Enhancing Cybersecurity Attack Detection Using Multiplayer Battle Game Optimizer with Hybridization of Deep Learning Models

K. Anitha[1,*], K. Rajiv Gandhi[2]

[1]Research Scholar, Department of Computer Applications, Alagappa University, Karaikudi, 630003, Tamilnadu, India
[2]Assistant Professor, Department of Computer Science , Government Arts and Science College For Women, Paramakudi, 623707, Tamilnadu, India

Emails: anitha.kalimuthu@gmail.com; dr.krajiv84@gmail.com

Abstract

Cybersecurity is advancing and the rate of cybercrime, which is always rising. Advanced attacks are measured as the novel normal as they are one of the more normal and extensive. Cybersecurity threats have risen promptly in many areas like healthcare, smart homes, energy, automation, agriculture, and industrial processes. An intrusion detection system (IDS) discovers intrusions by analyzing attack designs or mining signatures from system packets. To assess an IDS model, use Machine Learning (ML) and deep learning (DL) approaches for recognizing data traffic into malicious and healthy. ML and DL techniques has earned an extensive interest on countless applications and domains of study, mostly in Cybersecurity. With computing power and hardware becoming more available, ML and DL systems can be employed in order to classify and analyze corrupt actors from a massive group of accessible data. This manuscript presents an Enhancing Detection of Cybersecurity Attack Using Multiplayer Battle Game Optimizer with Hybrid Deep Learning (EDCA-MBGOHDL) technique. The main intention of the EDCA-MBGOHDL technique is to provide a robust framework for cyberattack detection using deep learning integrated with a hyperparameter tuning approach. At first, the feature selection process is implemented by applying improved Harris hawk optimization (IHHO) algorithm for ensuring that only the most relevant features are fed into the model. Furthermore, the hybrid of convolutional neural network, bidirectional long short-term memory and attention mechanism (CNN-BiLSTM-AM) model is employed for the classification of cybersecurity threats. Eventually, the multiplayer battle game optimizer (MBGO) algorithm adjusts the hyperparameter values of the CNN-BiLSTM-AM classifier optimally and outcomes in greater classification performance. The wide range of analysis of the EDCA-MBGOHDL technique takes place using a benchmark dataset. The outcomes pointed out the superior performance of the EDCA-MBGOHDL system across existing models

Keywords: Cybersecurity Attack Detection; Multiplayer Battle Game Optimizer; Hybrid Deep Learning Models; Feature Selection

1. Introduction

The large-scale development of the Internet of Things (IoT) in recent times has contributed to substantial growth in Industry 4.0, fog computing, and smart cities which execute the composite confidential data processing that can be protected against cyber security threats [1]. Cyber security threats have enlarged quickly in several fields, like healthcare, smart homes, agriculture, automation, industrial processes, and energy [2]. Thus, the broad variety of services and IoT sensor devices produce huge volumes of information that need security, privacy, and authentication. Formerly, classical frameworks and approaches were utilized to guarantee IoT security [3]. Cyber-threat events might be exposed but details of such incidents are generally not publicly available. Several reports

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

defined penetration testing as accompanied by private companies trying to connect from an external system to internal crucial cyber assets, for example, communication networks and programmable electronic devices [4].

Cyber security examines to techniques and technologies that defend the systems, computers, data, and programs, from being assaulted, accessed, or damaged by illegal individuals [5]. Cyber security covers several conditions corporate to mobile computing and might be separated into various fields like 1. Intruders from access to a computer or security system that aims to prevent cyber-attackers; 2. Application security that keeps devices and cyber-attacks or software free of hazards; 3. Information security mainly takes the privacy and security of related information; 4. Operational security examines to the techniques for safeguarding and handling effects of information [6]. Classical cyber security solutions contain anti-virus software, an Intrusion Detection System (IDS), or a firewall in computer security methods and networks [7]. Rule-based approaches, whereas fast and simple to execute, may not compensate for noisy or incomplete data and are complex to update [8]. To overwhelm these difficulties, statistics-based methods has been projected to allow the process of inaccurate data; nevertheless, such approaches entail a higher computational cost and have a limited capability to handle huge amounts of data [9]. The application of diverse Artificial Intelligence (AI) approaches for identifying cyber security threats has gained in popularity for many years. In recent times, Deep Learning (DL) and Machine learning (ML) -based methods have been progressively studied owing to their capability to utilize difficult interpretation methods that can be trained on huge numbers of information to identify difficult intrusion patterns [10]. These new methods can address several difficult scientific tasks in classical computing technologies and generate opportunities. With the futuristic technology development, cyber security infrastructure will become outdated soon.

This manuscript presents an Enhancing Detection of Cybersecurity Attack Using Multiplayer Battle Game Optimizer with Hybrid Deep Learning (EDCA-MBGOHDL) technique. The main intention of the EDCA-MBGOHDL technique is to provide a robust framework for cyberattack detection using deep learning integrated with a hyperparameter tuning approach. At first, the feature selection process is implemented by applying improved Harris hawk optimization (IHHO) algorithm for ensuring that only the most relevant features are fed into the model. Furthermore, the hybrid of convolutional neural network, bidirectional long short-term memory and attention mechanism (CNN-BiLSTM-AM) model is employed for the classification of cybersecurity threats. Eventually, the multiplayer battle game optimizer (MBGO) algorithm adjusts the hyperparameter values of the CNN-BiLSTM-AM classifier optimally and outcomes in greater classification performance. The wide range of analysis of the EDCA-MBGOHDL technique takes place using a benchmark dataset.

2. Related Works

Duraibi and Alashjaee [11] introduced an Improved Mayfly Optimizer Algorithm with a Hybrid DL-based ID (IMFOHDL-ID) approach in an IoT framework. The proposed model primarily monitors data normalization as a preprocessing phase. Additionally, the IMFOHDL-ID method creates usage of the IMFO-based FS technique for selecting feature subset. For IDs, the presented model employs the LSTM-based Deep Stacked Sequence-to-Sequence AE (LSTM-DSSAE) method. To end with, the Dipper-Throated Optimizer Algorithm (DTOA) was used for the optimum hyper-parameters choice of the LSTM-DSSAE approach. Awajan [12] implements a new DL-based IDS for IoT mechanisms. This intelligent approach presents a 4 layers-deep Fully Connected (FC) system framework to identify harmful traffic that can originate threats on connected IoT devices. The projected approach is improved as a communication protocol-independent method to minimize employment difficulties. Maddu and Rao [13] improved a practical DL method depending on the Elman Recurrent Neural Networks (ERNN) and Res2Net method. This structure contains many stages and started by tackling the database class inequality concern with a Data Augmentation Generative Adversarial Network (DAGAN). Then, the Enhanced Honey Badger Algorithm (EHBA) and Res2net are utilized for feature extraction and selection. To end with, an ERNN-based model is utilized to classify and detect the intrusions in SDN.

Gueye et al. [14] projected a structure that steadily exceeds the advanced approaches performed by intrusion Detection that contains the dual classification of whether an intrusion happened or not, and several categorizations which classifies the diverse kinds of threat utilizing an embedded layers in a Neural Network (NN) method to register values. The greatest precision outcomes were attained with a Convolutional Neural Network (CNN). Vaiyapuri et al. [15] aim to design an Improved Reptile Search Optimizer with Ensemble DL-based Cyber security (IRSO-EDLCS) model in the IIoT surroundings. The projected IRSO-EDLCS model achieves the IRSO model-based FS (IRSO-FS) model. Furthermore, the IRSO-EDLCS model accomplishes an ensemble of 3 DL methods AE, Bi-directional GRU (Bi-GRU), and Deep Belief Network (DBN). The hyper-parameters tuning process is accomplished by a Modified Gray Wolf Optimizer (MGWO) method.

Prabakar et al. [16] projected a new model in cyber security-based system traffic examination and malicious threat recognition utilizing IoT-based AI models for maintainable smart cities. A traffic examination is implemented by utilizing a kernel vector which improves the transmission of data by decreasing system traffic. It can improve

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

energy efficacy with minimized traffic. Next, harmful threat recognition is implemented by utilizing adversarial Bayesian belief systems. Alrayes et al. [17] implement a Distributed Multiclass Cyberattack Detection utilizing Golden Jackal Optimizer with DL (DMCD-GJODL) model for IoT systems. In the DMCD-GJODL approach, the min-max scalar is mainly exploited to scale the input data. The DMCD-GJODL approach employs a Chaotic Crow Search Optimizer Algorithm (CSSOA) based FS method to select features. Also, the BiGRU method is utilized to classify and detect cyber threats.

3. Materials and Methods

This manuscript presents an EDCA-MBGOHDL technique. The main intention of the EDCA-MBGOHDL technique is to provide a robust framework for cyberattack detection using deep learning integrated with a hyperparameter tuning approach. The proposed model involves IHHO-based feature selection, CNN-BiLSTM-AM-based classification model and MBGO-based hyperparameter tuning process. Fig. 1 depicts the working procedure of EDCA-MBGOHDL technique.

A. Feature Selection

At first, the feature selection process is implemented by applying IHHO algorithm for ensuring that only the most relevant features are fed into the model [18]. During this HHO method, all Harris Hawk signifies possible solutions, and the objective followed in all iterations characterizes the global optimum solution. The search phase of this population is mostly separated into 3 phases: the transition stage between development and search, the research, and the development stage.

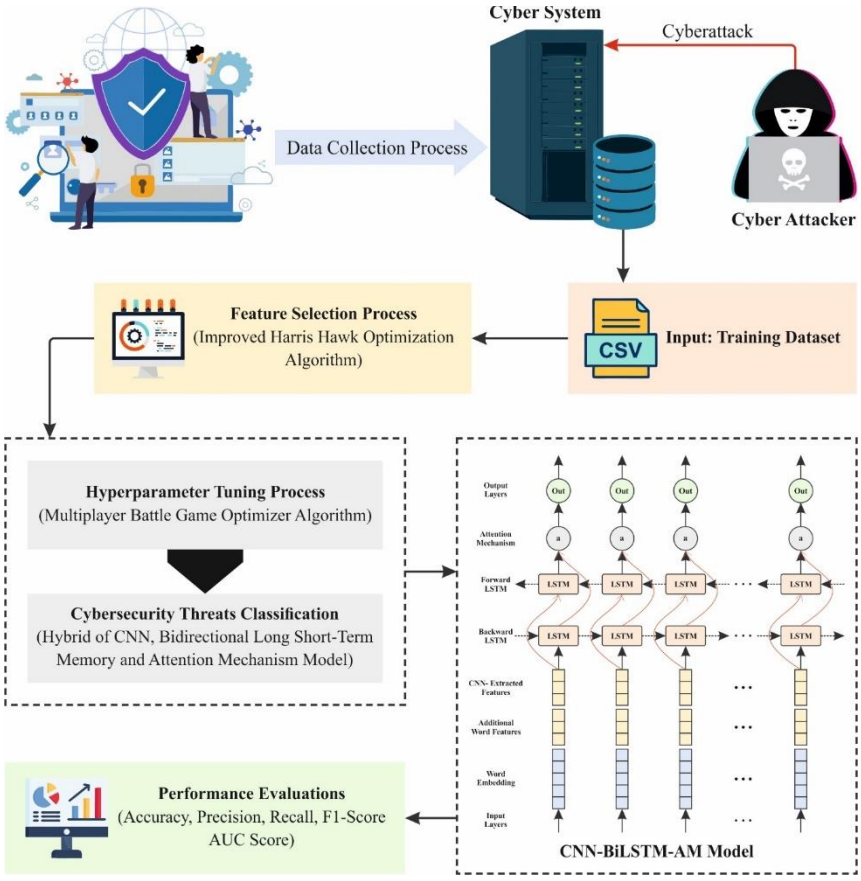

Figure 1. Overall working process of EDCA-MBGOHDL technique

During this first phase, the model implements a global search, using HHs arbitrarily distributed in the solution space, looking forward to chances inside the search range and looking for prey. In the search procedure, they would continually alter their locations depending on dual dissimilar tactics to manage the target. The mathematical representation is:

$$X(t+1) = \begin{cases} X_h(t) - r_1|X_h(t) - 2r_2X(t)| & q \geq 0.5 \\ \left(X_r(t) - X_m(t)\right) - r_3\left(LB + r_4(UB - LB)\right) & q < 0.5 \end{cases} \quad (1)$$

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

whereas t denotes present number of iterations, $X(t+1)$ represents location of $i\,th$ individual afterward the iteration t, X_h signifies individual locations selected at random after $t\,th$ iterations, r_1, r_2, r_3, r_4, and q symbolizes a constant among (0,1), $\cdot\, LB$ and UB characterize the lower and upper limits of the search area,$\cdot\, x_m(t)$ denotes the average location of population afterward $t\,th$ iterations. Its mathematical formulation is denoted below:

$$X_m(t) = \frac{1}{N}\sum_{i=1}^{N} X_i\,(t) \tag{2}$$

Here N refers to size of the population. The HHO method can adaptably be switched between development and exploration phases and can fine-tune its development tactic depending on the prey's residual energy. If $|E| \geq 1$, enter the search stage globally, If $|E| < 1$, arrive in the local development stage. The equations for escaping energy are:

$$E = 2E_0\left(1 - \frac{t}{T}\right) \tag{3}$$

Here E_0 characterizes the first energy condition of the prey, which represents arbitrary number of $[-1, 1]$. Since the iteration counts improve, the escape energy E declines. If the escaping energy $|E| < 1$, HHO arrives at the development stage and starts to execute a local search. According to the searching features of Harris eagles, 4 dissimilar tactics have been advanced to carry out local searches and relocate their locations, considering variables like the dimensions of E and whether the prey ran away or not. R denotes the possibility of escape. If $r < 0.5$, the prey safely evades,$\cdot\, r > 0.5$, Harris Eagle will perform a search. The escaping energy E has been applied to control the tactic. If $|E| \geq 0.5$, a soft encirclement approach can be accepted, If $|E| < 0.5$, accept a hard siege.

Soft Siege: If $r \geq 0.5$ and $|E| \geq 0.5$, the prey contains energy for escaping but finally fails, and the Harris Eagle accepts this approaches. Now, Harris Eagle will upgrade its location based on Eq. (4):

$$X(t+1) = \Delta X(t) - E|IX_r(t) - X(t)|\ \Delta X(t) = X_r(t) - X(t) \tag{4}$$

Here $X(t+1)$ characterizes the following updated location of HH, $\Delta X(t)$ signifies the distance among the present prey and HH, and I embodies the jumping strength of the prey, which is randomly generated numbers inside [0,2] that differs by iterations.

Hard Siege: When $r < 0.5$ and $|E| \geq 0.5$, the prey has no energy for escaping, and the HH isn't required to stay for a direct attack. Nowadays, utilize the subsequent equation to upgrade the present location:

$$X(t+1) = X_r(t) - E|\Delta X(t)| \tag{5}$$

Collected speed diving soft siege: If $r < 0.5$ and $|E| \geq 0.5$, the escaping energy is enough, and the prey safely evade. During the HHO model, the Levy flight unit is presented to mimic the jumping and escape behaviors of prey, equivalent to a novel location-updated tactic. By presenting it, the optimization capability and convergence speed of the methods are improved. In Levy's flight, the HH would follow the location updated tactic to guarantee that the searching range and direction encounter real requirements:

$$X(t+1) = \begin{cases} Y\ if\ F(Y) < F\big(X(t)\big) \\ Z\ if\ F(Z) < F\big(X(t)\big) \end{cases} \tag{6}$$

$$Y = X_r(t) - E|IX_r(t) - X(t)| \tag{7}$$

$$Z = Y + S \times LP(D) \tag{8}$$

Whereas D denotes dimension of the present problem, S refers to $1 \times D$ arbitrary row vector, and LF represents function of Levy flight that can be computed with Eq. (9):

$$LF(x) = 0.01 \times \frac{u \times \sigma}{\frac{1}{|v|^{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \tag{9}$$

Collected speed diving hard siege: If $r < 0.5$ and $|E| < 0.5$, when the prey can't escape. Upgrade the position based on the succeeding approach:

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

$$X(t+1) = \begin{cases} Y \; if \; P(Y) < P\big(X(t)\big) \\ Z \; if \; P(Z) < P\big(X(t)\big) \end{cases} \quad (10)$$

$$Y = X_r(t) - E|IX_r(t) - X_m(t)| \quad (11)$$

$$Z = Y + S \times LF(D) \quad (12)$$

The fitness function (FF) employed in the IHHO system is constructed to have a balance among the amount of designated features in every solution (minimum) and the classification accuracy (maximum) acquired by employing these nominated features, Eq. (13) represents the FF for estimating the solution.

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|C|} \quad (13)$$

Here, $\gamma_R(D)$ signifies the classifier rate of error of an assumed classifier. $|R|$ denotes to the cardinality of the select subset and $|C|$ denotes the total sum of features in the database, α and β are dual parameters relating to the status of classifier quality and sub-set length. $\in$ [1,0] and $\beta = 1 - \alpha$. Fig. 2 represents the flowchart of IHHO algorithm.

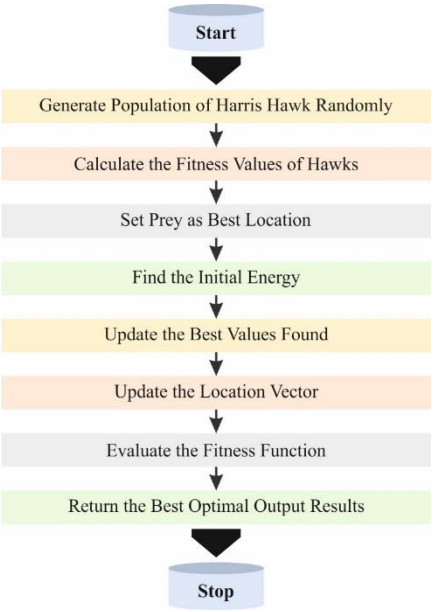

Figure 2. Flowchart of IHHO algorithm

B. Classification Model

Furthermore, the hybrid of CNN-BiLSTM-AM model is employed for the classification of cybersecurity threats. A CNN is a forward NN that is efficiently used for the time series [19]. A CNN is mostly comprised of 3 segments: a pooling, a convolutional, and a fully connected (FC) layers. It includes the features of local view and sharing of weight that importantly decreases the amount of model parameters and enhances the learning efficacy of the method. All convolutional layers have numerous convolutional kernels that are computed as demonstrated in Eq. (14). Afterward the information passes over the convolutional process of the layer of convolution, and the efficient data can be extracted. The pooling layer will handle the data from the convolution layer, minimize the feature size, decrease the training price of the system, and lastly handle an output over the FC layer.

$$Y_t = \tanh(x_t * w_t + b_t) \quad (14)$$

Whereas Y_t means output value, $\tanh$ refers tan activation function, x_t stands for vector of input, w_t represents convolutional kernel's weight, and b_t symbolizes convolution kernel's bias.

The theory of BiLSTM was initially presented in LSTM study that handles data sequences in dual ways (for instance, backward and forward). This method contains dual distinct hidden layers (HLs) linked to the similar output layer. The forward HLs (h_t^f) in the order of ascending $(1, 2, 3, \ldots, T)$ obtain the timing data after sending it to the HL (h_t^{bk}) i in the order of descending $(T, \ldots, 3, 2, 1)$ to obtain the time series information. The mathematic

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

representations of the output, forward, and backward layers are shown in Eqs. (14) ~ (16), and in recent times, BiLSTM systems have exposed improved functioning than one-directional LSTM in real application.

$$h_t^f = \tanh\left(w_{xh}^f x_t + w_{hh}^f h_{t-1}^f + b_h^f\right) \quad (15)$$

$$h_t^{bk} = \tanh\left(w_x^{bk} x_t + w_{hh}^{bk} h_{t+1}^{bk} + b_h^{bk}\right) \quad (16)$$

$$Y_t = w_{hy}^f h_t^f + w_{hy}^{bk} h_t^{bk} + b_y \quad (17)$$

Attention was presented by computing the attention probability distribution, the major data has been chosen from a larger amount of data, the main input is emphasized, and the conventional method is enhanced. In the same way, attention particularly concentrations on a few of the essential data avoids less significant data and allocates importance to the data, the computation procedure of attention is normally separated into 3 phases:

Computation of relevance or similarity between Key (input feature) and Query (output feature), as demonstrated in Eq. (18):

$$s_f = \tanh\left(w_h h_t + b_h\right) \quad (18)$$

Here w_h means attention mechanism's weight, b_h refers to attention mechanism's bias, h_t represents input vector, w_h and b_h denotes sharing weight in all layers.

The grades of the initial phase are standardized, and the attention scores were transformed to Eq. (19) with$SoftMax$:

$$a_t = \frac{\exp\left(s_f^T v\right)}{\sum_t \exp\left(s_f^T v\right)} \quad (19)$$

Here v represents the value of attention.

Based on the weight coefficient, which is performed to gain the last value of attention, as exposed in Eq. (20):

$$s = \sum_t a_t \, h_t \quad (20)$$

The CNN-BILSTM-AM hybrid method has been applied to forecast natural gas well manufacture, which mostly contains a CNN, input, BiLSTM, attention, and output layers. It makes overall usage of CNN to remove spatial characteristics of unique information and incorporates the attention mechanism for underlining the significant vector of features, so increasing the accuracy and computation of the Bi-LSTM approach.

C. Hyperparameter Tuning Model

Eventually, the MBGO algorithm adjusts the hyperparameter values of the CNN-BiLSTM-AM classifier optimally and outcomes in greater classification performance. Stimulated by the interactions and behaviors detected in multi-player battle games, MBGO approximately splits the optimizer procedure into dual distinct stages: the battle and the movement stages [20]. All these stages incorporate sensibly designed search agents, which imitate the tactical player's behaviors involved in the battle games. Movement stage: During this stage, the movements of the players are mainly affected in the safer zone, a predominant game element in numerous MBGs. Players must move directly near the safer zone to prevent loss from the outside setting. Encouraged by this novel gaming method, this method shows the present finest individual X_{best}^t as the middle of the safer zone, whereas t signifies the Euclidean distance and the iteration among X_{best}^t and the present poor individual X_{worst}^t helps as the basic radius. To present randomness levels, an arbitrary value $\delta \sim U(0.8,1.2)$ has been applied to intensify the radius. Eq. (21) describes the radius as shown.

$$R = (\|X_{best}^t - X_{worst}^t + \epsilon\| \quad).\delta \quad (21)$$

While $\| \|$ characterizes the Euclidean distance, and e refers to smaller value to avoid the radius from becoming $zero$. X_{best}^t makes off-spring individual, as stated in Eq. (22)

$$X_{new}^t = X_i^t + X_{best}^t \cdot \sin(2\pi r) \quad (22)$$

Here r denotes randomly generated number in the range of $(0,1)$. For individuals located outer the safer zone, Eq. (23) has been applied to accelerate the movement of the individual concerning possible regions.

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

$$X_{new,k}^{t} = \begin{cases} X_{i,k}^{t} + \theta, if \cdot r < 05 \\ X_{i,k}^{t} + \left(X_{bt,k}^{t}, -X_{i,k}^{t}\right) \cdot r, \text{otherwise} \end{cases} \quad (23)$$

Here θ means randomly produced value succeeding the normal standard distribution $N(O,1)$. More accurately, for every self-determining dimension k, every system in Eq. (23) contains an equivalent possibility of having a selected for making the off-spring individual.

Battle phase: It follows different behavior of the players after meeting arbitrary enemies in the game. However, the behaviors of the player might show unpredictability throughout the game, the final target residues exist till the game ends, and the removal of enemies to the greatest extent promising. This method streamlines real-game explanations and executes perfect limitations. Eqs. (24) and (25) offer mathematic representations for challenging best and poor enemies, correspondingly.

$$X_{new,k}^{t} = \begin{cases} X_{i,k}^{t} + dir_k \cdot r, ifr < 0.5 \\ X_{enemy,k}^{t} + dir_k r, otherwise \end{cases} \quad (24)$$

$$X_{new}^{t} = X_{i}^{t} + dir \cdot \cos(2\pi r) \quad (25)$$

whereas dir characterizes the variance vector between the jth individual X_i^t and the randomly chosen enemy X_{enemy}^t, as stated in Eq. (26).

$$dir = \begin{cases} X_i^t - X_{enemy}^t if X_i^t \; has \; a \; better \; fitness \; value \\ X_{enemy}^t - X_i^t, otherwise \end{cases} \quad (26)$$

Moreover, MBGO incorporates the embedded greedy selection to guarantee the elite's survival. Specially, after the off-spring individual can be made, Eq. (27).

$$X_i^{t+1} = \begin{cases} X_i^t, if \; X_i^t \; has \; a \; better \; fitness \; value \\ X_{new}^t \; otherwise \end{cases} \quad (27)$$

MBGO assistances from this embedded greedy range in rapid knowledge interchange from the upgraded population. Finally, the MBGO pseudocode is represented in Algorithm1.

Algorithm 1: Pseudocode of MBGO

Input: Size of population: N, Dimension: D, Maximum iteration: T_{max}

Output: Optimum: x_{best}^t

Function MBGO (N, D, T_{max})

Initializing the population arbitrarily

$t \leftarrow 0$

$x_{best}^t \leftarrow best\ (R)$

While $t < T_{max}$ do

Movement stage

for $i = 0$ to N do

Define the safe area utilizing Eq. (21)

if x_i^t is inside the safe area then

Constructing x_{new}^t utilizing Eq. (22)

end

else

Constructing x_{new}^t utilizing Eq. (23)

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

end

embedded greedy selection

end

Battle stage

for $i = 0$ to N do

Choose a arbitrary enemy x_{enemy}^{t} for x_{i}^{t}

if x_{enemy}^{t} has superior fitness value then

Constructing x_{new}^{t} using Eq. (24)

end

else

Constructing x_{new}^{t} using Eq. (25)

end

Embedded greedy selection

end

$x_{best}^{t} \leftarrow best(R)$

$t \leftarrow t + 1$

end

return x_{best}^{t}

The fitness selection (FS) is the substantial factor manipulating the efficiency of MBGO. The hyperparameter selection procedure engages the solution encoded technique to estimate the effectiveness of the candidate solution. Here, the MBGO reflects precision as the main measure to project the FF. Its mathematical formulation is expressed below:

$$Fitness = \max(P) \tag{28}$$

$$P = \frac{TP}{TP + FP} \tag{29}$$

Where, TP denote the positive value of true and FP represents the positive value of false.

4. Experimental Analysis

The performance evaluation of EDCA-MBGOHDL algorithm is tested under dual datasets such as CICIDS-2017 [21] and ToN-loT [22]. The CICIDS-2017 dataset has 10973 records under five classes as represented in Table 1. The total number of features are 78 but only 35 features are selected.

Table 1: Details of CICIDS-2017 Dataset

CICIDS-2017 Dataset	
Classes	No. of Records
"Benign"	2500
"DDoS"	2500
"DoS"	2500
"Bot"	1966
"Web Attack"	1507
Total Records	10973

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

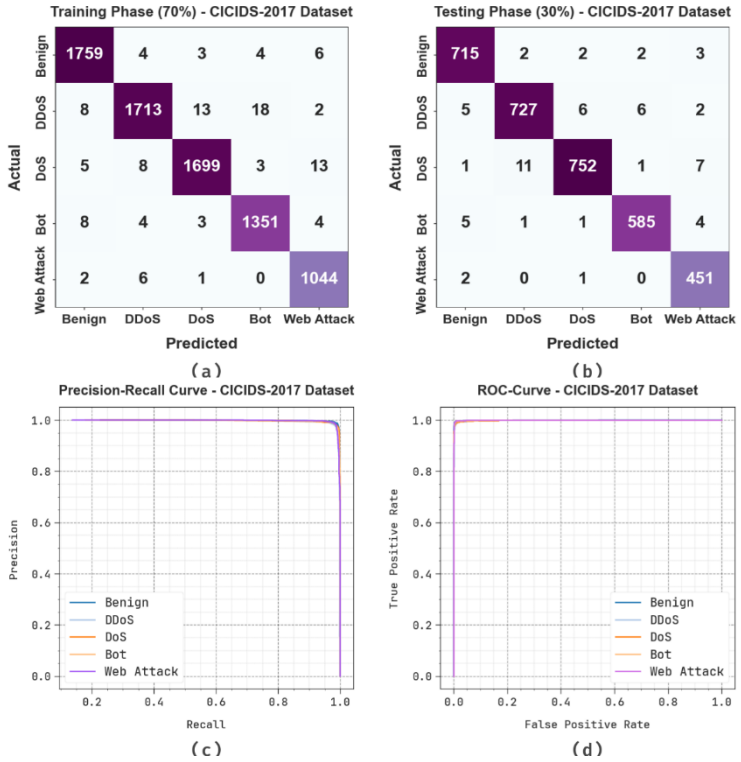

Figure 3. CICIDS-2017 dataset (a-b) confusion matrix, (c) curve of PR and (d) curve of ROC

The classifier results of the EDCA-MBGOHDL model based on CICIDS-2017 dataset is presented in Fig. 3. Figs. 3a-3b represent the confusion matrix, showcasing the precise classification of distinct class labels based on 70%TRPH and 30%TSPH. Fig. 3c shows the PR study which indicates enhanced performance through all classes. At last, Fig. 3d demonstrates the ROC study, signifying proficient results with robust ROC values for each class label.

Table 2 and Fig. 4 denotes the cybersecurity recognition of EDCA-MBGOHDL technique under CICIDS-2017 dataset. The performances indicate in which the EDCA-MBGOHDL system properly recognized the samples. With 70%TRPH, the EDCA-MBGOHDL model provides average $accu_y$, $prec_n$, $reca_l$, $F1_{score}$ and AUC_{score} of 99.40%, 98.42%, 98.56%, 98.49%, and 99.09%, respectively. Moreover, with 30%TSPH, the EDCA-MBGOHDL method offers average $accu_y$, $prec_n$, $reca_l$, $F1_{score}$ and AUC_{score} of 99.25%, 98.01%, 98.22%, 98.11%, and 98.88%, respectively.

Table 2: Cybersecurity detection of EDCA-MBGOHDL method under CICIDS-2017 dataset

Class Labels	$Accu_y$	$Prec_n$	$Reca_l$	$F1_{score}$	AUC_{score}
TRPH (70%)					
Benign	99.48	98.71	99.04	98.88	99.33
DDoS	99.18	98.73	97.66	98.19	98.65
DoS	99.36	98.84	98.32	98.58	98.99
Bot	99.43	98.18	98.61	98.40	99.11
Web Attack	99.56	97.66	99.15	98.40	99.38
Average	99.40	98.42	98.56	98.49	99.09
TSPH (30%)					
Benign	99.33	98.21	98.76	98.48	99.13
DDoS	99.00	98.11	97.45	97.78	98.45
DoS	99.09	98.69	97.41	98.04	98.51
Bot	99.39	98.48	98.15	98.32	98.91
Web Attack	99.42	96.57	99.34	97.94	99.39
Average	99.25	98.01	98.22	98.11	98.88

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

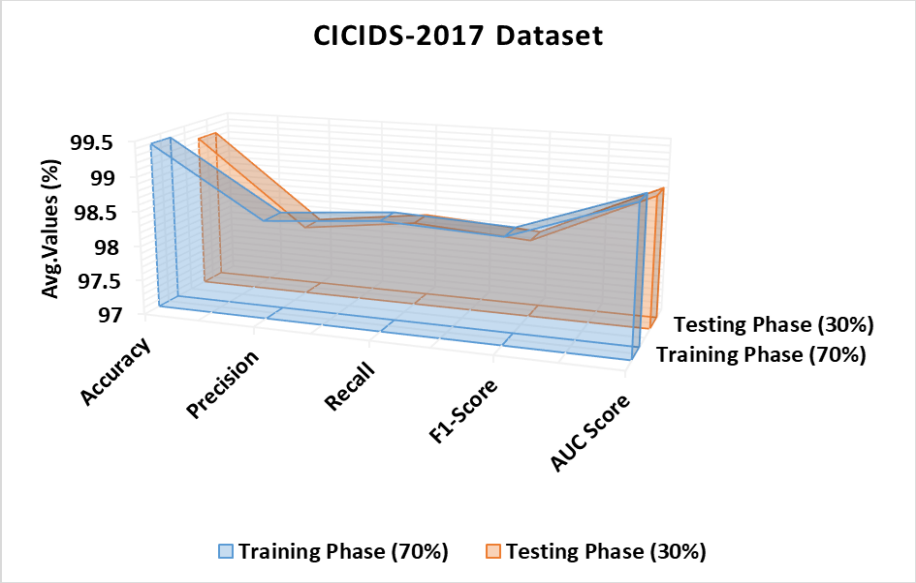

Figure 4. Average of EDCA-MBGOHDL method under CICIDS-2017 dataset

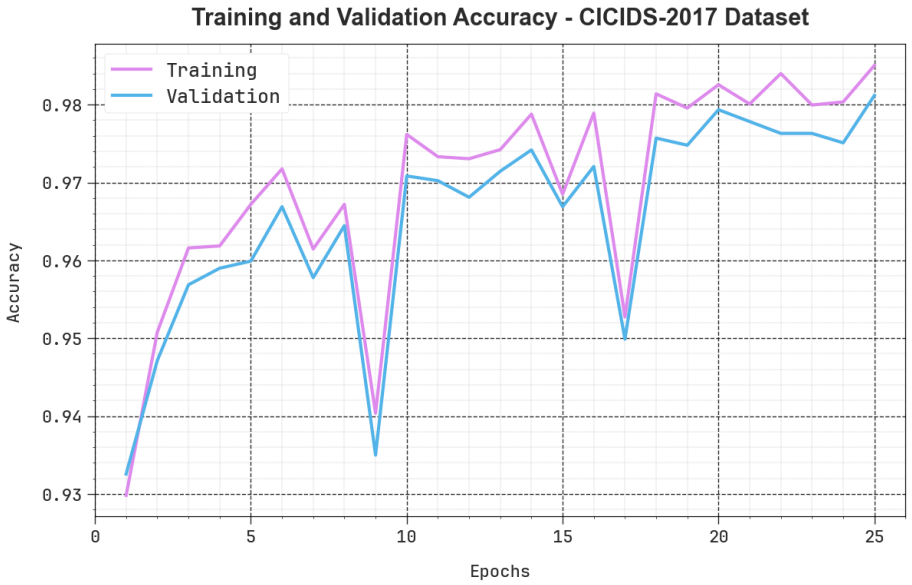

Figure 5. $Accu_y$ Curve of EDCA-MBGOHDL method under CICIDS-2017 dataset

In Fig. 5, the training (TRA) $accu_y$ and validation (VAL) $accu_y$ performances of the EDCA-MBGOHDL approach under CICIDS-2017 dataset is depicted. The $accu_y$ values are calculated within the range of 0-25 epochs. The figure highlighted in which the values of TRA and VAL $accu_y$ show an increasing trend, notifying the proficiency of the EDCA-MBGOHDL algorithm, which demonstrates higher performance above multiple iterations. Moreover, the TRA and VAL $accu_y$remains closer through the epochs specifying lesser overfitting and reveals maximum performance of the EDCA-MBGOHDL approach, ensuring continual calculation on unseen samples.

In Fig. 6, the TRA loss (TRALOS) and VAL loss (VALLOS) graph of the EDCA-MBGOHDL methodology under CICIDS-2017 dataset is revealed. The values of loss are computed across an interval of 0-25 epochs. It is illustrated in which the values of TRALOS and VALLOS signifies a decreasing trend, notifying the proficiency of the EDCA-MBGOHDL algorithm in harmonizing an equilibrium among generalize and data fitting. The constant decrease in values of loss moreover securities the maximal outcome of the EDCA-MBGOHDL model and tune the prediction results with time.

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

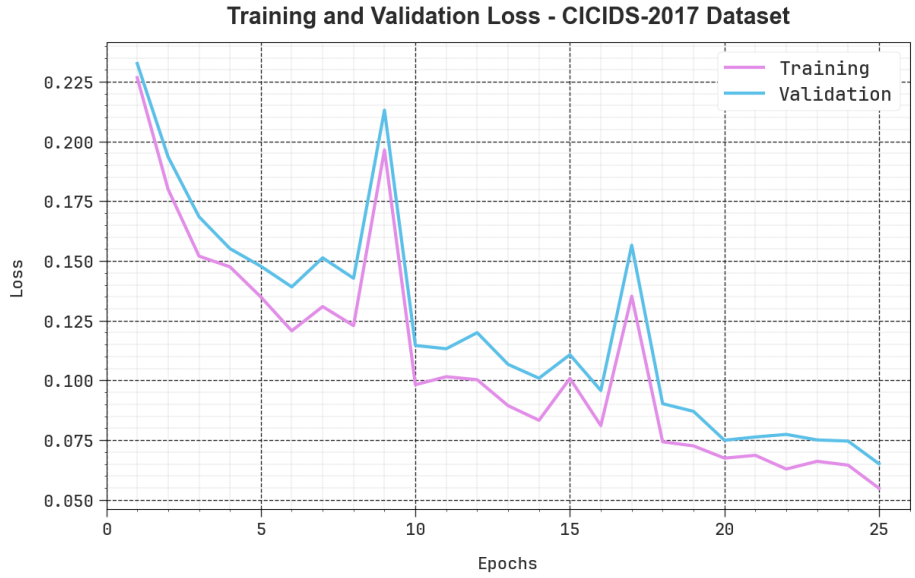

Figure 6. Loss curve of EDCA-MBGOHDL method using CICIDS-2017 dataset

Fig. 7 inspects the comparison results of the EDCA-MBGOHDL method under CICIDS-2017 dataset with the existing techniques [23, 24, and 25]. The results highlighted that the AE-GAN, GCNN-AE, Kernel SVM, DT, KNN and RF algorithms have reported worse performance. In the meantime, POADEL-ID model have gained closer outcomes. Additionally, the EDCA-MBGOHDL algorithm reported superior performance with enhanced$prec_n$, $reca_l$, $accu_y$, $and\ F1_{score}$ of 98.42%, 98.56%, 99.40%, and 98.49%, respectively.

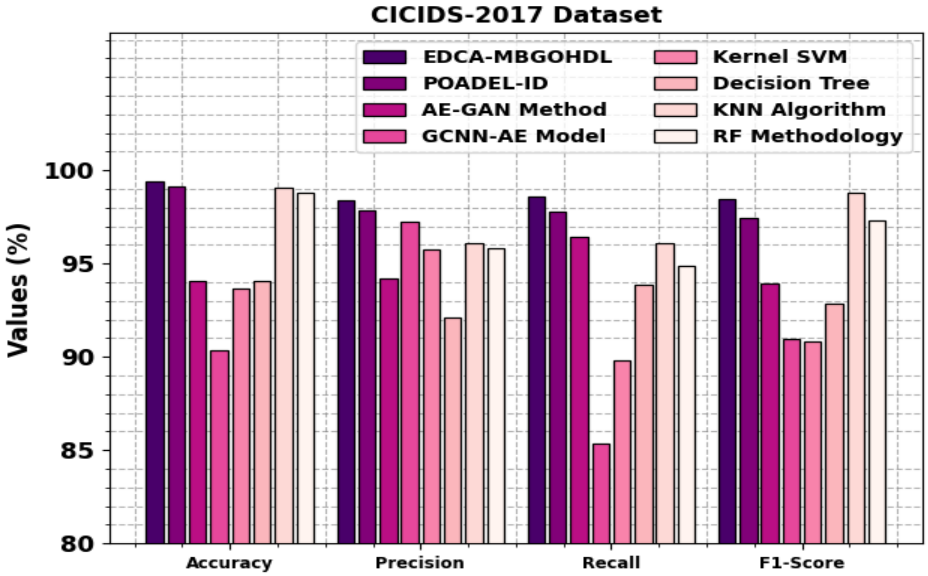

Figure 7. Comparative analysis of EDCA-MBGOHDL technique under CICIDS-2017 dataset

The ToN-loT dataset contains 119957 instances under nine classes as exposed in Table 3. The total number of features are 75 but only 40 features are selected.

Table 3: Details of ToN-loT Dataset

ToN-loT Dataset	
Classes	No. of Instances
“Normal”	78369
“MiTM”	336
“DoS”	5440

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

"DDoS"	5987
"Password"	6016
"Injection"	5867
"xss"	5951
"Ransomware"	5976
"Backdoor"	6015
Total Instances	119957

Fig. 8 presents the classifier outcomes of the EDCA-MBGOHDL system using ToN-loT dataset. Figs. 8a-8b reveal the confusion matrices with precise identification of all class labels under 70%TRPH and 30%TSPH. Fig. 8c shows the PR investigation which indicates improved performance through all classes. Eventually, Fig. 8d signifies the ROC investigation, exemplifying proficient results with robust ROC values for each class label.

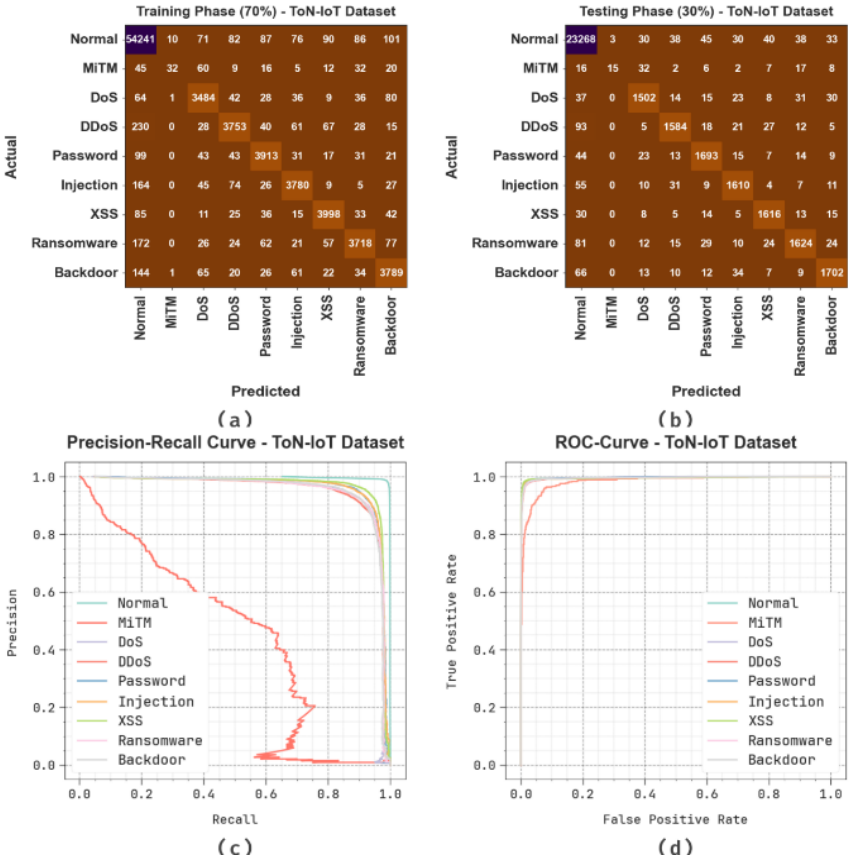

Figure 8. ToN-loT dataset (a-b) confusion matrices, (c) curve of PR and (d) curve of ROC

Table 4 and Fig. 9 illustrates the cybersecurity recognition of EDCA-MBGOHDL technique under ToN-loT dataset. The outcomes indicate that the EDCA-MBGOHDL algorithm properly designated the samples. With 70%TRPH, the EDCA-MBGOHDL approach provides average $accu_y$, $prec_n$, $reca_l$, $F1_{score}$ and AUC_{score} of 99.14%, 90.67%, 83.69%, 84.95%, and 91.50%, respectively. Additionally, with 30%TSPH, the EDCA-MBGOHDL method provides average $accu_y$, $prec_n$, $reca_l$, $F1_{score}$ and AUC_{score} of 99.15%, 91.94%, 83.90%, 85.20%, and 91.61%, correspondingly.

Table 4: Cybersecurity detection of EDCA-MBGOHDL method under ToN-loT dataset

Class Labels	$Accu_y$	$Prec_n$	$Reca_l$	$F1_{score}$	AUC_{score}
TRPH (70%)					
Normal	98.09	98.18	98.90	98.54	97.73
MiTM	99.75	72.73	13.85	23.27	56.92
DoS	99.23	90.89	92.17	91.53	95.87
DDoS	99.06	92.17	88.89	90.50	94.25

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

Password	99.28	92.42	93.21	92.81	96.40
Injection	99.22	92.51	91.53	92.02	95.57
XSS	99.37	93.39	94.18	93.78	96.91
Ransomware	99.14	92.88	89.44	91.13	94.54
Backdoor	99.10	90.82	91.04	90.93	95.28
Average	99.14	90.67	83.69	84.95	91.50
TSPH (30%)					
Normal	98.11	98.22	98.91	98.56	97.76
MiTM	99.74	83.33	14.29	24.39	57.14
DoS	99.19	91.87	90.48	91.17	95.05
DDoS	99.14	92.52	89.75	91.11	94.69
Password	99.24	91.96	93.12	92.54	96.35
Injection	99.26	92.00	92.69	92.34	96.14
XSS	99.41	92.87	94.72	93.79	97.18
Ransomware	99.07	92.01	89.28	90.62	94.43
Backdoor	99.21	92.65	91.85	92.25	95.73
Average	99.15	91.94	83.90	85.20	91.61

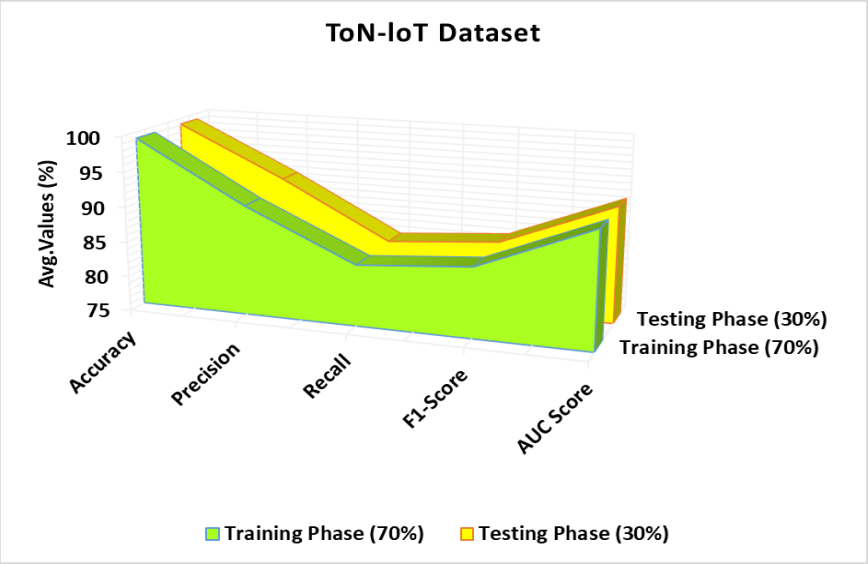

Figure 9. Average of EDCA-MBGOHDL method under ToN-loT dataset

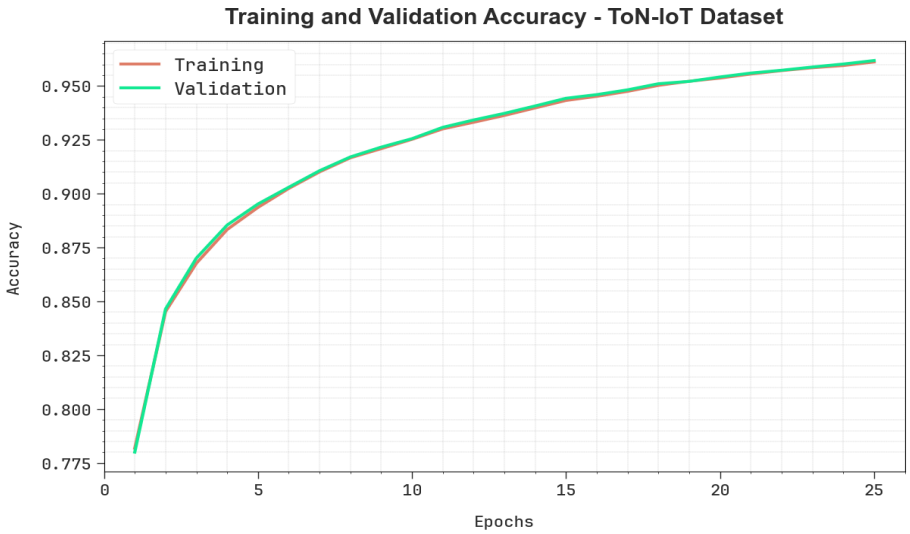

Figure 10. $Accu_y$ Curve of EDCA-MBGOHDL method under ToN-loT dataset

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

In Fig. 10, the TRA $accu_y$ and VAL $accu_y$ performances of the EDCA-MBGOHDL algorithm under ToN-loT dataset is exemplified. The $accu_y$ values are computed across within in the range of 0-25 epochs. The figure highlighted in which the values of TRA and VAL $accu_y$ shows a increasing trend, notifying the proficiency of the EDCA-MBGOHDL approach using maximum performance through multiple repetitions. In addition, the values of TRA and VAL $accu_y$ remain close through the epochs specifying diminish overfitting and reveals improved performance of the EDCA-MBGOHDL methodology, ensuring stable prediction on hidden samples.

In Fig. 11, the TRALOS and VALLOS graph of the EDCA-MBGOHDL system under ToN-loT dataset is shown. The values of loss are computed throughout 0-25 epochs. It is exposed that the TRALOS and VALLOS values demonstrate a decreasing trend, notifying the proficiency of the EDCA-MBGOHDL algorithm in harmonizing an equilibrium among generalization and data fitting. The consistent decrease in values of loss furthermore assurances the higher performance of the EDCA-MBGOHDL technique and tune the prediction results through time.

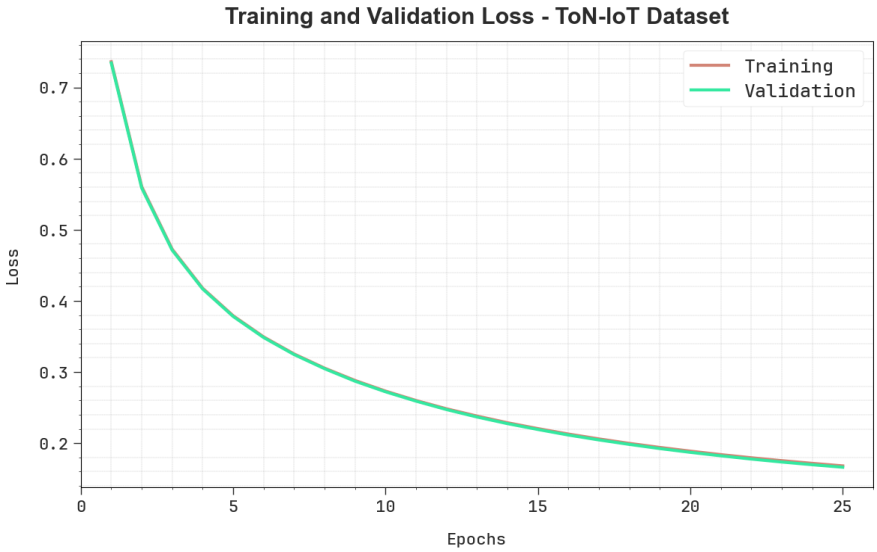

Figure 11. Loss curve of EDCA-MBGOHDL method under ToN-loT dataset

Fig. 12 studies the comparison results of the EDCA-MBGOHDL algorithm under ToN-loT dataset with the existing methodologies. The results highlighted that the IDS-EESAEE, RF, GDNN-AE, LSTM, GRU and GLSTM techniques have reported worse performance. Afterwards, EBWO-HDLID model have attained closer outcomes. Also, the EDCA-MBGOHDL technique reported higher performance with improved $prec_n$, $reca_l$, $accu_y$, $and\ F1_{score}$ of 91.94%, 83.90%, 99.15%, and 85.20%, respectively.

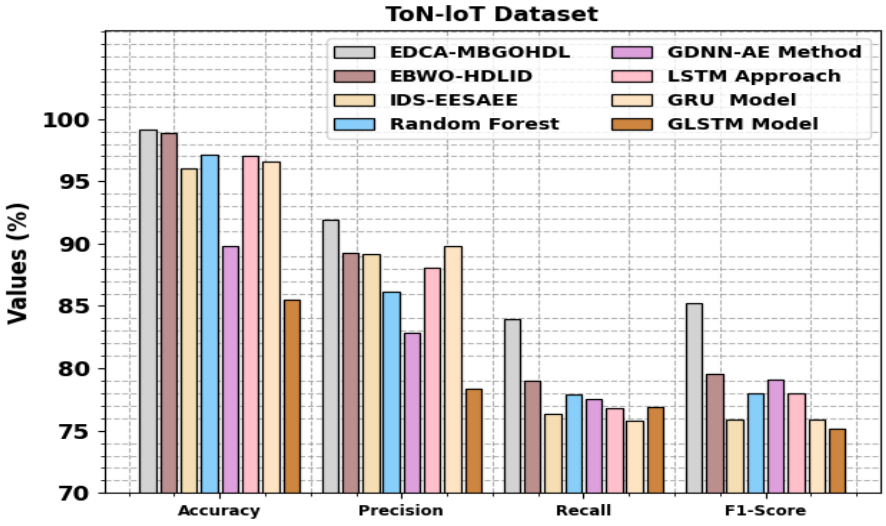

Figure 12. Comparative analysis of EDCA-MBGOHDL method under ToN-loT dataset

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

5. Conclusion

This manuscript presents an EDCA-MBGOHDL technique. The main intention of the EDCA-MBGOHDL technique is to provide a robust framework for cyberattack detection using deep learning integrated with a hyperparameter tuning approach. At first, the feature selection process is implemented by applying IHHO algorithm for ensuring that only the most relevant features are served into the model. Furthermore, the hybrid of CNN-BiLSTM-AM system is applied for the classification of cybersecurity threats. Eventually, the MBGO algorithm adjusts the hyperparameter values of the CNN-BiLSTM-AM classifier optimally and outcomes in greater classification performance. The wide range of analysis of the EDCA-MBGOHDL technique takes place using a benchmark dataset. The outcomes pointed out the better performance of the EDCA-MBGOHDL system across existing models.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] Ahsan, M., Nygard, K.E., Gomes, R., Chowdhury, M.M., Rifat, N. and Connolly, J.F., 2022. Cybersecurity threats and their mitigation approaches using Machine Learning—A Review. Journal of Cybersecurity and Privacy, 2(3), pp.527-555.

[2] Gümüşbaş, D., Yıldırım, T., Genovese, A. and Scotti, F., 2020. A comprehensive survey of databases and deep learning methods for cybersecurity and intrusion detection systems. IEEE Systems Journal, 15(2), pp.1717-1731.

[3] Trong, H.M.D., Le, D.T., Veyseh, A.P.B., Nguyễn, T. and Nguyen, T.H., 2020, November. Introducing a new dataset for event detection in cybersecurity texts. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 5381-5390).

[4] Sun, C.C., Cardenas, D.J.S., Hahn, A. and Liu, C.C., 2020. Intrusion detection for cybersecurity of smart meters. IEEE Transactions on Smart Grid, 12(1), pp.612-622.

[5] Khandpur, R.P., Ji, T., Jan, S., Wang, G., Lu, C.T. and Ramakrishnan, N., 2017, November. Crowdsourcing cybersecurity: Cyber-attack detection using social media. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (pp. 1049-1057).

[6] Narayanan, S.N., Ganesan, A., Joshi, K., Oates, T., Joshi, A. and Finin, T., 2018, October. Early detection of cybersecurity threats using collaborative cognition. In 2018 IEEE 4th international conference on collaboration and internet computing (CIC) (pp. 354-363). IEEE.

[7] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I.A., Chen, S., Liu, D. and Li, J., 2020. Performance comparison and current challenges of using machine learning techniques in cybersecurity. Energies, 13(10), p.2509.

[8] Handa, A., Sharma, A. and Shukla, S.K., 2019. Machine learning in cybersecurity: A review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(4), p.e1306.

[9] Mahdavifar, S. and Ghorbani, A.A., 2019. Application of deep learning to cybersecurity: A survey. Neurocomputing, 347, pp.149-176.

[10] Han, Y. , M., I. , Cai, W. (2019) . Dragonfly Algorithm with Gated Recurrent Unit for Cybersecurity in Social Networking. Journal of Cybersecurity and Information Management, 75-88.

[11] Duraibi, S. and Alashjaee, A.M., 2024. Enhancing Cyberattack Detection Using Dimensionality Reduction with Hybrid Deep Learning on Internet of Things Environment. IEEE Access.

[12] Awajan, A., 2023. A novel deep learning-based intrusion detection system for IOT networks. Computers, 12(2), p.34.

[13] Maddu, M. and Rao, Y.N., 2024. Res2Net-ERNN: deep learning based cyberattack classification in software defined network. Cluster Computing, pp.1-19.

[14] Gueye, T., Wang, Y., Rehman, M., Mushtaq, R.T. and Zahoor, S., 2023. A novel method to detect cyber-attacks in IoT/IIoT devices on the modbus protocol using deep learning. Cluster Computing, 26(5), pp.2947-2973.

[15] Vaiyapuri, T., Shankar, K., Rajendran, S., Kumar, S., Gaur, V., Gupta, D. and Alharbi, M., 2024. Automated cyberattack detection using optimal ensemble deep learning model. Transactions on Emerging Telecommunications Technologies, 35(4), p.e4899.

[16] Prabakar, D., Sundarrajan, M., Manikandan, R., Jhanjhi, N.Z., Masud, M. and Alqhatani, A., 2023. Energy analysis-based cyber-attack detection by IoT with artificial intelligence in a sustainable smart city. Sustainability, 15(7), p.6031.

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024

[17] Alrayes, F.S., Nemri, N., Aljaffan, N., Alshuhail, A., Alhashmi, A.A. and Mahmud, A., 2024. Distributed Multiclass Cyberattack Detection using Golden Jackal Optimization with Deep Learning Model for Securing IoT Networks. IEEE Access.

[18] Jia, D. and Wang, D., 2024, October. A Maximum Power Point Tracking (MPPT) Strategy Based on Harris Hawk Optimization (HHO) Algorithm. In Actuators (Vol. 13, No. 11, p. 431). MDPI.

[19] Gu, D., Zheng, R., Cheng, P., Zhou, S., Yan, G., Liu, H., Yang, K., Wang, J., Zhu, Y. and Liao, M., 2024. Single Well Production Prediction Model of Gas Reservoir Based on CNN-BILSTM-AM. Energies, 17(22), p.5674.

[20] Zhong, R., Xu, Y., Zhang, C. and Yu, J., 2025. Efficient multiplayer battle game optimizer for numerical optimization and adversarial robust neural architecture search. Alexandria Engineering Journal, 113, pp.150-168.

[21] https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset

[22] N. Moustafa, ‘‘A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets,’’ Sustain. Cities Soc., vol. 72, Sep. 2021, Art. no.102994.

[23] Park, C., Lee, J., Kim, Y., Park, J.G., Kim, H. and Hong, D., 2022. An enhanced AI-based network intrusion detection system using generative adversarial networks. IEEE Internet of Things Journal, 10(3), pp.2330-2345

[24] Alrayes, F.S., Aljebreen, M., Maray, M., Alshuhail, A., Alrslani, F.A. and Salama, A.S., 2024. Optimizing Security Protocol: A Synergy of Bio-inspired Planet Optimization Algorithm with Ensemble Learning-based Attack Detection for Connected and Autonomous Vehicles. IEEE Access.

[25] Aburasain, R.Y., 2024. Enhanced Black Widow Optimization with Hybrid Deep Learning Enabled Intrusion Detection in Internet of Things-based Smart Farming. IEEE Access.

DOI: https://doi.org/10.54216/JCIM.150210
Received: May 07, 2024 Revised: July 12, 2024 Accepted: October 24, 2024