AIMS Mathematics, 10(7): 16371–16392.
DOI: 10.3934/math.2025732
Received: 15 April 2025
Revised: 28 May 2025
Accepted: 09 June 2025
Published: 21 July 2025

https://www.aimspress.com/journal/Math

Research article

Blockchain-driven cybersecurity detection model using quantum bidirectional gated recurrent unit with metaheuristic algorithms on IoT environment

Abeer A. K. Alharbi*

Department of Information Systems, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11432, Saudi Arabia

* **Correspondence:** Email: AAkalharbi@imamu.edu.sa.

Abstract: The fast development of the Internet of Things (IoT) has led to a rising number of interrelated devices, creating novel chances for data automation and collection. This growth also presents unique cybersecurity threats. In IoT devices, cyberattacks can result in serious consequences such as unauthorized access, data breaches, and disruption of critical services. Thus, numerous research surveys have concentrated on the design of intrusion detection systems (IDS). Furthermore, blockchain (BC) technology is implemented to enhance security. BC delivers a robust device for securing data and transactions of IoT systems. By leveraging BC's technology, IoT systems benefit from improved privacy and security. In this study, I proposed a Blockchain Driven Cybersecurity Detection Using Metaheuristic Optimization Algorithms and a Deep Learning Model (BCCD-MOADLM). My main intention of the BCCD-MOADLM model was to detect and classify cybersecurity in an IoT environment using advanced optimization models. BC has emerged as a promising solution for improving the privacy and security of IoT networks. Furthermore, the data pre-processing stage applied z-score normalization to transform input data into a compatible format. The Osprey optimization algorithm (OOA) technique was employed for the dimensionality reduction. Moreover, the quantum bidirectional gated recurrent unit (QBiGRU) technique was used for cybersecurity classification. The parameter selection for QBiGRU was performed using the Pelican optimizer algorithm (POA) technique. The experimental evaluation of the BCCD-MOADLM approach was examined under the BoT-IoT dataset. The comparison outcomes of the BCCD-MOADLM approach portrayed a superior accuracy value of 99.32% over existing models.

Keywords: blockchain; cybersecurity; Internet of Things; Pelican Optimization Algorithm; feature selection; gated recurrent unit
Mathematics Subject Classification: 37H99, 92D20

1. Introduction

The IoT relates to a network of connected sensors, systems, and gadgets that exchange and communicate with data to enable multiple services and applications. The IoT has obtained considerable momentum recently and is being widely accepted through several regions comprising smart cities, smart homes, transportation, healthcare, agriculture, industrial automation, and so on [1]. Nevertheless, the development of IoT gadgets has also presented novel tasks in the context of cyberattacks. Owing to their prevalence and different applications, IoT gadgets are frequently aimed at cybercriminals looking to leverage vulnerabilities in these methods [2]. Cyber threats on IoT gadgets may lead to substantial impacts, like disruption, data breaches, and unauthorized access to essential services. Thus, safeguarding secure communication and system resilience is paramount, as data integrity and any security breach may have prevalent results affecting many aspects of urban life [3]. Conventional cybersecurity frequently falls short of addressing the unique demands of cognitive cities because of their heterogeneity, dynamic nature, and scale [4]. BC technology has been employed in various valuable fields and incorporated with diverse evolving technologies to offer trustworthy and secure methods [5]. Figure 1 depicts the general structure of BC technology in IoT devices.

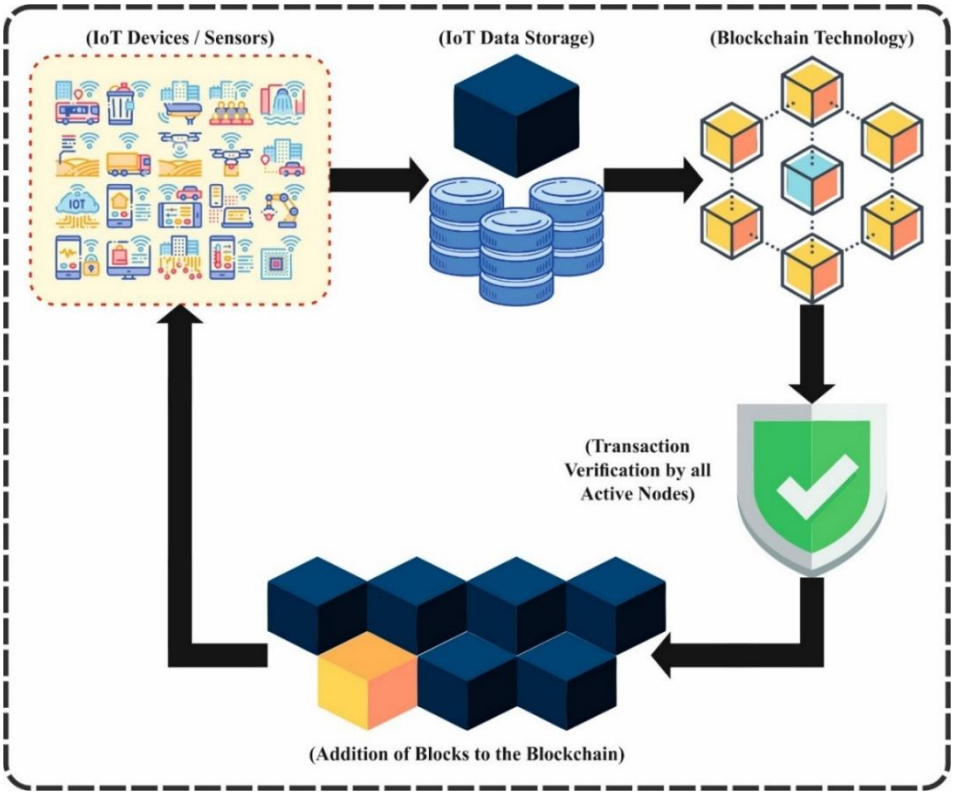

Figure 1. General structure of BC in IoT devices.

BC can significantly improve the safety and confidentiality of data sharing in IoT-assisted smart

grids, enhancing reliability and interoperability in these interconnected settings [6]. BC technology has become a possible solution to improve the confidentiality and safety of IoT methods. BC is a distributed ledger technology that permits transparent, tamper-proof, and secure transactions [7]. By utilizing IoT systems, BC technology capitalizes on better privacy and security. It provides a few benefits contributing to IoT system security [8]. For the last several years, deep learning (DL), artificial intelligence (AI), and machine learning (ML) have been utilized to upgrade IoT security by automated threat recognition [9]. Nevertheless, these models are leveraged by attackers to progress more advanced threats aiming at IoT gadgets. Moreover, organizations may attain timely intervention and proactive threat detection by training DL and ML-based techniques to identify multiple cyber threats [10].

Here, I propose Blockchain Driven Cybersecurity Detection Using Metaheuristic Optimization Algorithms and a Deep Learning Model (BCCD-MOADLM). My main intention of the BCCD-MOADLM model is to detect and classify cybersecurity in an IoT environment using advanced optimization models. BC has emerged as a promising solution for improving the privacy and security of IoT networks. Furthermore, the data pre-processing stage applies z-score normalization to transform input data into a compatible format. The Osprey optimization algorithm (OOA) technique is employed for the dimensionality reduction. Moreover, the quantum bidirectional gated recurrent unit (QBiGRU) technique is used for cybersecurity classification. The parameter selection for QBiGRU is performed using the Pelican optimizer algorithm (POA) technique. The experimental evaluation of the BCCD-MOADLM approach is examined under the BoT-IoT dataset.

- The BCCD-MOADLM model applies z-score normalization to standardize the data, improving the quality and consistency of input features. This pre-processing step assists in mitigating the impact of outliers and scales the data for enhanced training efficiency and model accuracy in cybersecurity classification.
- The BCCD-MOADLM method incorporates the OOA model to perform effectual feature selection and dimensionality reduction, which assists in eliminating irrelevant and redundant data. This process enhances computational efficiency and the overall performance of the cybersecurity classification system.
- The BCCD-MOADLM technique employs the QBiGRU technique to capture intrinsic temporal dependencies in cybersecurity data, enabling more accurate threat classification. This approach utilizes quantum-inspired techniques to improve learning capabilities and detection performance.
- The BCCD-MOADLM methodology implements the POA method for efficient hyperparameter tuning, which optimizes the learning process and enhances overall model accuracy. This optimization ensures the model adapts effectively to complex cybersecurity data patterns, improving detection reliability.
- The novelty of the BCCD-MOADLM method is integrating the quantum-inspired QBiGRU model with metaheuristic optimization algorithms such as OOA for feature selection and POA for hyperparameter tuning. This integrated approach effectively improves the model's capability of capturing intrinsic patterns and adapting to diverse cybersecurity threats. As a result, it achieves superior detection accuracy and robustness compared to conventional methods.

2. Literature

Ntizikira et al. [11] developed an innovative method for IoT security utilizing CCTV cameras and detection of emotion intrusion prevention and detection. Emotions are identified employing CNN. Depending on identified emotions, irregularities are categorized based on pre-defined standards:

Individuals meeting circumstances are classified, and some conditions are labeled suspicious and deliberated as normal. Dahiya et al. [12] projected an innovative BC-based security method structure that was particularly intended for the medical care sector employing IoT. By utilizing BC technology's transparency, immutability, and inherent security, the projected structure gives a strong solution to secure medical care data integrity and confidentiality. It enables secured patient data sharing between authorized existences. Dhanushkodi et al. [13] developed the bidirectional Gaussian hummingbird-optimized end-to-end BC (BGHO-E2EB) technique to classify and detect cyber threats in IoT settings. The presented method incorporates BC technology over Ethereum-based smart contracts to upgrade the integrity and security of data replacements in IoT systems. Furthermore, a Gaussian artificial hummingbird algorithm (GAHA) model is utilized for optimum feature selection (FS), reducing computational load and data dimensionality. The Bi-LSTM technique enhances the model's ability by precisely categorizing and detecting cyber-attacks depending on selected features. The Adam optimizer is employed for effective parameter tuning in the Bi-LSTM technique, guaranteeing better performance in recognizing cyber threats. Akbar et al. [14] projected a complete cyber-security trust method, which gives multi-risk protection for securing data transmission in CC, guaranteeing a higher degree of data security and privacy. The Cybersecurity Trust technique utilizes BC to determine a tamper-resistant and transparent ledger of each data transaction in the cloud. The multi-risk protection model combines MAES or QKD to give multi-layered defense mechanisms. MATLAB is utilized to organize rigorous experiments. Archana et al. [15] projected the BC-driven clinical image encryption model employing a reformed honey badger optimizer with the ensemble chaotic methods. These highly secured data are stored in BC, safeguarding the image security in edge nodes. Pichandi et al. [16] introduced an innovative method for upgrading system security depending on malicious activity recognition employing ML and a data-driven structure. Reinforcement reward shaping analysis and Gaussian symmetric encryption are employed in the cloud IoT system examination of malicious activities. Thus, the cloud IoT structure-based network security analysis depends on BC cryptanalysis.

Alkhonaini et al. [17] presented a new sandpiper optimizer with a hybrid DL-based ID (SPOHDL-ID) from a BC-aided IoT framework. The projected SPOHDL-ID model, FS from the system traffic data, employs the SPO method. Moreover, the presented model applies the HDL technique for IDS with the help of the CNN-SAE method. BSOA is employed for the parameter tuning procedure to increase the detection consequences of the CNN-SAE. Rathnamala et al. [18] improved IoT botnet attack detection using a stacked DL model, achieving high accuracy and outperforming existing methods. Rai et al. [19] improved the security and privacy of nuclear energy applications by integrating BC with IoT, addressing key challenges, presenting a secure data management framework, and exploring future directions like scalability and post-quantum cryptography. Sakthivelu and Vinoth Kumar [20] developed an optimized DL method for accurate real-time cyberthreat detection and classification, attaining superior performance over existing methods. Tightiz and Yoo [21] developed a quantum-enhanced, explainable model for accurately predicting monitoring timeframes for permanent pacemaker implantation (PPMI) after transcatheter aortic valve replacement (TAVR), enhancing clinical decision-making through advanced AI and expert integration. Popoola et al. [22] improved IoT intrusion detection by addressing class imbalance using a multi-stage DL method, enhancing the detection of minority-class attacks and overall network security. Hassan, Mohamad, and Muchtar [23] evaluated and classified advanced detection methods for black and grey-hole attacks in MANETs, highlighting the role of ML and optimization techniques in improving network security and resilience. Payyampally, Thampi, and Mukherjee [24] reviewed IDS techniques for IoT security, highlighting recent AI trends, evaluation metrics, datasets, and practical applications to guide users in

choosing effective solutions. Kumar et al. [25] improved botnet attack detection in IoT using a stacked DL technique, achieving high accuracy and outperforming existing methods to improve cybersecurity. Nazir et al. [26] optimized ML and hybrid neural models for accurate IoT threat detection across datasets, enhancing security and guiding future work. AboulEla et al. [27] aimed to review and categorize AI-based cybersecurity approaches for the Internet of Medical Things (IoMT), concentrating on techniques, datasets, evaluation metrics, and emerging research trends.

Despite significant improvements, several limitations and research gaps exist. Many models encounter high computational complexity and scalability challenges in resource-constrained IoT environments. Handling class imbalance and evolving attack patterns is inadequate, affecting detection accuracy. Integration of BC with IoT lacks standardized frameworks for interoperability and real-time processing. Furthermore, most studies emphasize accuracy but overlook the explainability and interpretability of AI-driven IDS. A research gap exists in validating these techniques on diverse, real-world datasets under dynamic network conditions. Addressing these gaps is significant for developing robust, efficient, and practical cybersecurity solutions for IoT and related domains.

3. Proposed methodology

In this study, I develop a BCCD-MOADLM model. My main intention of the model is to detect and classify cybersecurity in an IoT environment using advanced optimization models. To accomplish that, the proposed BCCD-MOADLM approach involves various stages, such as BC technology, data pre-processing, FS, classification, and a parameter tuning model. Figure 2 represents the overall workflow of the BCCD-MOADLM approach.

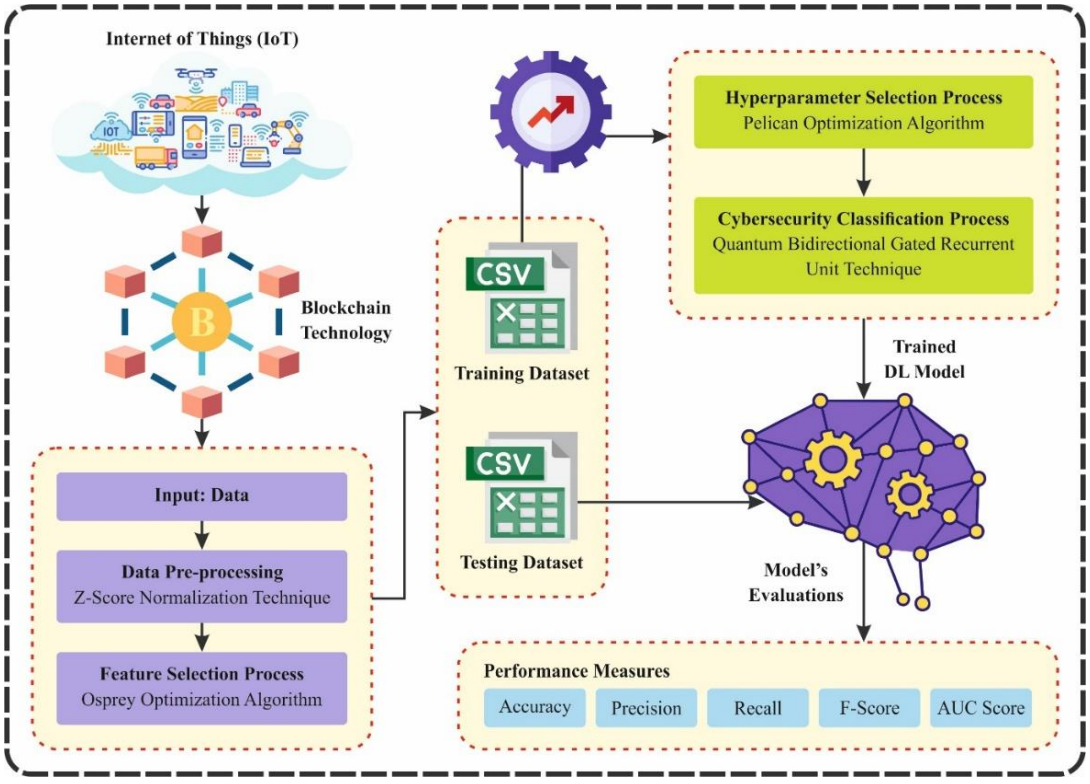

Figure 2. Overall working process of the BCCD-MOADLM approach.

3.1. BC technology

First, BC technology was developed as a probable solution for improving the privacy and security of IoT networks [28]. BC comprises a decentralized distributed dataset method supported by each node in the BC. Consisting of a data block series produced utilizing cryptographic methods, they all act as blocks inside the BC. The data chain was designed by systematically connecting blocks in the order of their group. This hash value acts as a unique identifier for all blocks in the BC, permitting the recognition of the linked block inside the BC over the hash value of its parent block, verified in the block header. This chain is built over a sequence of hash links among all blocks and its parent block. Therefore, the BC is capably tailored to create decentralized networks while also giving resilience against tampering. Furtherm, cryptography technology is incorporated to guarantee security.

3.2. Z-score normalization

Next, the data pre-processing stage applies z-score normalization to transform input data into a compatible format [29]. This model is chosen for its efficiency in standardizing data by converting features to have a mean of zero and a standard deviation of one, which assists in mitigating bias caused by varying scales across features. This technique is highly efficient in real-world cybersecurity and less sensitive to outliers. This method ensures that all features contribute equally during model training, improving convergence speed and stability. Furthermore, this model is simple to use and appropriate for massive datasets usually seen in IoT and cybersecurity applications. Overall, it improves performance by maintaining the underlying data distribution while normalizing feature scales effectively.

It is a statistical model generally employed in data pre-processing for cybersecurity recognition in IoT networks. It standardizes the data by transmuting it into a distribution with a standard deviation (SD) of one and a mean of zero, which removes the bias produced by features with fluctuating measures. In the framework of IoT security, Z-score normalization aids in enhancing the performance of ML techniques, particularly when trading with outliers or anomalies that may specify potential cybersecurity attacks. This certifies that features like traffic patterns, sensor data, or network activities are on a similar scale, improving recognition accuracy.

3.3. OOA-based dimensionality reduction process

For the dimensionality reduction process, the proposed BCCD-MOADLM model utilizes OOA [30]. This model was chosen for its robust global search capability and efficient convergence, which helps in effectually selecting the most relevant features from high-dimensional cybersecurity data. Compared to conventional methods like PCA or manual feature selection, OOA dynamically balances exploration and exploitation, mitigating the risk of getting trapped in local optima. This results in enhanced feature subsets that improve model accuracy while mitigating computational complexity. OOA's metaheuristic nature enables it to adapt to complex, nonlinear relationships in data, making it highly appropriate for growing cyber threat patterns. Its flexibility and scalability make it a robust choice for real-time IoT security systems where timely and precise feature selection is critical. Additionally, OOA assists in reducing dimensionality without sacrificing accuracy, resulting in faster convergence and enhanced model performance in dynamic environments.

The OOA is stimulated by the osprey's natural searching behaviors, recognizing the fish's location, hunting, and carrying fish. In OOA, the hunting fish behavior is mathematically shown as the exploration stage, and the carrying fish behavior is expressed as the exploitation stage. This

architecture enables OOA to balance exploitation and exploration and gain the best solution for optimization problems. The particulars of OOA are provided as shown.

3.3.1. Initialization stage

Set the size of the population N and the entire iteration counts T. Initializing the population of solutions

$$X = \{X_1, X_2, \ldots, X_N\}. \tag{1}$$

$X(0) =$ Initializing utilizing randomly generated values inside the described search area in Eq (1).

3.3.2. Fitness evaluation

Calculate the primary population according to the fitness function (FF) F:

$$F_i = F(X_i), \forall i = 1,2, \ldots, N. \tag{2}$$

Iteration method (repeat till convergence or maximal iterations attained). Set iteration $t = 1$.
Stage 1: Updated Fitness Probability (FP): Upgrade the FP of all ospreys according to their fitness value:

$$FP_i = \frac{F_i}{\sum_{j=1}^{N} F_j}, \forall i = 1,2, \ldots, N. \tag{3}$$

Stage 1.1: Choose Fish and Upgrade Osprey location establish the chosen fish SF by ith osprey randomly. Compute the novel location of ith osprey utilizing:

$$X_i^{t+1} = X_i^t + r_1 \cdot (X_{SF} - X_i^t). \tag{4}$$

On the other hand, r_1 refers to randomly generated numbers among $(0,1)$. X_{SF} denotes the chosen fish location. Upgrade the position of the osprey:

$$X_i^{t+1} = X_i^t + r_2 \cdot (X_{best} - X_i^t). \tag{5}$$

Here, r_2 is another randomly generated number among $(0,1)$. X_{best} stands for a present optimal solution from the population.
Stage 2: Upgrade Location for Exploitation: Compute the novel location utilizing other rules according to the exploitation stage:

$$X_i^{t+1} = X_i^t + r_3 \cdot (X_{best} - X_i^t + \varepsilon). \tag{6}$$

Now, r_3 represents random coefficients control exploitation. ε signifies a more minor perturbation feature

3.3.3. Termination check

Check if the iteration counts t attain the maximal T. If not, upgrade the optimal solution:

$$X_{best} = \arg\min_i F(X_i). \tag{7}$$

Set $t = t + 1$ and repeat till $t = T$.

3.3.4. Output the optimum solution

When the end condition is encountered, output the optimal candidate outcome established:

$$X_{best} = Best\ candidate\ after\ T\ iterations. \tag{8}$$

The FF deployed in the OOA model aims to balance the number of chosen features in all the solutions (least) and the accuracy of the classifier (highest) attained by employing these desired features. Equation (9) denotes the FF for evaluating the solutions.

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|C|}. \tag{9}$$

Here, $\gamma_R(D)$ signifies the error rate of the classifier. $|R|$ denotes the cardinality of the specific sub-set, and $|C|$ means the total number of features in the data; α and β are the two parameters that correspond to the significance of classifier quality and sub-set length. $\in$ [1,0] and $\beta = 1 - \alpha$.

3.4. QBiGRU-based classification process

In addition, the QBiGRU is employed for the cybersecurity classification process [31]. This model is chosen for its capability to capture past and future dependencies in sequential cybersecurity data, improving the understanding of complex temporal patterns. This model integrates quantum computing principles, enhancing information processing efficiency and enabling the model to handle high-dimensional data more effectively. This results in faster convergence and better generalization of noisy or imbalanced datasets, which are standard in IoT environments. Additionally, its bidirectional structure enables it to analyze data context from both directions, resulting in more accurate threat classification. Overall, QBiGRU presents an improved detection performance, robustness, and adaptability compared to classical neural network architectures.

The QBi-GRU is a bidirectional QGRU method, while QGRU means quantum GRU by presenting VQC into GRU to transform the input data's intermediary outcome occasionally. Assume x_t refers to the input vector characteristic at tth time; in QGRU, input data x_t and h_{t-1} are connected to the input vector $[h_{t-1}, x_t]$, and its data handling and state of the unit computation are as shown.
Update gate:

$$z_{qt} = \sigma\left(VQC_{update}\left(W_{qz} \cdot \left[h_{qt-1}, x_t\right] + b_{qz}\right)\right). \tag{10}$$

Reset gate:

$$r_{qt} = \sigma(VQC_{reset}(W_{qr} \cdot [h_{qt-1}, x_c] + b_{qr})). \tag{11}$$

Present memory state:

$$\tilde{h}_{\mathrm{qt}} = \tanh\left(VQC_{out}\left(W_{qh} \cdot \left[r_{qt}^{\star} h_{qt-1}, x_t\right] + b_{qh}\right)\right). \tag{12}$$

Last memory:

$$h_{qt} = (VQC\left(1 - z_{qt}\right)^{\star} h_{qt-1} + z_{qt}^{\star}\tilde{h}_{\mathrm{qt}}\right). \tag{13}$$

Here, $*$ and $\cdot$ represent Hadamard and dot product processes, respectively; correspondingly, $VQC_{update}(.)$, $VQC_{reset}(.)$, and $VQC_{out}(.)$ are three generic VQC processes to characterize the managing in the quantum circuits. h_{qt-1}, h_{qt} are preceding. Hidden state (HL), z_{qt}, $and\ r_{qt}$ are

reset and update gates, $\tilde{h}_{\mathrm{qt}}$ denotes data that is required to be upgraded in the present unit, $W_{qh}, W_{qz},$ and W_{qr} and b_{qh}, b_{qz}, and b_{qr} represent three weighted matrices and biases of the current memory state, update gate, and reset gate, respectively, and $\sigma(\cdot)$ and $\tanh(\cdot)$ are dual nonlinear activation functions, as illustrated in Figure 3. Figure 3 depicts the structure of the BiGRU model.

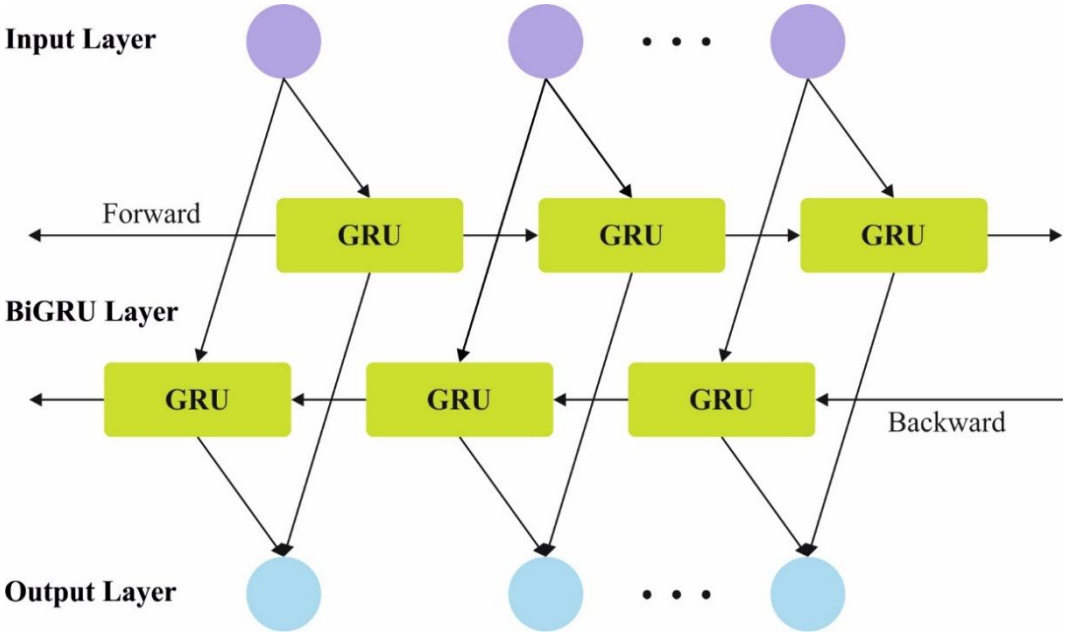

Figure 3. Structure of the BiGRU model.

In Eqs (10)–(13), VQC_{update} and VQC_{reset} are applied to delete and manipulate data, respectively. $[r_{qt} * h_{qt-1}, x_t]$ are handled by the traditional and the VQC_{out} layers to compute the candidate HL $\tilde{h}_{\mathrm{qt}}$. From Eq (13), the final HL h_{qt} is gained due to the output block calculation, comprising z_{qt}, the candidate HL h_{qt}, and the previous HL $\tilde{h}_{\mathrm{qt}}$. The output of each variation layer is the n-dimensional vector akin to Pauli-Z expectancy values of all qubits, while n denotes qubit counts in the quantum circuits, $\sigma(\cdot)$ and $\tanh(\cdot)$; finally, the quantum calculation aimed to enhance the non-linearity that is advantageous to learn composite patterns in the data. The HL of the QBi-GRU component encodes as demonstrated,

$$h_{qt} = [\overrightarrow{h_{qt}} \oplus \overleftarrow{h_{qt}}]. \tag{14}$$

Here, '$\oplus$' is a connecting process, $\overrightarrow{h_{qt}} = QGRU(x_t, \overrightarrow{h_{qt-t}})$ and $\overleftarrow{h_{qt}} = QGRU(x_t, \overleftarrow{h_{qt-1}})$ represent the output of dual QGRUs at tth time and equivalent to h_{qt} in Eqs (10)–(13). The QBi-GRU output is as shown:

$$u_{qdt} = \tanh(W_{qd} h_{qt} + b_{qd}). \tag{15}$$

Now, u_{qgt} denotes the corpus unit; instead of straightforwardly embedding the exposing content as text embedding, W_{qd} and b_{qd} indicate weight and corresponding bias, respectively.

3.5. POA-based hyperparameter tuning model

Finally, the parameter choice of the QBiGRU model is implemented using the POA method [32].

This model is chosen for its efficiency in balancing exploration and exploitation, enabling it to find optimal parameter settings that significantly improve model performance. Compared to conventional tuning methods, such as grid or random search, POA converges faster and avoids getting trapped in local minima, resulting in more accurate and robust models. Its metaheuristic nature enables it to adaptively navigate complex search spaces, which is specifically beneficial for tuning DL methods with various parameters. The technique's simplicity, scalability, and robust optimization capabilities make it appropriate for real-time IoT cybersecurity applications where timely and precise hyperparameter adjustment is crucial for maintaining detection accuracy and reliability.

The POA is a random heuristic optimizer model, which imitates the strategy and behavior of pelicans throughout hunting. Its local calculation performance and searchability are validated using multimodal and unimodal functions. This model uses a population matrix to recognize members of the pelican population, while all rows of the matrix characterize candidate solutions and all columns characterize presented values for the problem variables. At the start of the implementation, each member is required to be initialized randomly, and the objective function (OF) values and population matrix are characterized as shown:

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_1 \\ \vdots \\ X_p \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1q} \\ \vdots & \ddots & \vdots & & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{iq} \\ \vdots & & \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pj} & \cdots & x_{pq} \end{bmatrix} \tag{16}$$

$$Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_i \\ \vdots \\ Y_p \end{bmatrix} = \begin{bmatrix} Y(X_1) \\ \vdots \\ Y(X_i) \\ \vdots \\ Y(X_p) \end{bmatrix}. \tag{17}$$

X refers to the population matrix of the pelican, X_j denotes the pelican, p signifies population member counts, and q represents problem variable counts. Y stands for OF vector, and Y_i denotes a value of the OF of the ith candidate solution. The pelican's feeding procedure primarily contains dual phases: Development and exploration. During the initial phase of feeding, the model mimics the mathematical movement of individual populations to the prey locations, as demonstrated:

$$x_{ij}^{v_1} = \begin{cases} x_{ij} + rand(v_j - I \cdot x_{ij}), & Y_v < Y_i \\ x_{ij} + rand(x_{ij} - p_j), & else \end{cases}. \tag{18}$$

Here, $x_{ij}^{v_1}$ denotes the novel state of an ith pelican in the jth dimension in the exploration stage; $rand$ represents a randomly generated number in the interval of $[0,1]$; I signify an arbitrary integer of 1 or 2; v_j refers to a target location in the jth dimension; and Y_v symbolizes OF value. The implementation procedure is as shown:

$$x'_{ij} = \begin{cases} x_{ij}^{v_1}, & Y_i^{v_1} < Y_i \\ x_{ij}, & else \end{cases}. \tag{19}$$

Now, x'_{ij} signifies the novel state of the ith pelican, and $Y_i^{v_1}$ symbolizes the OF value derived from the order of exploration. In the development stage, the pelican searching procedure is as shown:

$$x_{ij}^{v_2} = x_{ij} + \gamma \left(1 - \frac{\ell}{\delta}\right) \cdot (2 \cdot rand - 1) \cdot t_{ij}. \tag{20}$$

Here, $x_{ij}^{v_2}$ denotes the novel state of an ith pelican in the mining stage; $\gamma = 0.2; \gamma(1 - \ell/\delta)$ designates the locality radius of $x_{ij}; \delta$ indicates the maximal iteration counts; and ℓ refers to the iteration counter. The present location's updated expression is as demonstrated:

$$x_{ij}'' = \begin{cases} x_{ij}^{v_2}, & Y_i^{v_2} < Y_i \\ X_{ij}, & \text{else} \end{cases}. \tag{21}$$

Here, x_{ij}'' symbolizes the novel state of the ith pelican, and $Y_i^{v_2}$ signifies the value of the OF within the development stage. The POA technique originates from an FF to achieve better classification results. It decides a positive number to signify the better outcome of the candidate solution. Here, the reduction of the error ratio of the classifier is measured as FF. Its mathematical formulation is given in Eq (22).

$$fitness(x_i) = ClassifierErrorRate(x_i)$$
$$= \frac{no.of\ misclassified\ samples}{Total\ no.of\ samples} \times 100. \tag{22}$$

4. Experimental analysis

The experimental analysis of the BCCD-MOADLM model is examined under the BoT-IoT dataset [33]. This dataset covers 2056 samples below five classes, as depicted in Table 1. The total number of features presented in this dataset is 10, but only 8 are selected.

Table 1. Details of the BoT-IoT dataset.

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

Figure 4 establishes the confusion matrix formed by the BCCD-MOADLM approach below 80%:20% and 70%:30% of the training phase (TRPH) and testing phase (TSPH), respectively. The outcomes indicate that the BCCD-MOADLM methodology detects and identifies each class specifically.

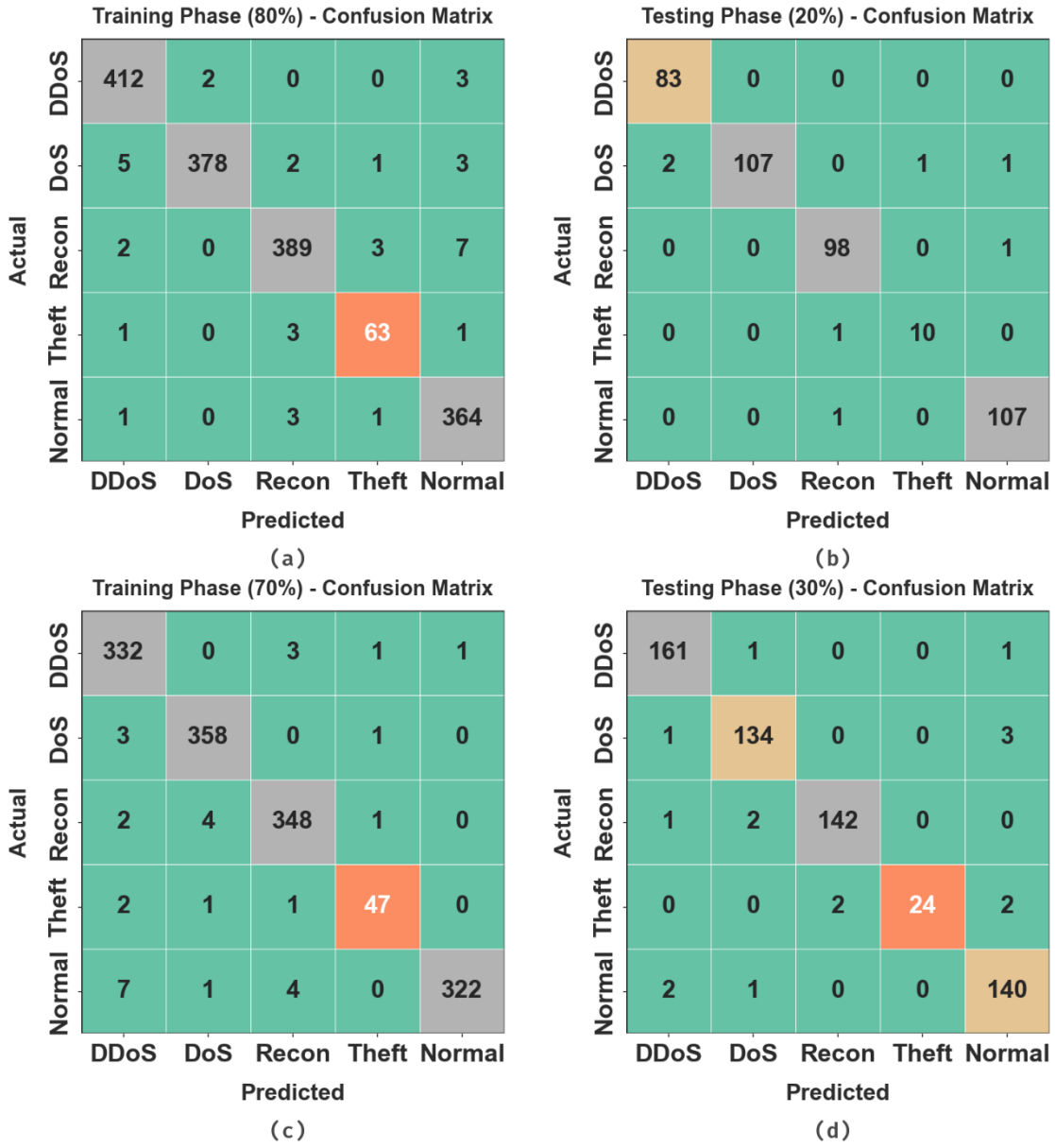

Figure 4. Confusion matrix of the BCCD-MOADLM model. (a-b) 80%TRPH and 20%TSPH. (c-d) 70%TRPH and 30%TSPH.

Table 2 and Figure 5 signify the cybersecurity detection of the BCCD-MOADLM approach below 80%:20% and 70%:30% of TRPH/TSPH, respectively. The results detailed that the BCCD-MOADLM approach has properly classified all the dissimilar class labels. Based on 80% TRPH, the proposed BCCD-MOADLM technique attained an average $accu_y$ of 99.08%, $prec_n$ of 96.85%, $reca_l$ of 96.85%, F_{score} of 96.85%, AUC_{score} of 98.13%, and Kappa of 98.19%. Besides, depending on 20% TSPH, the proposed BCCD-MOADLM technique attained an average $accu_y$ of 99.32%, $prec_n$ of 96.94%, $reca_l$ of 97.07%, F_{score} of 97.00%, AUC_{score} of 98.32%, and Kappa of 98.40%. Moreover, based on 70% TRPH, the proposed BCCD-MOADLM technique achieved an average $accu_y$ of 99.11%, $prec_n$ of 97.15%, $reca_l$ of 96.80%, F_{score} of 96.96%, AUC_{score} of 98.11%, and Kappa of 98.18%. Also, depending on 20% TSPH, the proposed BCCD-MOADLM technique attained an average $accu_y$ of 98.96%, $prec_n$ of 97.84%, $reca_l$ of 95.48%, F_{score} of 96.55%, AUC_{score} of 97.40%, and 97.47%.

Table 2. Cybersecurity detection of BCCD-MOADLM model under 80:20 and 70:30 of TRPH/TSPH, respectively.

Class Labels	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	AUC_{score}	Kappa
TRPH (80%)						
DDoS	99.15	97.86	98.80	98.33	99.03	99.09
DoS	99.21	99.47	97.17	98.31	98.51	98.58
Recon	98.78	97.98	97.01	97.49	98.18	98.23
Theft	99.39	92.65	92.65	92.65	96.16	96.21
Normal	98.84	96.30	98.64	97.46	98.77	98.85
Average	**99.08**	**96.85**	**96.85**	**96.85**	**98.13**	**98.19**
TSPH (20%)						
DDoS	99.51	97.65	100.00	98.81	99.70	99.78
DoS	99.03	100.00	96.40	98.17	98.20	98.28
Recon	99.27	98.00	98.99	98.49	99.18	99.25
Theft	99.51	90.91	90.91	90.91	95.33	95.40
Normal	99.27	98.17	99.07	98.62	99.21	99.27
Average	**99.32**	**96.94**	**97.07**	**97.00**	**98.32**	**98.40**
TRPH (70%)						
DDoS	98.68	95.95	98.52	97.22	98.62	98.69
DoS	99.31	98.35	98.90	98.62	99.17	99.24
Recon	98.96	97.75	98.03	97.89	98.65	98.71
Theft	99.51	94.00	92.16	93.07	95.97	96.02
Normal	99.10	99.69	96.41	98.02	98.16	98.24
Average	**99.11**	**97.15**	**96.80**	**96.96**	**98.11**	**98.18**
TSPH (30%)						
DDoS	99.03	97.58	98.77	98.17	98.95	99.02
DoS	98.70	97.10	97.10	97.10	98.13	98.19
Recon	99.19	98.61	97.93	98.27	98.75	98.83
Theft	99.35	100.00	85.71	92.31	92.86	92.92
Normal	98.54	95.89	97.90	96.89	98.32	98.38
Average	**98.96**	**97.84**	**95.48**	**96.55**	**97.40**	**97.47**

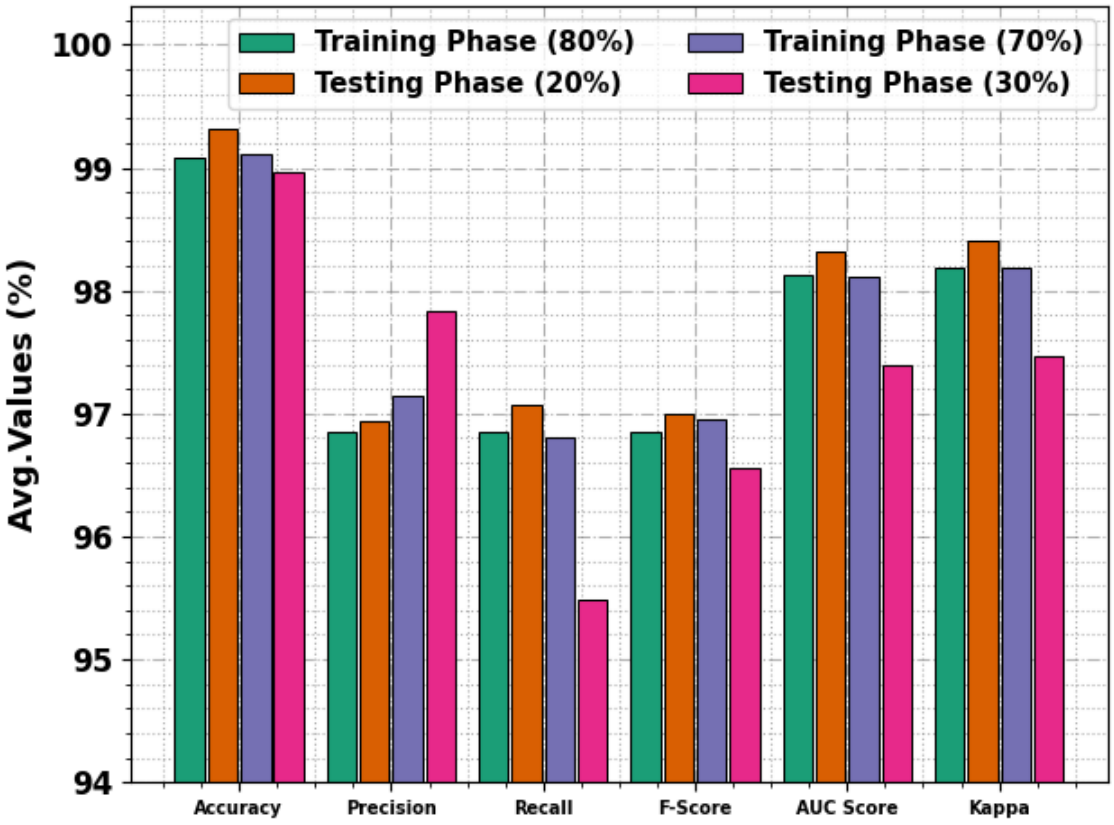

Figure 5. Average of the BCCD-MOADLM model under 80%:20% and 70%:30% of TRPH/TSPH, respectively.

Figure 6 illustrates the training (TRA) $accu_y$ and validation (VAL) $accu_y$ analyses of the BCCD-MOADLM methodology below 80%TRPH and 20%TSPH. The $accu_y$ analysis was computed within the range of 0–25 epochs. The figure highlights that the TRA and VAL $accu_y$ analysis exhibited an increasing trend, which informed the capacity of the BCCD-MOADLM method with superior outcomes across iterations. Furthermore, the TRA and VAL $accu_y$ leftovers were closer across the epochs, which identified inferior overfitting and exhibited maximum performance of the BCCD-MOADLM method, assuring reliable prediction for hidden samples.

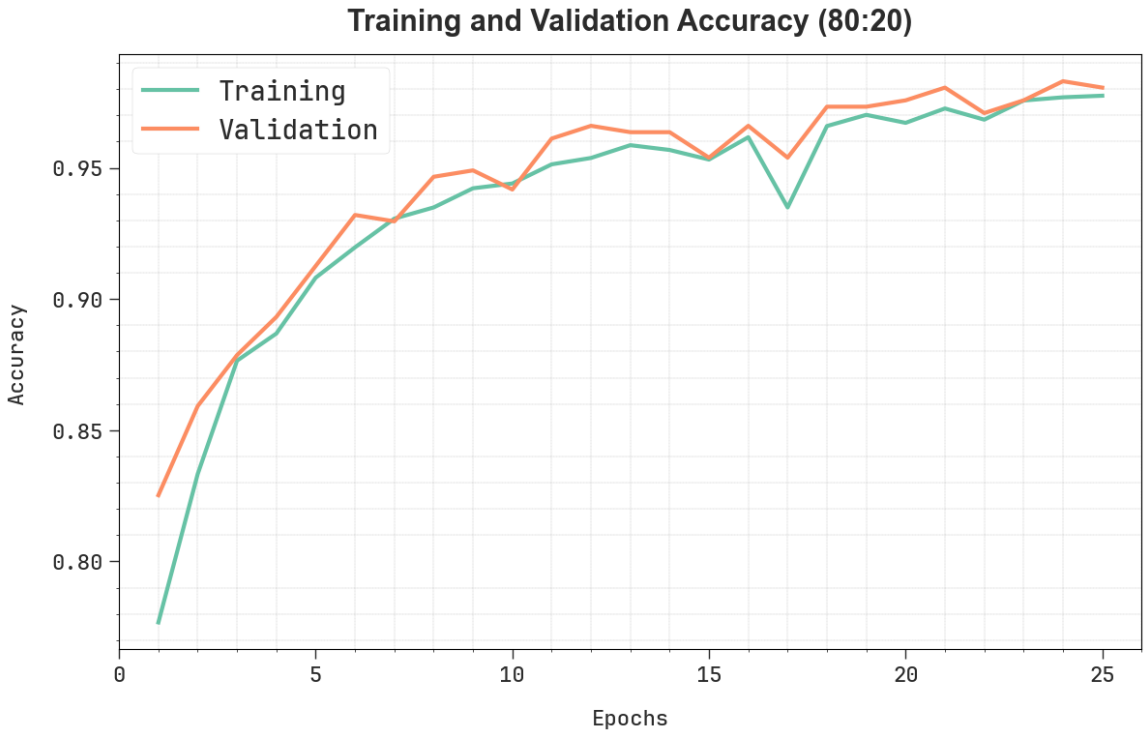

Figure 6. $Accu_y$ analysis of the BCCD-MOADLM method below 80%TRPH and 20%TSPH.

Figure 7 shows the TRA loss (TRALOS) and VAL loss (VALLOS) curves of the BCCD-MOADLM approach below 80%TRPH and 20%TSPH. The loss values were computed across an interval of 0–25 epochs. The TRALOS and VALLOS analyses exemplified a diminishing trend, notifying the capacity of the BCCD-MOADLM technique to balance a trade-off between simplification and data fitting. The constant reduction in loss values guaranteed the higher outcome of the BCCD-MOADLM technique and tuned the prediction results over time.

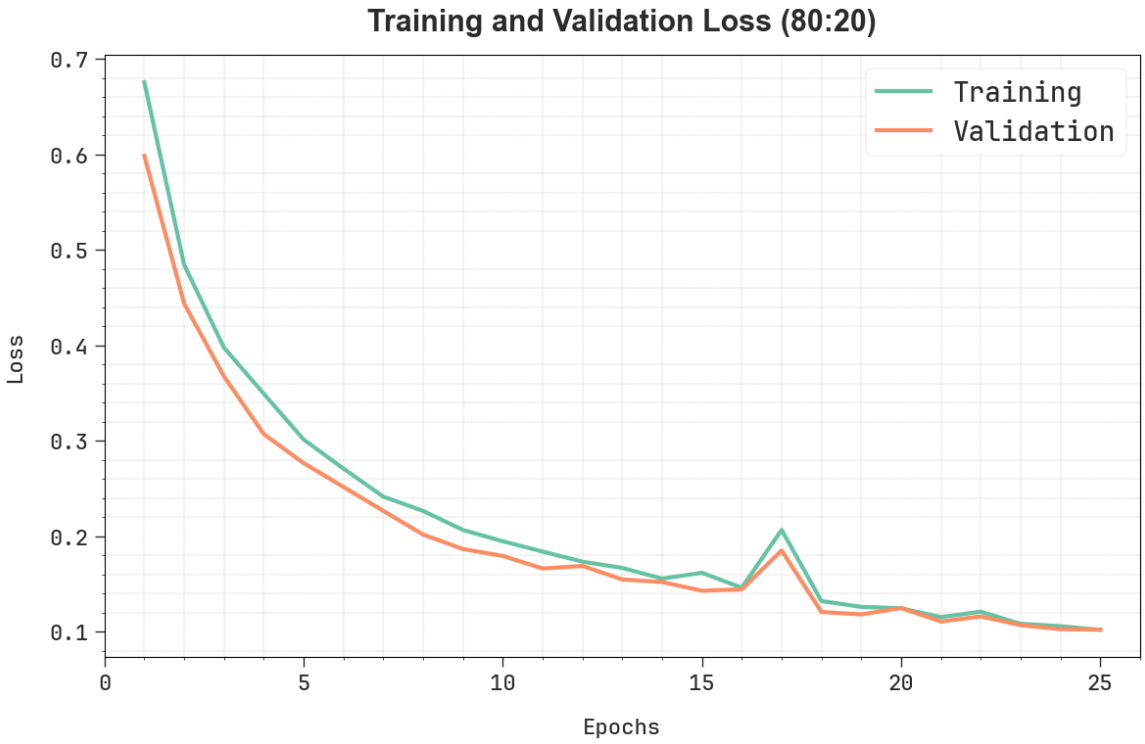

Figure 7. Loss graph of the BCCD-MOADLM method below 80%TRPH and 20%TSPH.

Table 3 and Figure 8 show the comparative results of the BCCD-MOADLM approach with existing models [34–36]. The results highlighted that the Linear SVC, naïve Bayes (NB), SVM, RNN, and ensemble bag (EB) techniques reported lesser performance. Moreover, the decision tree (DT) method and LSTM approaches accomplished closer outcomes. The BCCD-MOADLM methodology exhibited improved performance with maximal $prec_n$, $reca_l$, $accu_y$, and F_{score} of 96.94%, 97.07%, 99.32%, and 97.00%, respectively.

Table 3. Comparative outcome of the BCCD-MOADLM model with existing approaches [34–36].

Models	$\boldsymbol{Accu_y}$	$\boldsymbol{Prec_n}$	$\boldsymbol{Reca_l}$	$\boldsymbol{F_{score}}$
Linear SVC	93.78	89.83	92.22	91.44
NB	95.11	96.47	90.78	90.38
SVM Method	88.30	90.69	94.61	93.04
RNN Algorithm	97.90	96.34	96.61	95.28
LSTM Model	98.79	95.36	91.32	95.69
DT	99.00	92.02	91.37	90.58
EB	97.59	92.32	95.49	90.70
BCCD-MOADLM	99.32	96.94	97.07	97.00

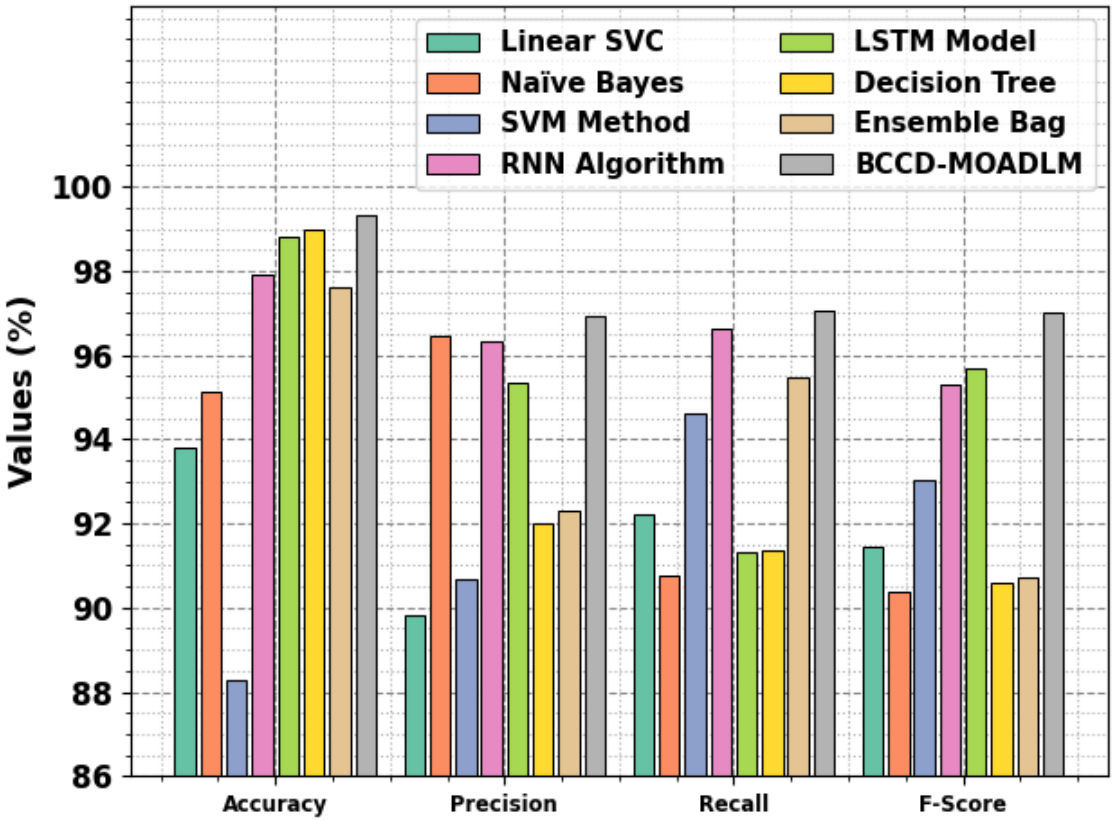

Figure 8. Comparative analysis of the BCCD-MOADLM model with existing approaches.

Table 4 and Figure 9 display the running time (RT) result of the BCCD-MOADLM technique. Based on RT, the BCCD-MOADLM technique attained a lower RT of 15.98sec, while the Linear SVC, NB, SVM, RNN, LSTM, DT, and EB methods achieved greater RT values of 25.41sec, 31.24sec, 37.46sec, 36.69sec, 34.35sec, 25.83sec, and 32.87sec, respectively.

Table 4. RT outcome of the BCCD-MOADLM method with existing techniques.

Methods	**RT (sec)**
Linear SVC	25.41
NB	31.24
SVM Method	37.46
RNN Algorithm	36.69
LSTM Model	34.35
DT	25.83
EB	32.87
BCCD-MOADLM	15.98

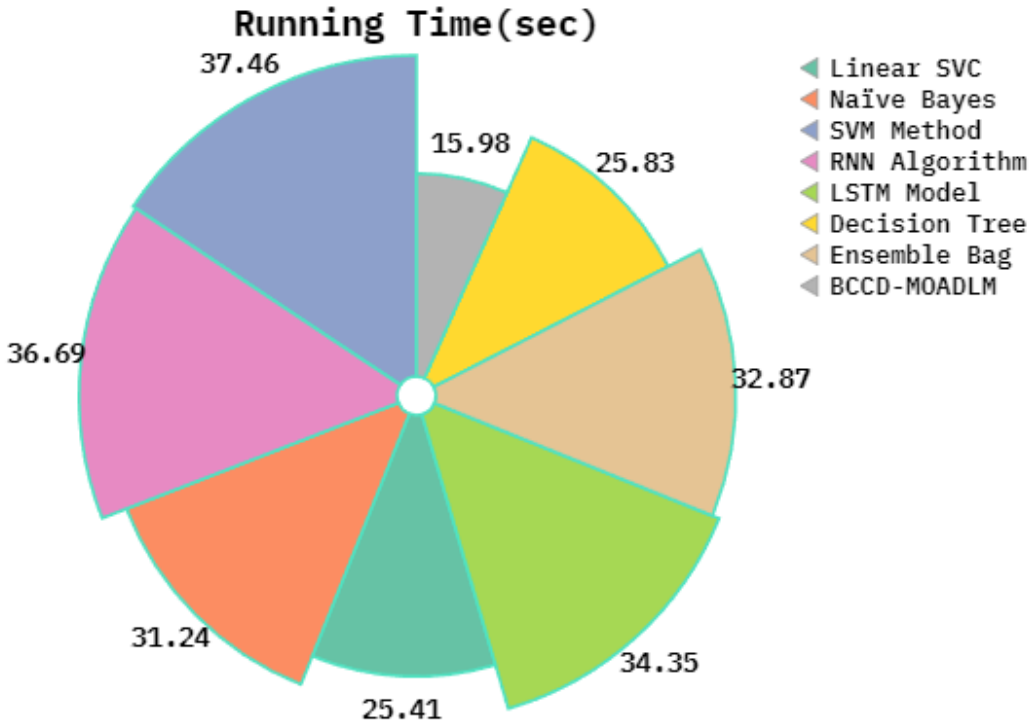

Figure 9. RT outcome of the BCCD-MOADLM method with existing techniques.

Table 5 and Figure 10 illustrate the error analysis of the BCCD-MOADLM technique with existing models. The SVM method attained the highest $accu_y$ at 11.70% but exhibited moderate $prec_n$ at 9.31% and a relatively low $reca_l$ of 5.39%, resulting in an F_{score} of 6.96%. The Linear SVC followed with an $accu_y$ of 6.22% and a $prec_n$ of 10.17%, yet it suffered from a lower $reca_l$ of 7.78% and an F_{score} of 8.56%. NB performed poorly across all metrics with an $accu_y$ of 4.89%, $prec_n$ of 3.53%, $reca_l$ of 9.22%, and F_{score} of 9.62%. The RNN approach achieved an $accu_y$ of 2.10% with lower values in $prec_n$ and $reca_l$ at 3.66% and 3.39%, respectively. Similarly, the LSTM model scored 1.21% in $accu_y$ and delivered an F_{score} of 4.31%. DT and EB performed marginally better with F_{score} of 9.42% and 9.30%, respectively. The BCCD-MOADLM model illustrated the lowest error values across all metrics with an $accu_y$ of 0.68%, $prec_n$ of 3.06%, $reca_l$ of 2.93%, and F_{score} of 3.00%, highlighting effective error minimization compared to conventional algorithms.

Table 5. Error analysis of the BCCD-MOADLM methodology with existing models.

Algorithm	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}
Linear SVC	6.22	10.17	7.78	8.56
NB	4.89	3.53	9.22	9.62
SVM Method	11.70	9.31	5.39	6.96
RNN Algorithm	2.10	3.66	3.39	4.72
LSTM Model	1.21	4.64	8.68	4.31
DT	1.00	7.98	8.63	9.42
EB	2.41	7.68	4.51	9.30
BCCD-MOADLM	0.68	3.06	2.93	3.00

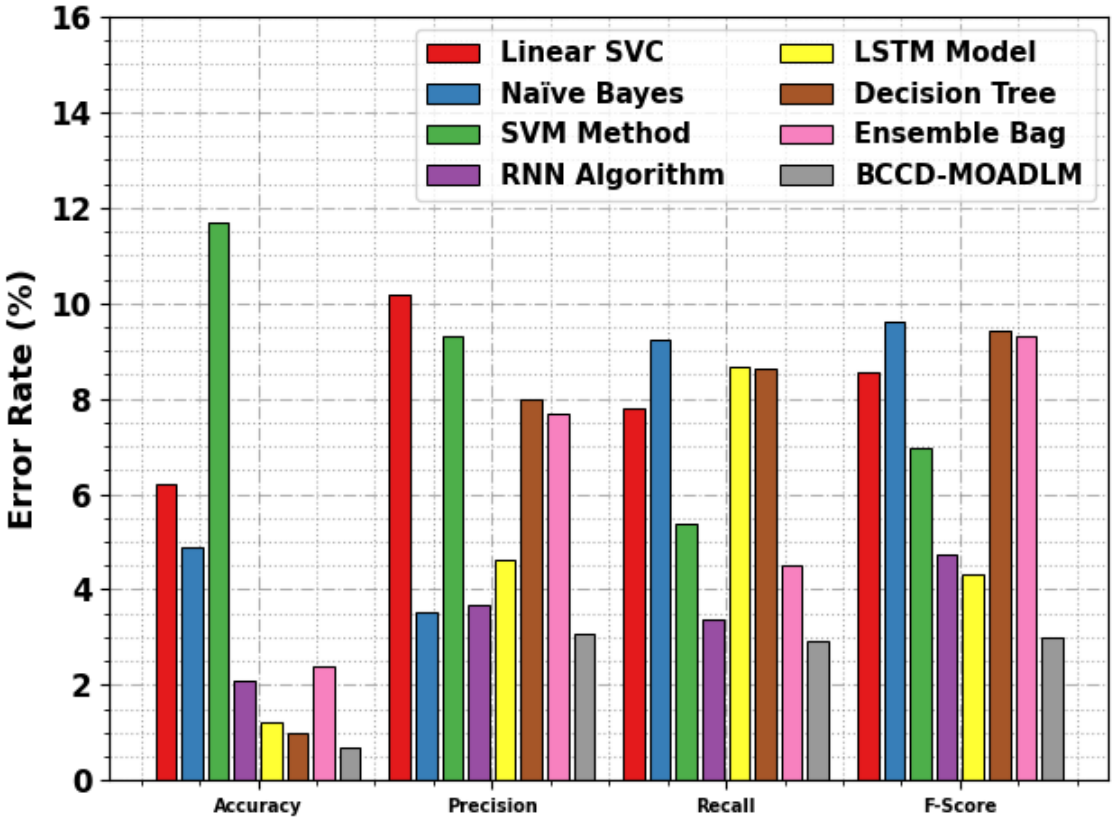

Figure 10. Error analysis of the BCCD-MOADLM methodology with existing models.

5. Conclusions

In this study, I develop a BCCD-MOADLM model. My main intention of the BCCD-MOADLM model is to detect and classify cybersecurity in an IoT environment using advanced optimization models. BC technology has been developed as a probable solution for improving the privacy and security of IoT networks. Next, the data pre-processing stage applies z-score normalization to transform input data into a compatible format. Thereafter, the proposed BCCD-MOADLM model designs OOA for the dimensionality reduction process. In addition, the QBiGRU technique is employed for the cybersecurity classification process. Finally, the POA design implements the parameter choice of the QBiGRU model. The experimental evaluation of the BCCD-MOADLM approach is examined under the BoT-IoT dataset. The comparison outcomes of the BCCD-MOADLM approach portrays a superior accuracy value of 99.32% over existing models.

Data availability statement

The data supporting this study's findings are openly available in the Kaggle repository at https://research.unsw.edu.au/projects/bot-iot-dataset, reference number [33].

Use of Generative-AI tools declaration

The author declares he has not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The author declares no conflicts of interest in this paper.

References

1. O. Abdulkader, A. M. Bamhdi, V. Thayananthan, F. Elbouraey, B. Al-Ghamdi, A lightweight blockchain based cybersecurity for IoT environments, In: *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, IEEE, 2019, 139−144. https://doi.org/10.1109/CSCloud/EdgeCom.2019.000-5
2. A. A. A. El-Latif, B. Abd-El-Atty, I. Mehmood, K. Muhammad, S. E. Venegas-Andraca, J. Peng, Quantum-inspired blockchain-based cybersecurity: securing smart edge utilities in IoT-based smart cities, *Inform. Process. Manag.*, **58** (2021), 102549. https://doi.org/10.1016/j.ipm.2021.102549
3. F. Alkurdi, I. Elgendi, K. S. Munasinghe, D. Sharma, A. Jamalipour, Blockchain in IoT security: A survey, In: *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*, IEEE, 2018, 1−4. https://doi.org/10.1109/ATNAC.2018.8615409
4. K. M. Giannoutakis, G. Spathoulas, C. K. Filelis-Papadopoulos, A. Collen, M. Anagnostopoulos, K. Votis, et al., A blockchain solution for enhancing cybersecurity defence of IoT, In: *2020 IEEE international conference on blockchain (blockchain)*, IEEE, 2020, 490−495. https://doi.org/10.1109/Blockchain50366.2020.00071
5. R. Alajlan, N. Alhumam, M. Frikha, Cybersecurity for blockchain-based IoT systems: A review, *Appl. Sci.*, **13** (2023), 7432. https://doi.org/10.3390/app13137432
6. W. Li, S. Tug, W. Meng, Y. Wang, Designing collaborative blockchained signature-based intrusion detection in IoT environments, *Future Gener. Comput. Syst.*, **96** (2019), 481−489. https://doi.org/10.1016/j.future.2019.02.064
7. B. Alotaibi, Utilizing blockchain to overcome cyber security concerns in the internet of things: A review, *IEEE Sens. J.*, **19** (2019), 10953−10971. https://doi.org/10.1109/JSEN.2019.2935035
8. P. Kumar, R. Kumar, A. Aljuhani, D. Javeed, A. Jolfaei, A. N. Islam, Digital twin-driven SDN for smart grid: A deep learning integrated blockchain for cybersecurity, *Sol. Energy*, **263** (2023), 111921. https://doi.org/10.1016/j.solener.2023.111921
9. M. Gimenez-Aguilar, J. M. De Fuentes, L. Gonzalez-Manzano, D. Arroyo, Achieving cybersecurity in blockchain-based systems: A survey, *Future Gener. Comput. Syst.*, **124** (2021), 91−118. https://doi.org/10.1016/j.future.2021.05.007
10. V. Nagamalla, R. K. Sanapala, Integrating predictive big data analytics with behavioral machine learning models for proactive threat intelligence in industrial IoT cybersecurity, *Int. J. Wirel. Ad Hoc Commun.*, **7** (2023). https://doi.org/10.54216/IJWAC.070201
11. E. Ntizikira, L. Wang, J. Chen, K. Saleem, Enhancing IoT security through emotion recognition and blockchain-driven intrusion prevention, *Internet Things*, **29** (2025), 101442. https://doi.org/10.1016/j.iot.2024.101442
12. R. Dahiya, L. Samal, D. Samal, J. Kumar, V. Sharma, D. K. Sahni, et al., A blockchain based security system framework in healthcare domain using IoT, *J. Electr. Syst.*, **20** (2024), 2039−2050. https://doi.org/10.52783/jes.1805
13. K. Dhanushkodi, K. Venkataramani, N. P. K R, R. Sethuraman, BGHO-E2EB model: Enhancing IoT security with gaussian artificial hummingbird optimization and blockchain technology, *T. Emerg. Telecommun. T.*, **36** (2025), e70037. https://doi.org/10.1002/ett.70037

14. M. Akbar, M. M. Waseem, S. H. Mehanoor, P. Barmavatu, Blockchain-based cyber-security trust model with multi-risk protection scheme for secure data transmission in cloud computing, *Cluster Comput.*, 2024, 1−15. https://doi.org/10.1007/s10586-024-04481-9
15. G. Archana, R. Goyal, K. M. Kumar, Blockchain-driven optimized chaotic encryption scheme for medical image transmission in IoT-Edge environment, *Int. J. Comput. Intell. Syst.*, **18** (2025), 11. https://doi.org/10.1007/s44196-024-00731-1
16. K. V. Pichandi, S. Qamar, R. Manikandan, *Network security enhancement in data-driven intelligent architecture based on cloud IoT blockchain cryptanalysis*, Data-Driven Systems and Intelligent Applications, CRC Press, 2024, 137−154. https://doi.org/10.1201/9781003388449-8
17. S. Rathnamala, T. Vijayashanthi, M. Prabhananthakumar, A. Panthakkan, S. Atalla, W. Mansoor, Enhanced hybrid deep learning approach for botnet attacks detection in IoT environment, *arXiv preprint*, 2025.
18. H. M. Rai, K. K. Shukla, L. Tightiz, S. Padmanaban, Enhancing data security and privacy in energy applications: Integrating IoT and blockchain technologies, *Heliyon*, **10** (2024). https://doi.org/10.1016/j.heliyon.2024.e38917
19. U. Sakthivelu, C. N. S. V. Kumar, Enhanced cyberthreat detection and classification with CMCOADL-TDC using deep learning and optimization techniques, *Tehnički Vjesnik*, **32** (2025), 891−899. https://doi.org/10.17559/TV-20240619001788
20. L. Tightiz, J. Yoo, Quantum-fuzzy expert timeframe predictor for post-TAVR monitoring, *Mathematics*, **12** (2024), 2625. https://doi.org/10.3390/math12172625
21. S. I. Popoola, Y. Tsado, A. A. Ogunjinmi, E. Sanchez-Velazquez, Y. Peng, D. B. Rawat, Multi-stage deep learning for intrusion detection in industrial internet of things, *IEEE Access*, **13** (2025), 60532−60555. https://doi.org/10.1109/ACCESS.2025.3557959
22. S. M. Hassan, M. M. Mohamad, F. B. Muchtar, Advanced intrusion detection in MANETs: A survey of machine learning and optimization techniques for mitigating black/gray hole attacks, *IEEE Access*, **12** (2024), 150046−150090. https://doi.org/10.1109/ACCESS.2024.3457682
23. A. Payyampally, S. M. Thampi, P. Mukherjee, *Advanced intrusion detection techniques and trends in IoT networks*, Securing the Connected World: Exploring Emerging Threats and Innovative Solutions, Springer, Cham, 2025, 47−113. https://doi.org/10.1007/978-3-031-82826-3_3
24. A. K. Kumar, S. Rathnamala, T. Vijayashanthi, M. Prabhananthakumar, A. Panthakkan, S. Atalla, et al., Enhanced hybrid deep learning approach for botnet attacks detection in IoT environment, In: *2024 7th International Conference on Signal Processing and Information Security (ICSPIS)*, IEEE, 2024, 1−6. https://doi.org/10.1109/ICSPIS63676.2024.10812621
25. A. Nazir, J. He, N. Zhu, A. Wajahat, F. Ullah, S. Qureshi, et al., Empirical evaluation of ensemble learning and hybrid CNN-LSTM for IoT threat detection on heterogeneous datasets, *J. Supercomput.*, **81** (2025), 775. https://doi.org/10.1007/s11227-025-07255-1
26. S. AboulEla, N. Ibrahim, S. Shehmir, A. Yadav, R. Kashef, Navigating the cyber threat landscape: An in-depth analysis of attack detection within IoT ecosystems, *AI*, **5** (2024), 704−732. https://doi.org/10.3390/ai5020037
27. M. A. Alkhonaini, M. A. Alohali, M. Aljebreen, M. M. Eltahir, M. H. Alanazi, A. Yafoz, et al., Sandpiper optimization with hybrid deep learning model for blockchain-assisted intrusion detection in IoT environment, *Alex. Eng. J.*, **112** (2025), 49−62. https://doi.org/10.1016/j.aej.2024.10.032
28. W. Wang, B. Yan, B. Chai, R. Shen, A. Dong, J. Yu, EBIAS: ECC-enabled blockchain-based identity authentication scheme for IoT device, *High-Confid. Comput.*, **5** (2025), 100240. https://doi.org/10.1016/j.hcc.2024.100240

29. R. Saranya, S. S. Priscila, *Development of intrusion detection using logistic regression with various pre-processing approaches*, International Conference on Advancements in Smart Computing and Information Security, Cham: Springer Nature Switzerland, 2023, 302−312. https://doi.org/10.1007/978-3-031-59097-9_22
30. A. Gaurav, B. B. Gupta, R. W. Attar, A. Alhomoud, V. Arya, K. T. Chui, Driver identification in advanced transportation systems using osprey and salp swarm optimized random forest model, *Sci. Rep.*, **15** (2025), 2453. https://doi.org/10.1038/s41598-022-12509-6
31. T. Zhang, C. Yu, S. Zhang, CA-SQBG: Cross-attention guided Siamese quantum BiGRU for drug-drug interaction extraction, *Comput. Biol. Med.*, **186** (2025), 109655. https://doi.org/10.1016/j.compbiomed.2025.109655
32. S. Wu, X. Cheng, H. Li, sEMG signal-based human lower-limb intention recognition algorithm using improved extreme learning machine, *J. Adv. Comput. Intell.*, **29** (2025), 53−63. https://doi.org/10.20965/jaciii.2025.p0053
33. https://research.unsw.edu.au/projects/bot-iot-dataset
34. M. Rahman, S. Al Shakil, M. R. Mustakim, A survey on intrusion detection system in IoT networks, *Cyber Secur. Appl.*, **3** (2025). https://doi.org/10.1016/j.csa.2024.100082
35. M. Zeeshan, Q. Riaz, M. A. Bilal, M. K. Shahzad, H. Jabeen, S. A. Haider, et al., Protocol-based deep intrusion detection for DoS and DDoS attacks using UNSW-NB15 and Bot-IoT data-sets, *IEEE Access*, **10** (2021), 2269−2283. https://doi.org/10.1109/ACCESS.2021.3137201
36. S. Alosaimi, S. M. Almutairi, An intrusion detection system using BoT-IoT, *Appl. Sci.*, **13** (2023), 5427. https://doi.org/10.3390/app13095427

AIMS Press