

Next-Generation Cyberattack Detection for Industrial IoT using Extreme Learning Machine with Optimization Algorithm

Maha Farouk Sabir^{1,*}

¹ Information Systems Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; msaber@kau.edu.sa

Abstract


The reliability of an industrial Internet of Things (IIoT) system is a significant end-user preference. Preserving network reliability is vital to void the loss of life. A trustworthy IIoT network incorporates the safety features of IT trustworthiness-security, safety, resilience, reliability, and privacy. Traditional security techniques and tools are not sufficient to protect the platform of IIoT owing to the variance in protocols, restricted upgrade opportunities, divergence in protocols, and earliest forms of the operating system employed in the industrial systems. With the unexpected and diversification behaviors of cyber-security attacks, classical cyber-attack recognition methods have some crucial challenges with enlarging huge data with inaccurate classification methods, unappropriated feature selection (FS) and extraction, and high computation time in prediction. This study develops an Advanced Cyberattack Detection for Industrial IoT using the Binary Salp Swarm Algorithm (ACDIIOT-BSSA) technique. The projected ACDIIOT-BSSA method mainly addresses the classification and identification of attack recognition in achieving cyber security. The first phase of data pre-processing is implemented to alter the input data into the relevant format. Next, the proposed ACDIIOT-BSSA approach achieves feature selection progress utilizing the binary salp swarm algorithm (BSSA) algorithm. For attack recognition, the ACDIIOT-BSSA method uses extreme learning machine (ELM) technique. Finally, arithmetic optimization algorithm (AOA) is deployed as a hyperparameter optimizer for the ELM method. To inspect the improved performance of the proposed ACDIIOT-BSSA approach, a wide range of experiments were done. The empirical findings reported a better outcome of the ACDIIOT-BSSA method over other existing techniques.


Keywords

Industrial Internet of Things, Cyberattack Detection, Arithmetic Optimization Algorithm, Feature Selection, Deep Learning

Proceedings of SNSFAIT 2024: International Symposium on Securing Next-Generation Systems using Future Artificial Intelligence Technologies, Delhi, India, August 08-09th, 2024

^{1*} Corresponding author: Maha Farouk Sabir

 (msaber@kau.edu.sa)

 <https://orcid.org/0000-0002-4233-1647>



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

.0).

1. Introduction

Cybersecurity plays a dangerous part under industrial control systems (ICSs) observant versus possible malicious activity and ensures the continuous functionalism of crucial national frameworks [1]. The requests of Industry 4.0 which is extremely automatic and has minimum human intrusion leads to the growth of incorporation of the Industrial Internet of Things (IIoT) within industrialized processes. The dependence on connected systems has developed significantly, then constructing industrialized network control methods more sensitive to cyber-attacks [2]. Therefore, the significance of executing robust cyber-security functionalism or protocols inside ICSs has become an imperative concern. The smart organization in the IIoT environment is exposed to several cyber-attacks such as Man-in-the-Middle attacks (MiM), DDoS, Infiltration attacks, Backdoors, and so on. Such attacks can break the integrity and confidentiality of data in that network [3]. An Intrusion Detection System (IDS) can be a safety device for protecting data traffic. It works for the next route of safety which protects the networks. IDS detects the networks in all admission points and identifies some intrusion in the packets running into the channel event that it signals the particular authority [4]. IDS is normally used afterward as a firewall, and it appears as an enhanced place for its arrangement. IDS were well-known mostly in two groups termed Signature-based IDS (SIDS) and Anomaly-based IDS (AIDS) [5]. SIDS mechanism with pattern toning method for example which scans the data packet toward malicious content with attack patterns. Having a pre-defined database or list of the signatures or patterns or the well-known attacks, what employ them by relating the data packets with them to detect the well-known attack [6].

Conventional cyber-security schemes safeguard users and devices via IDS firewalls, user authentication, anti-virus software, and data encryption [7]. The usage of a Machine Learning (ML) model for detecting malignant network traffic, anomalous behaviors, and challenges in computer schemes in an IDS becomes inadequate [8]. However, traditional MLs lack automated feature engineering, hold a lower detection level, and are not effective in identifying minor alternatives to present attacks. This has generated a deliberate DL model for improving cyber-security schemes [9]. DL is an ML subfield, which has grown high recognition in several fields owing to its development in precision in difficult tasks and the latest expansions in software and hardware [10]. DL methods increase cyber-security schemes preventing attacks by detecting patterns, which are diverse from standard behavior.

This study develops an Advanced Cyberattack Detection for Industrial IoT using Binary Salp Swarm Algorithm (ACDIIOT-BSSA) technique. The projected ACDIIOT-BSSA method mainly addresses the classification and identification of attack recognition in achieving cyber security. The first phase of data pre-processing is implemented to alter the input data into the relevant format. Next, the proposed ACDIIOT-BSSA approach achieves feature selection progress utilizing the binary salp swarm algorithm (BSSA) algorithm. For attack recognition, the ACDIIOT-BSSA method uses extreme learning machine (ELM) technique. Finally, arithmetic optimization algorithm (AOA) is deployed as a hyperparameter optimizer for the ELM method. The empirical findings reported a better outcome of the ACDIIOT-BSSA method over other existing techniques.

2. Literature Review

In [11], 2 different DL methods are used namely CNNs and Deep Belief Networks (DBNs) considered as hybrid classifications, to generate methods for identifying attacks in IoT enabled cyber physical methods. Also, this study aims to propose a novel hybrid optimizer method named “Seagull Adapted Elephant Herding Optimizer” (SAEHO) to fine-tune the hybrid classification weights. The “Hybrid Classification + SAEHO” method extracts the feature extraction datasets as input and identifies the networks as both benign or attacked. Li *et al.* [12] develop feasible solutions based on federated sequence learning (FSL) with cyberattack recognition abilities. In federated frameworks, FSL creates a collective global method unless violating local data unity. Exploitation of the local sequential model, FSL seizes the inherent industry time series response. In addition, data heterogeneity between distributed consumers is also regarded that is significant for maintaining a robust but delicate attack recognition. Durairaj *et al.* [13] use the DBNs which is one of the DL methods with few enhancements. To enhance the precision of the detections, a rule based recognition method is included to improve the recognition of intruders by utilizing DBN. The presented method is followed by the layer microgrid structure, which forms the system flexibility and simple towards the execution. The presented 2 attacks, like Denial of Service attacks and False Data Injection, are produced by Greedy Algorithms and are identified by the presented method.

Mohy-Eddine [14] designed an intrusion detection method using ML and feature engineering for IIoT security. The method incorporates Isolation Forest (IF) through Pearson's Correlation Coefficient (PCC) to decrease the forecast time and computing cost. IF is used to identify and delete anomalies from the dataset. The method employs PCC to select the most proper feature. IF and PCC are used interchangeably (IFPCC and PCCIF). The RF classification is executed to improve IDS performance. Kunang *et al.* [15] introduced a hybrid DL method. This method utilized unsupervised methods to mine features and data dimensions, then a neural network for classifications. Various methods are utilized to identify the efficacy of the DL based IoT IDS by 2 feature extraction scenarios. The initial stage utilized AE variations like deep AE (DAE), deep LSTM AE (LSTM-DAE), and deep convolutional AE. The second stage utilized stack methods for feature extractions, containing stacked AE and deep belief networks.

3. Materials and Methods

In this study, we have developed a novel ACDIIOT-BSSA technique. The projected ACDIIOT-BSSA method mainly addresses the classification and identification of attack recognition in achieving cyber security. To accomplish that, the ACDIIOT-BSSA technique has data normalization, BSSA based FS, ELM based attack detection, and AOA based parameter selection are illustrated in Fig. 1.

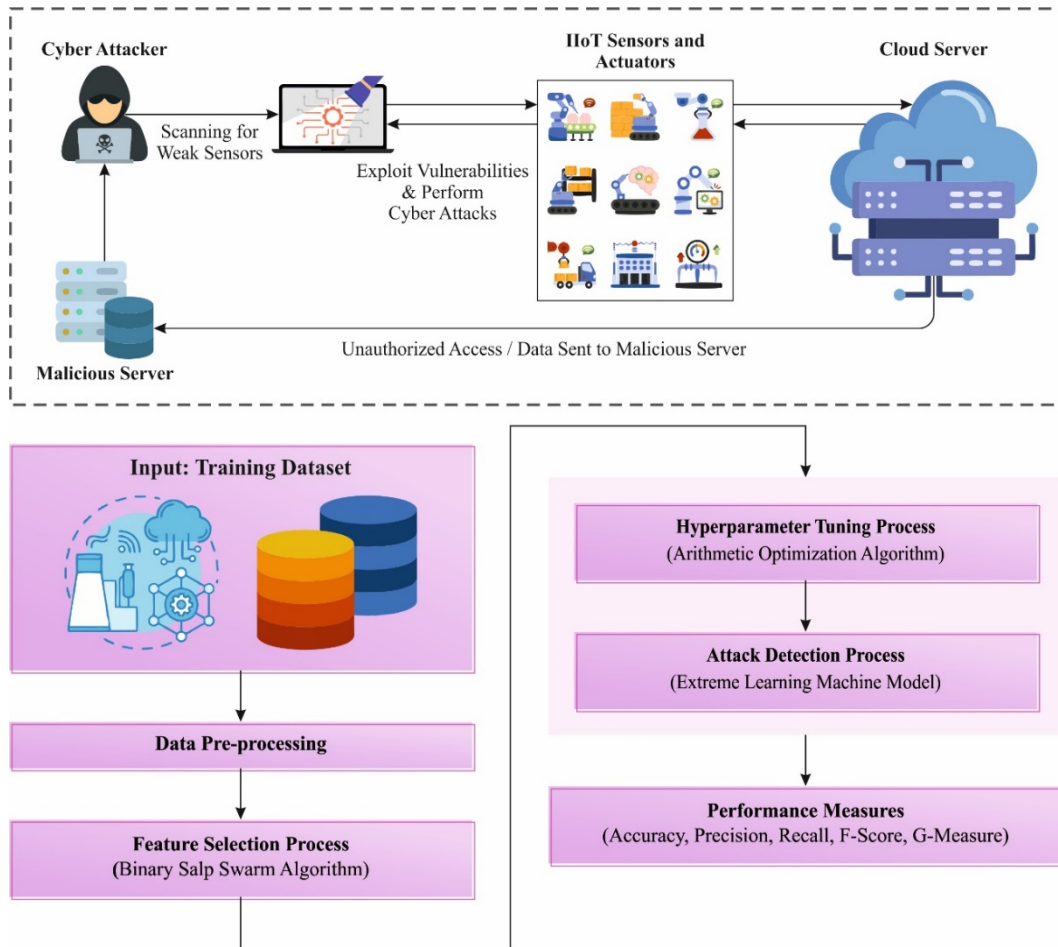


Figure 1: Workflow of ACDIOT-BSSA technique

3.1. Data Preprocessing

At primary phase of data, pre-processing is implemented to alter the input data into relevant format. Data pre-processing utilizing Linear Scaling Normalization (LSN) is essential in cyber-security for attack detection, as it converts values of features into a constant range, usually between 0 and 1. This normalization certifies that every feature pays similarly to the recognition method, averting any distinct feature from controlling the analysis. By standardizing the data, LSN improves the model's capability to exactly detect potential and anomaly threats.

3.2. Feature Selection Process

Next, the proposed ACDIOT-BSSA approach achieves feature selection progress utilizing the BSSA algorithm. Mirjalili et al. presented the SSA based on the group behavior of salps in the ocean [16]. In SSA, the swarm of salps forages and moves in a chain structure, and the leader and follower are two different roles of salps. The individual at the forefront of the sales chain serves as a leader, whereas the others serve as followers. The leaders lead the salp chain direction, whereas the follower follows the preceding leader. Thereby, the leader explores the food source, and follower moves to the leader. This enables the salp chain to have stronger local

exploitation and global exploration capabilities. Similar to other swarm-based techniques, the salp position can be described by a d -dimensional vector, where d is the dimension number of optimization problems. Next, the swarm of salps is described by a $N \times d$ matrix, where N is the size of swarm.

$$X = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^n & x_2^n & \cdots & x_d^n \end{bmatrix} \quad (1)$$

The leader used to update the location by using the following expression

$$x_j^1 = \begin{cases} F_j + c_1 \left((ub_j - lb_j)c_2 + lb_j \right) & \text{if } c_3 \geq 0.5 \\ F_j - c_1 \left((ub_j - lb_j)c_2 + lb_j \right) & \text{else} \end{cases} \quad (2)$$

In Eq. (2), x_j^1 is the j^{th} dimension vector of first salp position, viz., leader of salps. F denotes the food source position. c_1 is a crucial parameter. It has the function of balancing exploration and exploitation capability of SSA. c_2 and c_3 , that define the stepsize and movement direction of the leader, correspondingly, are two randomly generated values within $[0,1]$. ub_j and lb_j are the upper and the lower boundaries of the j^{th} dimension, correspondingly.

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \quad (3)$$

Where l and L are the existing and the maximum iteration count.

A follower is used to update the location using the following equation:

$$x_j^i = \frac{1}{2}(x_j^i + x_j^{i-1}) \quad (4)$$

In Eq. (4), x_j^i is the j^{th} dimension vector of i^{th} follower salp position.

The SSA was initially introduced to resolve the optimization problems. Meanwhile, FS is a discrete optimization problem, and SSA could not efficiently handle it. To overcome these issues, BSSA was introduced. In BSSA, the component of position vector should be mapped to 0 or 1 after all the iterations. The mapping model of location vector is given below:

$$S(x_j^i) = \frac{1}{1 + \exp^{-x_j^i}} \quad (5)$$

$$newx_j^i = \begin{cases} 1 & \text{if } rand \geq S(x_j^i) \\ 0 & \text{else} \end{cases} \quad (6)$$

Where x_j^i is the j^{th} dimension vector of location representing i^{th} salp; $rand$ shows the uniformly distributed random number within $[0,1]$; sigmoid function S is the possibility of choosing a candidate features; $newx_j^i$ indicates the j^{th} vector dimension of i^{th} salp position.

The fitness function (FF) examines the classifier precision and the chosen feature numbers. It maximizes the classification precision and minimizes the chosen feature set sizes. Hence, the subsequent FF is utilized to compute a particular solution, as presented in Eq. (7).

$$Fitness = \alpha * ErrorRate + (1 - \alpha) * \frac{\#SF}{\#All_F} \quad (7)$$

Where *Error Rate* indicates the classifier error rate utilizing the chosen features. *ErrorRate* Is computed as the ratio of wrong classified to the number of classifiers produced, represented as a value among 0 and 1. (*ErrorRate* Is complementary of the classifier precision),

$\#SF$ indicates the amount of chosen features and $\#All_F$ indicates the entire amount of features in the novel dataset. α is utilized to manage the significance of classifier subset length and quality. In our experimentations, the value α is 0.9.

3.3. Attack Detection using ELM Classifier

For attack recognition, the ACDIIOT-BSSA method uses ELM technique. The proposed ELM technique aims to resolve the slower training problems with classical FFNN model [17]. The slow training problems are tracked back to the iterative training owing to its gradient-based learning algorithm. Rather than training the network through iterative training, ELM arbitrarily selects the nodes in the HL of single hidden layer feedforward neural network (SHFN) and later defines the output weight through the analysis. Thus, the training time can be considerably decreased while providing better generalizability, though the architecture of NN remains the same.

Consider (x_i, y_i) , a set of observable samples X and the expected output Y , thus $x_i = [x_{i1}, \dots, x_{iu}]^T \in \mathbb{R}^u$ and $y_j = [y_{j1}, \dots, y_{jv}]^T \in \mathbb{R}^v$. N is the number of the observations, $s(x)$ is the activation function, and h is the amount of hidden nodes.

$$o_j = \sum_{i=1}^h \alpha_i s(w_i \cdot x_j + th_i) \quad (8)$$

In Eq. (8), o_j refers to the output of j^{th} nodes at the output layer, $j = 1, \dots, N$, $w_i = [w_{i1}, \dots, w_{iu}]^T$ indicates the weight vectors between i^{th} hidden nodes and the input nodes. $\alpha_i = [\alpha_{i1}, \dots, \alpha_{im}]^T$ and th_i are the threshold values of i^{th} hidden nodes. The symbol \cdot denotes the inner product of w_i and x_i . The SHFN can calculate the desired output of N samples with zero means using

$$y_j = \sum_{i=1}^h \alpha_i s(w_i \cdot x_j + th_i) \quad (9)$$

$$Y = B\alpha,$$

Where B refers to the output matrix of HL.

$$E = \sum_{j=1}^N \left(\sum_{i=1}^h \alpha_i s(w_i \cdot x_j + th_i) - y_j \right)^2 \quad (10)$$

The parameters such as W , the vector form of w_i , α , and b are updated iteratively to reduce the error from gradient-based algorithm,

$$W_n = W_{n-1} - \beta \frac{\partial E(W)}{\partial W} \quad (11)$$

Where β refers to the learning rate. Usually, Backpropagation is utilized as a learning model thus errors are forwarded back to parameter optimization. If β is small, then it takes long time for the learning model to be converged. On the other hand, a large β might result in divergence or instability. Other perplexing issues are gradient-based learning and Local minima. The dissimilarity between backpropagation and ELM algorithms lies mainly in the technique for updating parameters. For ELM, the neuron count in the HL is the primary factor that defines the ELM performance.

3.4. Hyperparameter Optimizer

Finally, the AOA is deployed as a hyperparameter optimizer for the ELM method. The AOA is a new metaheuristic algorithm based on the statistical properties of the four basic arithmetical operators such as multiplication (M), division (D), subtraction (S), and addition (A) [18]. The two processes that constitute optimization algorithms in AOA are exploitation and exploration of mathematical modeling of AOA. The hierarchy of arithmetical operations together with the domination from the external to inside. The Math Optimizer Accelerated (MOA) operator is a coefficient in the search process.

$$MOA(C_{Iter}) = \text{Min} + C_{Iter} * \left(\frac{\text{Max} - \text{Min}}{M_{Iter}} \right) \quad (12)$$

Where $MOA(C_{Iter})$ is the function value at t^{th} iterations. C_{Iter} is the existing iteration, ranging from 1 to the maximal value. “ Min ” and “ Max ” are the minimal and maximal values correspondingly.

The exploration operator of AOA explores the search region randomly on different approaches and areas to search for the best solution according to the (D) and (M) search strategies.

$$x_{ij}(C_{Iter} + 1) = \left\{ \begin{array}{l} \frac{\text{best}(x_j)}{MOP + \epsilon} * ((UB_j - LB_j) * \mu + LB_j), r_2 < 0.5 \\ \text{best}(x_j) * MOP * ((UB_j - LB_j) * \mu + LB_j), \text{otherwise} \end{array} \right\} \quad (13)$$

$x_{ii}(C_{Iter} + 1)$ is the i^{th} solution in i^{th} position at the existing iteration, and (x_j) is the j^{th} location in the optimum solution. UB_j and LB_j are the upper and lower boundaries of the j^{th} position and is a small integer number. The search process can be transformed by C where it denotes the existing iteration, $Iter$ refers to the setting of the control parameter set as 0.5.

$$MOP(C_{Iter}) = 1 - \frac{C_{Iter}^{1/\alpha}}{M_{Iter}^{1/\alpha}} \quad (14)$$

In Eq. (14), C_{Iter} indicates the existing iteration, (M_{Iter}) shows the maximal iteration number and MOP (Math Optimizer Probability) is a coefficient. $MOP(C_{Iter})$ is the function value at the t^{th} iteration. The delicate parameter α is the exploitation accuracy through the iteration at 5.

The (S) and (A) search strategies are utilized by the exploitation operators of AOA to exhaustively explore the search area in the dense places and approach to search for the best solution.

$$x_{i,j}(C_{Iter} + 1) = \left\{ \begin{array}{l} \text{best}(x_j) - MOP * ((UB_j - LB_j) * \mu + LB_j) r_3 < 0.5 \\ \text{best}(x_j) + MOP * ((UB_j - LB_j) * \mu + LB_j), \text{otherwise} \end{array} \right\} \quad (15)$$

The fitness selection is the significant feature affecting the presentation of the AOA . The hyper-parameter selection model covers the solution encode method to compute the candidate solution efficiency. In this study, the AOA finds precision as the main criterion to develop the FF which could be expressed as follows.

$$\text{Fitness} = \max(P) \quad (16)$$

$$P = \frac{TP}{TP + FP} \quad (17)$$

From the formulation, TP and FP represent true and false positive values respectively.

4. Performance Validation

The experimental validation outcomes of the ACDIIOT-BSSA approach are examined using EdgeIoTset dataset [19]. The dataset comprises 10500 samples under seven class labels defined in Table 1.

Table 1 : Details of dataset

IoT Traffic	Type of Event	Data Record
Normal	Normal	1500
	DDoS-UDP	1500
	SQL-injection	1500
Attack	DDoS-TCP	1500
	Password	1500
	Port-scanning	1500
	Ransomware	1500
Total Number of Record		10500

In Table 2 and Fig. 2, the overall cyberattack detection results of the ACDIIOT-BSSA model under 70%TRAP and 30%TESP are demonstrated. The table values stated that the ACDIIOT-BSSA method can find the samples proficiently. With 70%TRAP, the ACDIIOT-BSSA methodology offers average $accu_y$ of 95.00%, $prec_n$ of 82.53%, $reca_l$ of 82.51%, F_{score} of 82.51%, and $G_{measure}$ of 82.51%. Followed by, with 30%TESP, the ACDIIOT-BSSA technique provides average $accu_y$ of 95.56%, $prec_n$ of 84.47%, $reca_l$ of 84.45%, F_{score} of 84.43%, and $G_{measure}$ of 84.45%.

In Fig. 3, the training and validation accuracy outcomes of the ACDIIOT-BSSA approach can be exhibited. The precision values are calculated for 0-25 epoch counts. This figure emphasized that the training and validation accuracy values display reliable trend that indicated the capability of the ACDIIOT-BSSA method with better performance over numerous iterations. In addition, the training accuracy and validation accuracy stay nearer over the epoch count that denoted less minimum overfitting and shows superior performance of the ACDIIOT-BSSA technique, ensuring continuous prediction on hidden instances.

Table 2

Cyberattack detection outcome of ACDIIOT-BSSA model under 70%TRAP and 30%TESP

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	$G_{Measure}$
-------	----------	----------	----------	-------------	---------------

TRAP (70%)					
Normal	95.52	84.27	84.51	84.39	84.39
DDoS-UDP	95.51	82.99	85.71	84.33	84.34
SQL-Injection	94.65	81.01	81.40	81.21	81.21
DDoS-TCP	95.46	85.64	81.70	83.63	83.65
Password	94.99	82.21	83.89	83.04	83.05
Port-Scanning	94.79	81.78	80.91	81.34	81.35
Ransomware	94.08	79.79	79.42	79.61	79.61
Average	95.00	82.53	82.51	82.51	82.51
TESP (30%)					
Normal	95.65	83.73	86.16	84.93	84.94
DDoS-UDP	96.03	85.84	87.50	86.66	86.66
SQL-Injection	95.68	85.91	84.03	84.96	84.96
DDoS-TCP	96.92	90.34	88.16	89.23	89.24
Password	94.76	78.81	83.80	81.23	81.27
Port-Scanning	95.21	86.27	80.56	83.31	83.36
Ransomware	94.70	80.41	80.97	80.69	80.69
Average	95.56	84.47	84.45	84.43	84.45

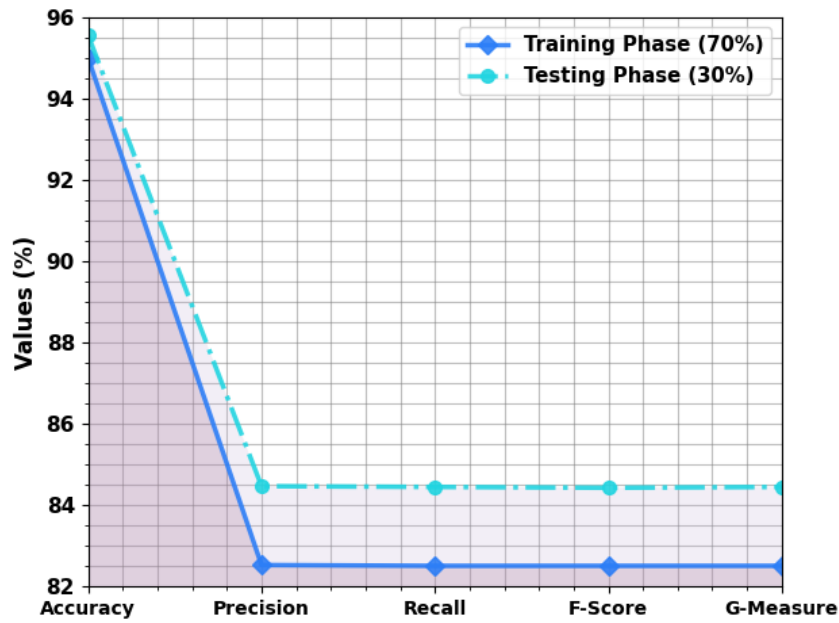


Figure 2: Average outcome of ACDIIOT-BSSA model under 70%TRAP and 30%TESP

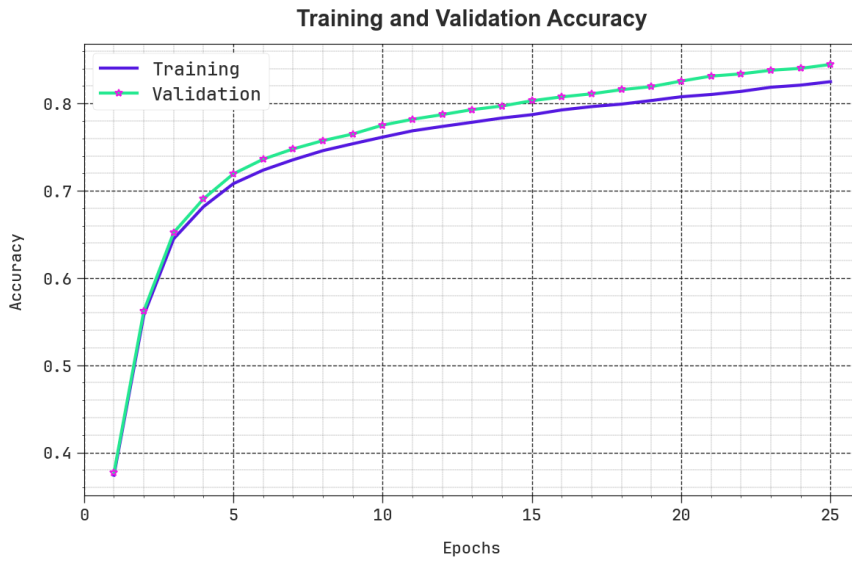


Figure 3: $Accu_y$ curve of ACDIIOT-BSSA model

In Fig. 4, the training and validation loss graph of the ACDIIOT-BSSA technique was demonstrated. The loss values are calculated for 0-25 epoch counts. It is depicted that the training and validation accuracy values indicated a reducing trend that announced the capacity of the ACDIIOT-BSSA technique to balance a trade-off between generalization and data fitting. The consistent decrease in loss values also assures the better performance of the ACDIIOT-BSSA approach and tuning of the prediction outcomes on time.

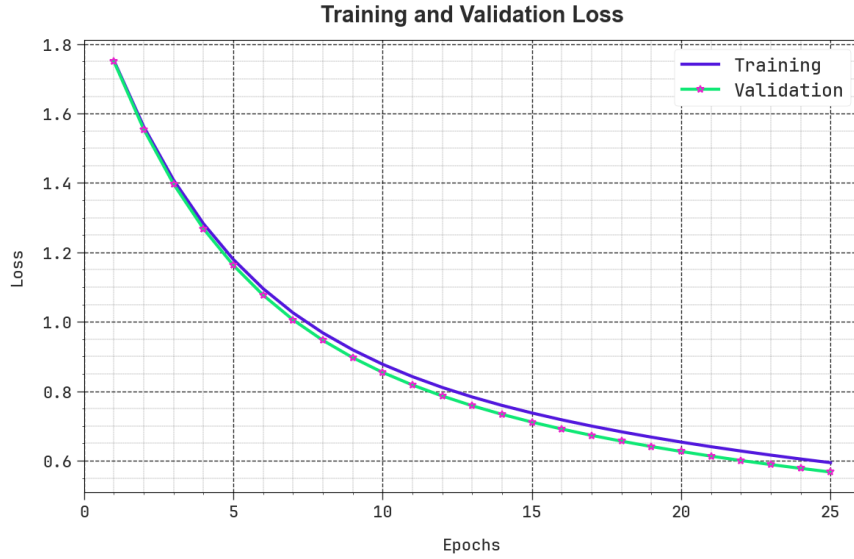


Figure 4: Loss curve of ACDIIOT-BSSA model

In Table 3 and Fig. 5, an overall comparative analysis of the ACDIIOT-BSSA approach is noticeably portrayed [20] compared with recent techniques [21-22]. The outcomes depicted that the RF, SVM, and KNN techniques have demonstrated ineffectual recognition outcomes with least $accu_y$ of 80.83%, 73.01%, and 69.33%, respectively. Meanwhile, the DNN methodology has displayed significant performance with $accu_y$ of 94.67%, $prec_n$ of 75.81%, $reca_l$ of 73.80%, and F_{score} of 70.08%. In addition, the Inception time technique has successfully performed reasonable results with $accu_y$ of 94.94%, $prec_n$ of 70.24%, $reca_l$ of 74.20%, and F_{score} of 68.27%. Lastly, the ACDIIOT-BSSA method exhibits better performance with improved $accu_y$ of 95.56%, $prec_n$ of 84.47%, $reca_l$ of 84.45%, and F_{score} of 84.43%. Therefore, the ACDIIOT-BSSA approach was used for superior cyberattack recognition in the IIoT environment.

Table 3

Comparison outcome of ACDIIOT-BSSA method with other models

Model	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}
Random Forest	80.83	73.01	69.33	77.72
SVM Model	77.61	76.66	76.31	77.63
KNN Classifier	79.18	71.33	75.04	75.09
DNN Algorithm	94.67	75.81	73.80	70.08
Inception Time	94.94	70.24	74.20	68.27

ACDIOT-BSSA	95.56	84.47	84.45	84.43
-------------	-------	-------	-------	-------

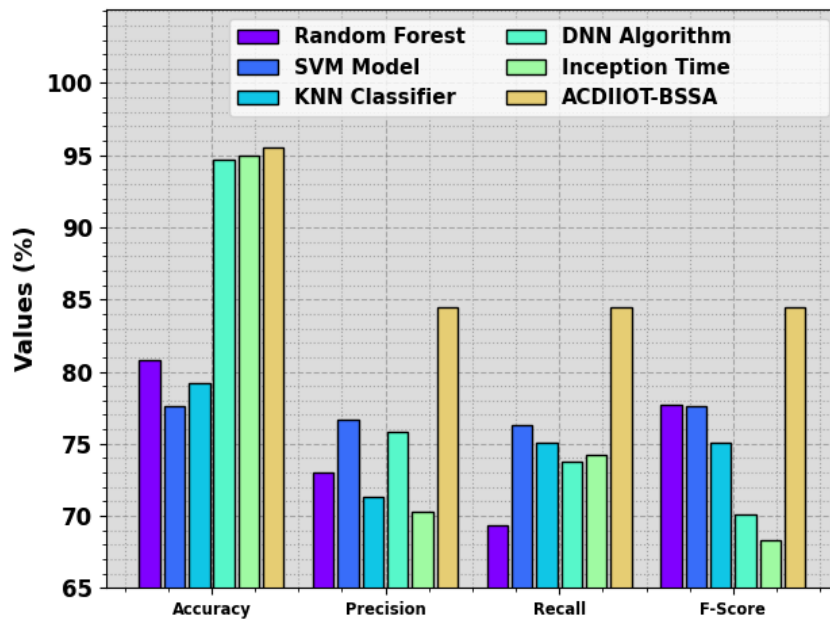


Figure 5: Comparison outcome of ACDIOT-BSSA model with other models

5. Conclusion

In this study, we have developed a novel ACDIOT-BSSA technique. The projected ACDIOT-BSSA method mainly addresses the classification and identification of attack recognition in achieving cyber security. The first phase of data pre-processing is implemented to alter the input data into the relevant format. Next, the proposed ACDIOT-BSSA approach achieves feature selection progress utilizing the BSSA algorithm. For attack recognition, the ACDIOT-BSSA method uses ELM technique. Finally, the AOA is deployed as a hyperparameter optimizer for the ELM method. To inspect the improved performance of the proposed ACDIOT-BSSA approach, a wide range of experiments were done. The empirical findings reported a better outcome of the ACDIOT-BSSA method over other existing techniques

References

- [1] N. Mhaisen, N. Fetais, A. Massoud, Secure smart contract-enabled control of battery energy storage systems against cyber-attacks, *Alexandria Engineering Journal*, 58.4, (2019) 1291-1300.
- [2] J. Zhang, L. Pan, Q.L. Han, C. Chen, S. Wen, Y. Xiang, Deep learning based attack detection for cyber-physical system cybersecurity: A survey, *IEEE/CAA Journal of Automatica Sinica*, 9.3, (2021) 377-391.

- [3] M. Dehghani, T. Niknam, M. Ghiasi, N. Bayati, M. Savaghebi, Cyber-attack detection in dc microgrids based on deep machine learning and wavelet singular values approach, *Electronics*, 10.16, (2021) 1914.
- [4] E. B. Ashary, L. A. Maghrabi, S. Jambi, R. B. Ashari, A. G. Fayoumi, A. S. Al-Ghamdi, M. Ragab, Enhancing Resilience in Next-Generation Wireless Networks Through Deep Learning for Security Enhancement, *IEEE Transactions on Consumer Electronics*. 2024
- [5] O. Ajayi, M. Cherian, T. Saadawi, Secured Cyber-Attack Signatures Distribution using Blockchain Technology, 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), New York, NY, USA, (2019) 482-488.
- [6] V. Kelli, Sarigiannidis, V. Argyriou, T. Lagkas, V. Vitsas, A Cyber Resilience Framework for NG-IoT Healthcare Using Machine Learning and Blockchain, ICC 2021 - IEEE International Conference on Communications, Montreal, QC, Canada, (2021) 1-6.
- [7] A. O. Khadidos, Z. M. AlKubaisy, A. O. Khadidos, K. H. Alyoubi, A. M. Alshareef, M. Ragab, Binary Hunter Prey Optimization with Machine Learning Based Cybersecurity Solution on Internet of Things Environment, *Sensors*, *Sensors*, 23. 16 (2023) 7207.
- [8] R. M. A. Ujjan, Z. Pervez, K. Dahal, Snort Based Collaborative Intrusion Detection System Using Blockchain in SDN, 2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), Island of Ulkulhas, Maldives, (2019) 1-8.
- [9] M. Ghiasi, M. Dehghani, T. Niknam, A. Kavousi-Fard, P. Siano, H.H. Alhelou, Cyber-attack detection and cyber-security enhancement in smart DC-microgrid based on blockchain technology and Hilbert Huang transform, *IEEE Access*, 9, (2021) 29429-29440.
- [10] O. Ajayi, T. Saadawi, Blockchain-Based Architecture for Secured Cyber-Attack Features Exchange, 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), New York, NY, USA, (2020) 100-107.
- [11] Sagu, N. S. Gill, Gulia, I. Priyadarshini, J. M. Chatterjee, Hybrid Optimization Algorithm for Detection of Security Attacks in IoT-Enabled Cyber-Physical Systems, in *IEEE Transactions on Big Data*, doi: 10.1109/TBDATA.2024.3372368.
- [12] F. Li, J. Lin, H. Han, FSL: federated sequential learning-based cyberattack detection for Industrial Internet of Things, *Industrial Artificial Intelligence*, 1.1, (2023) 4.
- [13] D. Durairaj, T.K. Venkatasamy, A. Mehbodniya, S. Umar, T. Alam, Intrusion detection and mitigation of attacks in microgrid using enhanced deep belief network, *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, 46.1, (2024) 1519-1541.
- [14] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrou, Y. Farhaoui, An ensemble learning based intrusion detection model for industrial IoT security, *Big Data Mining and Analytics*, 6.3, (2023) 273-287.
- [15] Y.N. Kunang, S. Nurmaini, D. Stiawan, B.Y. Suprpto, An end-to-end intrusion detection system with IoT dataset using deep learning with unsupervised feature extraction, *International Journal of Information Security*, (2024) 1-30.
- [16] B. Wei, X. Jin, L. Deng, Y. Huang, H. Wu, Feature selection via a multi-swarm salp swarm algorithm, *Electronic Research Archive*, 32.5, (2024) 3588-3617.
- [17] X. Yu, Z. Ren, D.S. Guttery, Y.D. Zhang, DF-dRVFL: A novel deep feature based classifier for breast mass classification, *Multimedia Tools and Applications*, 83.5, (2024) 14393-14422.

- [18] H. Abdelfattah, A.O. Aseeri, M. Abd Elaziz, Optimized FOPID controller for nuclear research reactor using enhanced planet optimization algorithm, Alexandria Engineering Journal, 97, (2024) 267-282.
- [19] <https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cybersecurity-dataset-of-iiot>
- [20] Tareq, B.M. Elbagoury, S. El-Regaily, E.S.M. El-Horbaty, Analysis of ton-iiot, unw-nb15, and edge-iiot datasets using dl in cybersecurity for iiot, Applied Sciences, 12.19, (2022) 9572.
- [21] I. Katib, M. Ragab, Blockchain-assisted hybrid harris hawks optimization based deep DDoS attack detection in the IoT environment, Mathematics, 11.8 (2023) 1887.
- [22] L. A. Maghrabi, I.R. Alzahrani, D. Alsalman, Z. M. AlKubaisy, D. Hamed, M. Ragab. Golden Jackal Optimization with a Deep Learning-Based Cybersecurity Solution in Industrial Internet of Things Systems, Electronics 2023, 12(19), 4091.