IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 August 2023, accepted 3 October 2023, date of publication 5 October 2023, date of current version 24 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3322369

RESEARCH ARTICLE

Hybrid Metaheuristics With Machine Learning Based Botnet Detection in Cloud Assisted Internet of Things Environment

LATIFAH ALMUQREN[1], HAMED ALQAHTANI[2], SUMAYH S. ALJAMEEL[3], AHMED S. SALAMA[4], ISHFAQ YASEEN[5], AND AMANI A. ALNEIL[5]

[1]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[2]Department of Information Systems, College of Computer Science, Center of Artificial Intelligence, Unit of Cybersecurity, King Khalid University, Abha 62529, Saudi Arabia
[3]Saudi Aramco Cybersecurity Chair, Computer Science Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam 31441, Saudi Arabia
[4]Department of Electrical Engineering, Faculty of Engineering and Technology, Future University in Egypt, New Cairo 11845, Egypt
[5]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia

Corresponding author: Ishfaq Yaseen (i.yaseen@psau.edu.sa)

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through large group Research Project under grant number (RGP2/159 /44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R349), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. We Would like to thank SAUDI ARAMCO Cybersecurity Chair for funding this project. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444). This study is partially funded by the Future University in Egypt (FUE).

ABSTRACT Botnet detection in a cloud-aided Internet of Things (IoT) environment is a tedious process, meanwhile, IoT gadgets are extremely vulnerable to attacks due to poor security practices and limited computing resources. In the cloud-aided IoT environment, Botnet can be identified by monitoring network traffic and analyzing it for signs of malicious activity. It can be performed by using intrusion detection systems, machine learning (ML) algorithms, and other security tools that are devised for identifying known botnet behaviors and signatures. Therefore, this study presents a Hybrid Metaheuristics with Machine Learning based Botnet Detection (HMMLB-BND) method in the Cloud Aided IoT environment. The projected HMMLB-BND technique focuses on the detection and classification of Botnet attacks in the cloud-based IoT environment. In the presented HMMLB-BND technique, modified firefly optimization (MFFO) algorithm for feature selection purposes. The HMMLB-BND algorithm uses a hybrid convolutional neural network (CNN)-quasi-recurrent neural network (QRNN) module for botnet detection. For the optimal hyperparameter tuning process, the chaotic butterfly optimization algorithm (CBOA) is employed. A series of simulations were made on the N-BaIoT dataset and the experimental outcomes stated the significance of the HMMLB-BND technique over other existing approaches.

INDEX TERMS Deep learning, cloud computing, Internet of Things, cybersecurity, botnet detection.

I. INTRODUCTION

Nowadays, cloud computing (CC) has grabbed the attention of the research community [1]. Its role in offering on-demand resources and services has unlocked its way into various technological atmospheres such as data centers, power systems and intelligent transportation, video delivery system, and earthquake command system [2]. Various design objectives which include energy consumption, fairness, reliability and fault tolerance are regarded in the model of the CC system. But security is considered to be the critical design objective in this domain. IoT often appears in the CC ecosystem [3]. This technology compiles geographically dispersed cyber-enabled systems or cyber-physical devices to offer strategic services. Similar to the case of CC [4], security becomes the significant goal in the model of the IoT platform. Cyber threats targeting

The associate editor coordinating the review of this manuscript and approving it for publication was Sangsoon Lim.

IoT gadgets found to be rising with the development of IoT [5]. Many IoT gadgets are linked to the internet, allowing a lack of security control abuse [6]. Several security threats are aimed at the IoT, which includes several susceptibilities. As the IoT is prone to various attacks, it is significant to categorize the attacks and appropriate vulnerabilities to study the IoT. Through certain research, it is established that routing, jamming, sinkhole, DoS, wormhole, a man in the middle, worm attacks [7], flooding, and virus probably occur in an IoT system. To be specific, DoS attacks and flooding take place in production IoT platforms [8].

Botnet attack is now increasingly gaining popularity [9]. Service disruption and resource depletion are some of the damages caused by the botnet. AI is commonly utilized to find such IoT attacks [10]. The intrusion detection system (IDS) is an application that can be used to oversee network traffic actions to detect malicious actions. The IDS are classified into two categories per the detection system named anomaly-based and signature-based detection systems [11]. The first one is to use the network behaviours from established baselines. This method will be suited to detect unknown and known malicious events. The next one applies particular patterns from the network (e.g., a sequence of bytes). And later it compares these sequences with current signature databases [12]. Compared to traditional ML methods, the current study noted that the deep learning (DL) method finds IoT assaults fruitfully. But the cloud layer only has the resource to run these algorithms [13]. On top of that, these methods are not active in some cases, like remote live functioning, since the system has been assumed to form realistic decisions faster.

The convergence of IoT devices and cloud services poses several complexities because of huge scale, heterogeneity, and dynamic nature of the ecosystem. The sheer volume of different IoT devices, each with distinct communication protocols and abilities, makes it difficult to devise universal detection methods. Encrypted communication among the devices and cloud services further complicate the examination of network traffic for signs of botnet activity. The limited resources of IoT devices obstruct the design of resource-intensive detection approaches, requiring the design of lightweight yet effective approaches. The distributed nature of botnets and its capability to mimic legitimate device behavior make pinpointing malicious activities and command-and-control nodes challenging. Rapid development of the attack approaches, integrated with the absence of uniform security standards across IoT devices, exacerbate the difficulty of botnet detection.

To resolve these issues, this study designs a Hybrid Metaheuristics with Machine Learning based Botnet Detection (HMMLB-BND) method in the Cloud Assisted IoT environment. In the presented HMMLB-BND approach, the modified firefly optimization (MFFO) technique is used for feature selection (FS) purposes. For botnet detection, the HMMLB-BND technique uses a hybrid convolutional neural network (CNN)-quasi-recurrent neural network (QRNN) model. For the optimal hyperparameter tuning process, the chaotic butterfly optimization algorithm (CBOA) is applied. To demonstrate the enhanced performance of the HMMLB-BND technique, a series of simulations were made on the N-BaIoT dataset. In short, the key contributions are listed as follows.

- Develop a new HMMLB-BND technique comprising MFFO based feature subset selection, CNN-QRNN classification, and CBOA based hyperparameter tuning for botnet detection has been developed. To the best of our knowledge, the HMMLB-BND technique never existed in the literature.
- Present MFFO algorithm for the feature selection process, which resolves the ineffective exploration ability and local optima problem.
- Hyperparameter tuning of the CNN-QRNN model using CBOA helps to improve the overall predictive performance on unseen training data.

II. RELATED WORKS

Vinayakumar et al. [14] based on a two-level DL structure, a botnet detection system is presented for semantically determining Botnet and legal activities at the application layer of the domain name system (DNS). A primary first level of structure, with the use of a Siamese network, depends on a pre-defined threshold, the measure of similarity of DNS queries will be predicted to opt the frequent DNS data across an Ethernet connection. In Shorman et al. [15], an innovative unsupervised evolutionary IoT botnet recognition algorithm was devised. The algorithm mainly detects IoT botnet attacks in IoT devices using the effectiveness of a recent SI method named GWO for optimizing the hyperparameter of the OCSVM and concurrently recognising the attributes that define the IoT botnet issue optimally.

The authors [16] introduced a potential DL-based Botnet attack-detecting approach that can deal with vastly imbalanced network traffic datasets. To be Specific, to achieve class balance, SMOTE makes more minority samples, while DRNN learned hierarchical feature representation in balanced network traffic datasets for effectuating discriminative classifying procedure. The authors in [17] devise a botnet detection method utilizing the barnacle's mating optimizer including ML (BND-BMOML) for the IoT platform. This proposed method has focused on the recognition and identification of botnets in IoT platforms. Initially, a data standardization method is followed by the BND-BMOML algorithm follows for effectuating this. In the above-mentioned method, for opting for a valuable feature set, the BMO approach was used. To detect botnets, this study uses the BND-BMOML method in an Elman NN (ENN) method.

In [18], ML approaches were utilized to support the prevention and detection of bot attacks. In this study for the selection of the best features, An Ensemble Classifier Algorithm includes Stacking Process (ECASP) was devised that is given as input to the ML classifiers for forecasting the performance of botnet identification. Catillo et al. [19] modelled a new

IoT-driven cross-device approach, which permits learning single IDSs rather than several separate methods atop the traffic of various IoT gadgets. Because of its extensive applicability for unanticipated attacks, a semi-supervised method was implemented. The solution relies upon deep AE, which has trained a single DNN with the normal traffic from many IoT gadgets. Habibi et al. [20] apply the CTGAN method, and the existing GAN approaches in tabular data modelling and generation to solve the limitations. Khan et al. [21] offered a robust and lightweight DL structure for detecting intrusions that has the computation ability to be gradually reduced and installed as a localized threat identification within IoT gadgets. Also, the presented Hybrid method was compared against a benchmark ANN method.

Banati and Bajaj [22] examine a novel FS method, which integrates the RST with a nature-simulated 'firefly' technique. This technique inspires the attraction method of real fireflies that guides the FS method. Coelho et al. [23] establish an enhanced FA system integrated with chaotic sequences (FAC) executed to reliability-redundancy optimizer. Alzahrani and Bamhdi [24] introduce a robust model specifically to support identifying botnet attacks on IoT devices. It is done by newly integrating the model of CNN with LSMT (CNN-LSTM) methodology for detecting 2 general and serious IoT attacks (BASHLITE and Mirai) on 4 kinds of security cameras.

Bhayo et al. [25] examine a ML-based system for detecting DDoS attacks from SDN-WISE IoT controller. The authors are combined a ML-based detection element as to controller and set up a testbed platform for simulating DDoS attack traffic generation. The traffic can captured by a logging system along with the SDN-WISE controller that writes network logs into log file that is pre-processing and converting into database. Siddiqui et al. [26] offers a widespread survey analysis the published studies on SDN-based structures for addressing IoT management problems in the sizes of fault tolerance, scalability, load balancing, energy management, and security service provisioning in the IoT networks. Khalid et al. [27] purposes to solve the intricacy of policies management, forged policy, dissemination, central management, automation, and tracking of access control policies of IoT nodes and offers a trackable and auditable access control policy management system which prevent forged policy dissemination by executing SDN and BC technology in IoT platform.

III. THE PROPOSED MODEL

In this study, we have established a novel HMMLB-BND approach in the Cloud Aided IoT environment. The projected HMMLB-BND technique focuses on the detection and classification of Botnet attacks from the cloud-based IoT environment. In the presented HMMLB-BND approach, the MFFO algorithm for FS purposes is applied. To detect and classify botnets properly the CBOA with CNN-QRNN model is used. Fig. 1 defines the overall process of the HMMLB-BND method.

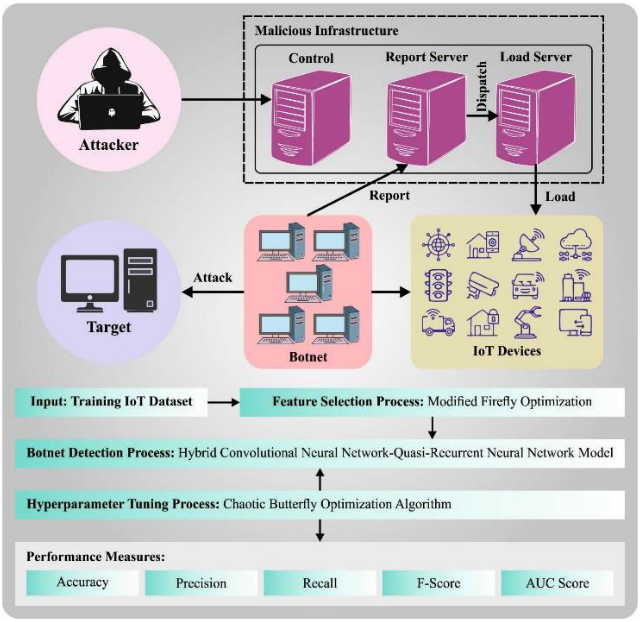
Malicious Infrastructure
Control
Report Server
Dispatch
Load Server
Attacker
Report
Load
Attack
Target
Botnet
IoT Devices
Input: Training IoT Dataset
Feature Selection Process: Modified Firefly Optimization
Botnet Detection Process: Hybrid Convolutional Neural Network-Quasi-Recurrent Neural Network Model
Hyperparameter Tuning Process: Chaotic Butterfly Optimization Algorithm
Performance Measures:
Accuracy
Precision
Recall
F-Score
AUC Score

FIGURE 1. Overall process of the HMMLB-BND system.

A. FEATURE SELECTION USING MFFO ALGORITHM

Firstly, the MFFO algorithm is used for the optimal selection of feature subsets. The FFO is a nature-inspired optimizer technique motivated by the flashing behaviors of fireflies [28]. It claims that the flash pattern produced by every firefly exists naturally via a unique bioluminescence method. The significant function of flashing generated by the bioluminescence process is the mating partner and attracting the prey. The succeeding four assumptions have been made to simplify the FFO: (i) each firefly is considered to be unisex, (ii) bright firefly attracts those with lesser brightness, (iii) the prey movement was assumed to be random if there is relatively no brighter firefly, and (iv) luminousness value is considered that main function that assists in simplifying the maximization problem. The FFO algorithm has two main challenges: attractiveness formulation and light intensity variation. Therefore, it is presumed that firefly brightness analyses its attractiveness.

$$I\left(r\right)=\frac{Is}{r^{2}} \tag{1}$$

In Eq. (1) Is signifies fixed intensity. Consider a medium that has a fixed light absorption coefficient and I_o original luminous intensity, then the light intensity is modelled by:

$$I=I_{o}e^{-\gamma r^{2}} \tag{2}$$

Eqs. (1) and (2) are used to present a Gaussian form of luminous intensity that is given below:

$$I\left(r\right)=\frac{I_{o}}{1}+\gamma r^{2} \tag{3}$$

The firefly attractiveness as a function of luminous intensity can be defined by the modification of fireflies that can be formulated by:

$$\beta(r) = \beta_0 e^{-\gamma r^2} \tag{4}$$

In 2D space, the distance between two fireflies is determined by the Cartesian distance:

$$r_{i_j} = \left|x_i - x_j\right| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{5}$$

As we discussed, the brighter firefly attracts the specific, less bright one. Therefore, i_{th} firefly movement can be mathematically formulated by:

$$x_j = x_j + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_j\right) + \alpha\,(rand - 0.50) \tag{6}$$

where α signifies the randomization parameter, and *rand* signifies a uniformly distributed random number within [0, 1]. The FFO technique is a powerful optimization technique utilized in an optimization problem. But, it has poor search capability and suffers from the local optima problem. An adapted version of the FFO is established, namely the MFFO algorithm by presenting the subsequent modification in the FFO algorithm: search capability is enhanced, and the local optimal problems can be solved. A comprehensive explanation is shown in the following.

The poor search capability and trapped in local optima are solved by presenting two modifications in the FFO, named MFFO technique: (1) the overall population of fireflies is stimulated towards the direction of global optimal or best solution; (2) population diversity can be enhanced by presenting two mutation operations and three crossover operations. Correspondingly, the entire firefly population can be enhanced in every iteration by presenting certain assumptions. The comprehensive overview is given below:

Consider that X_{best}^{itr} denotes the best individual, and X_{worst}^{itr} denotes the worse individual in the firefly population at every iteration. During the firefly population for i_{th} firefly, three more fireflies are selected randomly as X_{q1}, X_{q2}, and X_{q3} so that $q_1 \neq q_2 \neq q_3 \neq i$. The two newly generated individuals are given as follows:

$$\begin{aligned} X_{mutate1} &= X_{q_1} + \Delta \times (q_2 - q_3) \\ X_{mutate2} &= X_{mutate1} + \Delta(X_{best}^{itr} - X_{worst}^{itr}) \end{aligned} \tag{7}$$

Now Δ denotes a randomly generated value within [0, 1]. By using $X_{mutate1}$ and $X_{mutate2}$, the five fireflies are created by the following:

$$X_{best_1} = \left[x_{best_1}, x_{best_2}, \ldots, x_{best_d}\right] \tag{8}$$

$$x_{improve_{1,j}} = \begin{cases} x_{mute_{1,j}} & if\ k_1 \leq k_2 \\ x_{best_{1,j}} & if\ k_1 > k_2 \end{cases} \tag{9}$$

$$x_{improve_{2,j}} = \begin{cases} x_{mute_{1,j}} & if\ k_3 \leq k_2 \\ x_j & if\ k_3 > k_2 \end{cases} \tag{10}$$

$$x_{improve_{3,j}} = \begin{cases} x_{best_{1,j}} & if\ k_4 \leq k_3 \\ x_j & if\ k_4 > k_3 \end{cases} \tag{11}$$

$$x_{improve_{4,j}} = \begin{cases} x_{mute_{1,j}} & if\ k_5 \leq k_4 \\ x_{mute_{2,j}} & if\ k_5 > k_4 \end{cases} \tag{12}$$

$$x_{improve_{5,j}} = \psi \times X_{worst} + \zeta\,(X_{best} - X_{worst}) \tag{13}$$

F k_1: k_5, ψ, and ζ characterize random variable ranges from zero to one.

For each firefly, the objective function can be defined, and the i_{th} firefly will be replaced by the firefly having the smaller objective function. When the i_{th} firefly has a main function small than the optimally attained firefly, then the replacement cannot be done. The α random parameter controls the random search ability whereas the neighboring fireflies are not noticeable to the selected firefly. The α monitor and control the movements of every firefly selected randomly amongst [0, 1]. The value of α through the global search space leads to an optimum solution, whereas the smaller value of α promotes local search. Thereby, an optimum value of α fulfils the balance of local and global searching. A novel adaptive control mechanism can be devised for improving the search ability (global and local) to accomplish this balance. Moreover, the process runs for multiple epochs, and the heuristic function for every epoch is attained by the following:

$$\alpha_{itr+1} = (1/2k_{\max})^{1/k_{\max}} \alpha_{itr} \tag{14}$$

where *itr* signifies iteration value ranges from 1 to $k_{\max}$.

The fitness function of the MFFO algorithm is intended to have a balance between the classification performance (highest) and the count of features selected in every solution (lowest) attained by the features selected, Eq. (15) signifies the fitness function to estimate the solution.

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|C|} \tag{15}$$

where the two parameters respective to the importance of classification quality and subset length are α and β, $\gamma_R(D)$ characterizes the classification error rate. $|R|$ denotes the cardinality of the selected subset and $|C|$ shows the overall amount of features in the dataset. $\in$ [1,0] and $\beta = 1 - \alpha$.

B. BOTNET DETECTION USING CNN-QRNN MODEL

For botnet detection and classification, the CNN-QRNN model is used. The CNN-QRNN architecture comprises FC layers, a1D convolution layer, and a QRNN [29]. Initially, the 1D convolution layer chooses the spatial feature and generates feature maps that can be treated by using the activation function. Due to its fast convergence of the GD model, the ReLu function is applied in the convolutional layer, making it a better option for the CNN-QRNN method. Next, the feature maps can be processed using the second layer that exploits the Maxpooling function. The pooling layer removes irrelevant features and decreases dimensionality. In both layers of QRNN, the hidden size characterizes the output dimension and count of hidden modules. The major problem of NN is over-fitting which implies a model learns the data effectively. As a result, the model cannot find variants in novel datasets.

Thus, to prevent overfitting, we added a dropout layer. Fig. 2 illustrates the architecture of the CNN-QRNN technique.

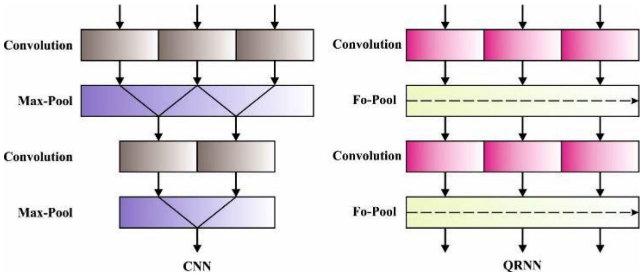

FIGURE 2. **Structure of CNN-QRNN.**

Next, max-pooling and 1D convolutional layers are used for extracting spatiotemporal features. The outcome of CNN is fed into Flatten layer the FC input layer converts the outcome of pooling layers into a single vector that is input for the following layers. Lastly, the dense layer, also known as the FC layer, with SoftMax function was utilized to categorize the threat by computing the probability for all the classes.

C. PARAMETER TUNING USING CBOA

The utilization of the CBOA model for hyperparameter tuning helps in attaining improved performance on botnet attack detection. The learning rate is the hyperparameter tuned by the CBOA. The BOA is a swarm-based metaheuristic algorithm based on the information-sharing and foraging behavior of butterflies (BFs) [30]. Due to its performance, BOA was used in different fields of optimization problems. The magnitude of BF produces an odor smell with intensity once it moves. The other BFs attracted towards BF based on the magnitude of fragrances. The fragrance of all the BFs is illustrated in Eq. (16).

$$pf_i = cI^a \tag{16}$$

where c and a parameters are a power exponent that signifies the degree of fragrance absorption and the sensor modality correspondingly. pf_i signifies the perceived magnitude of fragrances, and I represent the fragrance intensity.

Butterflies movement: The movement of BFs can be dependent upon 3 stages as follow.

-Global searching stage. All the BFs emit fragrance once it moves and other BFs take with it based on their magnitude of fragrances. This procedure is termed global searching and is determined as:

$$x_i^{t+1} = x_i^t + \left(r^2 \times g^* - x_i^t\right) \times f_i \tag{17}$$

In which x_i^t defines the vector that signifies the BF (solution) at iteration t, g^* signifies the entire better results, r stands for the random number in 0 and 1, and f_i denotes the fragrance of $i_t h$ BF. During this stage, the g primary value is the location of the minimal fitness of every solution and it is computed by allocating a fitness value to all the solutions and determining the minimal fitness afterwards upgrading the g (position) based on the minimal fitness. Besides, *ther* value could not be utilized for calculating fitness, it can be managed by the p (switching probability) value and primary value of $p = 0.8$. The r value has been related to the p-value for controlling the BF but moving to a better solution with minimal fitness from local/global searching.

-Local searching stage. If the BFs lose the sense of the fragrance of other BFs, they can be moved arbitrarily from the searching space. The procedure is termed local searching and it could be determined as:

$$x_i^{t+1} = x_i^t + \left(r^2 \times x_{j^t} - x_k^t\right) \times f_i \tag{18}$$

whereas x_j^t, x_{k^t} implies the 2 vectors that signify 2 various BFs from a similar population. -Solution estimation. The fragrance intensity of BFs defines their main function. The BF attract the other BFs based on their magnitude of fragrances.

The projected CBOA technique depends upon the combination of chaotic maps from the typical BOA. Essential stages of the presented CBOA are given in the following. Appeal the chaotic maps to upgrade BF places rather than utilizing random variables so as far as will enhance the performance of CBOA. Eqs. (2) and (3) are altered by exchanging r^2 by C_j as follows:

$$x_i^{t+1} = x_i^t + \left(C_j \times g^* - x_i^t\right) \times f_i \tag{19}$$

$$x_i^{t+1} = x_i^t + \left(C_j \times x_{j^t} - x_k^t\right) \times f_i \tag{20}$$

whereas C_j denotes the chaotic map and $j = 1, 2, \ldots, 10$. Noticeably the C_j values can be chaotic, created utilizing 10 chaotic maps that can be exchanged with r value for obtaining best outcomes and minimal fitness than novel technique utilize random value.

Fitness selection is a key factor in the CBOA system. Solution encoding is used to evaluate the goodness of the solution candidate. Then, the accuracy value is the primary condition exploited for devising a fitness function.

$$Fitness = \max(P) \tag{21}$$

$$P = \frac{TP}{TP + FP} \tag{22}$$

where TP represent the true positive and FP symbolizes the false positive value.

IV. RESULTS AND DISCUSSION

In this work, the botnet detection results of the HMMLB-BND method are studied on the N-BaIoT [31] Dataset. It includes 17001 instances with three class labels as given in Table 1. The proposed model is simulated using Python 3.6.5 tool on PC i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD. The parameter settings are given as follows: learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU.

Fig. 3 represents the confusion matrices of the HMMLB-BND method tested under distinct sizes of the TRP and TSP. The results denote that the HMMLB-BND system has identified the botnets proficiently under all TRP and TSP.

Algorithm 1 Pseudocode of BOA

```
Fixed the primary values of population size $n$(BFs), switch probability $\rho$z, $c$ sensory modality, parameters $a$z(power exponent), and maximal count of iterations Max$_{itr}$.
Set $t$:= 0.
for ($i = 1 : i \leq n$) d
    Make a primary population (BFs) $x_i^t$ arbitrarily.
    Measured the fitness function of all the BFs (solutions)($x_i^t$)z.
    Compute the fragrance for $x_i^t$ as in Eq. (16).
    Allocate the entire optimum BF (solution) $g^*$.
end for
repeat
Set $t = t$+1.
for ($i$= 1 :$i \leq n$)$do$
    Make random numbers $r$, $r \in [0, 1]$.
    if ($r < \rho$)z then
        Move BFs nearby the better BF $g^*$ as in Eq. (17).
    else
        Move BFs arbitrarily.
    endif
    Estimate the fitness function of all the BFs (solutions) ($x_i^t$)z.
    Allocate the entire optimum solution
    $g^*$.
end for
Upgrade the value of parameters $c$= [0.01, 0.25].
until ($t >$ Max$_{itr}$).
Display the optimum solution
$g^*$.
```

TABLE 1. Details of database.

Class	No. of Instances
Benign	5000
Mirai	7001
Gafgyt	5000
Total Number of Instances	17001

Table 2 reports the overall outcomes of the HMMLB-BND method on 80:20 and 70:20 of TRS/TSS. In Fig. 4, the botnet classification outcome of the HMMLB-BND technique can be examined on 80% of TRP. The outcomes represented that the HMMLB-BND system recognizes the botnets effectually under all classes. In addition, it is noticed that the HMMLB-BND method obtains an average $accu_y$ of 98.20%, $prec_n$ of 97.14%, $reca_l$ of 97.08%, F_{score} of 97.10%, and AUC_{score} of 97.87%.

In Fig. 5, the botnet classification performance of the HMMLB-BND system can be examined on 20% of TSP. The outcomes signified that the HMMLB-BND technique recognizes the botnets effectually under all classes. Besides, it can be noticed that the HMMLB-BND system attains an average $accu_y$ of 97.96%, $prec_n$ of 96.79%, $reca_l$ of 96.75%, F_{score} of 96.77%, and AUC_{score} of 97.62%.

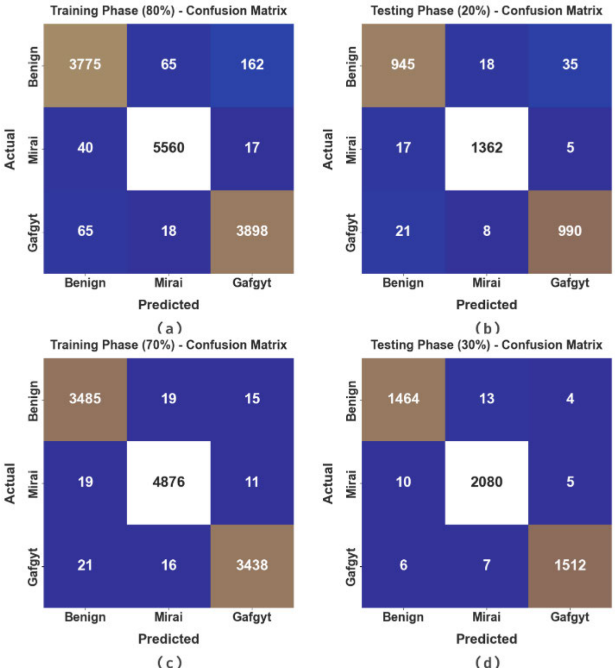

FIGURE 3. Confusion matrices of HMMLB-BND method (a-b) TRP/TSP of 80:20 and (c-d) TRP/TSP of 70:30.

TABLE 2. Botnet classifier outcome of HMMLB-BND method on 80:20 of TRP/TSP.

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	AUC_{score}
Training Phase (80%)					
Benign	97.56	97.29	94.33	95.79	96.62
Mirai	98.97	98.53	98.99	98.76	98.97
Gafgyt	98.07	95.61	97.92	96.75	98.03
Average	98.20	97.14	97.08	97.10	97.87
Testing Phase (20%)					
Benign	97.32	96.13	94.69	95.41	96.55
Mirai	98.59	98.13	98.41	98.27	98.56
Gafgyt	97.97	96.12	97.15	96.63	97.74
Average	97.96	96.79	96.75	96.77	97.62
Training Phase (70%)					
Benign	99.38	98.87	99.03	98.95	99.28
Mirai	99.45	99.29	99.39	99.34	99.44
Gafgyt	99.47	99.25	98.94	99.09	99.31
Average	99.43	99.13	99.12	99.13	99.35
Testing Phase (30%)					
Benign	99.35	98.92	98.85	98.89	99.21
Mirai	99.31	99.05	99.28	99.17	99.31
Gafgyt	99.57	99.41	99.15	99.28	99.45
Average	99.41	99.12	99.09	99.11	99.32

In Fig. 6, the botnet classification result of the HMMLB-BND technique can be examined on 70% of TRP. The results inferred that the HMMLB-BND system recognizes the botnets effectively under all classes. Moreover, it is clear that the HMMLB-BND method gains an average $accu_y$ of 99.43%, $prec_n$ of 99.13%, $reca_l$ of 99.12%, F_{score} of 99.13%, and AUC_{score} of 99.35%.

In Fig. 7, the botnet classification outcome of the HMMLB-BND method can be examined on 30% of TSP. The result stated that the HMMLB-BND technique

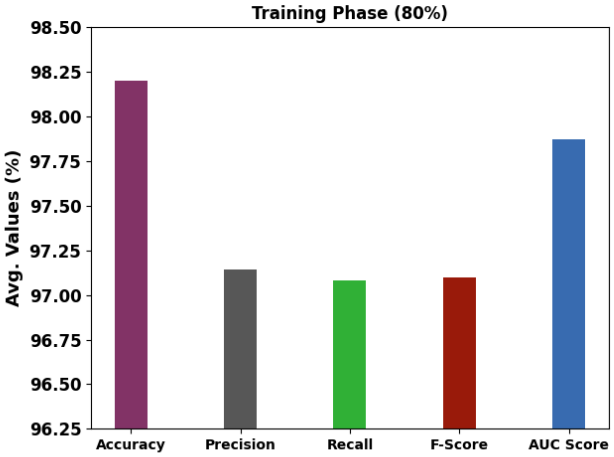

FIGURE 4. Average outcome of HMMLB-BND approach on 80% of TRP.

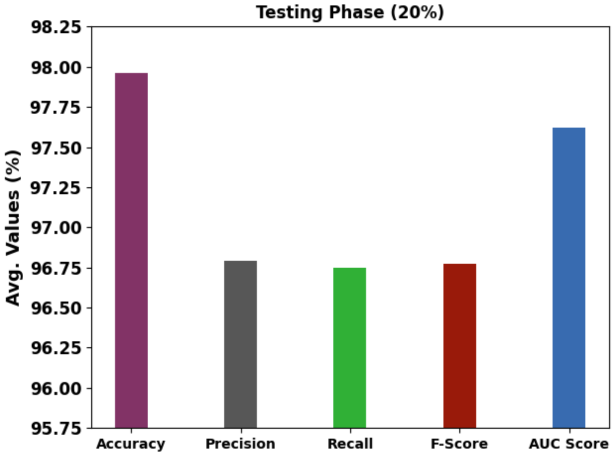

FIGURE 5. Average outcome of HMMLB-BND approach on 20% of TSP.

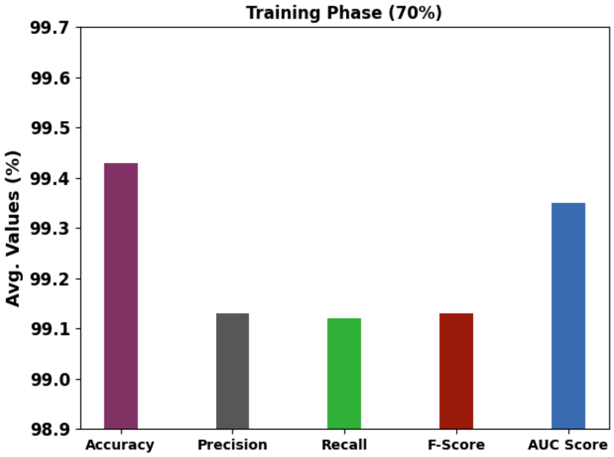

FIGURE 6. Average outcome of HMMLB-BND approach on 70% of TRP.

recognizes the botnets effectually under all classes. Furthermore, it can be evident that the HMMLB-BND technique obtains an average $accu_y$ of 99.41%, $prec_n$ of 99.12%, $recal_l$ of 99.09%, F_{score} of 99.11%, and AUC_{score} of 99.32%.

The TACY and VACY of the HMMLB-BND system are investigated on Botnet recognition performance in Fig. 8. The figure referred that the HMMLB-BND method has displayed better results with enhanced values of TACY and VACY. It is clear that the HMMLB-BND algorithm has accomplished the highest TACY outcomes.

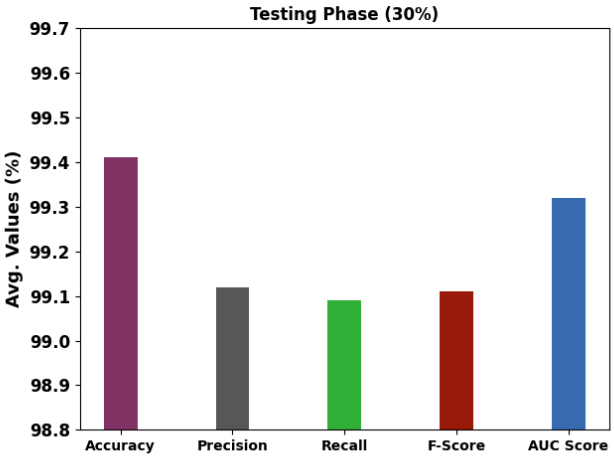

FIGURE 7. Average outcome of HMMLB-BND approach on 30% of TSP.

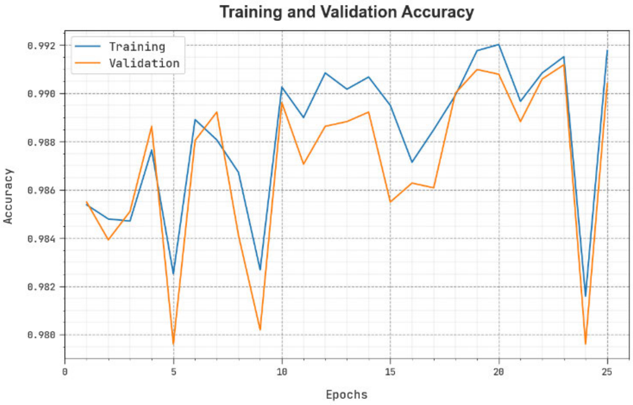

FIGURE 8. TACY and VACY outcomes of the HMMLB-BND method.

The TLOS and VLOS of the HMMLB-BND system are tested on Botnet recognition performance in Fig. 9. The figure stated that the HMMLB-BND methodology has exposed superior performance with lesser values of TLOS and VLOS. The HMMLB-BND technique has resulted in minimal VLOS outcomes.

An evident precision-recall study of the HMMLB-BND technique in the test database is exposed in Fig. 10. The figure referred that the HMMLB-BND method has led to better values of precision-recall values in three classes.

To assure the improvised performance of the HMMLB-BND technique, a brief comparison study with recent approaches was made in Table 3 and Fig. 11 [17]. The results

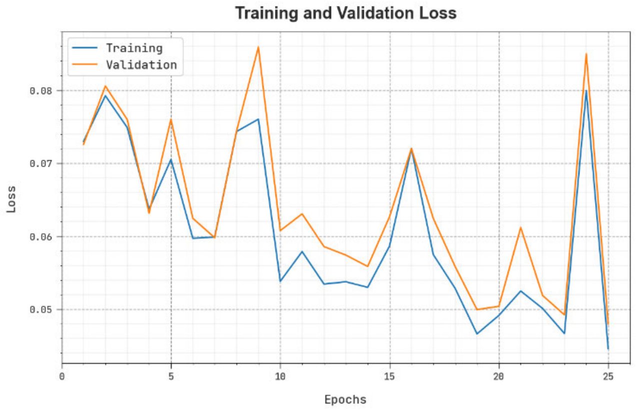

FIGURE 9. TLOS and VLOS outcomes of the HMMLB-BND approach.

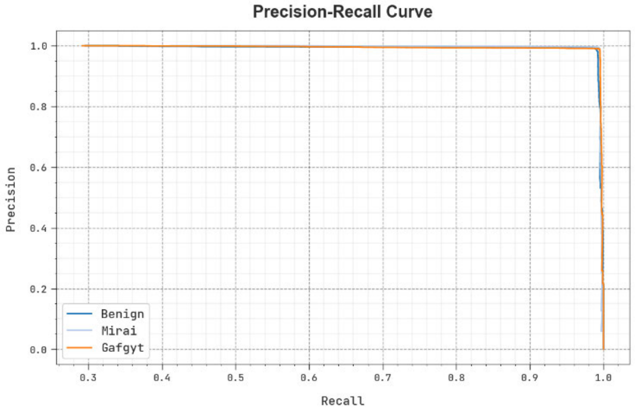

FIGURE 10. Precision-recall outcome of HMMLB-BND approach.

imply improvements in the HMMLB-BND technique in terms of several measures. The outcomes stated that the LSTM and CNN-RNN approaches reach the least outcomes while the DNN-LSTM, LSTM-CNN, and DNN models accomplish nearer classification performance.

TABLE 3. Comparative outcome of HMMLB-BND method with existing algorithms.

Methods	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}
HMMLB-BND	99.43	99.13	99.12	99.13
BND-BMOML	99.04	98.67	98.66	98.70
DNN-LSTM	98.84	98.10	97.99	97.86
LSTM	96.89	95.70	94.35	94.95
CNN-RNN	96.19	93.73	97.35	93.81
LSTM-CNN	98.60	96.74	97.42	95.90
DNN	98.52	96.74	96.16	94.51

Next, the BND-BMODL model results in considerable outcomes with $accu_y$, $prec_n$, $reca_l$, and F_{score} of 99.04%, 98.67%, 98.66%, and 98.70% respectively. But the HMMLB-BND technique reaches maximum performance with $accu_y$, $prec_n$, $reca_l$, and F_{score} of 99.43%, 99.13%, 99.12%, and 99.13% correspondingly. These outcomes demonstrate the superior performance of the HMMLB-BND technique in the botnet detection process.

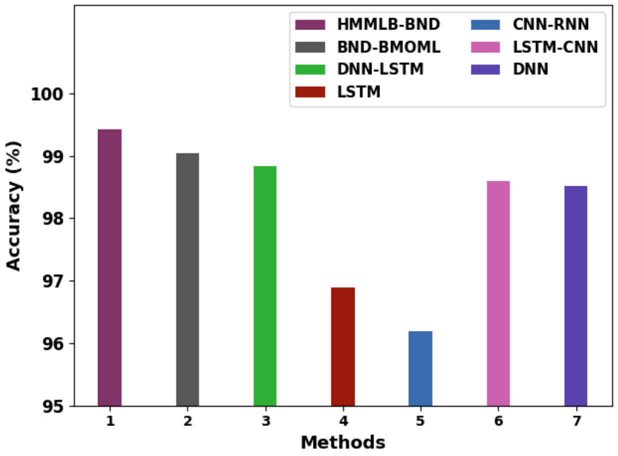

FIGURE 11. Comparative outcome of HMMLB-BND approach with recent algorithms.

V. CONCLUSION

In this study, we have established a novel HMMLB-BND method in the Cloud Aided IoT environment. The projected HMMLB-BND technique focuses on the detection and classification of botnet attacks in the cloud-based IoT platform. In the presented HMMLB-BND technique, the MFFO algorithm for FS purposes is applied. To detect and classify botnets properly the CBOA with CNN-QRNN model is used. The utilization of the CBOA model helps in attaining improved performance on botnet attack detection. A series of simulations were made on the N-BaIoT dataset to demonstrate the higher performance of the HMMLB-BND technique. The experimental outcomes stated the significance of the HMMLB-BND technique over other existing approaches. In the future, ensemble deep-learning classifiers can extend the performance of the HMMLB-BND algorithm. Besides, future work can investigate the computation complexity of the proposed model. In addition, class imbalance data handing problem will be addressed in future.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through large group Research Project under grant number (RGP2/159 /44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R349), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. We Would like to thank SAUDI ARAMCO Cybersecurity Chair for funding this project. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444). This study is partially funded by the Future University in Egypt (FUE).

REFERENCES

[1] Z. Chen, "Research on internet security situation awareness prediction technology based on improved RBF neural network algorithm," *J. Comput. Cogn. Eng.*, vol. 1, no. 3, pp. 103–108, Mar. 2022.

[2] K. Shinan, K. Alsubhi, A. Alzahrani, and M. U. Ashraf, "Machine learning-based botnet detection in software-defined network: A systematic review," *Symmetry*, vol. 13, no. 5, p. 866, May 2021.

[3] S. Namasudra, R. G. Crespo, and S. Kumar, "Introduction to the special section on advances of machine learning in cybersecurity (VSI-mlsec)," *Comput. Electr. Eng.*, vol. 100, May 2022, Art. no. 108048.

[4] A. Gutub, "Boosting image watermarking authenticity spreading secrecy from counting-based secret-sharing," *CAAI Trans. Intell. Technol.*, vol. 8, no. 2, pp. 440–452, 2023.

[5] S. Das and S. Namasudra, "Multiauthority CP-ABE-based access control model for IoT-enabled healthcare infrastructure," *IEEE Trans. Ind. Informat.*, vol. 19, no. 1, pp. 821–829, Jan. 2023.

[6] A. A. Laghari, A. A. Khan, R. Alkanhel, H. Elmannai, and S. Bourouis, "Lightweight-BIoV: Blockchain distributed ledger technology (BDLT) for Internet of Vehicles (IoVs)," *Electronics*, vol. 12, no. 3, p. 677, Jan. 2023.

[7] M. Waqas, K. Kumar, A. A. Laghari, U. Saeed, M. M. Rind, A. A. Shaikh, F. Hussain, A. Rai, and A. Q. Qazi, "Botnet attack detection in Internet of Things devices over cloud environment via machine learning," *Concurrency Comput., Pract. Exp.*, vol. 34, no. 4, Feb. 2022, Art. no. e6662.

[8] A. A. Laghari, X. Zhang, Z. A. Shaikh, A. Khan, V. V. Estrela, and S. Izadi, "A review on quality of experience (QoE) in cloud computing," *J. Reliable Intell. Environ.*, pp. 1–15, Jun. 2023.

[9] M. Alauthman, N. Aslam, M. Al-kasassbeh, S. Khan, A. Al-Qerem, and K.-K. Raymond Choo, "An efficient reinforcement learning-based botnet detection approach," *J. Netw. Comput. Appl.*, vol. 150, Jan. 2020, Art. no. 102479.

[10] S. Sarkar, K. Saha, S. Namasudra, and P. Roy, "An efficient and time saving web service based Android application," *SSRG Int. J. Comput. Sci. Eng.*, vol. 2, no. 8, pp. 18–21, 2015.

[11] A. Wani, S. Revathi, and R. Khaliq, "SDN-based intrusion detection system for IoT using deep learning classifier (IDSIoT-SDL)," *CAAI Trans. Intell. Technol.*, vol. 6, no. 3, pp. 281–290, Sep. 2021.

[12] F. Sattari, A. H. Farooqi, Z. Qadir, B. Raza, H. Nazari, and M. Almutiry, "A hybrid deep learning approach for bottleneck detection in IoT," *IEEE Access*, vol. 10, pp. 77039–77053, 2022.

[13] M. Habib, I. Aljarah, H. Faris, and S. Mirjalili, "Multi-objective particle swarm optimization for botnet detection in Internet of Things," in *Evolutionary Machine Learning Techniques*. Singapore: Springer, 2020, pp. 203–229.

[14] R. Vinayakumar, M. Alazab, S. Srinivasan, Q.-V. Pham, S. K. Padannayil, and K. Simran, "A visualized botnet detection system based deep learning for the Internet of Things networks of smart cities," *IEEE Trans. Ind. Appl.*, vol. 56, no. 4, pp. 4436–4456, Jul. 2020.

[15] A. Al Shorman, H. Faris, and I. Aljarah, "Unsupervised intelligent system based on one class support vector machine and grey wolf optimization for IoT botnet detection," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 7, pp. 2809–2825, Jul. 2020.

[16] S. I. Popoola, B. Adebisi, R. Ande, M. Hammoudeh, K. Anoh, and A. A. Atayero, "SMOTE-DRNN: A deep learning algorithm for botnet detection in the Internet-of-Things networks," *Sensors*, vol. 21, no. 9, p. 2985, Apr. 2021.

[17] F. S. Alrayes, M. Maray, A. Gaddah, A. Yafoz, R. Alsini, O. Alghushairy, H. Mohsen, and A. Motwakel, "Modeling of botnet detection using barnacles mating optimizer with machine learning model for Internet of Things environment," *Electronics*, vol. 11, no. 20, p. 3411, Oct. 2022.

[18] S. Srinivasan and P. Deepalakshmi, "Enhancing the security in cyber-world by detecting the botnets using ensemble classification based machine learning," *Meas., Sensors*, vol. 25, Feb. 2023, Art. no. 100624.

[19] M. Catillo, A. Pecchia, and U. Villano, "A deep learning method for lightweight and cross-device IoT botnet detection," *Appl. Sci.*, vol. 13, no. 2, p. 837, Jan. 2023.

[20] O. Habibi, M. Chemmakha, and M. Lazaar, "Imbalanced tabular data modelization using CTGAN and machine learning to improve IoT botnet attacks detection," *Eng. Appl. Artif. Intell.*, vol. 118, Feb. 2023, Art. no. 105669.

[21] S. Khan and A. B. Mailewa, "Discover botnets in IoT sensor networks: A lightweight deep learning framework with hybrid self-organizing maps," *Microprocessors Microsyst.*, vol. 97, Mar. 2023, Art. no. 104753.

[22] H. Banati and M. Bajaj, "Fire fly based feature selection approach," *Int. J. Comput. Sci. Issues*, vol. 8, no. 4, p. 473, 2011.

[23] L. D. S. Coelho, D. L. D. A. Bernert, and V. C. Mariani, "A chaotic firefly algorithm applied to reliability-redundancy optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2011, pp. 517–521.

[24] M. Y. Alzahrani and A. M. Bamhdi, "Hybrid deep-learning model to detect botnet attacks over Internet of Things environments," *Soft Comput.*, vol. 26, no. 16, pp. 7721–7735, Aug. 2022.

[25] J. Bhayo, S. A. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, "Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks," *Eng. Appl. Artif. Intell.*, vol. 123, Aug. 2023, Art. no. 106432.

[26] S. Siddiqui, S. Hameed, S. A. Shah, I. Ahmad, A. Aneiba, D. Draheim, and S. Dustdar, "Toward software-defined networking-based IoT frameworks: A systematic literature review, taxonomy, open challenges and prospects," *IEEE Access*, vol. 10, pp. 70850–70901, 2022.

[27] M. Khalid, S. Hameed, A. Qadir, S. A. Shah, and D. Draheim, "Towards SDN-based smart contract solution for IoT access control," *Comput. Commun.*, vol. 198, pp. 1–31, Jan. 2023.

[28] G. Hafeez, I. Khan, S. Jan, I. A. Shah, F. A. Khan, and A. Derhab, "A novel hybrid load forecasting framework with intelligent feature engineering and optimization algorithm in smart grid," *Appl. Energy*, vol. 299, Oct. 2021, Art. no. 117178.

[29] N. Al-Taleb and N. Saqib, "Towards a hybrid machine learning model for intelligent cyber threat identification in smart city environments," *Appl. Sci.*, vol. 12, no. 4, p. 1863, Feb. 2022.

[30] A. A. Awad, A. F. Ali, and T. Gaber, "Feature selection method based on chaotic maps and butterfly optimization algorithm," in *Proc. Int. Conf. Artif. Intell. Comput. Vis. (AICV)*. Cham, Switzerland: Springer, Mar. 2020, pp. 159–169.

[31] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12–22, Jul. 2018.