scientific reports

Check for updates

OPEN

Blockchain enhanced distributed denial of service detection in IoT using deep learning and evolutionary computation

V. V. S. H. Prasad[1], Swathi Sowmya Bavirthi[2], C. S. S. Anupama[3], E. Laxmi Lydia[4], K. Sathesh Kumar[5], Khalid Ammar[6]✉ & Mohamad Khairi Ishak[6]

The Internet of Things (IoT) is emerging as a new trend mainly employed in developing numerous vital applications. These applications endure on a federal storage framework primarily concerned with multiple issues. Blockchain technology (BC) is one of the supportive methods for developing IoT-based applications. It is employed to solve the problems encountered in IoT applications. The attack Distributed Denial of Service (DDoS) is one of the leading security attacks in IoT systems. Attackers can effortlessly develop the exposures of IoT gadgets and restrain them as fragments of botnets to commence DDoS threats. The IoT devices are said to be resource-constrained with computing resources and restricted memory. As a developing technology, BC holds the possibility of resolving security problems in IoT. This paper proposes the Metaheuristic-Optimized Blockchain Framework for Attack Detection using a Deep Learning Model (MOBCF-ADDLM) method. The main intention of the MOBCF-ADDLM method is to deliver an effective method for detecting DDoS threats in an IoT environment using advanced techniques. The BC technology is initially applied to mitigate DDoS attacks by presenting decentralized security solutions. Furthermore, data preprocessing utilizes the min-max scaling method to convert input data into a beneficial format. Additionally, feature selection (FS) is performed using the Aquila optimizer (AO) technique to recognize the most relevant features from input data. The attack classification process employs the deep belief network (DBN) technique. Finally, the red panda optimizer (RPO) model modifies the hyper-parameter values of the DBN model optimally and results in higher classification performance. A wide range of experiments with the MOBCF-ADDLM approach is performed under the BoT-IoT Binary and Multiclass datasets. The performance validation of the MOBCF-ADDLM approach portrayed a superior accuracy value of 99.22% over existing models.

Keywords Blockchain, DDoS attack, Deep learning, IoT, Red panda optimizer

During the last decade, the IoT has become vital to real-time applications. It performs a pivotal role in providing critical services that have become essential in everyday life[1]. IoT has considerably impacted different sectors, including smart homes, cities, and medical care. It augments these regions by presenting the substance of consistent technologies[2]. As vast amounts of data exchange and the propagation of gadgets continue to increase, guaranteeing strong protection and safety has become essential for effective resource management. Several investigators have managed their efforts to deal with these challenges, focusing on incorporating BC with IoT[3]. BC technology is developing decentralization-related structures. The distributed computing system employs a point-to-point computing system to address billions of transactions created by IoT methods. Nevertheless, various fields are connected to IoT, including identity management, privacy, security, and more[4]. Among

[1]Department of Mechanical Engineering, Institute of Aeronautical Engineering, Hyderabad, Telangana, India. [2]Department of IT, Chaitanya Bharathi Institute of Technology, Gandipet, Hyderabad 500075, India. [3]Department of Electronics and Instrumentation Engineering, V. R. Siddhartha Engineering College, Deemed to be University, Vijayawada 520007, India. [4]Department of Computer Science and Engineering, Vignan's Institute of Engineering for Women, Visakhapatnam, AP, India. [5]Department of Computer Science and Engineering, Alliance College of Engineering and Design, Alliance University - Central Campus, Bengaluru 562106, India. [6]Department of Electrical and Computer Engineering, College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates. ✉email: k.ammar@ajman.ac.ae

security concerns, DDoS threats pose a severe attack. With the faster development of IoT settings, updating and enhancing related system security extents has become essential to oppose these attacks effectively[5]. Figure 1 signifies the general structure of IoT-based DDoS attacks. IoT gadgets are resource-constrained with restricted computing resources and memory, security protection is absent in these gadgets. To mitigate and detect DDoS attacks in IoT, many solutions are projected[6]. The BC concept is derived from Bitcoin. It does not depend upon a third person to verify or store information around distributed nodes. It is a decentralized data structure formed by a sequence of blocks. BC is employed in multiple regions, comprising e-commerce, IoT, smart contracts, and more[7]. With decentralization, audibility, persistency, and anonymity features, BC technology precludes DDoS attacks in IoT.

Even though BC is a favourable technology for safeguarding the innovation and management of technology, it endures few susceptibilities linked to data confidentiality[8]. Innovative methods and technologies like intrusion detection systems (IDS) should be endorsed to address these concerns. IDSs have a significant challenge: observing the anomalous performance in the host or system[9]. The existing IDSs are ineffective in recognizing the vast array of attacks. Collective IDs have a specific capability of determining some attacks and moving for more processing. IDSs are classified into dual kinds depending on where the IDSs are employed, that is, host-based ID systems (HIDS) and network-based ID systems (NIDS)[10]. The application of multiple deep learning (DL) methodologies for recognizing the threats with dual classification and categorizing diverse threats with multiple-class classification has become an active investigation region.

This paper proposes the Metaheuristic-Optimized Blockchain Framework for Attack Detection using a Deep Learning Model (MOBCF-ADDLM) method. The main intention of the MOBCF-ADDLM method is to deliver an effective method for detecting DDoS threats in an IoT environment using advanced techniques. The BC technology is initially applied to mitigate DDoS attacks by presenting decentralized security solutions. Furthermore, data preprocessing utilizes the min-max scaling method to convert input data into a beneficial format. Additionally, feature selection (FS) is performed using the Aquila optimizer (AO) technique to recognize the most relevant features from input data. The attack classification process employs the deep belief network (DBN) technique. Finally, the red panda optimizer (RPO) model modifies the hyper-parameter values of the DBN model optimally and results in higher classification performance. A wide range of experiments with the MOBCF-ADDLM approach is performed under the BoT-IoT Binary and Multiclass datasets. The key contribution of the MOBCF-ADDLM approach is listed below.

- The MOBCF-ADDLM model utilizes BC to establish a decentralized, tamper-proof infrastructure that improves security against DDoS attacks. This approach enables real-time resistance and logging of malicious attempts. It strengthens the detection framework's trustworthiness while mitigating reliance on centralized systems. It also contributes to reducing single points of failure common in conventional architectures.
- The MOBCF-ADDLM method employs min-max scaling during data preprocessing to normalize input features within a defined range, which improves model stability and accelerates convergence. This preprocessing

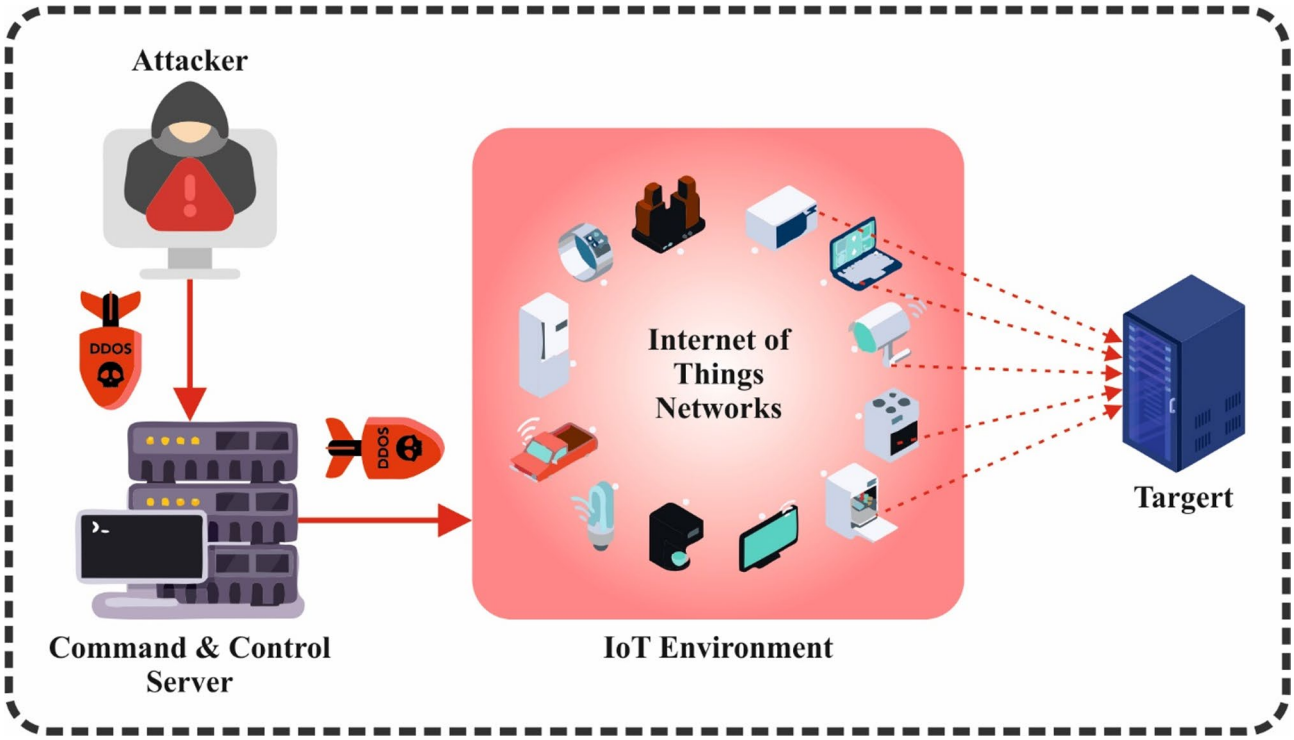

Fig. 1. IoT networks-based DDoS attack detection.

step confirms that no feature dominates due to scale differences, improving overall learning efficiency. It also contributes to cleaner data representation, assisting in more accurate DDoS attack detection.
- The MOBCF-ADDLM approach implements the AO model to perform effective FS by detecting the most relevant attributes from the dataset, thereby mitigating dimensionality and computational overhead. This selection process improves the model's focus on impactful features, enhancing detection performance. It also assists in developing a more efficient and interpretable DDoS detection system.
- The MOBCF-ADDLM methodology utilizes the DBN method to accurately detect DDoS attacks by capturing intrinsic patterns in network traffic. At the same time, the RPO model is used to fine-tune hyperparameters for optimal performance. This integration improves detection precision and model generalization across varying attack scenarios. It also contributes to constructing a robust and adaptive intrusion detection framework.
- The novelty of the MOBCF-ADDLM model is in the unique integration of BC technology to provide decentralized security, incorporated with a hybrid AO-DBN-RPO model that integrates advanced FS, DL, and optimization techniques. This approach ensures accuracy and scalability in DDoS attack detection, addressing the limitations of conventional methods. By utilizing these cutting-edge technologies, the solution presents improved robustness and adaptability in dynamic attack environments.

The article's structure is as follows: Section "Related works" reviews the literature, Section "The proposed methodology" describes the proposed method, Section "Experimental Validation" presents the evaluation of results, and Section "Conclusion" offers the study's conclusions.

Related works

Vijay Anand et al.[11] presented an innovative security structure that utilizes BC technology and lightweight cryptography for protecting electronic health records (EHRs). The projected model utilizes an amended elliptic scheme (AES) for protected key generation, lattice homomorphic re-encryption (LHoRe) for data encryption, and an improved Merkel tree (IMM) hashing methodology for data integrity. Key optimizer is accomplished using the Opposition-Based Coati Optimizer model, whereas encoded data is stored in the IPFS. Ilakkiya and Rajaram[12] developed an innovative DAG-BC structure for MANET-IoT security. The methodology presented the secure trust-based Dijikstra's model with several criteria for safeguarding data transfer. A deep packet examination was employed to recognize intrusions blocked by a blocking mechanism. Indrason et al.[13] developed an automated and secured e-voting structure like MBCSD-IoT. A multi-level voting structure is intended for this objective, where most nodes contain a BC-assisted SDN-based IoT framework. BC is rooted in the IoT method to offer dependable security. In MBCSD-IoT, four stages of hierarchies are intended: country, booth, state, and district-level layers. Sharma et al.[14] presented a progressive method to deal with these challenges around a multi-code trust and BC structure. Employing the immutable and decentralized features of BC technology and a multi-code-driven trust mechanism, this structure focused on generating robust, resilient, and secure settings for IoT gadgets. This model utilizes the transparency of BC to foster validation and trust in system transactions, considerably decreasing the attack surface for DDoS threats. Additionally, the integration of multi-code methods strengthens the extent of safety. Kiran and Nalini[15] developed the SprakGrid model. The projected study contains four consecutive stages: Primarily, the method implements user authentication to guarantee that legal users employ the elliptic curve-based chaos theory model that creates a secret key and stores it in BC. Then, query scheduling is implemented for resource finding, leveraging the soft actor-critic model by deliberating parameters of 3Ps that are accomplished by spark setting and scheduling optimum resources depending on the service request.

The author[16] presented a PoAh-enabled structure of FL for DDoS threat recognition in IoT. In addition, BC is employed in the authentication layer with PoAh to ensure performance validation, higher security, and data authentication in IoT. Halim et al.[17] developed a Chain of Things (CoT), a BC-based structure to improve IoT security applications. By employing decentralization of BC, transparent and immutable assets, and reliable data sharing, CoT safeguards secure communication and tamper-proof logging of IoT transactions. The presented structure combines smart contracts for scalability and automated policy enforcement, dealing with heterogeneous settings and dynamics of IoT. Ohri et al.[18] introduced an innovative model combining Proof of Work (PoW) and Ethereum smart contracts to handle these challenges. Employing DoS attacks, BC with SDN, authentication, and spoofing threats are effectually reduced. This method discovers the possibility of Ethereum BC technology, specifically its PoW consensus algorithm and smart contracts, to improve access control inside multi-SDN settings. Kachavimath and Narayan[19] developed a robust DDoS attack detection model for SDN utilizing an ensemble learning (EL) method that integrates extreme gradient boosting (XGBoost) and histogram-based gradient boosting (HGBClassifier) techniques with optimal feature selection for high detection accuracy. The model is examined by utilizing decision tree (DT), logistic regression (LR), random forest (RF), and an ensemble classifier approach. Sumathi and Rajesh[20] proposed a hybrid grey wolf optimizer (GWO) + back propagation network (BPN) + self-organizing map (SOM) IDS for improved DDoS attack detection in cloud computing environments, addressing challenges such as overfitting, detection delay, and high false positive rate (FPR) through advanced feature selection and hyperparameter tuning techniques. Abdullah et al.[21] proposed a federated learning-based DoS attack detection and classification model (FLDoSADC-DTL) technique for BC-assisted IIoT environments. The model utilizes a sand cat swarm algorithm (SCSA) for feature selection, a stacked auto-encoder (SAE) for detection, and a black widow optimization algorithm (BWOA) technique for hyperparameter tuning.

Sokkalingam and Ramakrishnan[22] proposed a hybrid ML-based IDS that utilizes 10-fold cross-validation for feature selection and fine-tunes support vector machine (SVM) parameters using an integration of Harris Hawks optimization (HHO) and particle swarm optimization (PSO) models. Saraswathi and Dayana[23] developed a robust IDS system for 6G networks by utilizing an LSTM-RNN model integrated with the NADAM optimizer

to detect growing cyber-attack patterns and address gradient vanishing issues. Sumathi, Rajesh, and Lim[24] presented an efficient IDS for DDoS attack detection utilizing a long short-term memory (LSTM) recurrent neural network and autoencoder-decoder-based DL technique, with optimized network parameters by using a hybrid HHO and PSO method. Wazid et al.[25] developed a secure DL-based malware attack detection model (SDLMA-IITS) methodology for IoT-enabled intelligent transportation systems, integrating explainable AI (XAI) to enhance detection efficiency and security analysis. Sumathi and Rajesh[26] developed a hybrid artificial neural network (ANN)-based IDS for DDoS attack detection using a backpropagation neural network (BPN) and multilayer perceptron (MLP), optimized with a hybrid HHO-PSO model for feature selection and tuning. Almseidin et al.[27] introduced a hybrid detection model that investigates the capability to integrate PSO and GWO techniques to improve the DNN architecture to detect the Sunburst attack. The PSO model is employed for optimizing the learning rate and the number of hidden layers (HLs), while the GWO method is used to optimize the neuron weight. Sumathi and Rajesh[28] detected DDoS attacks, specifically TCP SYN flood attacks, utilizing data mining and ML approaches while computing performance metrics. Mehmood et al.[29] proposed a model to improve DDoS attack detection in software-defined networks (SDN) utilizing MLP and CNN models, enhanced with SHapley Additive exPlanations (SHAP)-based feature selection and Bayesian optimization (BO) model for accurate and efficient performance.

The limitations of the existing studies are high FPR, delayed convergence, overfitting, and poor scalability in complex environments, namely SDN, IoT, and cloud. Various IDS models face difficulty with real-time detection, adaptability to growing attack patterns, and multi-source DDoS detection. The lack of unified frameworks incorporating BC, DL, and ML with efficient FS and hyperparameter tuning is evident. Furthermore, most works lack explainability and transparency in predictions. A research gap exists in developing lightweight, explainable, and scalable IDS architectures with incorporated optimization and low detection latency across heterogeneous infrastructures.

The proposed methodology

This manuscript proposes the MOBCF-ADDLM model. The main intention of the MOBCF-ADDLM methodology is to deliver an effective technique for identifying DDoS threats in an IoT environment using advanced techniques. It involves various phases, including BC technology, min-max scaling, the FS process, attack classification, and parameter tuning. Figure 2 exemplifies the entire flow of the MOBCF-ADDLM methodology.

BC technology in DDoS attack

The BC technology is initially applied to mitigate DDoS attacks by presenting decentralized security solutions[30]. BC presents crucial merits in mitigating DDoS attacks, specifically in decentralized environments like IoT networks. Unlike conventional centralized solutions, BC confirms data integrity and tamper-proof records, making it difficult for attackers to manipulate or disrupt the detection system. Additionally, the decentralized behaviour of BC eliminates single points of failure, improving resilience against large-scale DDoS attacks. BC provides real-time automated responses to attacks by utilizing smart contracts and consensus mechanisms, thereby enhancing the efficiency of the detection and mitigating latency. BC also shows excellence over conventional centralized or less secure methods, giving both security and scalability in distributed systems. Figure 3 illustrates the structure of the BC technology.

The BC is a homologous system of removing nodes. All nodes promote the reproduction of economic transactions as a record and confirm the continuous transactions using the agreement device incorporated into the BC. Then, the transaction depends on a cryptographic hash model and becomes an assessment of the ledger, and the process is repeated. The hash is essential in joining the block; therefore, all blocks include its hash and the following hash of blocks. The strength of these transmissions is the compelling liabilities, which makes it firmer for the invader to interrupt utilizing the record. The invader requires the cooperation of the mainstream of the nodes to adjust the ledger's stability. As a particular news source stated, BC platforms have become the base of the most noteworthy DDoS attacks. To prevent the repetition of such attacks, it is essential to identify the relevant platform applications to prevent forthcoming chapters. Let us address how the attackers produced attacks of DDoS on the IoT platform. The BC typically does not require a reliable third party to transact with its nodes. In the same way, BC allows tamper-proof and secure transactions; therefore, someone is capable of proving them. In general, there are dual DDoS attacks, such as the transaction attack of floods and the attack of UDPs. While in the transaction attacking flood, the invader transfers many spam transactions with proper mining, conveying fees, and underlining the transactions for a block. The BC gives original blocks for the chain.

Min-Max scaling

Furthermore, the min-max scaling method is employed for the data preprocessing process for converting input data into a beneficial format[31]. This method is chosen due to its effectualness in normalizing input features between a constant range [0, 1], ensuring uniformity across all attributes. This prevents any one feature from dominating the model due to scale differences, resulting in more balanced training. Unlike other techniques, such as Z-score normalization, which assumes a Gaussian distribution, Min-Max scaling does not require assumptions about the underlying distribution of the data, making it appropriate for diverse datasets. It also enhances the performance of optimization algorithms by accelerating convergence. The model is advantageous in DL techniques such as DBNs, where stable and well-scaled data is significant for accurate and efficient training.

The normalization of features is implemented over Min-max scaling, succeeding the given Eq. (1). This model guarantees that each feature is rescaled to fit inside the interval [0,1], thus determining that the minimal and maximal values of some variable or feature should be $(0,1)$, correspondingly.

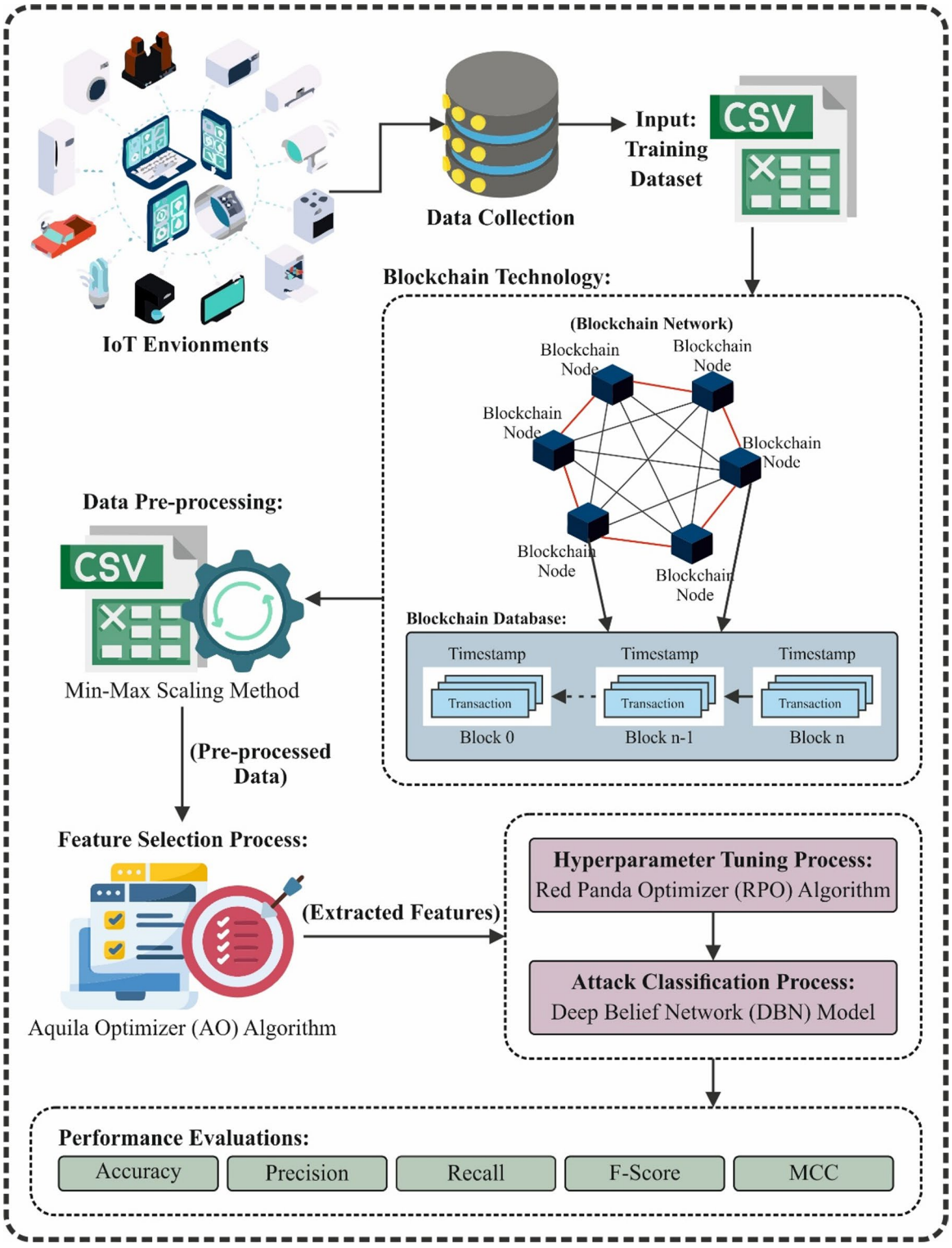

Fig. 2. Overall flow of MOBCF-ADDLM approach.

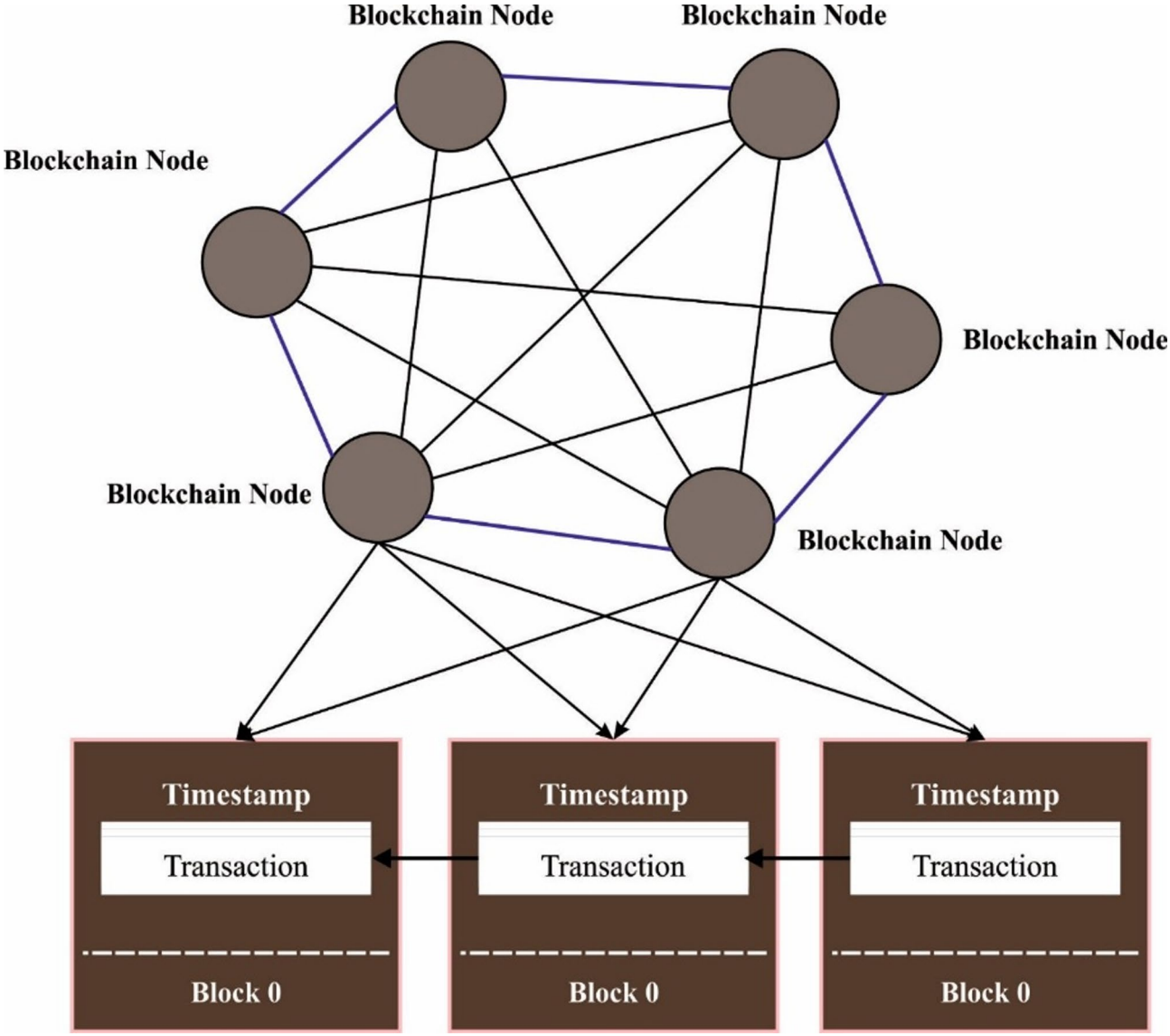

Fig. 3. BC framework.

$$Cscaled = \frac{C - C_{\min}}{C_{\max} - C_{\min}} \tag{1}$$

C_{min} and C_{max} characterize the minimal and maximal values of the feature vectors, respectively.

AO-based FS

Besides, the AO model implements the FS process to recognize the most relevant features from input data[32]. This model is selected for its superior capability to explore massive, high-dimensional search spaces. Unlike conventional techniques such as genetic algorithms or random search, this model can effectually detect the most relevant features and balance exploration and exploitation, thereby enhancing its convergence speed and robustness. This enables AO to choose optimal features that improve model accuracy while mitigating dimensionality, significantly improving computational efficiency. Additionally, the adaptability of the AO model to diverse dataset types and its robust performance in complex tasks make it a powerful tool in DDoS detection. By concentrating on the most relevant features, AO improves the capability of the technique to generalize and detect attacks accurately across varying conditions. Figure 4 specifies the steps involved in the AO model.

The AO model is stimulated by hawks' searching behaviour, imitating their agility and efficiency in the prey-searching process. The model generates an agent population that iteratively updates its locations inside the searching region to approach or reach the global best gradually. AO combines numerous optimizer approaches, containing random search, Levy flight (LF) stages, and swarm intelligence, improving global searching abilities and avoiding local best entrapment. The particular mathematical modelling procedure of the AO model is described below.

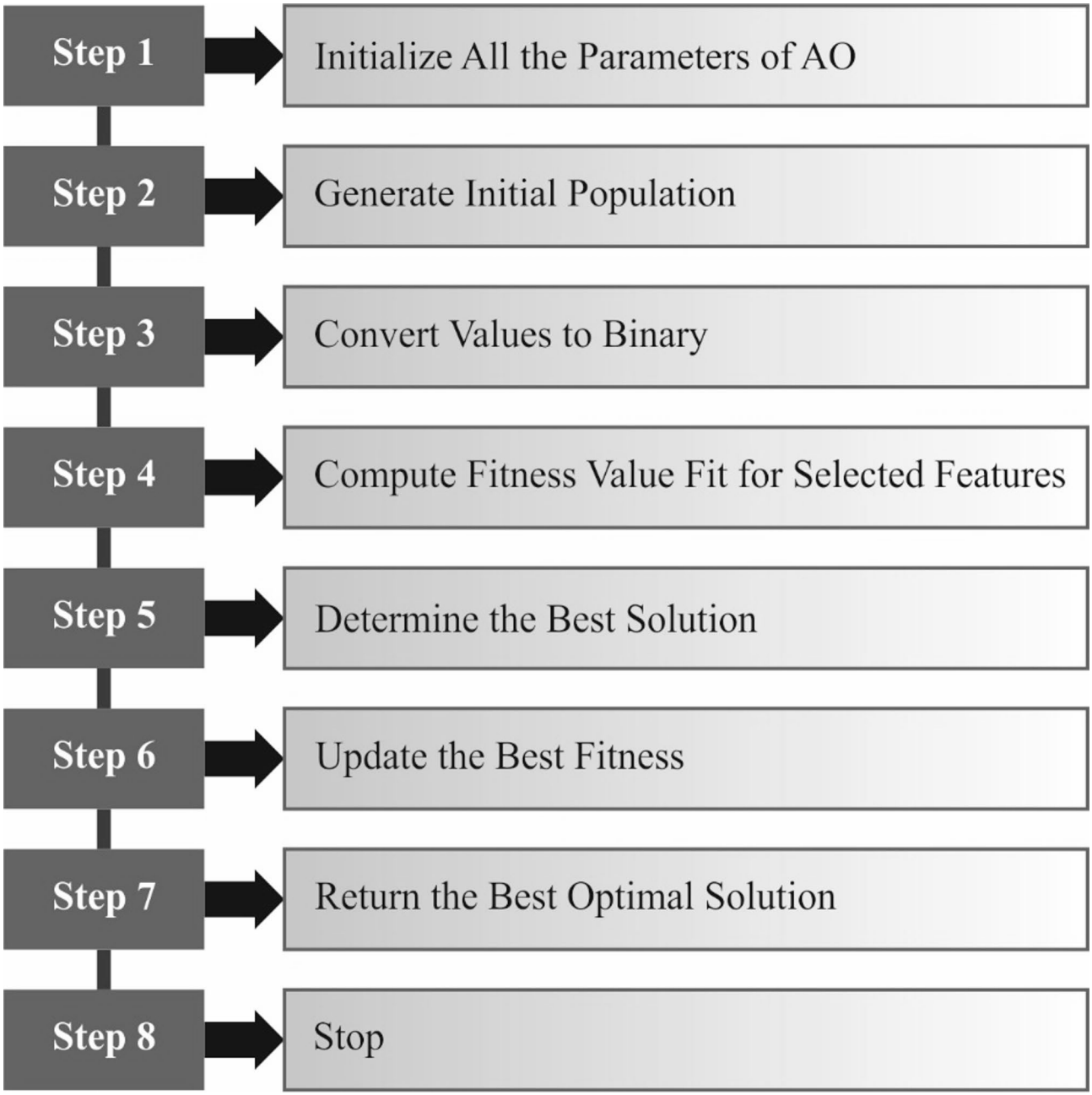

Fig. 4. Steps involved in the AO technique.

Initialization Stage

During this first stage, the AO model randomly creates a set of hawk agents (solution vectors) to generate the primary population. All hawk agents' location vector x_i is uniformly distributed inside the described search space. The number of hawk agents within the population is fixed to N. For all hawk agent's x_i $(i = 1,2, \ldots, N)$, every dimension $x_{i,j} (j = 1,2, \ldots, d)$ of its location vector is randomly initialized inside the stated limits $[lb_j, ub_j]$, is demonstrated:

$$x_{i,j} = lb_j + rand() \times (ub_j - lb_j) \tag{2}$$

Now, $rand$ () characterizes a randomly generated uniform number inside the range $[0,1]$.

The fitness value f_i for every hawk agent's location vector x_i is computed utilizing the objective function $f(x)$, is stated as:

$$f_i = f(x_i) \tag{3}$$

According to optimizer objectives, fitness values are measured to identify the optimal agent X^* in the present population, as presented:

$$x^* = \begin{cases} arg\min_{X_i} f(x_j), \ if\ minimizing \\ arg\max_{X_i} f(x_j), \ if\ maximizing \end{cases} \tag{4}$$

LF step calculation

LF is a random stage with a long-tailed distribution used to improve the model's global searching ability. The computation equation for the LF stage L is shown:

$$L = \frac{u}{|v|^{1/\beta}} \times multiplier \tag{5}$$

Whereas $u \sim N(0, \sigma^2)$: u denotes a normally distributed arbitrary number with standard deviation σ and mean 0; $v \sim N(0,1)$: v is an abnormally distributed arbitrary number with standard deviation one and mean 0; β denotes Levy index, commonly among (1, 3), controlling stage distribution features; σ denote stage scale parameter, computed as:

$$\sigma = \left(\frac{\Gamma(1+\beta)\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)\beta\, 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{\frac{1}{\beta}} \tag{6}$$

Now, $\Gamma(\bullet)$ signifies the Gamma function.

Position Update.

The AO model iteratively upgrades the hawk agents' location vectors to approach the global best solution slowly. It is separated into exploitation and exploration stages, established by the present iteration count. The exploitation and exploration stages are presented.

(1) Exploration Stage

The model concentrates on a global search in the exploration stage to prevent trapping in local bests. It mainly upgrades hawk locations utilizing the succeeding dual tactics:

Tactic 1: Moving toward the best global direction, as presented in Eq. (7).

$$x_i^{new} = x^* \times \left(1 - \frac{t}{T}\right) + rand() \times (\bar{x} - x^*) \tag{7}$$

Now, t signifies the present iteration count, T denotes the maximal iterations, $rand()$ is a randomly generated number in $[0,1]$, and $\overline{X}$ is the population's average location vector.

Tactic 2: LF incorporated with arbitrary agent impact, as described:

$$x_i^{new} = x^* \times L + x_j + rand() \times (y - x) \tag{8}$$

Now, L denotes LF step length, χ_j refers to the location vector of an arbitrarily chosen hawk agent, and x and y are pre-defined parameter vectors.

(2) Exploitation Stage

In this stage, the model underlines local search to speed up convergence. It mainly updates hawk locations utilizing the following dual tactics:

Tactic 1: Fine-tune depending on average location and global optimal solution, as presented in Eq. (9):

$$x_i^{new} = \alpha \times (x^* - \overline{x}) - rand() \times (rand() \times (ub - lb) + lb) \times \delta \tag{9}$$

Whereas α and δ indicate controller parameters, ub and lb represent decision variables' lower and upper limits.

Tactic 2: Complete adjustment according to the

$$x_i^{new} = QF \times x^* - (g_2 \times x_j \times rand()) - g_2 \times L + rand() \times g_1 \tag{10}$$

Now, $QF = t^{\frac{2\times rand()-1}{(1-T)^2}}$, $g_1 = 2 \times rand(-1,\ g_2 = 2 \times \left(1 - \frac{t}{T}\right)$.

Boundary Correction

Afterwards, the position was updated, and clipping was used to guarantee that Hawk agent location vectors x continued inside pre-defined limits.

$$x_j^{new} = clip\left(x_j^{new}, lb, ub\right) \tag{11}$$

Now, the clipping process limits every dimension of x_i^{new} inside the $[lb_j, ub_j]$ range.

Fitness Selection and Evaluation.

The fitness value $f(x_i^{new})$ of upgrade hawk agents is calculated. According to optimizer goals, the novel fitness is compared to the original, preserving the excellent agents, as demonstrated in Eq. (12).

$$x_i = \begin{cases} x_i^{new}, \ if \ improved \\ x_i, \ otherwise \end{cases} \tag{12}$$

At the end of every generation, the population is re-ranked to upgrade the global optimal solution X^*. The model ends after the maximal iterations T are made or another convergence condition is encountered. Output the global optimal solution X^* and its fitness value $f(x^*)$.

The fitness function (FF) reveals the classification accuracy and the volume of chosen features. It exploits the accuracy of classification and lessens the set size of FSs. So, the FF mentioned below is employed to assess individual solutions, as revealed in Eq. (13).

$$Fitness = \alpha \times ErrorRate + (1-\alpha) \times \frac{\#SF}{\#All\_F} \tag{13}$$

While $ErrorRate$ is the classification rate of error utilizing the chosen features. $ErrorRate$ is intended as the ratio of incorrect categories between 0 and 1. $\#SF$ represents the number of preferred features, and $\#All\_F$ means the total amount of characteristics in an original dataset. α is generally applied to control classifier quality and sub-set length prominence.

Attack classification using the DBN model

The DBN technique is employed for the attack classification process[33]. This technique is chosen because it can model complex, high-dimensional data through a multi-layered, unsupervised learning process. The model outperforms automatically extracting hierarchical features, which is crucial for detecting complex patterns in DDoS attack traffic that conventional models might miss. The model shows excellence in capturing non-linear data relationships, resulting in more accurate classifications. Their DL architecture enables the model to enhance over time with massive datasets, making them highly adaptable for growing attack patterns. Moreover, DBNs are prevalent for their robust generalization capabilities, making them more resilient to overfitting and more reliable in real-world scenarios. DBNs are excellent for accurate, scalable attack detection in dynamic environments. Figure 5 depicts the framework of the DBN model.

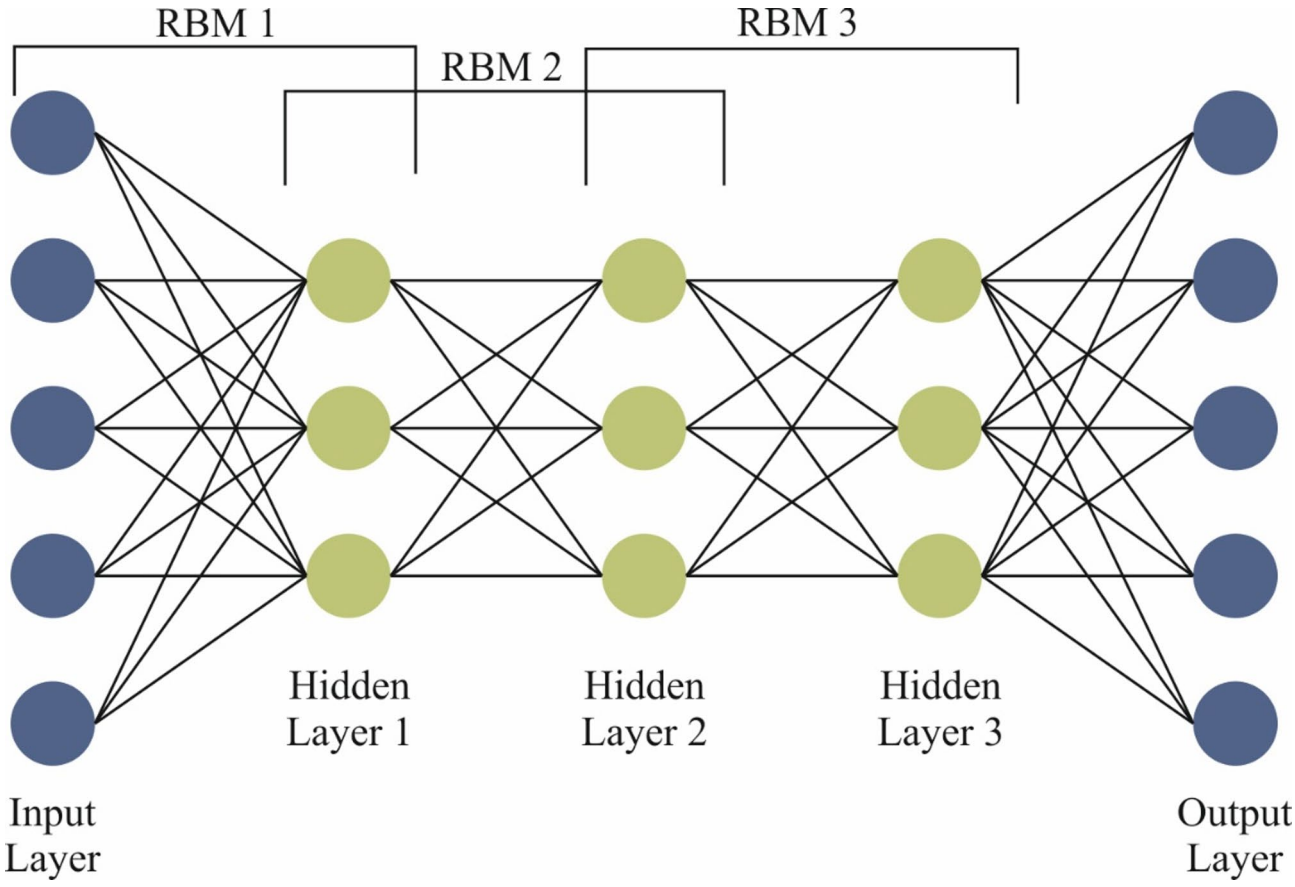

Fig. 5. Framework of DBN.

DBN is an unsupervised ANN model that contains many stacked restricted Boltzmann machines (RBM). The RBM includes dual layers: the HL and the visible layer. During a DL model moulded by stacking RBMs, the output layer HL of the preceding RBM functions as the input layer for the succeeding RBM component, making sequential stacking RBMs to determine the framework of DBN. Eventually, an additional output layer was integrated into the final DBN structure.

$$p\left(v, h^1, h^2, \ldots, h^j\right) = p\left(v|h^1\right) p\left(h^1|h^2\right) \ldots .p\left(h^{j-1}|h^j\right) \tag{14}$$

Here, $p\left(h^{j-1}\middle| h^j\right), j = 2,3,\ldots\ l$ is the conditional probability distribution among $(j-1)$ and j^{th} HL. The energy relations among hidden and visible neurons are depicted.

$$E\left(v,\ h;\theta\right) = -\sum_{i=1}^{m} v_i b_i - \sum_{j=1}^{n} c_j h_j - \sum_{i=1}^{m}\sum_{j=1}^{n} v_i h_j \omega_{\ ij} \tag{15}$$

Here, $\theta = \{\omega_{\ ij},\ b_i,\ c_j\}$ is the network parameters; $\omega_{\ ij}$ depicts the weight between the HL and explicit layer neurons; b_i signifies the bias of the layer neuron; and c_j is the bias of the HL neuron. m specifies the neuron counts in the visible layer, and n denotes neuron counts in HL.

The input X of the DBN model is demonstrated:

$$X = v = [v_1,\ v_2,\ v_3,\ \ldots\ ,\ v_m] \tag{16}$$

In Eq. (16), X specifies the set of relevant parameters.

$$p\left(v,\ h\right) = \frac{1}{Z} e^{-E(v,h)} \tag{17}$$

Here, Z represents the allocation function employed to fine-tune the allocation value of $e^{-E(v,h)}$. The neurons in HL and visible layers are fully connected in either direction. The hidden units are proficient in acquiring the co-relations of higher-order data offered in the visible layer. The activation function of system hidden and explicit neurons is calculated:

$$p\left(v_i = 1|h\right) = \sigma\ \left(\sum\nolimits_{j=1}^{n} \omega_{\ ij} h_j + b_i\right) \tag{18}$$

$$p\left(h_j = 1|v\right) = \sigma\ \left(\sum\nolimits_{i=1}^{m} \omega_{\ ij} v_i + c_j\right) \tag{19}$$

Here $\sigma\ (x)$ represents function of sigmoid that are demonstrated:

$$\sigma\ (x) = \frac{1}{1+e^{-x}} \tag{20}$$

To train RBM related to the divergence learning model for finding the optimum weight $\omega_{\ ij}$ and constantly upgrade the attained novel biases and weights.

$$\omega_{\ ij}^{(s)} \leftarrow\ \omega_{\ ij}^{(s-1)} + \lambda\ \left(p\left(h_i|v_i;\theta\right) v_i - p\left(h_{i+1}|v_{i+1};\theta\right) v_{i+1}\right) \tag{21}$$

$$c_i \leftarrow\ c_{i-1} + \lambda\ \left(h_i - h_{i+1}\right) \tag{22}$$

$$b_j \leftarrow\ b_{j-1} + \lambda\ \left(v_i - v_{i+1}\right) \tag{23}$$

Here, λ represents the learning rate, and s is the RBM counts.

Parameter optimizer using RPO technique

Finally, the RPO technique modifies the hyper-parameter values[34]. The model effectively balances exploration and exploitation in complex optimization tasks. The red pandas are known for their intellectual behaviour and can navigate high-dimensional search spaces to find optimal hyperparameters that improve the model's performance. Compared to conventional optimizers like grid search or even some evolutionary algorithms, RPO presents faster convergence and avoids local minima more effectually. Its lightweight computational behaviour is appropriate for large-scale models such as DBNs utilized in DDoS detection. By fine-tuning model parameters precisely, RPO significantly improves detection accuracy and reduces training time, making it a robust and scalable choice for real-time applications. Figure 6 demonstrates the RPO technique.

One of the nature-inspired advanced models is named RPO. RP's foraging approaches and ability to climb trees were applied as the primary source of ideas for the RPO's design. Its main design idea is stimulated by the dual natural behaviours of the red pandas: searching and climbing trees to relax. Dual stages of the recommended RPO model are modelled mathematically: an exploration stage depending on the searching approach of RPs and an exploitation stage according to the movements of RPs as they climb trees. The main benefit of the proposed model is that it doesn't need a parameter adjustment process since no controller parameter is applied to mathematical models. It is an animal that generally lives at night. Due to its extraordinary climbing abilities, it

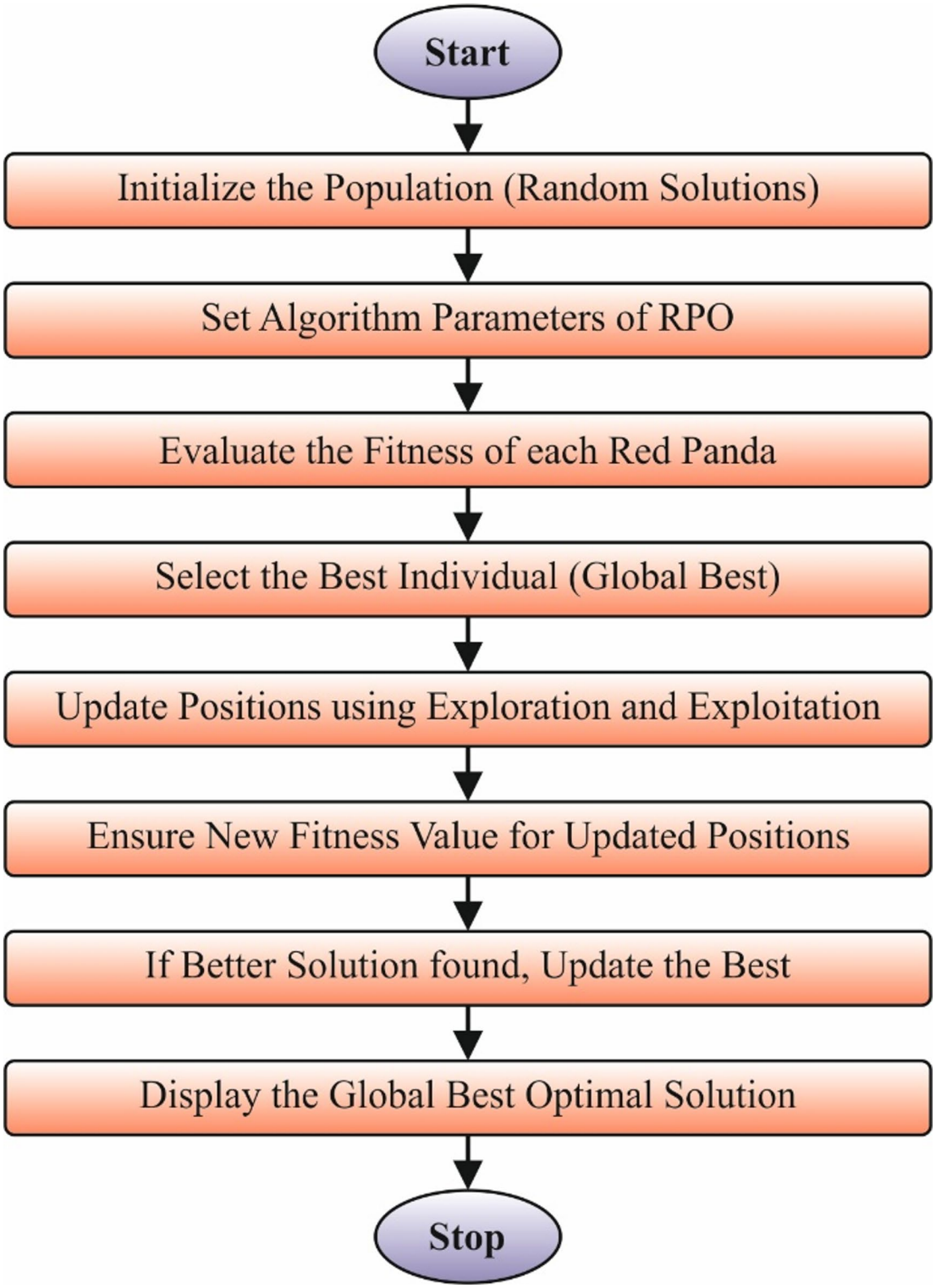

Fig. 6. Framework of DBN.

spends the whole day resting and sleeping in the highest places, like trees. Mathematical representations notified the RPO model's design of the RP's natural behaviours. The mathematical modelling of updated promising solutions in the exploitation and exploration stages is explained, utilizing the imitation of the RP's natural behaviours.

Initialization.

1. Initialize the population of red pandas with random positions.

2. Compute the fitness of each red panda.

3. Set iteration counter $t = 1$.

4. While $t \leq maximum$ iterations do:

   a. For every red panda, update its position using the exploration phase.

   b. Compute the fitness of the updated positions.

   c. Update positions using the exploitation phase.

   d. Compute the fitness of the new positions.

   e. Update the optimal solution found so far.

   f. Increment iteration counter t.

5. Return the optimal solution found.

Algorithm 1. RPO model.

From a mathematical view, a vector is applied to model every RP or candidate solution. Based on Eq. (24), a matrix is applied to mathematically model the RPs of the model population. RP's first location in the searching region is arbitrarily established at the beginning of RPO performance utilizing Eq. (25).

$$X = \begin{bmatrix} x_1 \\ ? \\ x_i \\ ? \\ x_N \end{bmatrix} \tag{24}$$

$$\chi_{ii} = lb_j + r_{ij}\left(ub_j - lb_j\right),\ i = 1,2,\ldots,N, j = 1,2,\ldots,M \tag{25}$$

Whereas X refers to the population matrix of RPs' positions, x_i denotes $ith\ RP$, N means RPs counts, x_{ij} is its jth size (problem variable), M presents the problem variable counts, r_{ij} are randomly generated numbers within the range [0,1], ub_j, and lb_j represents upper and lower limits of the jth problem variable, correspondingly. A matrix is applied to exemplify the collection of estimated values for the objective function following Eq. (26).

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix} \tag{26}$$

As the algorithm executes, the solution presented becomes the optimal candidate solution discovered in the model's iterations. Upgrading candidate solutions in the recommended RPO has dual stages: exploitation and exploration.

Exploration:

During the investigation of the targeted functional values, Eq. (27) is used to model the collection of recommended food source places for all RPs. The equivalent RP should arbitrarily select an either suggested place as its eating place.

$$PF_i = \{X_k | k \in \{1,2,\ldots N\}\ and\ F_k < F_i\} \cup \{X_{best}\} \tag{27}$$

Meanwhile, PF_i denotes a collection of recommended food sources for RP, and X_{best} represents RP's location with the objective function's finest value. The following stage is to compute a novel location utilizing Eqs. (28) and (29).

$$X_i^{p1} : X_{ii}^{p1} = x_{ii} + r.\left(SF_{ii} - I.x_{ij}\right) \tag{28}$$

$$X_i = \begin{cases} X_i^{p1}, if\ F_i^1 < F_i; \\ X_i,\ else \end{cases} \tag{29}$$

Whereas X_i^{p1} stands for original place of the ith RP according to the RPO's initial step, $X_{i,j}^{p1}$ is its jth dimension, F_i^{p1} showing this value of the objective function, SF_i symbolizes chosen food resource for $ith\ RP$, SF_{ii} specifies its jth dimension, r stands for the randomly generated number within the range (0,1), and I denote arbitrarily selected integer from the set {1,2}.

Exploitation:

The RP's capacity for climbing trees and living is classified in the next phase of the RPO. They spend their whole days sleeping on trees. It has small movements after it moves nearer to and climbs the tree, which improves the capability of the presented RPO model to exploit and locate search in regions that make progress. Utilizing Eq. (30), an original place is primarily specified for all RPs. Then, utilizing Eq. (31), this novel location substitutes the previous place of the corresponding RPs when the objective function value is improved.

$$X_{i,j}^{p2} = x_{i,j} + \frac{lb_j + r.\left(ub_j - lb_j\right)}{t}, i = 1,2,\ \ldots\ ,\ N, j = 1,2,\ldots\ ,M, t = 1,2,\ \ldots\ ,T \tag{30}$$

$$X_i = \begin{cases} X_i^{p2}, if\ F_i^2 < F_i; \\ X_i,\ else \end{cases} \tag{31}$$

Whereas T refers to maximal iteration counts, $X_{i,j}^{p2}$ is its j^{th} dimension, F_i^{p2} specifies its value of the objective function, r denotes a randomly generated number within the range [0,1], b characterizes the model's iteration counter, and X_i^{p2} represents the novel location of the ith RP according to the second stage of RPO. Algorithm 1 specifies the RPO technique.

The RPO method presents the FF for attaining an enriched classification performance. It defines an optimistic numeral to epitomize the better outcome of the candidate solution.

$$\begin{aligned} fitness\left(x_i\right) &= ClassifierErrorRate\left(x_i\right) \\ &= \frac{no.\ of\ misclassified\ samples}{Total\ no.\ of\ samples} \times\ 100 \end{aligned} \tag{32}$$

Experimental validation

Here, the performance analysis of the MOBCF-ADDLM method is examined under dual datasets, namely the BoT-IoT Binary and Multiclass datasets[35]. The latter holds 34 features, but only 25 are selected. In contrast, the BoT-IoT Binary dataset contains 2056 samples under dual classes such as attack and normal, as shown in Table 1.

Figure 7 illustrates the classifier outcome of the MOBCF-ADDLM approach on the BoT-IoT Binary dataset. Figure 7a and b shows the confusion matrices with precise recognition of 2 classes below 70%TRAPHA and 30%TESPHA. Figure 7c depicts the PR values, demonstrating maximum outcomes all over classes. Figure 7d demonstrates the ROC analysis, showing proficient outcomes with a high value of ROC for two classes.

Table 2; Fig. 8 show the DDoS attack recognition of the MOBCF-ADDLM approach on the BoT-IoT Binary dataset. The results imply that the MOBCF-ADDLM approach correctly identified the samples. With 70%TRAPHA, the MOBCF-ADDLM approach presents average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC of 98.54%, 98.25%, 97.47%, 97.85%, and 95.72%, respectively. In addition, with 30%TESPHA, the MOBCF-ADDLM methodology presents average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC of 97.41%, 97.00%, 96.24%, 96.61%, and 93.24%, correspondingly.

Figure 9 demonstrates the training (TRAN) $accu_y$ and validation (VALN) $accu_y$ analysis of the MOBCF-ADDLM methodology on the BoT-IoT Binary dataset. The figure highlights that the TRAN and VALN $accu_y$ values exhibit an increasing trend, which indicates the capability of the MOBCF-ADDLM approach to have maximal outcomes over various iterations. Followed by the TRAN and VALN $accu_y$ remains closer over the epochs, which shows lesser overfitting and greater outcomes of the MOBCF-ADDLM approach, guaranteeing reliable prediction on hidden samples.

Figure 10 depicts the TRAN loss (TRANLOS) and VALN loss (VALNLOS) analysis of the MOBCF-ADDLM approach on the BoT-IoT Binary dataset. It is signified that the TRANLOS and VALNLOS values exemplify a

BoT-IoT Binary Dataset	
Classes	**No. of Samples**
"Attack"	1579
"Normal"	477
Total	**2056**

Table 1. Details of BoT-IoT binary Dataset.

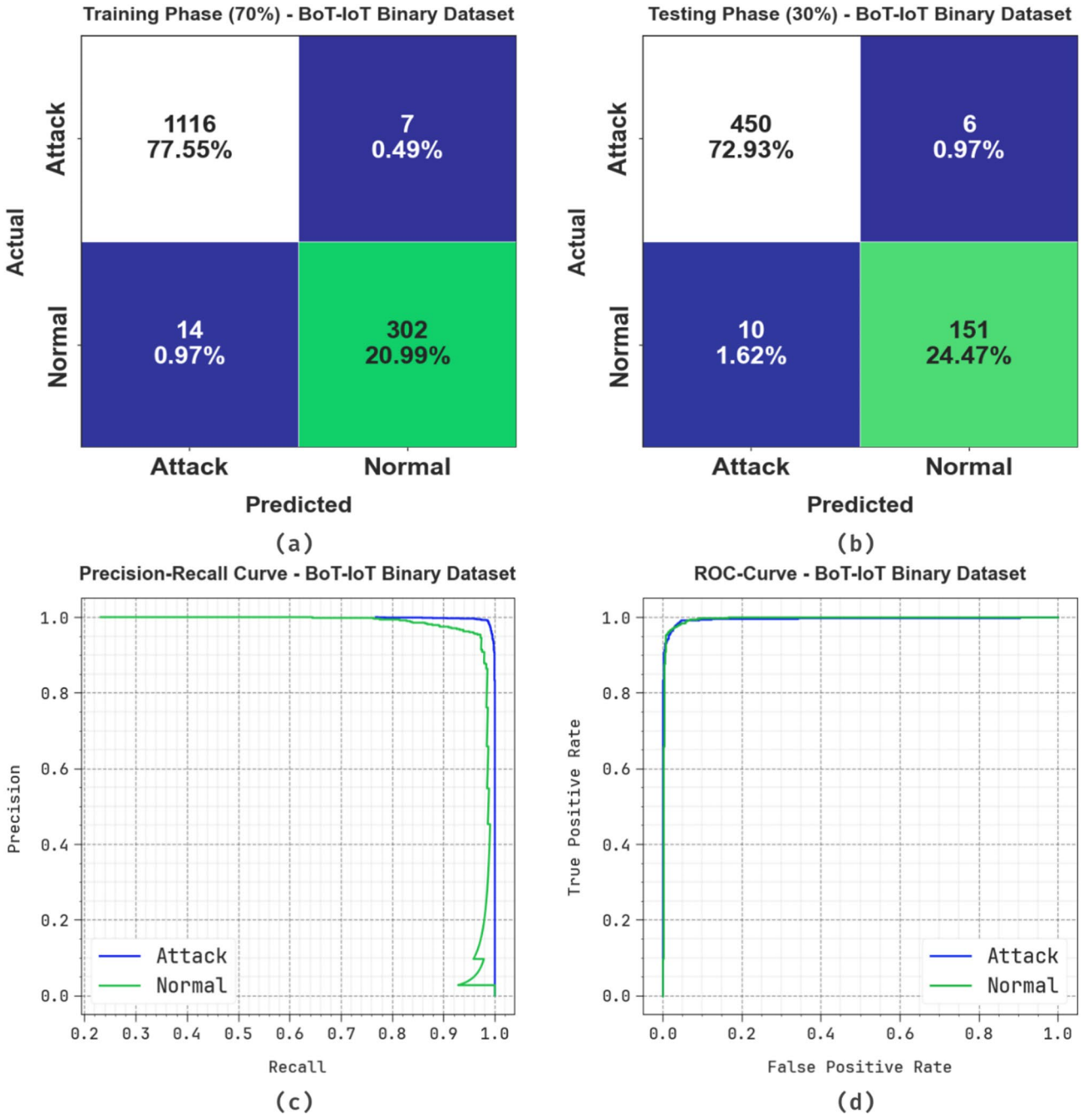

Fig. 7. BoT-IoT Binary dataset (**a**, **b**) confusion matrix and (**c**, **d**) PR and ROC curves.

reducing tendency, informing the ability of the MOBCF-ADDLM methodology to balance a trade-off between data fitting and simplification.

The BoT-IoT Multiclass dataset holds 2056 instances under five classes, such as DDoS, DoS, Recon, Theft, and Normal, as depicted in Table 3.

Figure 11 presents the classifier outcomes of the MOBCF-ADDLM on the BoT-IoT Multiclass dataset. Figure 11a and b displays the confusion matrices with perfect recognition of 5 classes below 70%TRAPHA and 30%TESPHA. Figure 11c shows the PR values, indicating optimal performance over all classes. Simultaneously, Fig. 11d illustrates the ROC values, signifying proficient outcomes with high ROC analysis for five classes.

Table 4; Fig. 12 represent the DDoS attack detection of MOBCF-ADDLM methodology on the BoT-IoT Multiclass dataset. The outcomes imply that the MOBCF-ADDLM methodology correctly recognized the samples. With 70%TRAPHA, the MOBCF-ADDLM technique presents average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC of 98.97%, 97.24%, 96.15%, 96.67%, and 96.02%, respectively. Besides, with 30%TESPHA, the MOBCF-ADDLM technique presents average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC of 99.22%, 98.38%, 95.83%, 96.98%, and 96.55%, correspondingly.

Classes	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	MCC
TRAPHA (70%)					
Attack	98.54	98.76	99.38	99.07	95.72
Normal	98.54	97.73	95.57	96.64	95.72
Average	**98.54**	**98.25**	**97.47**	**97.85**	**95.72**
TESPHA (30%)					
Attack	97.41	97.83	98.68	98.25	93.24
Normal	97.41	96.18	93.79	94.97	93.24
Average	**97.41**	**97.00**	**96.24**	**96.61**	**93.24**

Table 2. DDoS attack detection of MOBCF-ADDLM model on BoT-IoT binary dataset.

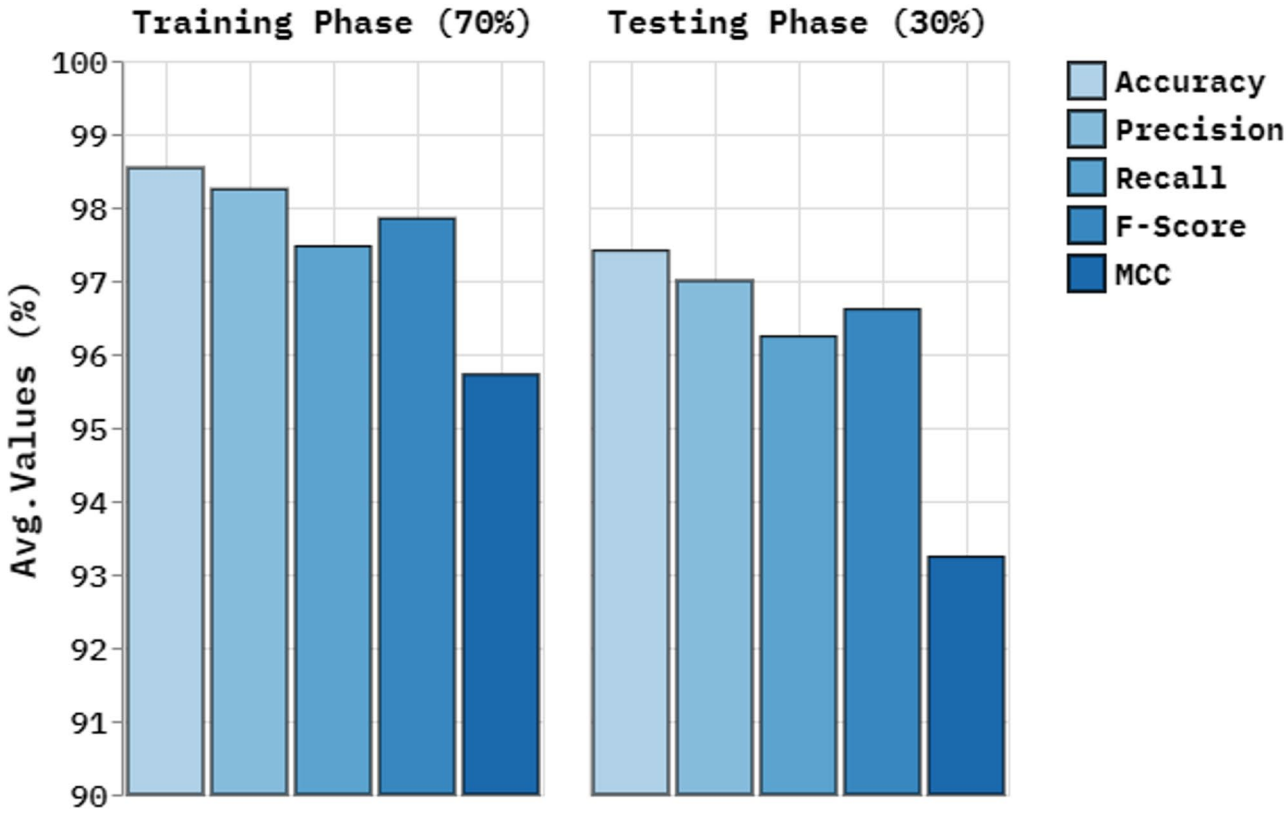

Fig. 8. Average of MOBCF-ADDLM model on BoT-IoT Binary dataset.

Figure 13 demonstrates the TRAN $accu_y$ and VALN $accu_y$ analysis of the MOBCF-ADDLM technique on the BoT-IoT Multiclass dataset. The outcome highlights that the TRAN and VALN $accu_y$ values display a rising trend, which indicates the capability of the MOBCF-ADDLM approach to have maximum outcomes across various iterations. The TRAN and VALN remain closer across the epochs, which identifies lesser overfitting and shows the optimal result of the MOBCF-ADDLM approach, promising continuous prediction on unseen instances.

Figure 14 shows the TRANLOS and VALNLOS analysis of the MOBCF-ADDLM model on the BoT-IoT Multiclass dataset. It is represented that the TRANLOS and VALNLOS analysis demonstrate a diminishing trend, informing the proficiency of the MOBCF-ADDLM technique in balancing a trade-off between simplification and data fitting.

Table 5; Fig. 15 inspect the comparison results of the MOBCF-ADDLM method with existing approaches[19,36,37]. The results emphasized that the LR, XGBoost, HGBClassifier, H3SC-DLIDS, AE-MLP, XGBoost, RF, DT, Bi-LSTM, and hybrid IDS approaches exhibited lesser performance. The proposed MOBCF-ADDLM technique exhibited superior performance with maximum $accu_y$, $prec_n$, $reca_l$ and F_{score} of 99.22%, 98.38%, 95.83%, and 96.98%, respectively.

Table 6; Fig. 16 show the processing time (PT) results of the MOBCF-ADDLM methodology with existing models. Based on PT, the MOBCF-ADDLM technique provides the worst PT of 9.31 s, while the LR, XGBoost,

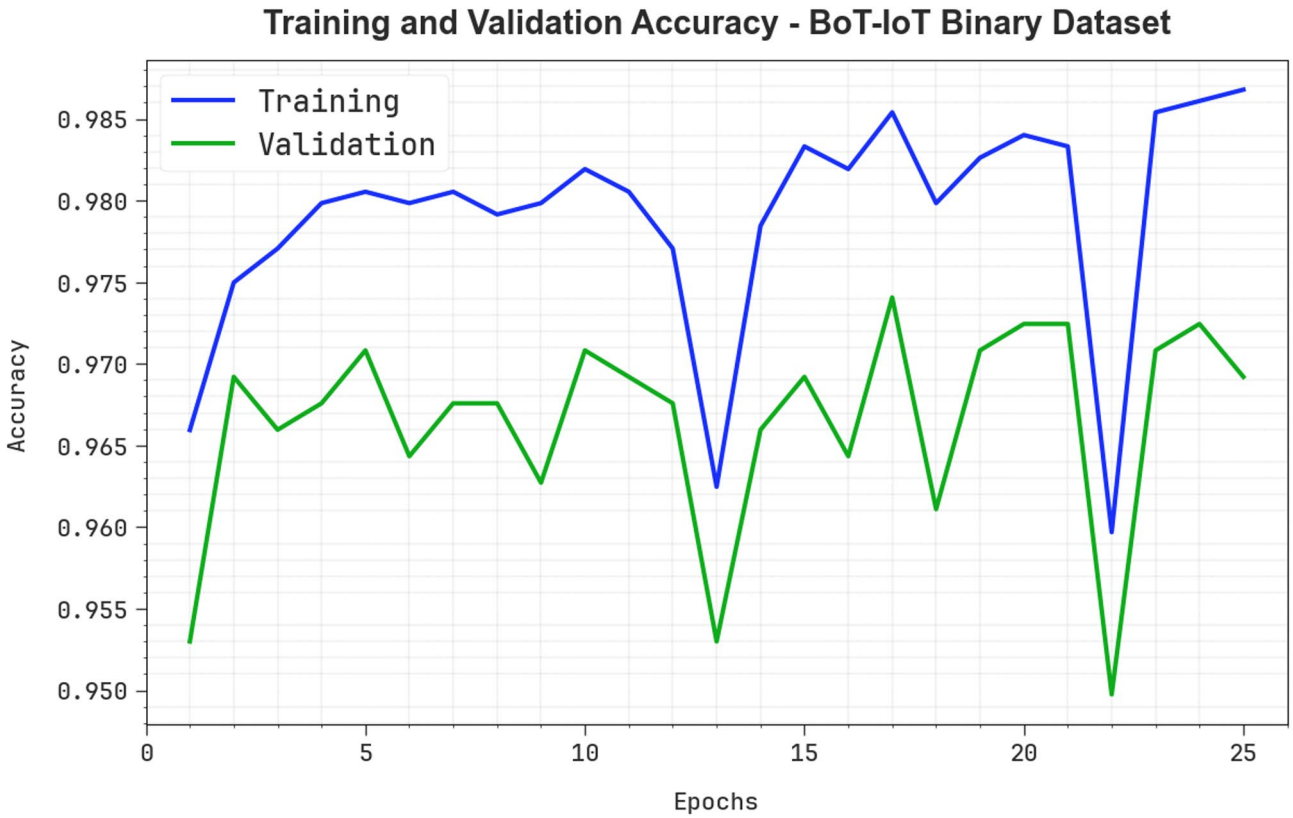

Fig. 9. $Accu_y$ curve of MOBCF-ADDLM model on BoT-IoT Binary dataset

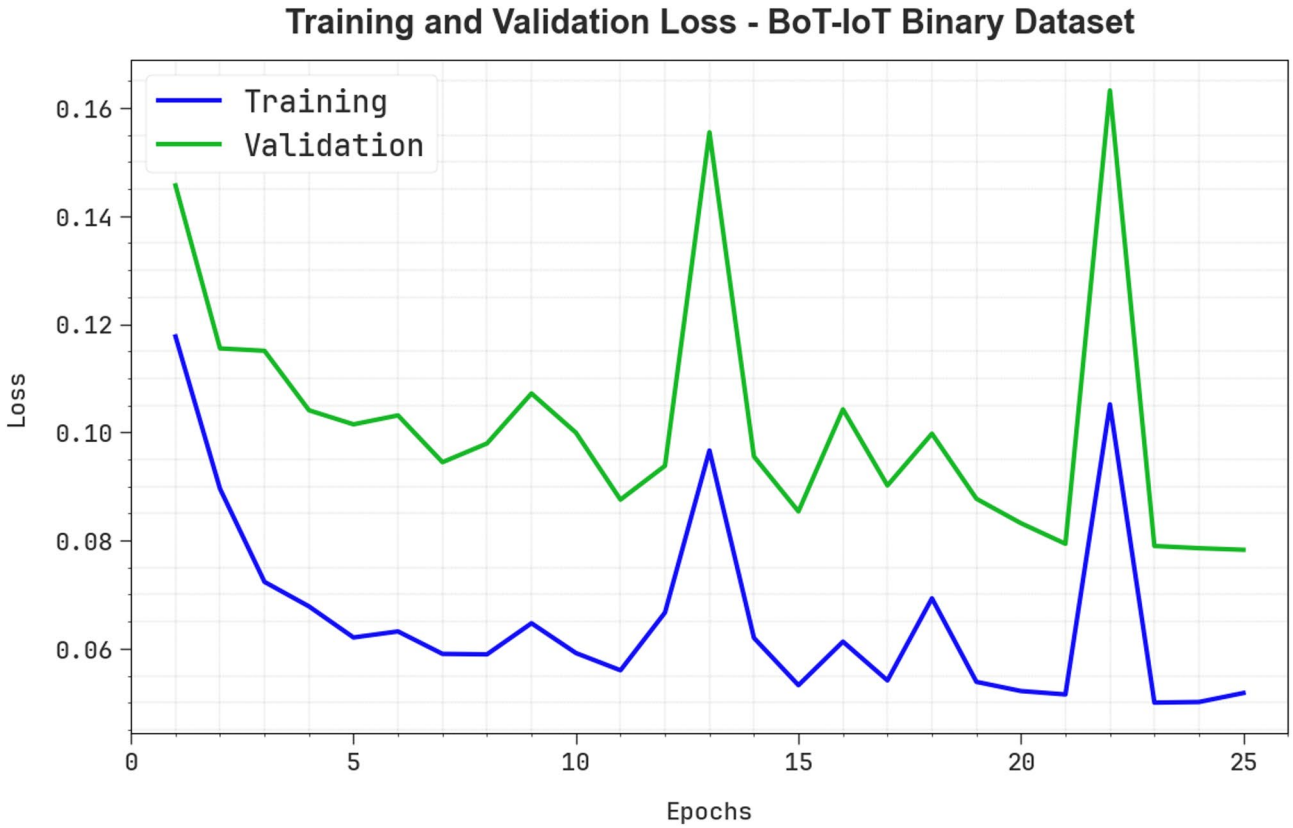

Fig. 10. Loss graph of MOBCF-ADDLM technique on BoT-IoT Binary dataset.

BoT-IoT Multiclass Dataset	
Classes	**No. of Samples**
"DDoS"	500
"DoS"	500
"Recon"	500
"Theft"	79
"Normal"	477
Total Samples	**2056**

Table 3. Details of BoT-IoT multiclass dataset.

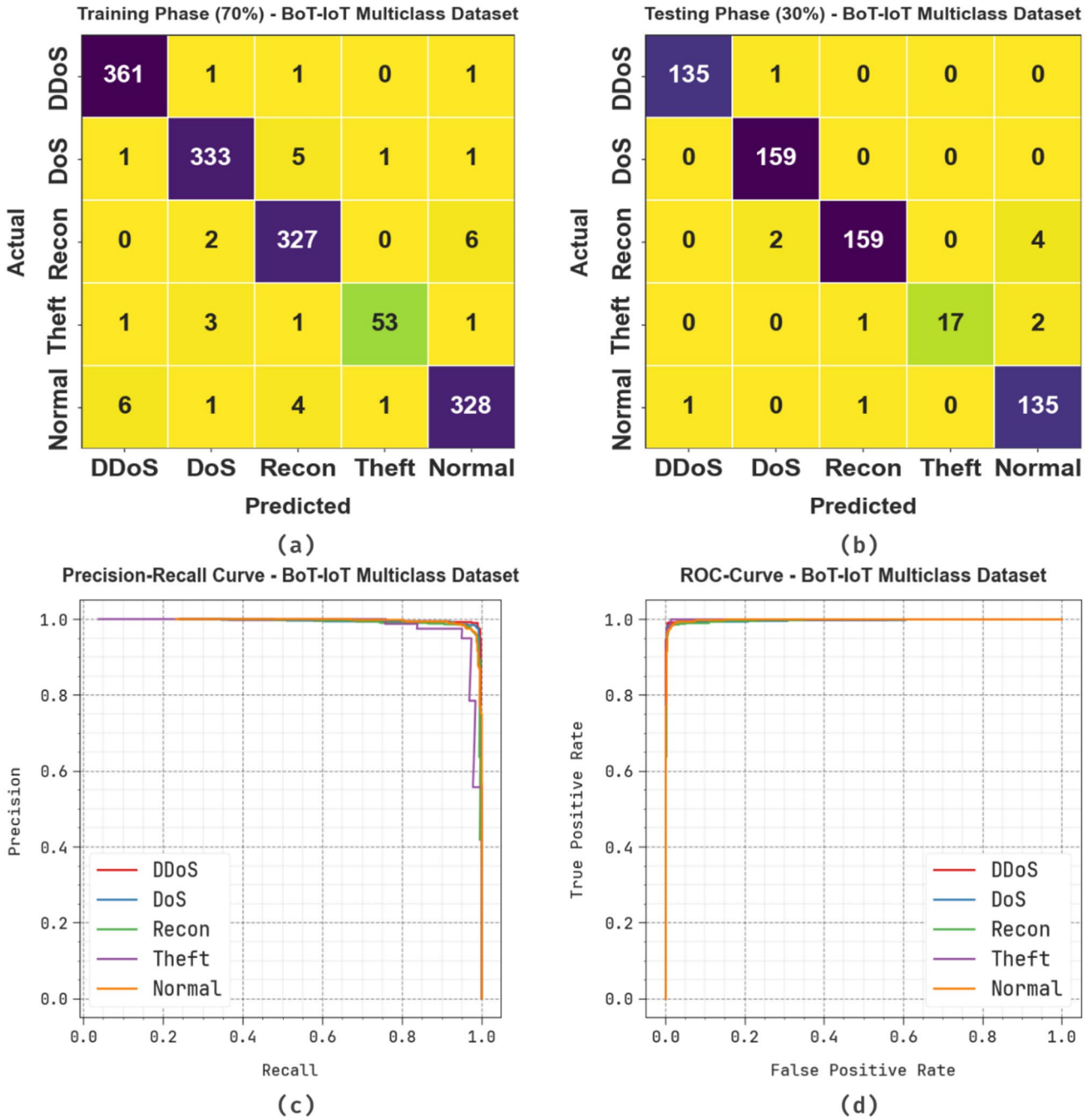

Fig. 11. BoT-IoT Multiclass dataset (**a**, **b**) confusion matrices and (**c**, **d**) PR and ROC curve.

Classes	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	MCC
TRAPHA (70%)					
DDoS	99.24	97.83	99.18	98.50	97.99
DoS	98.96	97.94	97.65	97.80	97.11
Recon	98.68	96.75	97.61	97.18	96.32
Theft	99.44	96.36	89.83	92.98	92.76
Normal	98.54	97.33	96.47	96.90	95.95
Average	**98.97**	**97.24**	**96.15**	**96.67**	**96.02**
TESPHA (30%)					
DDoS	99.68	99.26	99.26	99.26	99.06
DoS	99.51	98.15	100.00	99.07	98.74
Recon	98.70	98.76	96.36	97.55	96.68
Theft	99.51	100.00	85.00	91.89	91.96
Normal	98.70	95.74	98.54	97.12	96.30
Average	**99.22**	**98.38**	**95.83**	**96.98**	**96.55**

Table 4. DDoS attack detection of MOBCF-ADDLM model on BoT-IoT multiclass dataset.

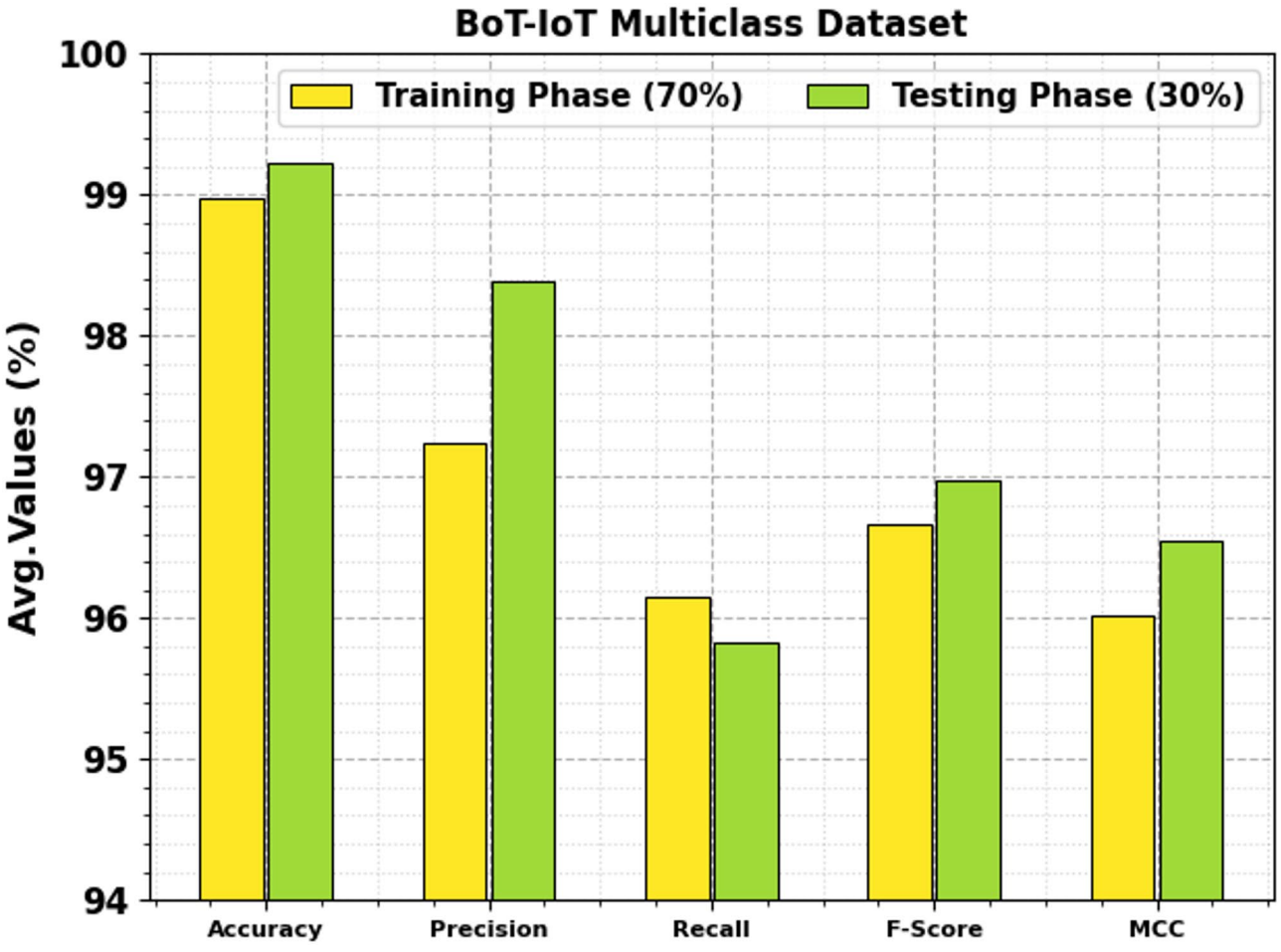

Fig. 12. Average of MOBCF-ADDLM model on BoT-IoT Multiclass dataset.

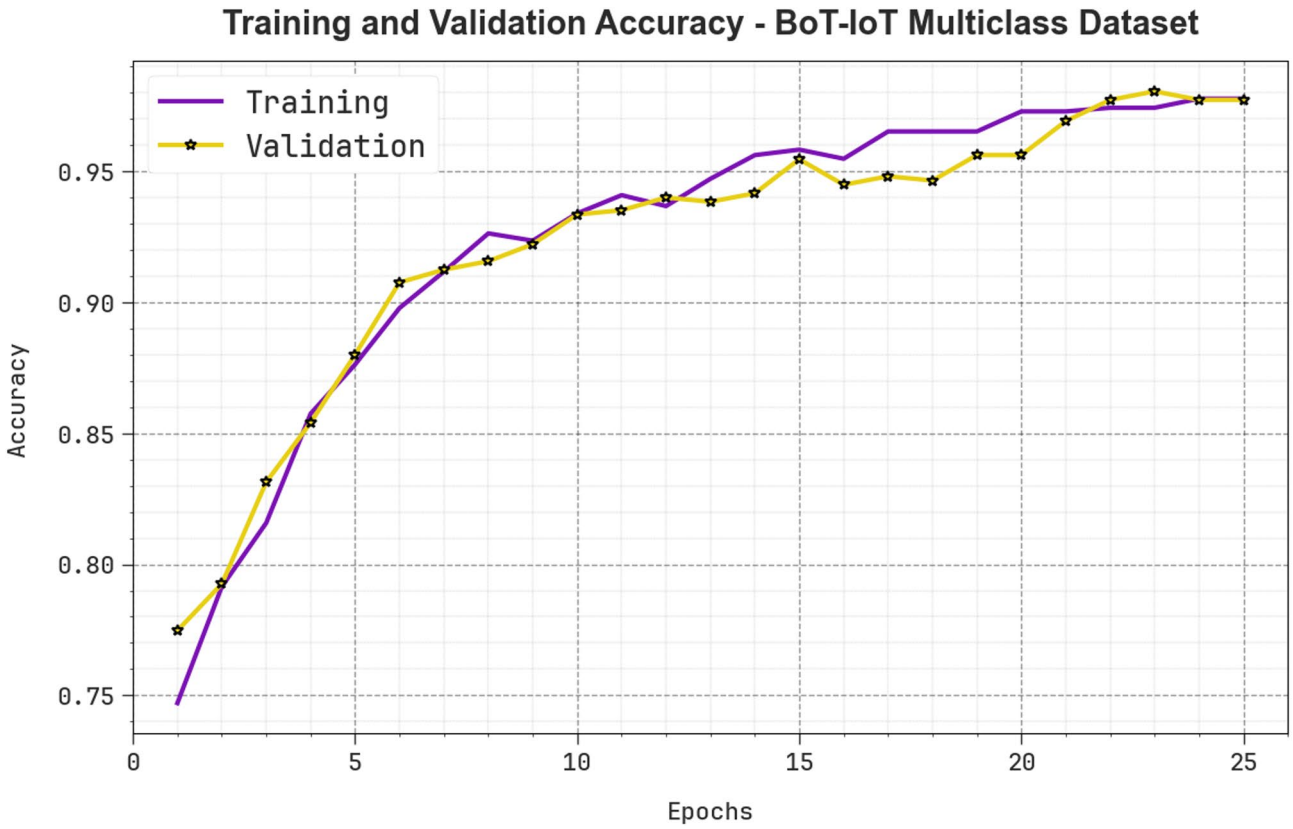

Fig. 13. $Accu_y$ curve of MOBCF-ADDLM model on BoT-IoT Multiclass dataset

HGBClassifier, H3SC-DLIDS, AE-MLP, XGBoost, RF, DT, Bi-LSTM, and hybrid IDS models attain greater PT values of 16.11 s, 12.98 s, 14.00 s, 14.68 s, 22.60 s, 15.28 s, 12.32 s, 13.96 s, 22.59 s, and 15.36 s, respectively.

Conclusion

In this novel, the MOBCF-ADDLM methodology is proposed. The main intention of MOBCF-ADDLM methodology is to deliver an effectual technique for recognizing DDoS threats in IoT environments using advanced techniques. First, the BC technology is applied to mitigate DDoS attacks by presenting decentralized security solutions. Furthermore, the data preprocessing stage employs a min-max scaling method for converting input data into a beneficial format. Moreover, the FS process uses the AO technique to recognize the most relevant features from input data. The DBN technique is employed for the attack classification process. Finally, the RPO model modifies the hyper-parameter values of the DBN model optimally, resulting in higher classification performance. A wide range of experiments with the MOBCF-ADDLM approach is performed under the BoT-IoT Binary and Multiclass datasets. The performance validation of the MOBCF-ADDLM approach portrayed a superior accuracy value of 99.22% over existing models. The limitations of the MOBCF-ADDLM approach comprise its evaluation on a single dataset, which may not fully capture the diversity of real-world DDoS attack patterns across diverse SDN environments. The model's performance may vary with growing attack strategies and unseen traffic behaviours. Furthermore, the study primarily concentrates on detection accuracy without extensively analyzing detection latency or scalability in large-scale deployments. Resource consumption during training and real-time inference also requires additional investigation. Future work may explore cross-dataset validation, adaptive learning mechanisms, integration with real-time SDN controllers, and improving computational overhead while maintaining detection precision.

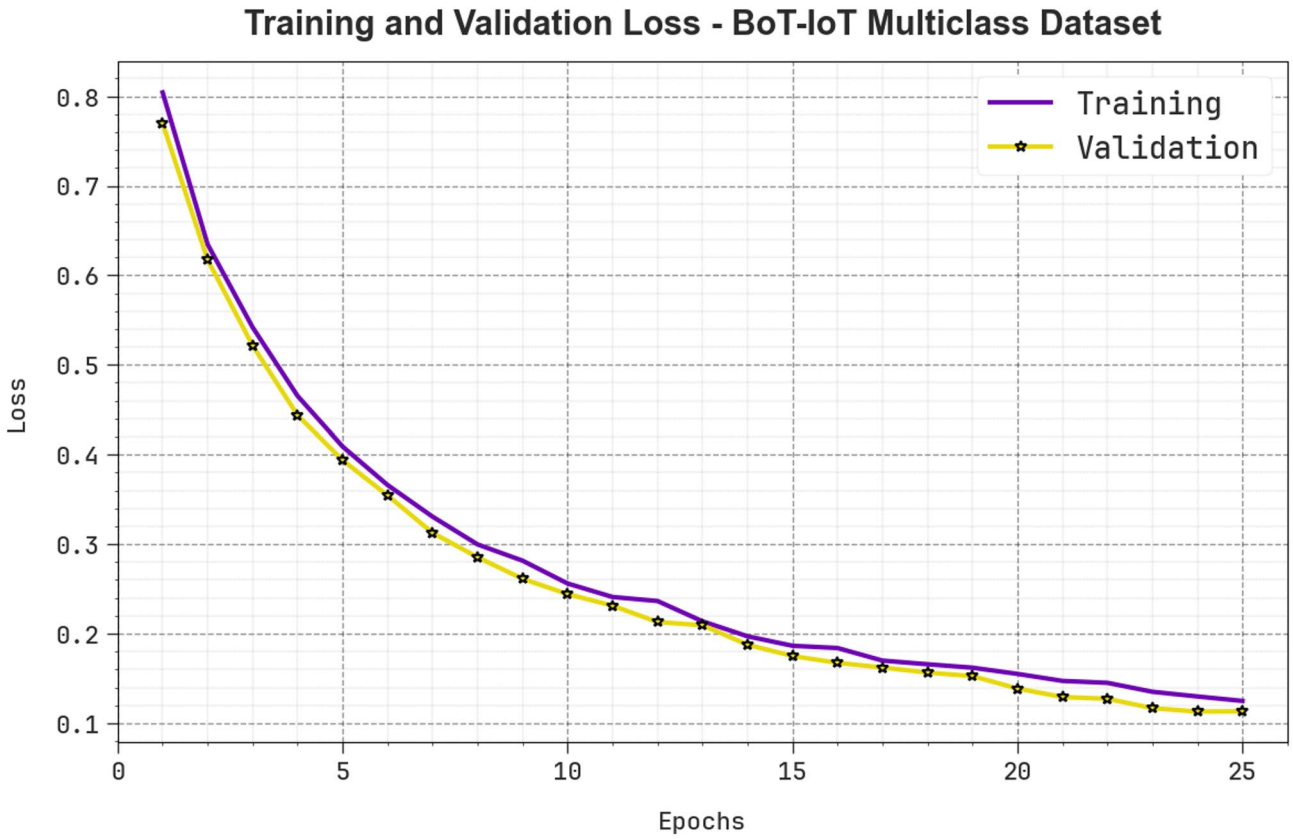

Fig. 14. Loss graph of MOBCF-ADDLM model on BoT-IoT Multiclass dataset.

Technique	$Accu_y$	$Prec_n$	$Recal_l$	F_{score}
MOBCF-ADDLM	99.22	98.38	95.83	96.98
LR	98.72	96.67	93.98	95.71
XGBoost	97.87	94.96	92.90	95.83
HGBClassifier	97.76	95.66	94.36	95.34
H3SC-DLIDS	99.07	96.68	95.20	96.06
AE-MLP Method	98.21	95.93	93.34	95.15
XGBoost Method	97.12	94.31	92.15	95.08
RF	97.02	95.00	93.72	94.59
DT	95.24	92.46	92.54	93.29
Bi-LSTM	97.43	95.83	94.93	95.55
Hybrid IDS	96.92	94.80	90.26	92.89

Table 5. Comparative outcomes of MOBCF-ADDLM technique with existing approaches[19,36]– [37].

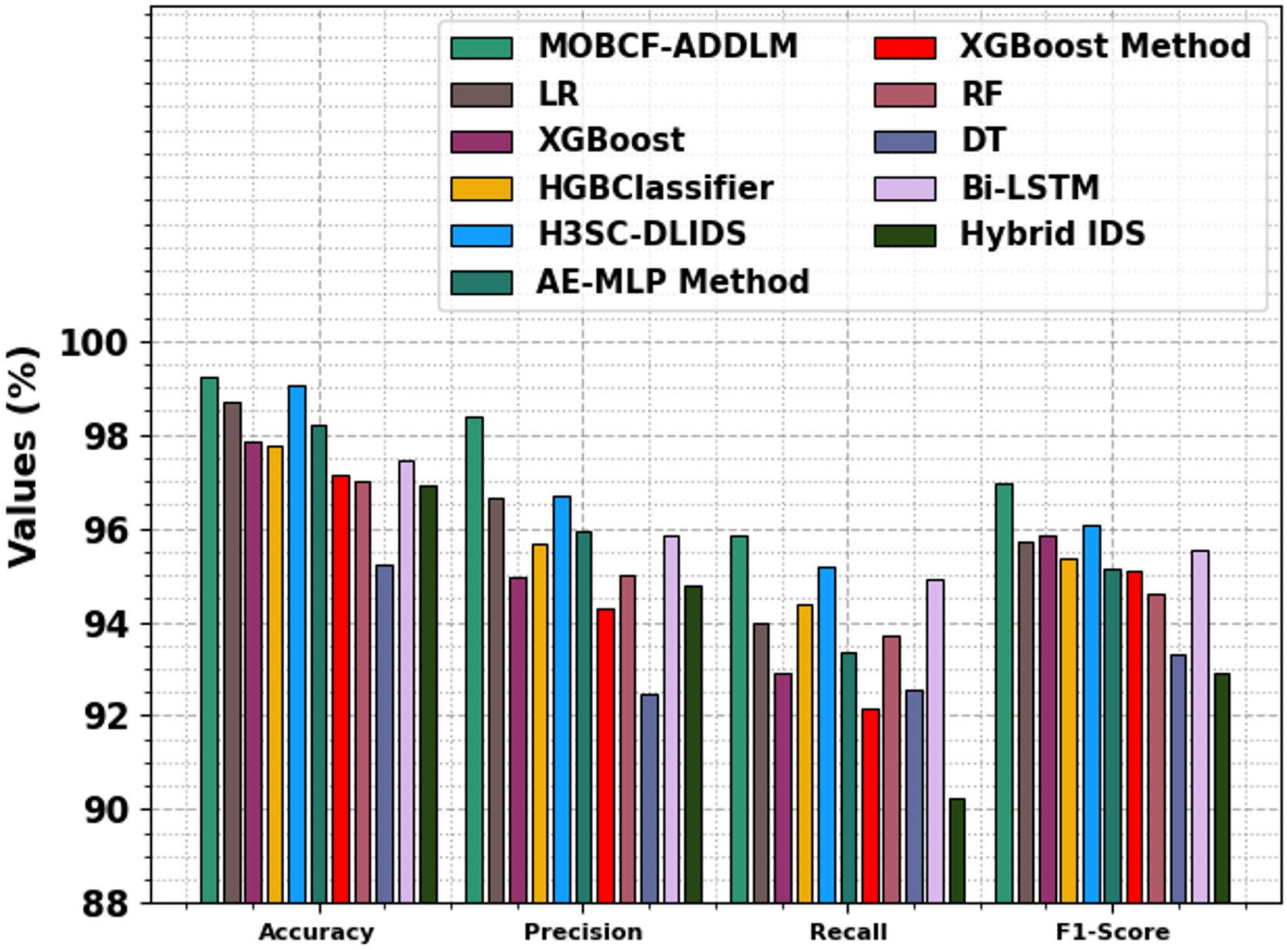

Fig. 15. Comparative results of MOBCF-ADDLM technique with existing approaches.

Methods	PT (sec)
MOBCF-ADDLM	9.31
LR	16.11
XGBoost	12.98
HGBClassifier	14.00
H3SC-DLIDS	14.68
AE-MLP Method	22.60
XGBoost Method	15.28
RF	12.32
DT	13.96
Bi-LSTM	22.59
Hybrid IDS	15.36

Table 6. PT result of MOBCF-ADDLM methodology with existing models.

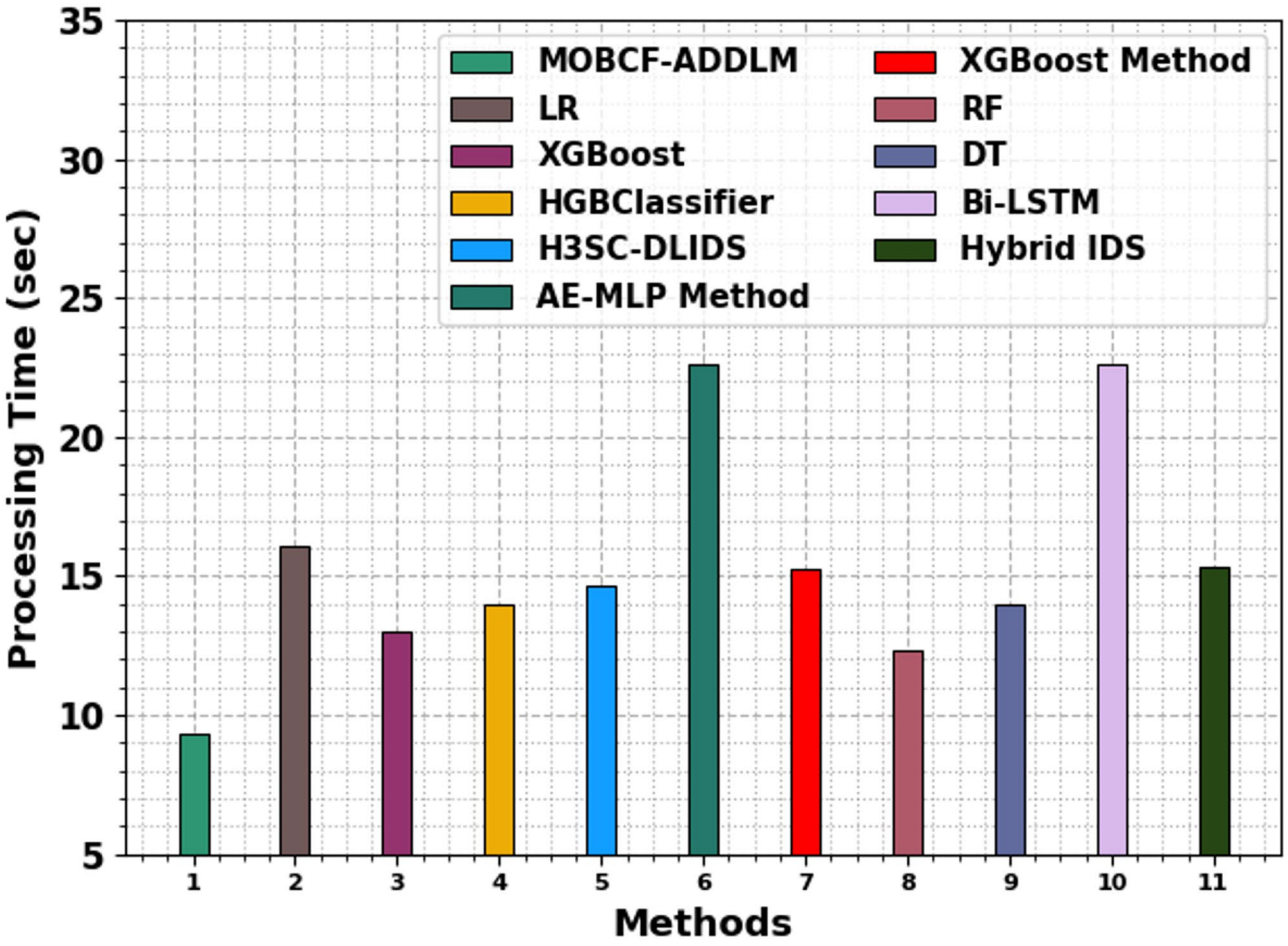

Fig. 16. PT outcome of MOBCF-ADDLM methodology with existing models.

Data availability

The data that support the findings of this study are openly available at https://research.unsw.edu.au/projects/bot-iot-dataset, reference number [35].

Received: 1 April 2025; Accepted: 9 June 2025
Published online: 02 July 2025

References

1. Ghadi, Y. Y. et al. Integration of federated learning with IoT for smart cities applications, challenges, and solutions. *PeerJ Comput. Sci.* **9**, e1657 (2023).
2. Hamouda, D., Ferrag, M. A., Benhamida, N. & Seridi, H. P. P. S. S. A privacy-preserving secure framework using blockchain-enabled federated deep learning for industrial IoTs. *Pervasive Mob. Comput.* **88**, 101738 (2022).
3. Al-Shammari, N. K., Syed, T. H. & Syed, M. B. An Edge–IoT framework and prototype based on blockchain for smart healthcare applications. *Eng. Technol. Appl. Sci. Res.* **11**, 7326–7331 (2021).
4. Shah, H. et al. Deep learning-based malicious smart contract and intrusion detection system for IoT environment. *Mathematics* **11**, 418 (2023).
5. Ghadi, Y. Y. et al. The role of blockchain to secure internet of medical things. *Sci. Rep.* **14** (1), 18422. (2024).
6. Kumar, R. et al. A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. *J. Parallel Distrib. Comput.* **164**, 55–68 (2022).
7. Ilyas, B., Kumar, A., Setitra, M. A., Bensalem, Z. A. & Lei, H. Prevention of DDoS attacks using an optimized deep learning approach in blockchain technology. *Trans. Emerg. Telecommun. Technol.* **34** (4), e4729 (2023).
8. Mohd, M. et al. Machine learning and deep learning approaches for detecting DDOS attacks in cloud environments. *Fusion Pract. Appl.* **17** (2), 79–97. https://doi.org/10.54216/fpa.170207 (2024).
9. Mazhar, T. et al. Analysis of integration of IoMT with blockchain: Issues, challenges and solutions. *Discov. Internet Things.* **4** (1), 1–36 (2024).
10. Revathi, N., Duraivel, A. N. & Prabu, S. A hybrid genetic algorithm and neural Network-Based cyber security approach for enhanced detection of DDoS and malware attacks in wide area networks. *J. Cybersecur. Inform. Manag.*, **14**(2). (2024).
11. Vijay Anand, R., Alagiri, I., Jayalakshmi, P., Brahmam, M. G. & Abdullah, A. B. Lattice homomorphic assisted privacy preserving electronic health records data transmission in internet of medical things using blockchain. *Trans. Emerg. Telecommun. Technol.* **36** (2), e70070 (2025).
12. Ilakkiya, N. & Rajaram, A. A secured trusted routing using the structure of a novel directed acyclic graph-blockchain in mobile ad hoc network internet of things environment. *Multimedia Tools Appl.*, pp. 1–26. (2024).

13. Indrason, N., Khongbuh, W., Baital, K. & Saha, G. MBCSD-IoT: A multi-level blockchain-assisted SDN-based IoT architecture for secured E-Voting system. *IEEE Trans. Netw. Sci. Eng.* (2025).
14. Sharma, K. V., Sarada, C., Vasavi, M. & Ambika, K. Securing IoT devices from DDoS attacks through blockchain and multi-code trust framework. In *E3S Web of Conferences* (Vol. 472, p. 03001). EDP Sciences. (2024).
15. Kiran, G. M. & Nalini, N. SparkGrid: Blockchain assisted secure query scheduling and dynamic risk assessment for live migration of services in Apache Spark-Based grid environment. *IET Blockchain*. **5** (1), e70004 (2025).
16. Park, J. H., Yotxay, S., Singh, S. K. & Park, J. H. PoAh-enabled federated learning architecture for DDoS attack detection in IoT networks. *Human-Centric Comput. Inf. Sci.*, **14**. (2024).
17. Halim, E., Reda, M., Maher, Y. & Younan, M. Chain of things (CoT): A Blockchain-based framework for Securing internet of things applications. *J. Comput. Commun.*. **4** (1), 1–18 (2025).
18. Ohri, P., Daniel, A., Neogi, S. G. & Muttoo, S. K. Blockchain-based security framework for mitigating network attacks in multi-SDN controller environment. *Int. J. Inform. Technol.*, pp. 1–13. (2024).
19. Kachavimath, A. V. & Narayan, D. G. An efficient DDoS attack detection in SDN using multi-feature selection and ensemble learning. *Procedia Comput. Sci.* **252**, 241–250 (2025).
20. Sumathi, S. & Rajesh, R. HybGBS: A hybrid neural network and grey Wolf optimizer for intrusion detection in a cloud computing environment. *Concurr. Comput. Pract. Exp.*. **36** (24), e8264 (2024).
21. Abdullah, M. et al. Federated learning with Blockchain on Denial-of-Service attacks detection and classification of edge IIoT networks using Deep Transfer Learning model. *Comput. Electr. Eng.* **124**, p. 110319. (2025).
22. Sokkalingam, S. & Ramakrishnan, R. An intelligent intrusion detection system for distributed denial of service attacks: A support vector machine with hybrid optimization algorithm based approach. *Concurr. Comput. Pract. Exp*. **34** (27), e7334 (2022).
23. Saraswathi, V. & Dayana, R. Enhancing security in next generation networks: A deep learning approach for intrusion detection. In *2025 4th International Conference on Sentiment Analysis and Deep Learning (ICSADL)* (pp. 870–877). IEEE. (2025).
24. Sumathi, S., Rajesh, R. & Lim, S. Recurrent and deep learning neural network models for DDoS attack detection. *J. Sensors*, **2022**(1), 8530312. (2022).
25. Wazid, M. et al. Explainable deep Learning-Enabled malware attack detection for IoT-Enabled intelligent transportation systems. *IEEE Trans. Intell. Transp. Syst.* (2025).
26. Sumathi, S. & Rajesh, R. A dynamic BPN-MLP neural network DDoS detection model using hybrid swarm intelligent framework. *Indian J. Sci. Technol.* **16** (43), 3890–3904 (2023).
27. Almseidin, M. et al. Hybrid deep neural network optimization with particle swarm and grey wolf algorithms for sunburst attack detection. *Computers*, **14**(3), 107. (2025).
28. Sumathi, S. & Rajesh, R. Comparative study on TCP SYN flood DDoS attack detection: A machine learning algorithm based approach. *WSEAS Trans. Syst. Control*. **16**, 584–591 (2021).
29. Mehmood, S. et al. Distributed denial of services (DDoS) attack detection in SDN using Optimizer-equipped CNN-MLP. *PLoS ONE*. **20** (1), e0312425 (2025).
30. Alkhammash, M. A metaheuristic approach to detecting and mitigating DDoS attacks in Blockchain-Integrated deep learning models for IoT applications. *IEEE Access* (2024).
31. Dutta, S. S., Sandeep, S., Nandhini, D. & Amutha, S. Hybrid quantum neural networks: Harnessing dressed quantum circuits for enhanced tsunami prediction via earthquake data fusion. *EPJ Quantum Technol.* **12**(1), p.4. (2025).
32. Zhang, H. Battery state of charge Estimation based on improved neural network. *J. Comput. Sci. Artif. Intell.* **2** (1), 36–44 (2025).
33. Zhou, L. et al. Fault diagnosis and data reconstruction of temperature sensors for wind turbine stator winding. *Shock Vibr.* **2025**(1), 4713545. (2025).
34. Elalfy, D. A., Gouda, E., Kotb, M. F., Bureš, V. & Sedhom, B. E. Frequency and voltage regulation enhancement for microgrids with electric vehicles based on red panda optimizer. *Energy Convers. Manag. X*, **25**, 100872. (2025).
35. https://research.unsw.edu.au/projects/bot-iot-dataset
36. Katib, I. & Ragab, M. Blockchain-assisted hybrid harris hawks optimization based deep DDoS attack detection in the IoT environment. *Mathematics*, **11**(8), 1887. (2023).
37. Zeeshan, M. et al. Protocol-based deep intrusion detection for Dos and Ddos attacks using unsw-nb15 and bot-iot datasets. *IEEE Access.* **10**, 2269–2283 (2021).

Acknowledgements

This work is partly supported by Ajman University, United Arab Emirates.

Author contributions

Conceptualization: VVSH PrasadData curation and Formal analysis: Swathi Sowmya Bavirthi, C.S.S. AnupamaInvestigation and Methodology: VVSH PrasadFunding Support, Khalid Ammar, Mohamad Khairi IshakProject administration and Resources: Supervision; E. Laxmi Lydia, Khalid AmmarValidation and Visualization: K. Sathesh Kumar, Mohamad Khairi IshakWriting—original draft, VVSH PrasadWriting—review and editing, Khalid Ammar, Mohamad Khairi IshakAll authors have read and agreed to the published version of the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Ethics approval

This article contains no studies with human participants performed by any authors.

Additional information

Correspondence and requests for materials should be addressed to K.A.