# Intrusion Detection in a Digital Twin-Enabled Secure Industrial Internet of Things Environment for Industrial Sustainability

**Mohammed Altaf Ahmed**
Department of Computer Engineering, College of Computer Engineering & Sciences, Prince Sattam bin Abdulaziz University, Alkharj, Saudi Arabia
m.altaf@psau.edu.sa (corresponding author)

**Suleman Alnatheer**
Department of Computer Engineering, College of Computer Engineering & Sciences, Prince Sattam bin Abdulaziz University, Alkharj, Saudi Arabia
s.alnatheer@psau.edu.sa

*Received: 5 January 2025 | Revised: 27 January 2025 and 3 February 2025 | Accepted: 5 February 2025*

 *DOI: https://doi.org/10.48084/etasr.10128*

**ABSTRACT**

**Research focuses on sustainable development for the smart industry environment, where new challenges emerge every day. Digital Twins (DT) have gained substantial attention from an industrial growth point of view. This is because it significantly contributes to the predictive maintenance, simulation, and optimization of the Industrial Internet of Things (IIoT), ensuring its sustainability in future industries that demand unprecedented flexibility. Current research discusses the possibility of using DT for intrusion detection in industrial systems. By integrating DT and IIoT, physical elements become virtual representations and enhance data analytics performance. However, a lack of trust between the parties involved and untrustworthy public communication channels can lead to various types of attacks and threats to ongoing communication. With this motivation in mind, this study develops a Binary Arithmetic Optimization Algorithm with Variational Recurrent Autoencoder-based Intrusion Detection (BAOA-VRAID) for DT-enabled secure IIoT environments. The proposed BAOA-VRAEID technique focuses on the integration of DT with the IIoT server, which collects industrial transaction data and helps to enhance the IIoT environment's security and communication privacy. The BAOA-VRAID technique uses BAOA to designate an optimal subset of features to detect intrusions. The VRAE classification model with the Harris Hawks Optimization (HHO) algorithm-based hyperparameter optimizer is used for intrusion detection. The BAOA-VRAID method was tested on a benchmark dataset, showing that it significantly outperformed other contemporary methods.**

***Keywords-digital twins; industrial IoT; intrusion detection; Al-Kharj industrial area; deep learning***

## I. INTRODUCTION

Industrial IoT (IIoT) refers to interconnected smart devices and computing resources that are installed on a network to reach high potential and enhance industrial and production processes [1]. IIoT adoption is based on eradicating complexity in management, device deployment, and connectivity. For instance, IIoT enables the tracking of objects in the supply chain from production to distribution. The fast evolution of IIoT has coincided with cyberattacks on dynamic structures, including smart grids, smart factories, etc. [2, 3]. To carry out malicious attacks, a hacker can use robust tools and methods, such as Denial of Service (DoS), Man-in-the-Middle (MiM), false code injection, Distributed DoS (DDoS), and firmware modification, to control the IIoT infrastructure [4, 5].

Digital Twin (DT) is an innovative digitalization approach that performs a real-time logical simulation method on physical substances [3, 6-9]. DT in IIoT environments helps in carrying out experiments in analyzing physical substances without deploying them before production. DT can find data discrepancies in the product life cycle by comparing virtual and physical entities and offering simulation data to physical entities, thus improving the testing and calibration processes [10]. These recurring procedures improve methods, allowing more accurate optimization, evaluation, and prediction of industrial operations [11]. Machine Learning (ML) is a widely used method to formulate an anomaly-based Intrusion Detection System (IDS). ML consists of two types, namely supervised and unsupervised learning, along with hybrid learning, called semi-supervised learning [12]. Unsupervised learning uses unlabeled samples for training, whereas

supervised learning demands characterized examples [13]. Semi-supervised methods use more uncategorized input instances and some categorized instances for training. ML techniques for detecting cyber intrusions are widely used due to their automated action [14]. However, cyber intrusions are changing uninterruptedly, which makes scalable recognition systems a demand. Different neural network models, such as CNN, LSTM, and DBN, are used by the Fuzzy-MIFS method to handle the complexity of IoT data [15]. The main contributions of the current study are:

- It proposes a Binary Arithmetic Optimization Algorithm with Variational Recurrent Autoencoder Intrusion Detection (BAOA-VRAID) method for DT-enabled secure IIoT environments.
- This research was initially targeted to the Alkharj, Saudi Arabia and was later generalized to be suitable for any urban area.
- The proposed BAOA-VRAEID technique focuses on the integration of DT with the IIoT server, which collects industrial transaction data and helps improve communication security and privacy in the IIoT environment.
- The performance of the proposed approach was tested on a benchmark dataset, showing significant results.

## II. THE PROPOSED MODEL

This study introduces a new BAOA-VRAEID technique for intrusion detection in a DT-enabled secure IIoT environment. The proposed technique focuses on the incorporation of DT with the IIoT server, which collects industrial transaction data and helps improve communication security. It encompasses three major subprocesses: BAOA-based feature selection, VRAE-based classification, and HHO-based hyperparameter tuning.

### *A. Digital Twin (DT) Model*

The DT model refers to a key element of IIoT that connects to the virtual network structure. It offers a runtime layout of virtual devices and a real-time simulation environment for physical processes [15]. The generated virtual IIoT platform is the same as its physical counterpart and presents different functionality, namely, control logic execution, physical device types, and network protocols. Simulation emulates the DT whenever it is necessary to replicate a physical element or execute control logic. In simulation mode, the DT works without its physical counterpart and allows users to look closely at test equipment, change processes, and even improve production operations. This is the same as virtual commissioning. The replication mode mimics data, which includes log files, network connections, and sensor readings from the physical environment. In addition, devices or sensors are connected directly to the IIoT twinning structure accessible at edge nodes.

### *B. Feature Selection with the BAOA Method*

The BAOA-VRAEID technique initially uses the BAOA to select an optimal subset of features. AOA is a metaheuristic algorithm consisting of four basic arithmetic operations, as given in [15], and it can give a better component through mathematical optimization on a series of solutions. As this operator can introduce change in a large order, the algorithm implements exploration using division and multiplication. Hence, subtraction and addition operators can be used for performing local search or exploitation, but this operator is not fit for local search due to its higher dispersion.

$x_i = [x_i^1, x_i^2, l_i]$ represents the initial solution, and AOA is a population-based technique that is randomly generated over a $d$-dimension search space as follows:

$$d_i = x_i' + r\left(x^j{}_{\max} - d_{\min}\right) = \{1,2,..,N\} \quad (1)$$

where $j = \{1,2,..,d\}$, $x_i'$ denotes the $j^{th}$ dimension of the $i^{th}$ solution, $r$ is a random number in the range from zero to one, $x_i$ denotes the $i^{th}$ solution, $x^j{}_{\max}$ and $x^j{}_{\min}$ denote the upper and the lower boundaries in the search space of the $j^{th}$ dimension, and $N$ represents the population size. $X$ is the initial solution:

$$X = \begin{Bmatrix} x_1^1 & \cdots & x_1^d \\ \vdots & \ddots & \vdots \\ x_N^1 & \cdots & x_N^d \end{Bmatrix} \quad (2)$$

A fitness function defines in the population the quality of every solution. The candidate solution with the highest fitness value across all iterations is the optimal solution. The decision on the choice of exploration and exploitation can be evaluated according to the Math Optimizer Accelerated (MOA), providing a coefficient based on existing iteration, as follows:

$$MOA(C_{Iter}) = \text{Min} + C_{Iter} \times \left(\frac{\text{Max} - \text{Min}}{M_{Iter}}\right) \quad (3)$$

Equation (3) represents the existing iteration, $Max$ signifies the maximum quantity of iterations, and $Min$ represents a minimal best value of MOA. MOA is developed to favor exploration at the initial stage and exploitation in the later iteration. A random value was produced, and its value was compared to MOA to perform exploration or exploitation.

#### *1) Exploration*

The solution space can be explored using multiplication and division operators. The multiplication or division operator is randomly chosen for exploration with equivalent probability. The new solution can be evaluated using:

$$x_i^j(C_{Iter}+1) = \{B(x^j) \div (MOP + \varepsilon) \times ((x^j{}_{\max} - x'{}_{\min}) \times \mu + x'{}_{\min}), r_2 \quad (4)$$

where $x_i^j(C_{Iter}+1)$ denotes the $j^{th}$ dimension of the $i^{th}$ solution at the following iteration, $B(x^j)$ denotes the $j^{th}$ dimension in the present optimum solution, $\varepsilon$ denotes a small non-zero number, $\mu$ indicates a control variable for adjusting the method of search and locate to the value 0.5, and $r_2$ denotes a random value generated in $[0, 1]$. MOP is evaluated by:

$$MOP\ (C_{Iter}) = 1 - \frac{C_{Iter}^{1/\alpha}}{M_{Iter}^{1/\alpha}} \quad (5)$$

where $\alpha$ indicates the sensitivity factor with its value fixed at 5.

*2) Exploitation*

The exploitation stage is a deep-rooted search near the optimum solution. Therefore, the operators leveraged are subtraction and addition. Similar to the exploration, the selection probability of the operator in exploitation can be equivalent. The new solution is evaluated using:

$$x_i'(C_{Iter}+1) = \{B(x^j) - MOP \times ((x^j_{\max} - xj_{\min}) \times \mu + x^j min), r_3 \quad (6)$$

where $r_3$ represents a random value within $[0, 1]$.

*C. Algorithm (BAOA-S/BAOA-V)*

The algorithms BAOA-S and BAOA-V of AOA are proposed, and the binary variants are given. According to Kennedy and Eberhart, the extended version of the technique might be effectively converted into binary using the Transfer Function (TF). TF converts into binary vectors from the real value vector. It describes changing the probability of components in solution vectors to zero or one based on the step vector value. Now, two TFs, a $V$-shaped (hyperbolic tangent) and an $S$-shaped (sigmoid), are employed to convert AOA into BAOA-S and BAOA-V, respectively.

$$TF_S(x!_i) = \frac{1}{1+e^{x_i^j}} \quad (7)$$

$$TF_V(x_i') = (\,|\tanh(x_i^j)|\,) \quad (8)$$

where $x_i^j$ denotes the $j^{th}$ dimension of the $i^{th}$ solution. Using (4) and (6), the location of the $i^{th}$ solution is updated based on the value of MOA, $r_2$ and $r_3$. The solution generates a real value from the step function. To transform it into the binary vector, $TF_S$ or $TF_V$ is employed on the generated solutions. According to (9) and (10) for the $V$-shaped and $S$-shaped TF, respectively, the actual value in the solution vector is mapped to either 0 or 1 depending on the values of the random numbers $r$ and TF. In this case, $r$ is a random number generated that belongs to the uniform distribution.

$$x_i^{j,b}(C_{lter}+1) = \{1, r\} \quad (9)$$

$$x_i^{j,b}(C_{lter}+1) = \{x_i^j, r\} \quad (10)$$

where $x_i^{j,b}(C_{Iter}+1)$ denotes the binary vector attained by mapping the real or step value solution vector $x_i'(C_{Iter}+1)$ using TF.

*D. Intrusion Detection using the VRAE Model*

This method applies VRAE to recognize intrusions in the IIoT environment. VRAE can be a recurring version of VAE to demonstrate sequential information, such as time series. Every step of time $t$, VRAE includes VAE having a stochastic latent variable $z_t$. The state $h_t$, which is a deterministic recurrent hidden state, captures temporal dependency, and $s_t$ is an observation. VRAE has four key steps: modeling generation, prior distribution, recurrence, and inference.

- Step 1: The latent variables $z_t$ at time step $t$ can be described by a prior distribution that is conditioned on the prior hidden recurrent state $h_{t-1}$:

$$z_t \sim N\left(\phi_\mu^{prior}(h_{t-1}), \phi_\sigma^{prior}(h_{t-1})\right) \quad (11)$$

where $N$ indicates the normal distribution.

- Step $G$: The observation $s_t$ is generated by conditioning it on latent variables $z_t$ and the prior hidden state $h_{t-1}$. The possibility of generating $s_t$ is.

$$s_t|z_t \sim N\left(\phi_\mu^{dec}(\phi^z(z_t), h_{t-1}), \phi_\sigma^{dec}(\phi^z(z_t), h_{t-1})\right) \quad (12)$$

- Step 2: Similarly, the posterior distribution in the inference step is described as follows.

$$z_t|s_t \sim N\left(\phi_\mu^{enc}(\phi^s(s_t), h_{t-1}), \phi_\sigma^{enc}(\phi^s(s_t), h_{t-1})\right) \quad (13)$$

- Step $R$: In the recurrence step, the hidden recurrent state $h_t$ can be gained from the previously hidden state $h_{t-1}$, the data variable $s_t$ and the latent variable $z_t$.

$$h_t = f_\theta([\phi^s(s_t), \phi^z(z_t)], h_{t-1}) \quad (14)$$

where $f_\theta(\cdot)$ is a recurrent neural network such as GRU and $\phi(\cdot)$ and denotes a feed-forward neural network.

*E. HHO Algorithm for Hyperparameter Tuning*

This study uses HHO for hyperparameter tuning. HHO, a new metaheuristic stochastic algorithm, defining Harris hawks' behavior, is characterized by superior coordination, which allows approaching, tracking, encircling, and attacking the prey. During hunting, a clever escape behavior, named surprise pounce, is implemented effectively. The HHO technique involves exploration and exploitation stages.

During the exploration phase, Harris hawks randomly search for prey based on the subsequent formula.

$$X(t+1) = \begin{Bmatrix} X_{rand}(t) - r_1|X_{rand}(t) - 2_{r2}X(t)|q \geq 0.5 \\ \left(X_{prey}(t) - X_m(t)\right) - r_3\left(LB + r_4(UB - LB)\right)q < 0.5 \end{Bmatrix} \quad (15)$$

At the $(t+1)$ iteration, the hawks are positioned at $X(t+1)$, the prey is positioned at $X_{prey}(t)$, $r1$ to $r4$ and $q$ are random values from zero to one, $X_{rand}(t)$ characterizes a hawk selected at an arbitrary location, and $X_m$ denotes the average position of the present hawk population, as follows:

$$X_m(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t) \quad (16)$$

where $X_i(t)$ represents every hawk's position at the $t$ iteration, and $N$ specifies the total hawk number. Before the exploitation stage, there is a transitional phase as soon as the discovery phase is completed. In the period of transition, the energy of the prey needs to be shaped based on:

$$E = 2E_0\left(1 - \frac{t}{T}\right) \quad (17)$$

where $E$ denotes the escape energy of the prey, $E_0$ indicates the initial energy state of the prey, and $T$ is the maximal number of iterations. $E_0$ ranges from $-1$ to 1 based on the physical fitness victim. This implies that victims lose energy while $E_0$ was heading toward $-1$. In the last stage, Harris hawks suddenly method the prey strategies of the attack. Now, if $E \geq 0.5$ and

$r \geq 0.5$, where $r$ is considered an escape probability, Harris hawks gradually use a soft besiege strategy for encircling the prey as follows:

$$X_i^{t+1} = \Delta X_i^t - E\left|IX_{prey} - x_i^t\right|, \Delta X_i^t = X_{prey} - X_i^t \quad (18)$$

where $X_i(t+i)$ represents a distance between a prey and a present individual, and $J$ denotes the strength of the prey jumping in the escape and is considered as a random value in the $[0, 2]$ interval. If $E < 0.5$, and $r \geq 0.5$, due to insufficient escape energy, the prey cannot escape, and the position of Harris hawks can be formulated by:

$$X_i^{t+1} = X_{prey} - E|\Delta X_i^t| \quad (19)$$

If $E \geq 0.5$ and $r < 0.5$, then Harris hawks do soft besiege with increasing quick dive strategies to confuse whether the prey has adequate energy to efficiently escape. The mathematical expression of the strategy is specified by:

$$X_i^{t+1} = \begin{cases} Y = x_{prey} - E\left|X_{prey} - X_i^t\right| & if\ (Y) < f(X_i^t) \\ Z = Y + S \times Levy(d) & if'(Z) < f(X_i^t) \end{cases} \quad (20)$$

where $d$ denotes the problem dimension and $S$ indicates the $1 \times d$ arbitrary vector. If $E < 0.5$ and $r < 0.5$, then the prey has inadequate escape energy and attacks the prey using:

$$X_i^{t+1} = \begin{cases} x_{prey} - E\left|JX_{prey} - X_m^t\right|, & if\ (Y) < f(X_i^t) \\ Z = Y + S \times Levy(d), & \text{if } (Z) < f(X_i^t) \end{cases} \quad (21)$$

In HHO, choosing a fitness is a critical factor. The assessment of the goodness of a candidate solution is performed by solution encryption. Then, precision is considered as the main condition to create a fitness function.

$$Fitness = \max(P) \quad (22)$$

$$P = \frac{TP}{TP+FP} \quad (23)$$

where $TP$ and $FP$ indicate the true and false positive values, respectively.

## III. RESULTS AND DISCUSSION

The CICIDS-2017 dataset was used for the experimental evaluation of the BAOA-VRAEID technique. The dataset includes 17500 samples and 7 class labels, as shown in Table I.

TABLE I. DATASET DETAILS

| Dataset class | Number of samples |
|---|---|
| BruteForce | 2500 |
| DoS | 2500 |
| WebAttacks | 2500 |
| Infiltration | 2500 |
| Bot | 2500 |
| DDoS | 2500 |
| PortScan | 2500 |
| Total No. of Samples | 17500 |

Figure 1 shows the confusion matrices for the proposed BAOA-VRAEID technique based on the CICIDS-2017 dataset. The results indicate that the BAOA-VRAEID technique identifies distinct intrusion types proficiently.

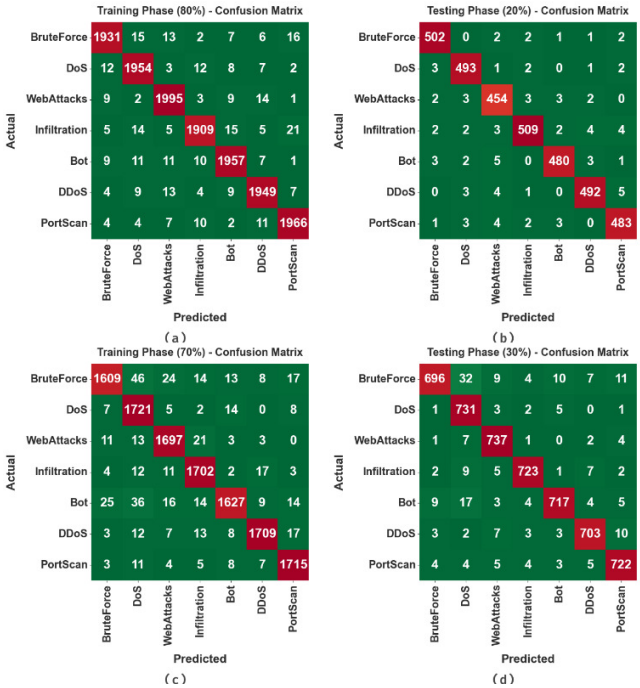


Fig. 1. Confusion matrices for the BAOA-VRAEID system using: (a-b) TRP/TSP of 80:20, (c-d) TRP/TSP of 70:30

Figure 2 and Table II show the IDS results of BAOA-VRAEID using 80:20 TRP/TSP. The results show that BAOA-VRAEID detects different kinds of intrusions proficiently. For instance, on 80% of TRP, the BAOA-VRAEID technique achieved an average $accu_{bal}$ of 99.31%, $prec_n$ of 97.58%, $reca_l$ of 97.58%, $F_{score}$ of 97.58%, and $G_{mean}$ of 98.58%. Meanwhile, on 20% of TSP, the BAOA-VRAEID method achieved an average $accu_{bal}$ of 99.29%, $prec_n$ of 97.50%, $reca_l$ of 97.51%, $F_{score}$ of 97.50%, and $G_{mean}$ of 98.54%. Table III and Figure 3 show the results of BAOA-VRAEID for 70:30 TRP/TSP.

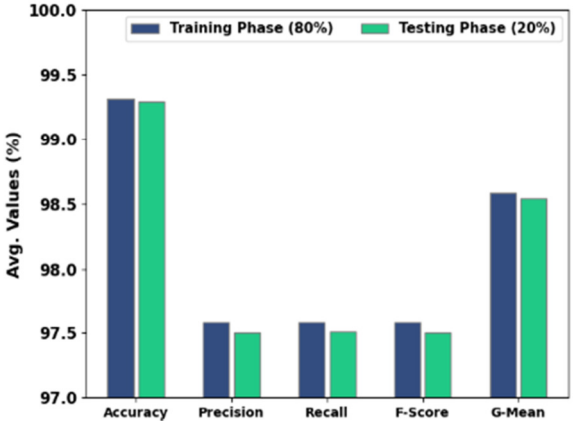


Fig. 2. Average results of BAOA-VRAEID on 80:20 TRP/TSP.

TABLE II. RESULTS OF BAOA-VRAEID ON 80:20 TRP/TSP

| Class | $Accu_{bal}$ | $Prec_n$ | $Reca_l$ | $F_{score}$ | $G_{mean}$ |
|---|---|---|---|---|---|
| **Training Phase (80%)** | | | | | |
| BruteForce | 99.27 | 97.82 | 97.04 | 97.43 | 98.33 |
| DoS | 99.29 | 97.26 | 97.80 | 97.53 | 98.67 |
| WebAttacks | 99.36 | 97.46 | 98.13 | 97.79 | 98.85 |
| Infiltration | 99.24 | 97.90 | 96.71 | 97.30 | 98.17 |
| Bot | 99.29 | 97.51 | 97.56 | 97.53 | 98.57 |
| DDoS | 99.31 | 97.50 | 97.69 | 97.60 | 98.63 |
| PortScan | 99.39 | 97.62 | 98.10 | 97.86 | 98.85 |
| Average | 99.31 | 97.58 | 97.58 | 97.58 | 98.58 |
| **Testing Phase (20%)** | | | | | |
| BruteForce | 99.46 | 97.86 | 98.43 | 98.14 | 99.03 |
| DoS | 99.37 | 97.43 | 98.21 | 97.82 | 98.88 |
| WebAttacks | 99.09 | 95.98 | 97.22 | 96.60 | 98.29 |
| Infiltration | 99.23 | 98.07 | 96.77 | 97.42 | 98.21 |
| Bot | 99.34 | 98.16 | 97.17 | 97.66 | 98.43 |
| DDoS | 99.31 | 97.81 | 97.43 | 97.62 | 98.52 |
| PortScan | 99.23 | 97.18 | 97.38 | 97.28 | 98.45 |
| Average | 99.29 | 97.50 | 97.51 | 97.50 | 98.54 |

TABLE III. RESULTS OF BAOA-VRAEID ON 70:30 TRP/TSP

| Class | $Accu_{bal}$ | $Prec_n$ | $Reca_l$ | $F_{score}$ | $G_{mean}$ |
|---|---|---|---|---|---|
| **Training Phase (70%)** | | | | | |
| BruteForce | 98.57 | 96.81 | 92.95 | 94.84 | 96.17 |
| DoS | 98.64 | 92.98 | 97.95 | 95.40 | 98.36 |
| WebAttacks | 99.04 | 96.20 | 97.08 | 96.64 | 98.22 |
| Infiltration | 99.04 | 96.10 | 97.20 | 96.65 | 98.27 |
| Bot | 98.68 | 97.13 | 93.45 | 95.26 | 96.45 |
| DDoS | 99.15 | 97.49 | 96.61 | 97.05 | 98.08 |
| PortScan | 99.21 | 96.67 | 97.83 | 97.25 | 98.63 |
| Average | 98.90 | 96.20 | 96.15 | 96.16 | 97.74 |
| **Testing Phase (30%)** | | | | | |
| BruteForce | 98.23 | 97.21 | 90.51 | 93.74 | 94.92 |
| DoS | 98.42 | 91.15 | 98.38 | 94.63 | 98.40 |
| WebAttacks | 99.10 | 95.84 | 98.01 | 96.91 | 98.64 |
| Infiltration | 99.16 | 97.57 | 96.53 | 97.05 | 98.05 |
| Bot | 98.78 | 97.02 | 94.47 | 95.73 | 96.96 |
| DDoS | 98.99 | 96.57 | 96.17 | 96.37 | 97.79 |
| PortScan | 98.90 | 95.63 | 96.65 | 96.14 | 97.95 |
| Average | 98.80 | 95.85 | 95.82 | 95.79 | 97.53 |

The experimental results show that the BAOA-VRAEID method categorizes different kinds of intrusions proficiently. For example, on 70% of TRP, the BAOA-VRAEID approach achieved an average $accu_{bal}$ of 98.90%, $prec_n$ of 96.20%, $reca_l$ of 96.15%, $F_{score}$ of 96.16%, and $G_{mean}$ of 97.74%. In the meantime, on 30% of TSP, the BAOA-VRAEID approach achieved an average $accu_{bal}$ of 98.80%, $prec_n$ of 95.85%, $reca_l$ of 95.82%, $F_{score}$ of 95.79%, and $G_{mean}$ of 97.53%.

Figure 4 shows how well the DT works when using the $T_{AC}$ and $V_{AC}$ of the BAOA-VRAEID method. The figure suggests that higher $T_{AC}$ and $V_{AC}$ values indicate better performance from the BAOA-VRAEID technique. Interestingly, the BAOA-VRAEID method achieved the highest $T_{AC}$ results.

Figure 5 illustrates the DT performance of the BAOA-VRAEID approach's $T_{LOS}$ and $V_{LOS}$, showing low $V_{LOS}$ results. Figure 6 presents the precision-recall curve for the BAOA-VRAEID method, showing that it improved the values of precision-recall under all considered classes.

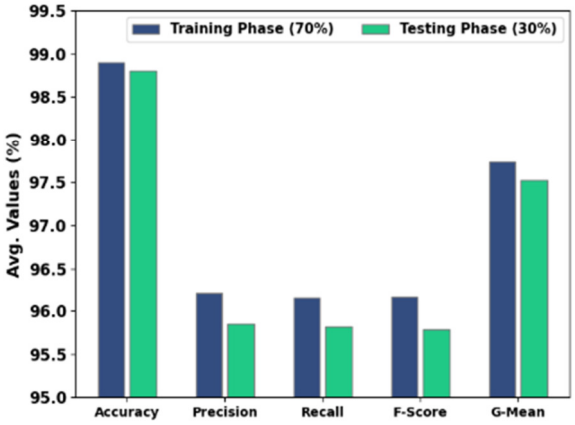


Fig. 3. Average results of BAOA-VRAEID on 70:30 TRP/TSP.


Fig. 4. TACY and VACY outcomes of the BAOA-VRAEID approach.

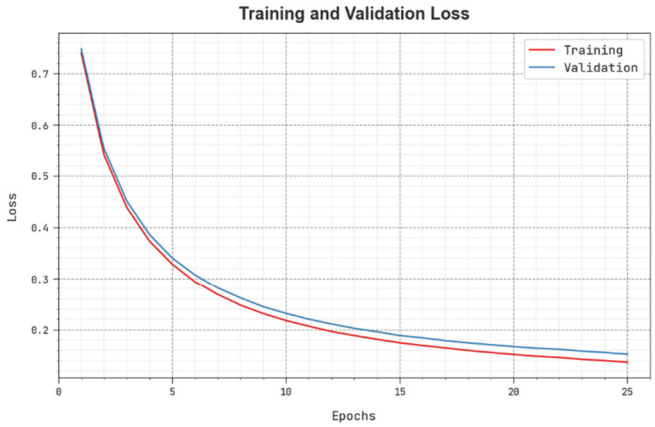


Fig. 5. $T_{LOS}$ and $V_{LOS}$ results of the BAOA-VRAEID approach.

To examine the performance of the BAOA-VRAEID technique, a wide comparison was carried out, as shown in Table IV and Figure 7. The KNN model achieved the lowest results, while the RF model had improved results. The stacked model achieved moderately improved classification results. The CNN-GRU and CNN-LSTM models achieved reasonable results. The proposed BAOA-VRAEID technique

outperformed the other models with a higher $accu_y$ of 99.31%, $prec_n$ of 97.58%, $reca_l$ of 97.58%, and $F1_{score}$ of 97.58%. Therefore, the proposed BAOA-VRAEID technique outperformed the other models.

TABLE IV. COMPARATIVE OUTCOME OF BAOA-VRAEID APPROACH WITH OTHER SYSTEMS

| Methods | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| **BAOA-VRAEID** | 99.31 | 97.58 | 97.58 | 97.58 |
| **RF Model** | 90.55 | 91.97 | 78.05 | 83.67 |
| **KNN Model** | 76.70 | 74.56 | 76.58 | 76.34 |
| **DT Model** | 90.25 | 90.70 | 78.08 | 82.15 |
| **CNN-GRU** | 98.48 | 90.46 | 95.00 | 97.17 |
| **CNN-LSTM** | 98.22 | 96.21 | 94.24 | 95.08 |
| **Stacked Model** | 93.06 | 93.09 | 86.58 | 89.80 |

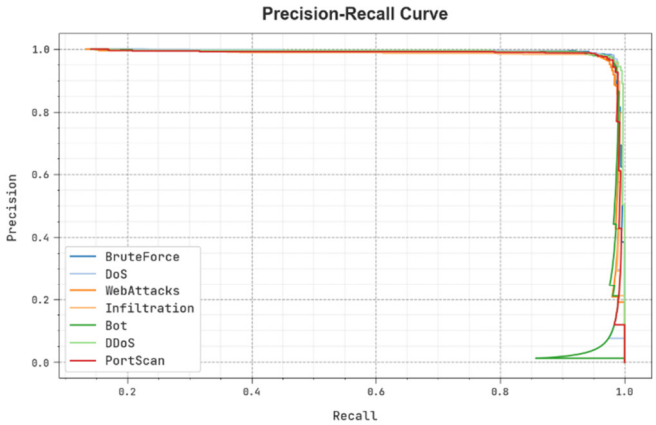


Fig. 6. Precision-recall results of the BAOA-VRAEID approach.

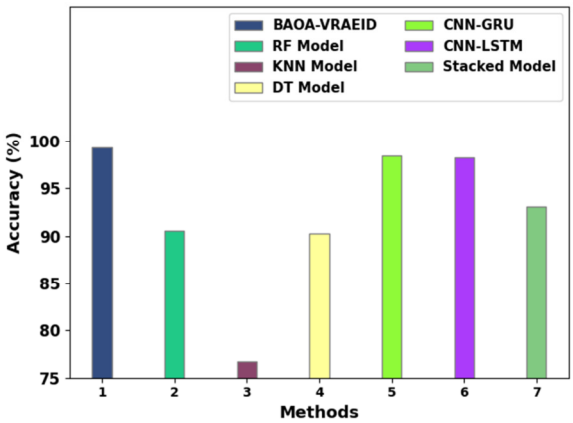


Fig. 7. Comparative results of the BAOA-VRAEID approach with other systems.

Table V and Figure 8 show a comparison of the proposed with other methods in terms of computation time. The BAOA-VRAEID approach outperformed other techniques with a CT of 11.37s. Therefore, the proposed method achieves enhanced performance in less time than other methods.

TABLE V. COMPUTATION TIME COMPARISON

| Methods | Computational Time (sec) |
|---|---|
| BAOA-VRAEID | 11.37 |
| RF Model | 14.15 |
| KNN Model | 22.20 |
| DT Model | 42.70 |
| CNN-GRU | 33.75 |
| CNN-LSTM | 16.43 |
| Stacked Model | 28.93 |

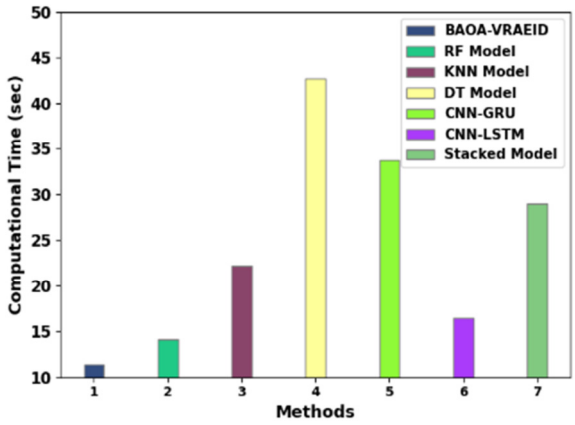


Fig. 8. Computation time comparison of BAOA-VRAEID and other methods.

## IV. CONCLUSION

This study introduced a modern BAOA-VRAEID method for intrusion detection in a DT-enabled secure IIoT environment. The proposed BAOA-VRAEID technique focused on the incorporation of the DT with the IIoT server, which collects industrial transaction data and helps to enhance communication security and privacy in the IIoT environment. For accurate classification and detection of intrusions, the proposed method used BAOA for feature selection, the HHO algorithm for hyperparameter tuning, and the VRAE classification model for intrusion detection in the IIoT environment. The proposed BAOA-VRAEID technique was evaluated on a benchmark dataset, and the results showed that it outperformed other recent approaches. The proposed BAOA-VRAEID technique provided significantly better results than existing methods in terms of accuracy (99.31%), precision (97.58%), recall (97.58%), and F1-Score (97.58%) with a low computational time. Therefore, the BAOA-VRAEID technique achieved significantly improved performance over other models. In the future, blockchain technology can increase the performance of the BAOA-VRAEID technique.

## ACKNOWLEDGMENT

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through project number PSAU/2024/01/28560.

## REFERENCES

[1] P. Empl and G. Pernul, "Digital-Twin-Based Security Analytics for the Internet of Things," *Information*, vol. 14, no. 2, Feb. 2023, Art. no. 95, https://doi.org/10.3390/info14020095.

[2] K. Xia *et al.*, "A digital twin to train deep reinforcement learning agent for smart manufacturing plants: Environment, interfaces and

intelligence," *Journal of Manufacturing Systems*, vol. 58, pp. 210–230, Jan. 2021, https://doi.org/10.1016/j.jmsy.2020.06.012.

[3] X. Li, H. Liu, W. Wang, Y. Zheng, H. Lv, and Z. Lv, "Big data analysis of the Internet of Things in the digital twins of smart city based on deep learning," *Future Generation Computer Systems*, vol. 128, pp. 167–177, Mar. 2022, https://doi.org/10.1016/j.future.2021.10.006.

[4] A. Potgantwar, S. Aggarwal, P. Pant, A. S. Rajawat, C. Chauhan, and V. N. Waghmare, "Secure Aspect of Digital Twin For Industry 4.0 Application Improvement Using Machine Learning." Social Science Research Network, Aug. 11, 2022, https://doi.org/10.2139/ssrn.4187977.

[5] S. R. Chhetri, S. Faezi, A. Canedo, and M. A. A. Faruque, "QUILT: quality inference from living digital twins in IoT-enabled manufacturing systems," in *Proceedings of the International Conference on Internet of Things Design and Implementation*, Montreal, Canada, Apr. 2019, pp. 237–248, https://doi.org/10.1145/3302505.3310085.

[6] J. Scheibmeir and Y. Malaiya, "Multi-Model Security and Social Media Analytics of the Digital Twin," *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 6, pp. 323–330, 2020, https://doi.org/10.25046/aj050639.

[7] M. O. Ozdogan, L. Carkacioglu, and B. Canberk, "Digital Twin Driven Blockchain Based Reliable and Efficient 6G Edge Network," in *2022 18th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Los Angeles, CA, USA, May 2022, pp. 342–348, https://doi.org/10.1109/DCOSS54816.2022.00062.

[8] M. M. Salim, A. K. Comivi, T. Nurbek, H. Park, and J. H. Park, "A Blockchain-Enabled Secure Digital Twin Framework for Early Botnet Detection in IIoT Environment," *Sensors*, vol. 22, no. 16, Jan. 2022, Art. no. 6133, https://doi.org/10.3390/s22166133.

[9] A. Bécue, E. Maia, L. Feeken, P. Borchers, and I. Praça, "A New Concept of Digital Twin Supporting Optimization and Resilience of Factories of the Future," *Applied Sciences*, vol. 10, no. 13, Jan. 2020, Art. no. 4482, https://doi.org/10.3390/app10134482.

[10] J. Lopez, J. E. Rubio, and C. Alcaraz, "Digital Twins for Intelligent Authorization in the B5G-Enabled Smart Grid," *IEEE Wireless Communications*, vol. 28, no. 2, pp. 48–55, Apr. 2021, https://doi.org/10.1109/MWC.001.2000336.

[11] Y. Wang, Z. Su, S. Guo, M. Dai, T. H. Luan, and Y. Liu, "A Survey on Digital Twins: Architecture, Enabling Technologies, Security and Privacy, and Future Prospects," *IEEE Internet of Things Journal*, vol. 10, no. 17, pp. 14965–14987, Sep. 2023, https://doi.org/10.1109/JIOT.2023.3263909.

[12] J. Tan, X. Sha, B. Dai, and T. Lu, "Wireless Technology and Protocol for IIoT and Digital Twins," in *2020 ITU Kaleidoscope: Industry-Driven Digital Transformation (ITU K)*, Ha Noi, Vietnam, Dec. 2020, pp. 1–8, https://doi.org/10.23919/ITUK50268.2020.9303189.

[13] G. Lampropoulos and K. Siakas, "Enhancing and securing cyber-physical systems and Industry 4.0 through digital twins: A critical review," *Journal of Software: Evolution and Process*, vol. 35, no. 7, 2023, Art. no. e2494, https://doi.org/10.1002/smr.2494.

[14] A. H. Abu Saq, A. Zainal, B. A. S. Al-Rimy, A. Alyami, and H. A. Abosaq, "Intrusion Detection in IoT using Gaussian Fuzzy Mutual Information-based Feature Selection," *Engineering, Technology & Applied Science Research*, vol. 14, no. 6, pp. 17564–17571, Dec. 2024, https://doi.org/10.48084/etasr.8268.

[15] P. Kumar, R. Kumar, A. Kumar, A. A. Franklin, S. Garg, and S. Singh, "Blockchain and Deep Learning for Secure Communication in Digital Twin Empowered Industrial IoT Network," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 2802–2813, Sep. 2023, https://doi.org/10.1109/TNSE.2022.3191601.