML, where the SVM procedure is trained utilizing an important IoT database, and identification of the attackers is offered. Moreover, by using the BC technique, the integrity of the attacker's record is obtained. Alevizos et al. [16] introduce a method of BIDPS that advances ZTA onto end-points. The BIDPS targets achieve two core results: Primarily, they identify and prevent attacker's methods and strategy as per MITRE's ATT&CK enterprise matrix previous than laterally movement stage and secondarily, strip trust out of the end-point itself and place it on a chain, therefore producing an unchallengeable model of obvious trust.

The authors [17] suggest a security design that incorporates the BC and SDNs techniques. The intended security design collected from IDS, like RSL-KNN, is the combination of KNN and RSL to protect against spurious instructions, which aims to the industrial control process and BICS that have to avoid misrouted attacks that tamper with the Open-Flow rules of the SDN allowed industrialized IoTs models. Abunadi et al. [18] proposed the techniques of FLBIC-CUAV on the IoTs platform. The suggested FLBIC-CUAV approach involved 3 main processes such as BC allowed secure communication, FL-based image classification, and clustering. The beetle swarm optimizer (BSO) approach with 3 input constraints is considered for the process of UAV cluster construction to cluster the UAVs and efficient communication. Additionally, BC allowed secure information transmission process proceeds to transfer the information from UAVs to cloud servers.

3 THE PROPOSED MODEL

In this study, we have focused on the design and development of automated intrusion detection using the BAFWO-MLIDS technique in the healthcare environment. The major purpose of the BAFWO-MLIDS technique is to apply BCT with IDS for enhanced security in the healthcare sector. The BCT enables to achieve secure data broadcast in the healthcare environment. In the presented BAFWO-MLIDS technique, a three-stage process is performed namely FWO based FS process, ENN-based detection, and BO-based parameter optimization. Fig. 1 displays the workflow of the BAFWO-MLIDS system.

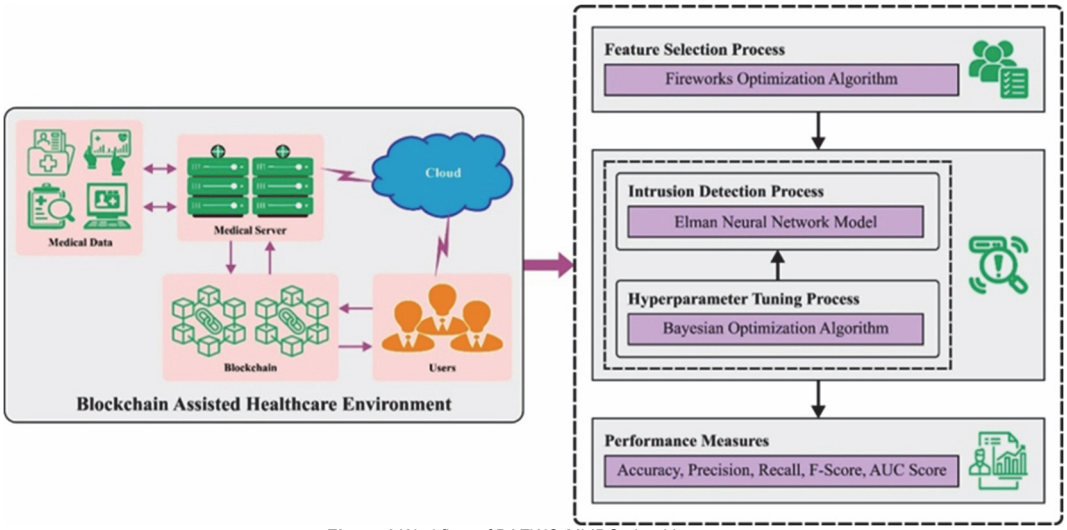

Figure 1 Workflow of BAFWO-MLIDS algorithm

3.1 BC Technology

The major purpose behind BCT is Decentralization and its transparent and distributed features of a ledger of the BCsuggest that the failure of a single node could not affect the complete system [19]. This design allows 2 parties to directly manage one another using encrypt and safety dependent upon code and algorithm security. Meanwhile, the parties participating in the transaction method are needed only to trust the method utilized for presenting mutual trust, it is not necessary for information about the reliability of parties. Furthermore, the structure did not need any security authorization by 3rd-parties as the method takes complete responsibility for each kind of

endorsement. Ethereum and Hyperledger Fabric are the 2 common BC application development environments. The basic technology is the same. The major difference between Hyperledger and Ethereum lies in the target users and the approach thatcan be designed. Public BCs and SCs are focused on applications, which are exploited by the common consumer. Hyperledger Fabric is an actual modular structure which is better adapted to business applications. It applies business logic more freely and provides great flexibility. The objective is to simplify trade and work procedures utilizing BCT, viz., to resolve the problems of inter-firm credit.

3.2 Feature Selection using FWO Algorithm

In the presented BAFWO-MLIDS technique, the FWO algorithm is applied for choosing features. FWO algorithm primarily comprises four stages: mutation, explosion, selection, and evaluation [20]. Especially, an explosion is the arbitrary searching process in the solution space about the firework (FW), and the FW location is a solution candidate for the storing place assignment for a non-traditional warehouse layout. The steps to FWO approach are given below.

First, a certain amount of FW positions is produced in the searching space that generates a collection of sparks via exploding.

Next, the spark position can be attained by mutation and explosion. An FW with low fitness exploded with fewer sparks with a large amplitude, whereas an FW with high fitness exploded with a great number of sparks with a small amplitude.

Then, the FW quality position was developed by FF.

The sparks and FWs with higher fitness are chosen as a location (candidate solution) for the next generation's FWs.

At last, the optimizer ends once the maximal amount of iterations can be attained.

The amount of sparks is based on the FWquality and is formulated by Eq. (1).

$$s_j = N_0 \times \frac{f_{\max} - f + \xi}{\sum_{j=1}^{n}\left(f_{\max} - f\right) + \xi} \tag{1}$$

where $f_{\max}$ denotes the maximal value of objective function amongst nFWs. ξ represent the smaller constant in the computer, which prevents a zero-division error. f indicates the overall objective function. j shows the number of FWs. N_0 shows the parameter controlling the overall amount of sparks produced by the nFWs. Bounds for s_j are designed to avoid the overwhelming effect of splendid FWs as follows.

$$\hat{s}_j = \begin{cases} \text{round}\ \left(a\ N_0\right) s_j < a\ N_0 \\ \text{round}\ \left(a\ N_0\right) s_j < a\ N_0, a < b < 1 \\ \text{round}\ \left(s_j\right)\ \text{otherwise} \end{cases} \tag{2}$$

where the parameters a and b are constant.

In comparison with the proposal of spark numbers, the amplitude of the optimum FW explosion is lesser than the bad one. The amplitude of explosion for all the FWs is derived as follows:

$$A_j = \hat{A} \times \frac{f - f_{\min} + \xi}{\sum_{j=1}^{n}\left(f - f_{\min}\right) + \xi} \tag{3}$$

In Eq. (3), $f_{\min}$ shows the minimal value of the main function amongst nFWs, and A refers to the maximal explosion amplitude.

The position of spark q_e^u produced by q_j^u was attained by arbitrarily setting w dimension $\left(1 \le e \le s_j, 1 \le u \le w\right)$ as follows:

$$q_e^u = q_j^u + A_j \times \text{rand}\left(-1, 1\right) \tag{4}$$

In Eq. (4), w shows the arbitrary dimension of sparks, $= \text{round}\left(d \times \text{rand}\left(0, 1\right)\right)$ and d denotes the FW dimensional q_j.

Furthermore, a Gaussian distribution with a mean and standard deviation (*SD*) of 1 can be used for defining the coefficient of an explosion to maintain the spark's diversity. In explosion generation, a specific count of sparks can be produced.

The present optimum position x^* is often preserved for the next explosion generation at the beginning of the explosion generation. Next, the $n-1$ location was chosen based on the distance to other locations for maintaining the diversity of sparks.

$$p\left(x_j\right) = \frac{R\left(q_j\right)}{\sum_{e \in K} R\left(q_e\right)} \tag{5}$$

In Eq. (5), $R\left(q_j\right)$ shows the distance among place q_j and other places q_e, and K denotes the set of existing places of FWs and sparks as follows:

$$R\left(q_j\right) = \sum_{e \in K} d\left(q_j, q_e\right) = \sum_{e \in K} \left\| q_j - q_e \right\| \tag{6}$$

The optimum storage location assignment was attained as the evaluation reaches the desired evaluation point.

In the proposed work, the objective was combined as a single objective formula to a preset weight to detect all main significance [21]. The FF was executed that mixes both objectives of FS:

$$Fitness\left(X\right) = a \times E\left(X\right) + \beta \times \left(1 - \frac{\left|R\right|}{\left|N\right|}\right) \tag{7}$$

In Eq. (7) $Fitness\left(X\right)$, shows the fitness value of subset X, $\left|R\right|$ and $\left|N\right|$ denotes the count of FS and original

features in the datasets, $E(X)$ indicates the classifier rate of errors by applying the FS from the X subset, α and β refers to the weight of the classifier error and decrease ratio, $\alpha \in [0, 1]$ and $\beta=(1-\alpha)$.

3.3 Intrusion Detection Using ENN Model

For intrusion detection, the ENN approach was employed. ENN is a recursive network introduced by the internal self-referencing layer [22]. Context layer, hidden layer (HL), input layer, and output layer are the four components of ENN. Fig. 2 exemplifies the architecture of ENN. The following is the ENN proposal.

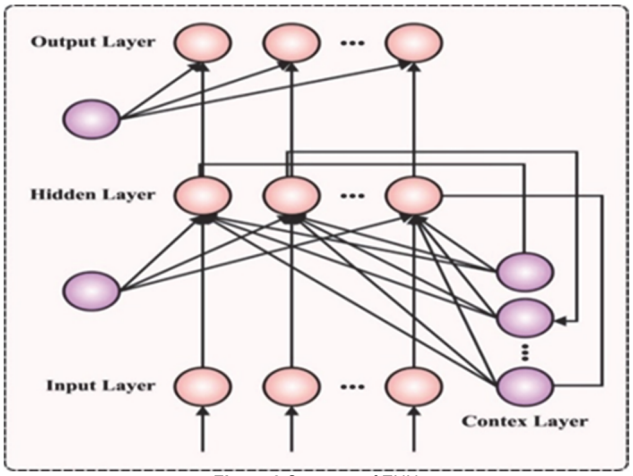

Figure 2 Structure of ENN

Input layer: the input value is the position tracking error e and its differential $\dot{e}$. The input and output of nodes are determined using Eq. (8).

$$\begin{cases} u_1^{(1)}(N) = e \\ u_1^{(1)}(N) = e \end{cases} \qquad o_1^{(1)}(N) = u_i^{(1)},\ i = 1, 2 \tag{8}$$

where N shows the iterations count and i denotes the count of neurons.

Context layer: in this layer, the nodes are expressed as follows.

$$u_k^{(2)} = \beta o_k^{(2)}(N-1) + o_j^{(3)}(N-1) \quad j = 1, 2, ..., 9 k = 1, 2, ..., 9 \tag{9}$$

In Eq. (9), $u_k^{(2)}$ and $o_k^{(2)}$ denote the input and output of k^{th} nodes in the layer, correspondingly. $o_j^{(3)}$ represents the output of the hidden layer. $0 \le \beta \le 1$ shows the self-connected feedback gain. j and k show the neuron counts from the context and HLs, correspondingly.

Hidden layer: in this layer, the nodes are determined as follows:

$$o_j^{(3)} = \frac{1}{e^{\theta j}},\ \theta_j = \sum_{i=1}^{2} o_i^{(1)} + \sum_{k=1}^{9} o_k^{(2)} \tag{10}$$

In Eq.(10), θ_j shows the sum of the resultant value of the input and the context layers. θ_j and $o_j^{(3)}$ denote the input and output of jth nodes in HL, correspondingly. The weight connection except for the hidden neurons to the output neurons is fixed to 1 for the convenience of calculation.

Output layer: in this layer, the input and output of nodes are given as follows:

$$u_1^{(4)} = \sum_{j=1}^{9} o_j^{(3)} w_j^3,\ u_2^{(4)} = \sum_{j=1}^{9} o_j^{(3)} v_j^3,\ o_1^{(4)} = u_1^{(4)}, o_2^{(4)} = u_2^{(4)} \tag{11}$$

In Eq. (11), w^3 and v_j^3 indicate the weight connected between the hidden and the output layers, correspondingly. o_1^4 and $o_2^{(4)}$ shows the output layer used to evaluate the uncertainty term $f(x)$ and the system failure function δ, correspondingly.

3.4 Parameter Tuning using BO Algorithm

Lastly, the BO algorithm chooses the hyperparameter values of the ENN approach. It is commonly used in the automated optimization of hyperparameters [23]. In contrast to conventional optimization algorithms, including grid search and random optimization, BO approaches take full advantage of the searched data and upgrade the hyperparameter for this reason which avoids the exploration of a wide range of invalid hyperparameter spaces. Thus, the BO technique has strong optimization ability and high search efficacy.

The Gaussian surrogate model is established to define the relationships between the objective function (x) and hyperparameter χ. Set up the searched dataset $D = \{x_j, y_i, i = 1, 2, ..., m\}$, the Gaussian proxy method was formulated by Eq. (12):

$$y(x) \sim GP(0, k(x, x_i)) \tag{12}$$

Where $k(*)$ denotes the covariance function and $GP(*)$ refers to the Gaussian process:

$$\kappa(x, x_i) = \exp\left(-\frac{1}{2}(x - x_i)^2\right) \tag{13}$$

The Gaussian model is capable of predicting the objective function under any hyperparameters. The acquisition function was established on the predictive SD and mean of the Gaussian surrogate method, and it can be maximized to define the next group of hyper-parameters that were measured. The three common acquisition functions are (x), $EI(x)$, and $UCB(x) / LCB(x)$:

$$PI(x) = \phi(z) = \phi\left(\frac{\mu(x) - f(x^*)}{\sigma}\right) \tag{14}$$

$$EI(x)=\begin{cases}\left[\mu(x)-f(x^*)\right]\phi(z)+\sigma(x)\phi(z) & (\sigma(x)>0)\\ 0 & (\sigma(x)>0)\end{cases} \quad (15)$$

$$\begin{cases}\mathrm{UCB}(x)=\mu(x)+\beta\sigma(x)\\ \mathrm{LCB}(x)=\mu(x)-\beta\sigma(x)\end{cases} \quad (16)$$

where $\mu(x)$ and $\sigma(x)$ show the predictive mean and SD of the method, correspondingly, ϕ denotes the uniform distribution cumulative distribution function, β is a constant, and $\beta\geq 0, f(x^*)$, shows the optimum value of the existing main function and $\phi(*)$ represent the probability density function of uniform distribution.

$$Fitness=\max(P) \quad (17)$$

$$P=\frac{TP}{TP+FP} \quad (18)$$

where TP and FP represent the true and false positive values.

4 RESULTS AND DISCUSSION

The IDS outcomes of the BAFWO-MLIDS approach can be tested on the BoT-IoT Database [24-27], which comprises a mixture of benign and botnet traffic, and the original PCAP files comprise over 72 million records. In this work, a set of 3673 samples was taken with 10 classes as depicted in Tab. 1.

Table 1 Details of database

Class	Labels	No. of Samples
Service Scanning	C1	400
OS Fingerprinting	C2	400
DDoS TCP	C3	400
DDoS UDP	C4	400
DDoS HTTP	C5	400
DoS TCP	C6	400
DoS UDP	C7	400
DoS HTTP	C8	400
Normal	C9	400
Keylogging	C10	73
Total Number of Samples		3673

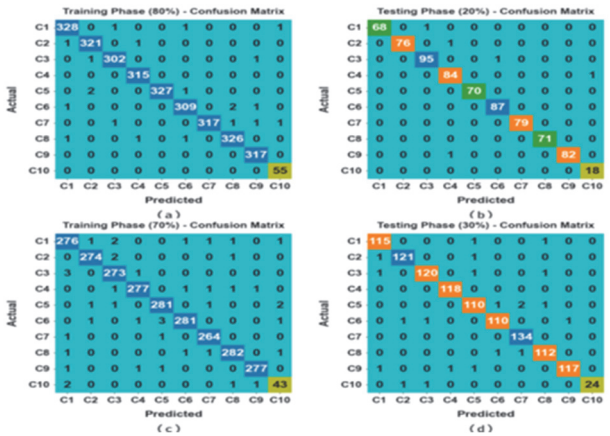

Figure 3 Confusion matrices of (a - b) 80:20 of TR set/TS set and (c - d) 70:30 of TR set/TS set

The intrusion detection results of the BAFWO-MLIDS technique are depicted in the procedure of confusion matrix in Fig. 3. The outcome indicates that the BAFWO-MLIDS technique reached effectual recognition of intrusions.

The intrusion detection findings of the BAFWO-MLIDS algorithm are depicted in Tab. 2 and Fig. 4, specifically for the 80:20 ratio of the training set to the test set. The BAFWO-MLIDS approach achieves an average accuracy, precision, recall, F-Score, and AUC-score of 99.86%, 99.06%, 99.35%, 99.20%, and 99.63% accordingly on 80% of the TR set. Furthermore, when applied to 20% of the TS set, the BAFWO-MLIDS technique yields an average accuracy, precision, recall, F-Score, and AUC-score of 99.86%, 99.02%, 99.38%, 99.19%, and 99.65% respectively.

Table 2 Intrusion detection outcome of BAFWO-MLID algorithm on 80:20 of TR set/TS set

Class Labels	Accuracy	Precision	Recall	*F* Score	AUC Score
Training Phase (80%)					
Service Scanning (C1)	99.80	99.09	99.09	99.09	99.49
OS Fingerprinting (C2)	99.83	99.07	99.38	99.23	99.63
DDoS TCP (C3)	99.86	99.34	99.34	99.34	99.63
DDoS UDP (C4)	99.93	99.37	100.00	99.68	99.96
DDoS HTTP (C5)	99.90	100.00	99.09	99.54	99.55
DoS TCP (C6)	99.76	99.04	98.72	98.88	99.30
DoS UDP (C7)	99.86	100.00	98.75	99.37	99.38
DoS HTTP (C8)	99.80	99.09	99.09	99.09	99.49
Normal (C9)	99.90	99.06	100.00	99.53	99.94
Keylogging (C10)	99.93	96.49	100.00	98.21	99.97
Average	99.86	99.06	99.35	99.20	99.63
Testing Phase (20%)					
Service Scanning (C1)	99.86	100.00	98.55	99.27	99.28
OS Fingerprinting (C2)	99.86	100.00	98.70	99.35	99.35
DDoS TCP (C3)	99.73	98.96	98.96	98.96	99.40
DDoS UDP (C4)	99.59	97.67	98.82	98.25	99.26
DDoS HTTP (C5)	100.00	100.00	100.00	100.00	100.00
DoS TCP (C6)	99.86	98.86	100.00	99.43	99.92
DoS UDP (C7)	100.00	100.00	100.00	100.00	100.00
DoS HTTP (C8)	100.00	100.00	100.00	100.00	100.00
Normal (C9)	99.86	100.00	98.80	99.39	99.40
Keylogging (C10)	99.86	94.74	100.00	97.30	99.93
Average	99.86	99.02	99.38	99.19	99.65

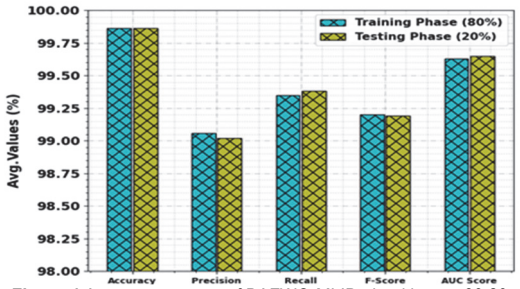

Figure 4 Average outcome of BAFWO-MLID algorithm on 80:20

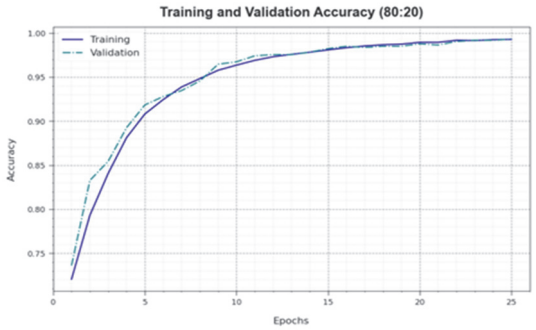

Figure 5 Accuarcy curve of BAFWO-MLID algorithm on 80:20

Fig. 6 and Fig. 7 depict the training accuracy (TR) and validation accuracy (VL) of the BAFWO-MLID system when the training set (TR set) and test set (TS set) are divided in a 70:30 ratio.

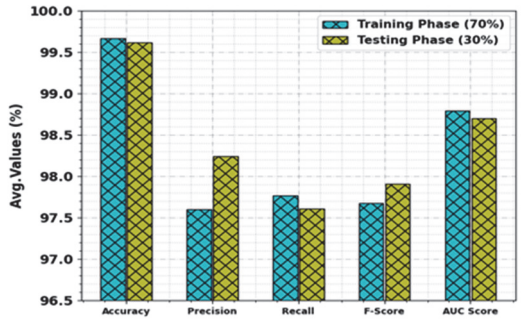

Figure 6 Average outcome of BAFWO-MLID algorithm on 70:30

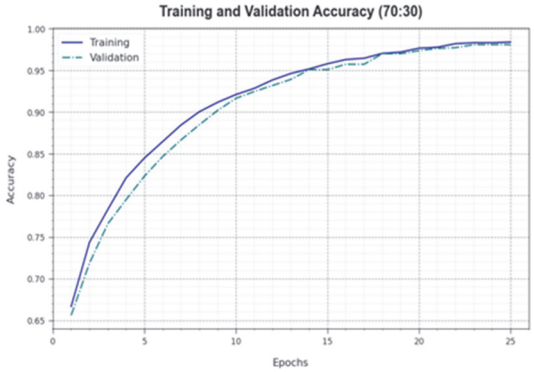

Figure 7 Accuracy curve of BAFWO-MLID algorithm on 70:30

In Tab. 2 and Fig. 4, the intrusion detection outcome of the BAFWO-MLIDS algorithm is portrayed at 80:20 of the TR set/TS set.

Table 3 Intrusion detection outcome of BAFWO-MLID algorithm on 70:30 of TR set/TS set

Class Labels	Accuracy	Precision	Recall	*F* Score	AUC Score
Training Phase (70%)					
Service Scanning (C1)	99.42	97.18	97.53	97.35	98.59
OS Fingerprinting (C2)	99.77	98.92	98.92	98.92	99.39
DDoS TCP (C3)	99.61	97.85	98.56	98.20	99.15
DDoS UDP (C4)	99.69	98.93	98.23	98.58	99.05
DDoS HTTP (C5)	99.61	98.25	98.25	98.25	99.02
DoS TCP (C6)	99.65	98.94	97.91	98.42	98.89
DoS UDP (C7)	99.77	98.51	99.25	98.88	99.54
DoS HTTP (C8)	99.69	98.60	98.60	98.60	99.21
Normal (C9)	99.81	99.28	98.93	99.11	99.42
Keylogging (C10)	99.65	89.58	91.49	90.53	95.65
Average	99.67	97.60	97.77	97.68	98.79
Testing Phase (30%)					
Service Scanning (C1)	99.55	97.46	98.29	97.87	98.99
OS Fingerprinting (C2)	99.64	98.37	98.37	98.37	99.08
DDoS TCP (C3)	99.55	98.36	97.56	97.96	98.68
DDoS UDP (C4)	99.91	99.16	100.00	99.58	99.95
DDoS HTTP (C5)	99.27	96.49	96.49	96.49	98.04
DoS TCP (C6)	99.55	98.21	97.35	97.78	98.57
DoS UDP (C7)	99.73	97.81	100.00	98.89	99.85
DoS HTTP (C8)	99.55	97.39	98.25	97.82	98.97
Normal (C9)	99.64	99.15	97.50	98.32	98.70
Key logging (C10)	99.82	100.00	92.31	96.00	96.15
Average	99.62	98.24	97.61	97.91	98.70

The Fig. 8 and Fig. 9 are the TR loss and VR loss outcomes of the BAFWO-MLID system on 70:30 of the TR set/TS set.

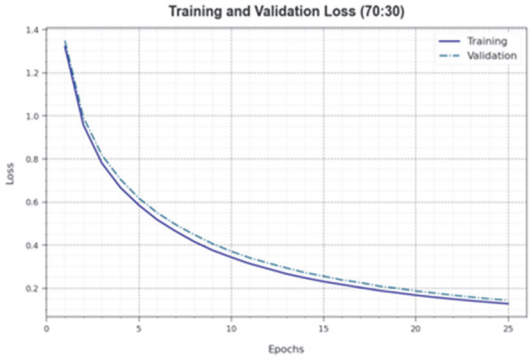

Figure 8 Loss curve of BAFWO-MLID algorithm on 70:30

The computation time (CT) results of the BAFWO-MLIDS technique are compared with other systems in Tab. 4 and Fig. 9 shows that the BAFWO-MLIDS technique requires less CT than existing ones. Based on training time (TRT), the BAFWO-MLIDS technique requires a lesser CT of 4.34 s while the DT, MLP ANN, CNN, GNB, SVM, and RNN models offer higher CT values. Finally, based on testing time (TST), the BAFWO-MLIDS algorithm needs a smaller CT of 0.81 s while the DT, MLP ANN, CNN, GNB, SVM, and RNN techniques give superior CT values.

Table 4 Comparative outcome of BAFWO-MLIDS algorithm with other methodologies

Classifiers	Accuracy	Precision	Recall	*F* Score
Decision Tree	99.19	97.06	98.98	98.03
MLP ANN	87.40	97.06	97.29	98.65
CNN Model	98.37	97.92	97.89	98.18
Gaussian Naive Bayes	99.14	98.61	97.70	97.23
SVM Model	99.15	97.55	98.50	98.76
RNN Model	98.31	98.37	98.93	98.02
BAFWO-MLIDS	99.86	99.06	99.35	99.20

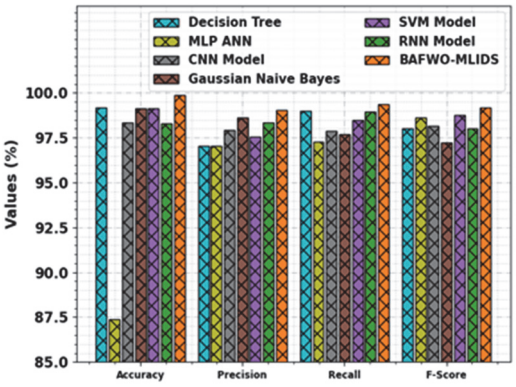

Figure 9 Comparative outcome of BAFWO-MLIDS algorithm with other methods

These results confirmed the supremacy of the BAFWO-MLIDS technique in the intrusion detection process.

5 CONCLUSION AND FUTURE WORK

In this study, we have focused on the design and development of automated intrusion detection using the BAFWO-MLIDS technique in the healthcare environment. The major purpose of the BAFWO-MLIDS algorithm is to apply BCT with IDS for enhanced security in the healthcare sector. The BCT enables to achieve secure data transmission in the healthcare platform. In the presented

BAFWO-MLIDS technique, a three-stage process is performed, namely FWO based FS process, ENN-based detection, and BO-based parameter optimization. In this work, the BO approach was executed to modify the parameters compared with the ENN model and thereby accomplish enhanced detection performance. The simulation results of the BAFWO-MLIDS approach can be inspected in a series of experiments and the obtained results ensured the greater efficiency of the BAFWO-MLIDS methodology compared with other approaches. In future, the efficiency of the BAFWO-MLIDS algorithm will be better by outlier detection systems. The performance of the BAFWO-MLIDS model can be enhanced in the future by the implementation of feature reduction techniques.

6 REFERENCES

[1] Kavitha, S., Uma Maheswari, N., & Venkatesh, R. (2023). Intelligent Intrusion Detection System using Enhanced Arithmetic Optimization Algorithm with Deep Learning Model. *Technical Gazette*, *30*(4), 1217-1224. https://doi.org/10.17559/TV-20221128071759

[2] Rathee, G., Kerrache, C. A., & Ferrag, M. A. (2022). A blockchain-based intrusion detection system using viterbi algorithm and indirect trust for iiot systems. *Journal of Sensor and Actuator Networks*, *11*(4), 71. https://doi.org/10.3390/jsan11040071

[3] Makhdoom, I., Zhou, I., Abolhasan, M., Lipman, J., & Ni, W. (2020). Privy Sharing: A blockchain - based framework for privacy-preserving and secure data sharing in smart cities. *Computers & Security*, *88*, 101653.

[4] Guha Roy, D. & Srirama, S. N. (2021). A blockchain-based cyber attack detection scheme for decentralized Internet of Things using software - defined network. *Software: practice and experience*, *51*(7), 1540-1556. https://doi.org/10.1002/spe.2972

[5] Javeed, D., Gao, T., Saeed, M. S., & Kumar, P. (2023). An Intrusion Detection System for Edge-Envisioned Smart Agriculture in Extreme Environment. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2023.3288544

[6] Meng, W., Li, W., Tug, S., & Tan, J. (2020). Towards blockchain-enabled single character frequency-based exclusive signature matching in IoT-assisted smart cities. *Journal of parallel and distributed computing*, *144*, 268-277. https://doi.org/10.1016/j.jpdc.2020.05.013

[7] Kumar, R., Kumar, P., Tripathi, R., Gupta, G. P., Garg, S., & Hassan, M. M. (2022). A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. *Journal of Parallel and Distributed Computing*, *164*, 55-68. https://doi.org/10.1016/j.jpdc.2022.01.030

[8] Abdulqadder, I. H., Zou, D., & Aziz, I. T. (2023). The DAG blockchain: A secure edge assisted honeypot for attack detection and multi-controller based load balancing in SDN 5G. *Future Generation Computer Systems*, *141*(7), 339-354. https://doi.org/10.1016/j.future.2022.11.008

[9] Liu, L. & Li, J. (2022). A Blockchain-assisted Collaborative Ensemble Learning for Network Intrusion Detection. In *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), IEEE,* 1042-1047. https://doi.org/10.1109/TrustCom56396.2022.00142

[10] Rahman, M. S., Khalil, I., Moustafa, N., Kalapaaking, A. P., & Bouras, A. (2021). A blockchain-enabled privacy-preserving verifiable query framework for securing cloud-assisted industrial internet of things systems. *IEEE Transactions on Industrial Informatics*, *18*(7), 5007-5017. https://doi.org/10.1109/TII.2021.3105527

[11] Kumaran, N. & Js, S. M. (2023). BRDO: Blockchain Assisted Intrusion Detection Using Optimized Deep Stacked Network. *Cybernetics and Systems*. https://doi.org/10.1080/01969722.2023.2175153

[12] Ahmed, M. A., Althubiti, S. A., Rao, D. N., Lydia, E. L., Cho, W., Joshi, G. P., & Kim, S. W. (2022). Blockchain Assisted Intrusion Detection System Using Differential Flower Pollination Model. *Computers*, *Materials & Continua*, *73*(3). 4695-4711. https://doi.org/10.32604/cmc.2022.032083

[13] Sharadqh, A. A., Hatamleh, H. A. M., Saloum, S. S., & Alawneh, T. A. (2023). Hybrid Chain: Blockchain Enabled Framework for Bi - Level Intrusion Detection and Graph - Based Mitigation for Security Provisioning in Edge Assisted IoT Environment. *IEEE Access*, *11*, 27433-27449. https://doi.org/10.1109/ACCESS.2023.3256277

[14] Alkadi, O., Moustafa, N., Turnbull, B., & Choo, K. K. R. (2020). A deep blockchain framework-enabled collaborative intrusion detection for protecting IoT and cloud networks. *IEEE Internet of Things Journal*, *8*(12), 9463-9472. https://doi.org/10.1109/JIOT.2020.2996590

[15] Ashraf, E., Areed, N. F., Salem, H., Abdelhay, E. H., & Farouk, A. (2022), June. Fidchain: Federated intrusion detection system for blockchain-enabled iot healthcare applications. *Healthcare MDPI*, *10*(6), 1110. https://doi.org/10.3390/healthcare10061110

[16] Cheema, M. A., Qureshi, H. K., Chrysostomou, C., & Lestas, M. (2020). Utilizing blockchain for distributed machine learning based intrusion detection in internet of things. *2020 16th international conference on distributed computing in sensor systems (DCOSS)*, *IEEE,* 429-435. https://doi.org/10.1109/DCOSS49796.2020.00074

[17] Alevizos, L., Eiza, M. H., Ta, V. T., Shi, Q., & Read, J. (2022). Blockchain - enabled intrusion detection and prevention system of APTs within zero trust architecture. *IEEE Access*, *10*, 89270-89288. https://doi.org/10.1109/ACCESS.2022.3200165

[18] Derhab, A., Guerroumi, M., Gumaei, A., Maglaras, L., Ferrag, M. A., Mukherjee, M., & Khan, F. A. (2019). Blockchain and random subspace learning - based IDS for SDN - enabled industrial IoT security. *Sensors*, *19*(14), 3119. https://doi.org/10.3390/s19143119

[19] Abunadi, I., Althobaiti, M. M., Al - Wesabi, F. N., Hilal, A. M., Medani, M., Hamza, M. A., Rizwanullah, M., & Zamani, A. S. (2022). Federated learning with blockchain assisted image classification for clustered UAV networks. *Comput. mater. contin*, *72*(1), 1195-1212. https://doi.org/10.32604/cmc.2022.025473

[20] Vinoth Kumar, K. & Balaganesh, D. (2022). Efficient Privacy - Preserving Red Deer Optimization Algorithm with Blockchain Technology for Clustered VANET. *Technical Gazette*, *29*(3), 813-817.

[21] Liang, C., Shanmugam, B., Azam, S., Karim, A., Islam, A., Zamani, M., Kavianpour, S., & Idris, N. B. (2020). Intrusion detection system for the internet of things based on blockchain and multi - agent systems. *Electronics*, *9*(7), 1120. https://doi.org/10.3390/electronics9071120

[22] Zhang, X., Mo, T., & Zhang, Y. (2023). Optimization of Storage Location Assignment for Non - Traditional Layout Warehouses Based on the Firework Algorithm. *Sustainability*, *15*(13), 10242. https://doi.org/10.3390/su151310242

[23] Mafarja, M., Thaher, T., Al-Betar, M. A., Too, J., Awadallah, M. A., Abu Doush, I., & Turabieh, H. (2023). Classification framework for faulty-software using enhanced exploratory whale optimizer-based feature selection scheme and random forest ensemble learning. *Applied Intelligence*, *53*, 18715-18757.

[24] Balakrishnan, S. & Vinoth Kumar, K. (2023). Hybrid Sine-Cosine Black Widow Spider Optimization based Route

Selection Protocol for Multihop Communication in IoT Assisted WSN. *Technical Gazette*, *30*(4), 1159-1165. https://doi.org/10.17559/TV-20230201000306

[25] Wu, Q., Zhu, Q., & Han, S. (2023). Elman Neural Network-Based Direct Lift Automatic Carrier Landing Nonsingular Terminal Sliding Mode Fault - Tolerant Control System Design. *Computational Intelligence and Neuroscience*. https://doi.org/10.1155/2023/3560441

[26] Wang, H., Wen, W., Zhang, Z., & Gao, N. (2023). Construction of Building Energy Consumption Prediction Model Based on Multi - Optimization Model. *Buildings*, *13*(7), 1677. https://doi.org/10.3390/buildings13071677

[27] Li, Y. & Zhanyong, W. (2023). A Cloud Based Network Intrusion Detection System. *Technical Gazette*, *29*(3), 987-992. https://doi.org/10.17559/TV-20211130024245

Contact information:

Sudhakar THIRUVENKATASAMY, Assistant Professor
(Corresponding author)
Department of Computer Science and Engineering,
Nandha College of Technology, Erode, Pincode 638052
E-mail: thiruvenkatasamys86@gmail.com

Rajappan SIVARAJ, Professor
Department of Computer Science and Engineering,
Nandha Engineering College, Erode, Pincode 638052
E-mail: rsivarajcse@gmail.com

Murugasamy VIJAYAKUMAR, Professor
Department of Computer Science and Engineering,
Sasurire College of Engineering, Vijayamangalam 638056
E-mail: tovijayakumar@gmail.com