Computers, Materials & Continua
DOI: 10.32604/cmc.2022.031613
Article

Tech Science Press

Artificial Intelligence Based Threat Detection in Industrial Internet of Things Environment

Fahad F. Alruwaili*

College of Computing and Information Technology, Shaqra University, Sharqa, Saudi Arabia
*Corresponding Author: Fahad F. Alruwaili. Email: alruwaili@su.edu.sa

Received: 22 April 2022; Accepted: 07 June 2022

Abstract: Internet of Things (IoT) is one of the hottest research topics in recent years, thanks to its dynamic working mechanism that integrates physical and digital world into a single system. IoT technology, applied in industries, is termed as Industrial IoT (IIoT). IIoT has been found to be highly susceptible to attacks from adversaries, based on the difficulties observed in IIoT and its increased dependency upon internet and communication network. Intentional or accidental attacks on these approaches result in catastrophic effects like power outage, denial of vital health services, disruption to civil service, etc., Thus, there is a need exists to develop a vibrant and powerful for identification and mitigation of security vulnerabilities in IIoT. In this view, the current study develops an AI-based Threat Detection and Classification model for IIoT, abbreviated as AITDC-IIoT model. The presented AITDC-IIoT model initially pre-processes the input data to transform it into a compatible format. In addition, Whale Optimization Algorithm based Feature Selection (WOA-FS) is used to elect the subset of features. Moreover, Cockroach Swarm Optimization (CSO) is employed with Random Vector Functional Link network (RVFL) technique for threat classification. Finally, CSO algorithm is applied to appropriately adjust the parameters related to RVFL model. The performance of the proposed AITDC-IIoT model was validated under benchmark datasets. The experimental results established the supremacy of the proposed AITDC-IIoT model over recent approaches.

Keywords: Security; industrial internet of things; threat detection; artificial intelligence; feature selection

1 Introduction

Internet of Things (IoT) has managed to pervade numerous domains from home automation to industries with crucial frameworks. The contributions of IoT are wide enough started from attaining the final cases or complementing/exchanging the processes involved in industrial control systems. The extensive applicability of IoT gadgets allows the industrial technologies to flourish, in industries with less technical maturity. Few appropriate instances are linked with exploitation of oil and electricity production while both the domains are straightforwardly linked with national cyberdefence [1].

Industrial Internet of Things (IIoT) combines multiple players such as sensors, gadgets, and physical machineries with internet. Then, it utilizes software to conduct deep analytics and convert huge volumes of both structured and unstructured data into powerful insights and information [2]. IIoT emphasizes the application of IoT in manufacturing zones since there is a growing interest among researchers to involve Machine-to-Machine (M2 M) transmission, big data, and Machine Learning (ML) in industry settings. IoT can also be applied in some other domains such as linking wastewater systems and manufacturing of robots, flow gauges, electric meters and other connected systems, and industrial gadgets. With the incorporation of IIoT, institutions as well as manufacturing hubs gain high efficiency and dependability upon its works [3]. Since IoT is capable of linking multiple gadgets with internet, it allows the identification of distinct threats to perform anomalous actions. There is an increasing number of loopholes and susceptibilities found in the protocol utilized by IIoT structure. If it encounters risks, sophisticated attacks can be made at IIoT environment using multiple methods [4]. The intentions of an attacker are multitude in nature such as gaining access to appropriate data, money theft, and source corruption [5].

IoT gadgets have special features with regard to transmission. So, whenever there is an attack made, it tends to provoke the decentralized assaults on any kind of structures [6]. These are the difficulties faced in designing an identification algorithm for IoT which are well known in traditional networks [7,8]. The main goal of machine learning technique is to empower the technologies so that it learns and performs estimation based on the information scheduled earlier. Though the usage of ML in identifying anomalous conduct is an established process, intruder identification domain has been mostly untouched [9]. In conventional techniques, anomaly recognition has been performed by statistical methodologies. Therefore, the increasing penetration of ML methods has unlocked new probabilities for the identification of outlier information, thanks to the accessibility of huge volume of information which might be leveraged using ML methods. In this perspective, such ML methods provide an alluring viewpoint to be applied in IoT application zones. It is challenging to make use of stationary models in this regard [10].

Aboelwafa et al. [11] proposed a novel attack detection methodology via Autoencoder (AE). The study exploited the sensor data in correlation with time and space to sequentially recognize the fabricated dataset. Furthermore, the fabricated dataset is refined by Denoising AE (DAE). The DAE dataset was cleaned in an efficient manner and produced clean datasets from the corrupted (attacked) information. Hassan et al. [12] developed a down sampler-encoder-based collective dataset generator. This model was to ensure the effective collection of real distribution of the attack model for large-scale IIoT attack surfaces. The presented downsampler-based data generator is upgraded simultaneously and confirmed at the time of training Deep Neural Network (DNN) discriminators so as to ensure robustness.

Qureshi et al. [13] presented a secure and novel architecture for identification of security threats in RPL-based IoT and IIoT systems. The presented architecture possesses the ability to identify Version number, HELLO-Flood, Blackhole, and Sinkhole attacks. Hassan et al. [14] enhanced the reliability of IIoT systems using a scalable and reliable cyberattack recognition method i.e., Supervisory Control and Data Acquisition (SCADA) technique. To be specific, an ensemble-learning method, related to the integration of Random Subspace (RS) learning model using Random Tree (RT), was presented to identify SCADA cyberattacks o through network traffic from SCADA-related IIoT architecture. The researchers in the literature [15–19] developed a detection module based on Stacked Variation Auto-Encoder (VAE) with Convolution Neural Network (CNN). This model has the capability to learn about hidden architecture of the scheme's activity and reveal its ransomware behaviour. Furthermore, a data augmentation technique was proposed based on VAE to generate a novel dataset that can be utilized in training a system and to improve the generalized abilities of the presented method.

The current study develops an AI-based Threat Detection and Classification model for IIoT, named AITDC-IIoT model. The presented AITDC-IIoT model initially pre-processes the input data and transforms it into a compatible format. Then, Whale Optimization Algorithm-based Feature Selection (WOA-FS) model has been involved to elect the subset of features. Moreover, Cockroach Swarm Optimization (CSO) is employed with Random Vector Functional Link network (RVFL) model for classification of threats. Finally, CSO algorithm is applied to appropriately adjust the parameters involved in RVFL model. The performance of the proposed AITDC-IIoT model was validated using benchmark datasets.

2 The Proposed Model

In this study, a new AITDC-IIoT model has been developed for proficient threat detection and classification using IIoT. The presented AITDC-IIoT model initially pre-processes the input data to convert it into a compatible format. Followed by, WOA-FS model is applied to elect the subset of features. At last, CSO is employed with RVFL model for classification of threats. Fig. 1 depicts the overall block diagram of AITDC-IIoT technique.

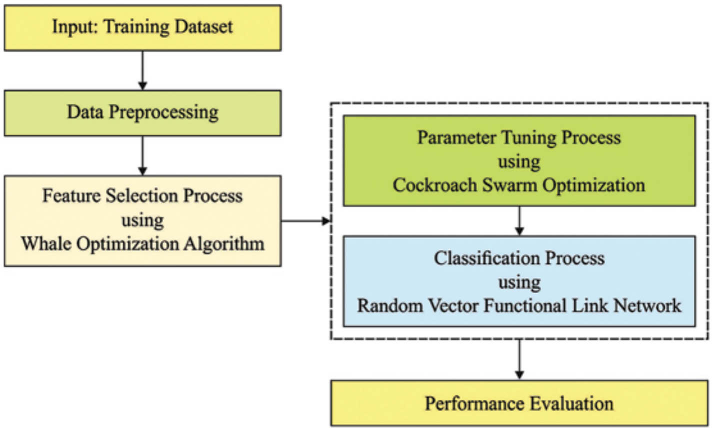

Input: Training Dataset
Data Preprocessing
Feature Selection Process using Whale Optimization Algorithm
Parameter Tuning Process using Cockroach Swarm Optimization
Classification Process using Random Vector Functional Link Network
Performance Evaluation

Figure 1: Block diagram of AITDC-IIoT technique

2.1 Feature Selection Module

In order to elect the features, WOA is applied in this study. In order to explore the most number of possible solutions for the problem from searching space, whale individuals are utilized from the community [20]. Three functions are applied in WOA such as hunting, encircling, and shrinking. During exploitation stage, both surrounding and shrinking functions are utilized. However, under exploration stage, the hunting function is utilized. To arrive at the optimal solution for Dimension Optimization problem (DO), the processes of i^{th} individual from c^{th} generation are utilized. Following processes are involved in WOA.

Encircling Operation

$$ESH_{ij}(c+1) = ESH_{*j}(c) - B \cdot O_{ij}(c) \tag{1}$$

Shrinking Operation

$$ESH_{ij}(c+1) = ESH_{*j}(c) + g^{et} \cdot \cos(2\pi t) \cdot O'_{ij}(c) \tag{2}$$

Hunting Operation

$$ESH_{ij}(c+1) = ESH_{kj}(c) - B \cdot O^{*}_{ij}(c) \tag{3}$$

$$B = 2\left(1 - \frac{c}{c_{\max}}\right) \cdot (2rd - 1) \tag{4}$$

The arbitrary number in the range of [0 1] is explained through (rd), The existing number of iterations is demonstrated as c, maximum number of iterations is explained as $c_{\max}$ and the positive vector of the optimum solution is denoted by $ESH_{-}(c)$. In order to define the logarithmical spiral shape, a constant e is utilized, and the arbitrary number from -1 and 1 is demonstrated as t. The arbitrary position vector $ESH(c)$ is chosen from the existing population. Three distances are subsequently found. At first, the primary distance is at $\left|O_{ij}(c) = \right| 2rd.ESH_{*j}(c) - ESH_{ij}(c)|$ while the secondary distance is at $O'_{ij}(c) = \left|ESH_{*j}(c) - ESH_{ij}(c)\right|$, and the tertiary distance is at $O^{*}_{ij}(c) = \left|2rd.ESH_{kj}(c) - ESH_{ij}(c)\right|$. Based on the probability p_{rob}, three Eqs. (1)–(3) are applied in WOA. The whale individuals are upgraded in Eq. (1), if $Prob < 0.5$ and $|B| < 1$, then the individuals are adjusted by Eq. (3), once $|B| \geq 1$. Eq. (2) is utilized for updating the individuals, if $p_{rob} \geq 0.5$.

In WOA, the whale moves from searching space to adapt to the position pointed in the space which is named as 'constant space'. The transformation can be done using S-shaped transfer function. The possibility of altering the location vector element from 0 to 1 is adapted by the transfer function. So, it forces the searching agent to move into a binary space. Fig. 2 depicts the flowchart of WOA.

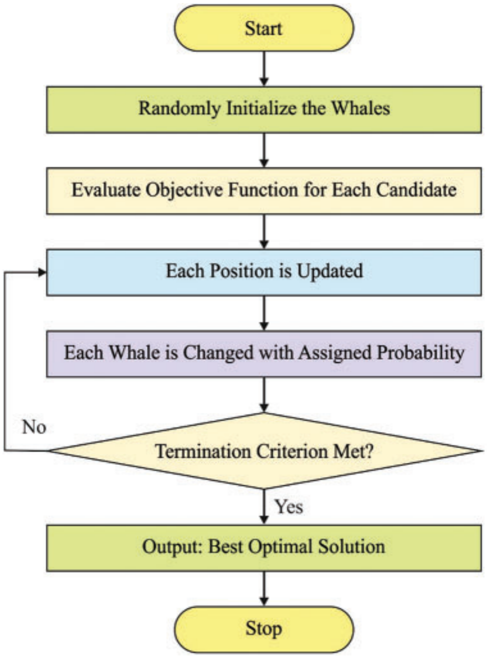

Start
Randomly Initialize the Whales
Evaluate Objective Function for Each Candidate
Each Position is Updated
Each Whale is Changed with Assigned Probability
No
Termination Criterion Met?
Yes
Output: Best Optimal Solution
Stop

Figure 2: WOA Flowchart

The S-shaped function is updated as demonstrated herewith.

$$y^k = \frac{1}{1 + e^{-v_i^k(t)}} \tag{5}$$

$$X_i^d = \begin{cases} 1 & if\ rand < \left(x_i^k (t+1)\right) \\ 0 & otherwise \end{cases} \tag{6}$$

2.2 Threat Classification Module

Once the features are selected, they are fed as input in RVFL model for classification purpose. RVFL model depends upon Single Layer Feed Forward Network (SLFN) [21]. In this method, the weights are arbitrarily initialized based on the node and weight is tuned with no iteration. Consider that RVFL network contains J improvement node and $\alpha = (\alpha_1, \cdots, \alpha_P)^t$ is the resultant weight, whereas $= J + n$. The activation function for j^{th} trained instance is determined as $G_l(x_i) = g(a_l, b, x_i)$ on the ℓ^{th} improvement layer to $\ell = 1, \ldots, J$ and $i =, \ldots, m$. Here, $a_l = (a_{(l1}, \ \ldots, \ a_{(lm)})^t$ and b correspond to weight as well as bias correspondingly. Accordingly, Hessian matrix is assumed as $H = \left[G_1(U) \ \cdots \ G_j(U)\right]$ as follows.

$$H = \begin{bmatrix} G_1(x_1) & \cdots & G_J(x_1) \\ \cdots & \cdots & \cdots \\ G_l(x_m) & \cdots & G_J(x_m) \end{bmatrix}.$$

The problem equation for RVFL is stated as

$$\min \|y - V\alpha\|^2 + \lambda \|\alpha\|^2 \tag{7}$$

whereas $V = [H \ U]$ and λ refers to the fixed positive constants. At this point, the gradient of Eq. (7) is defined in terms of α. Additionally, the gradient equates to 0 to determine the solution as follows.

$$\alpha = (V^t V + CI)^{-1} V^t y. \tag{8}$$

At novel instance x, the regressor evaluated for RVFL is as follows.

$$f(x) = [h(x)\, x]\, \alpha, \tag{9}$$

whereas $h(x) = [G_1(x) \cdots \ G,(x)]$.

2.3 Parameter Optimization Module

In this final stage, CSO algorithm is applied to appropriately adjust the parameters related to RVFL model [22–25]. The CSO model imitates cockroach behavior i.e., dispersing, ruthless, chase-swarming behaviors [26]. In D-dimension searching region R^D, a cockroach cluster consists of N cockroach individuals while i-th individual characterizes the D-dimension vector $x(i) = (xi1, xi2, \ldots, xiD)$, $(i = 1, 2, \ldots, \ N)$ and the individual position is the best possible solution.

Chase-Swarming Behavior:

$$x_i = \begin{cases} w.\ x_i + step.rand.\ (p_i - x_i)\,, \ x_i \neq p_i \\ w.\ x_i + step.rand.\ \left(p_g - x_i\right)\,, \ x_i = pi \end{cases} \tag{10}$$

In this equation, $w$indicates the inertia weight i.e., a constant step indicates a fixed value whereas rand denotes an arbitrary value that lies in the interval of $[0, 1]$.

$$p_i = Opt_j \left\{x_j, \ x\_i - x\_j \leq visual\right\} \tag{11}$$

$j = 1, 2, \ldots, N, i = 1, 2, \ldots N.$

$$p_i = Opt_j \{x_i\} \tag{12}$$

Whereas opt indicates the optimal value.

Dispersion Behaviour:

$$x_i = \mathfrak{x}_i + rand\,(1,\ D)\,,\ i = 1, 2, \ldots,\ N \tag{13}$$

Now rand(l, D) represents the *D*-dimension vector that is fixed to some extent.

Ruthless Behavior

$$X\mathrm{k} = pg \tag{14}$$

In this formula, k denotes an arbitrary value within $[1, N]$ and p_g indicates the global optimal location. The steps involved in Continual space Cockroach Swarm Optimization (CCSO) method are shown below.

1. Initialize cockroach swarm with uniform distribution of arbitrary numbers and set value for each parameter.
2. Search p_i and p_g using the Eqs. (11) and (12).
3. Implement chase-swarming by Eq. (10)
4. Implement dispersion behaviour by Eq. (13)
5. Implement ruthless behavior by Eq. (14)
6. Repeat the loop until the end condition is obtained.

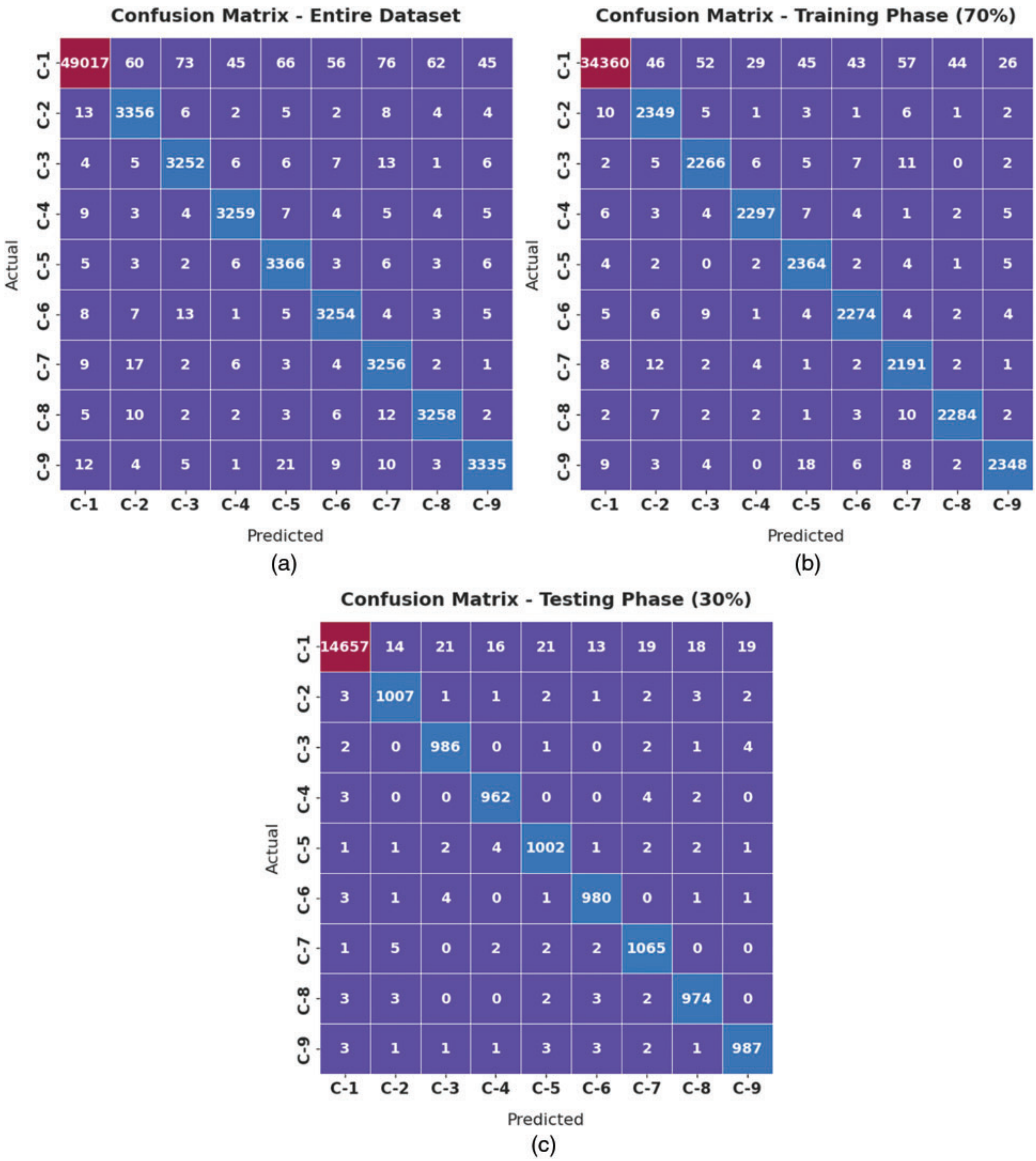
3 Experimental Validation

In this section, the proposed AITDC-IIoT model was experimentally validated using N-BaIoT dataset [27]. The dataset holds 76,200 samples under 9 class labels which are given in Tab. 1.

Table 1: Sample class labels

Class labels	Categories	No. of instances (Attack)
C-1	Benign	49500
C-2	Ack	3400
C-3	Scan	3300
C-4	SYN	3300
C-5	UDP	3400
C-6	UDP Plain	3300
C-7	Combo	3300
C-8	Junk	3300
C-9	TCP	3400
Total		76200

Fig. 3 demonstrates the set of confusion matrices generated by the proposed AITDC-IIoT model on test dataset. The figures imply that the proposed AITDC-IIoT model effectively recognized all the nine classes in the applied dataset.



Confusion Matrix - Entire Dataset

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9
C-1	49017	60	73	45	66	56	76	62	45
C-2	13	3356	6	2	5	2	8	4	4
C-3	4	5	3252	6	6	7	13	1	6
C-4	9	3	4	3259	7	4	5	4	5
C-5	5	3	2	6	3366	3	6	3	6
C-6	8	7	13	1	5	3254	4	3	5
C-7	9	17	2	6	3	4	3256	2	1
C-8	5	10	2	2	3	6	12	3258	2
C-9	12	4	5	1	21	9	10	3	3335

Confusion Matrix - Training Phase (70%)

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9
C-1	34360	46	52	29	45	43	57	44	26
C-2	10	2349	5	1	3	1	6	1	2
C-3	2	5	2266	6	5	7	11	0	2
C-4	6	3	4	2297	7	4	1	2	5
C-5	4	2	0	2	2364	2	4	1	5
C-6	5	6	9	1	4	2274	4	2	4
C-7	8	12	2	4	1	2	2191	2	1
C-8	2	7	2	2	1	3	10	2284	2
C-9	9	3	4	0	18	6	8	2	2348

Confusion Matrix - Testing Phase (30%)

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9
C-1	14657	14	21	16	21	13	19	18	19
C-2	3	1007	1	1	2	1	2	3	2
C-3	2	0	986	0	1	0	2	1	4
C-4	3	0	0	962	0	0	4	2	0
C-5	1	1	2	4	1002	1	2	2	1
C-6	3	1	4	0	1	980	0	1	1
C-7	1	5	0	2	2	2	1065	0	0
C-8	3	3	0	0	2	3	2	974	0
C-9	3	1	1	1	3	3	2	1	987

Figure 3: Confusion matrices generated by AITDC-IIoT technique for (a) entire dataset, (b) 70% of TR dataset, and (c) 30% of TS dataset

Tab. 2 illustrates the results offered by AITDC-IIoT model on threat classification in IIoT environment. The results indicate that the proposed AITDC-IIoT model gained significant results under all the classes. For instance, with entire dataset, the proposed AITDC-IIoT model categorized benign classes with $accu_y$, $prec_n$, $reca_l$, F_{score}, and Mathew Correlation Coefficient (MCC) values such as 99.28%, 99.87%, 99.02%, 99.44%, and 98.43% respectively. Simultaneously, with entire dataset, the AITDC-IIoT method categorized TCP class with $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.82%, 97.83%, 98.09%, 97.96%, and 97.86% respectively. Concurrently, with 70% of TR dataset, the presented AITDC-IIoT approach categorized benign classes with $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.27%, 99.87%, 99.01%, 99.44%, and 98.41% correspondingly. Meanwhile, with 70% of TR dataset, the proposed AITDC-IIoT system categorized TCP class with $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.82%, 98.04%, 97.91%, 97.98%, and 97.88% respectively. Eventually, with 30% of TS dataset, AITDC-IIoT model categorized benign class with $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.83%, 97.58%, 98.53%, 98.05%, and 97.96% correspondingly.

Table 2: Results of the analysis of AITDC-IIoT technique under different measures

Labels	Accuracy	Precision	Recall	F-Score	MCC
		Entire dataset			
Benign	99.28	99.87	99.02	99.44	98.43
Ack	99.80	96.85	98.71	97.77	97.67
Scan	99.80	96.81	98.55	97.67	97.57
SYN	99.86	97.93	98.76	98.34	98.27
UDP	99.80	96.67	99.00	97.82	97.72
UDP Plain	99.82	97.28	98.61	97.94	97.85
Combo	99.77	96.05	98.67	97.34	97.23
Junk	99.84	97.54	98.73	98.13	98.05
TCP	99.82	97.83	98.09	97.96	97.86
Average	99.75	97.43	98.68	98.05	97.85
		Training phase (70%)			
Benign	99.27	99.87	99.01	99.44	98.41
Ack	99.79	96.55	98.78	97.65	97.55
Scan	99.78	96.67	98.35	97.50	97.39
SYN	99.86	98.08	98.63	98.35	98.28
UDP	99.81	96.57	99.16	97.85	97.75
UDP Plain	99.81	97.10	98.48	97.79	97.69
Combo	99.75	95.59	98.56	97.05	96.94
Junk	99.84	97.69	98.75	98.22	98.14
TCP	99.82	98.04	97.91	97.98	97.88
Average	99.75	97.35	98.63	97.98	97.78
		Training phase (30%)			
Benign	99.83	97.58	98.53	98.05	97.96
Ack	99.83	97.14	99.00	98.06	97.98

(Continued)

Table 2: Continued

Labels	Accuracy	Precision	Recall	F-Score	MCC
Scan	99.86	97.57	99.07	98.31	98.24
SYN	99.80	96.91	98.62	97.76	97.65
UDP	99.85	97.71	98.89	98.29	98.22
UDP Plain	99.80	96.99	98.89	97.93	97.83
Combo	99.82	97.21	98.68	97.94	97.85
Junk	99.82	97.34	98.50	97.92	97.82
TCP	99.77	97.59	98.80	98.19	98.00
Average	99.83	97.58	98.53	98.05	97.96

Fig. 4 demonstrates the average threat classification outcomes achieved by the proposed AITDC-IIoT model. Upon entire dataset, AITDC-IIoT model achieved average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.75%, 97.43%, 98.68%, 98.05%, and 97.85% respectively. Moreover, on 70% of TR dataset, the proposed AITDC-IIoT technique offered average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.75%, 97.35%, 98.63%, 97.98%, and 97.78% correspondingly. Furthermore, on 30% of TS dataset, the presented AITDC-IIoT model provided average $accu_y$, $prec_n$, $reca_l$, F_{score}, and MCC values such as 99.83%, 97.58%, 98.53%, 98.05%, and 97.96% correspondingly.

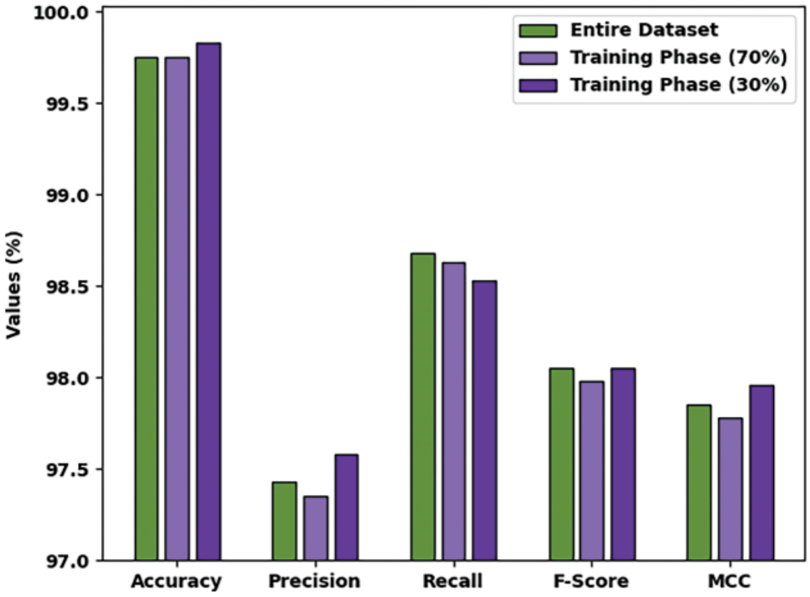


Figure 4: Average analysis results of AITDC-IIoT technique under different measures

A brief precision-recall analysis was conducted upon AITDC-IIoT approach on test dataset and the results are depicted in Fig. 5. As per the figure, it is clear that the proposed AITDC-IIoT method accomplished maximum precision-recall performance under different number of class labels.

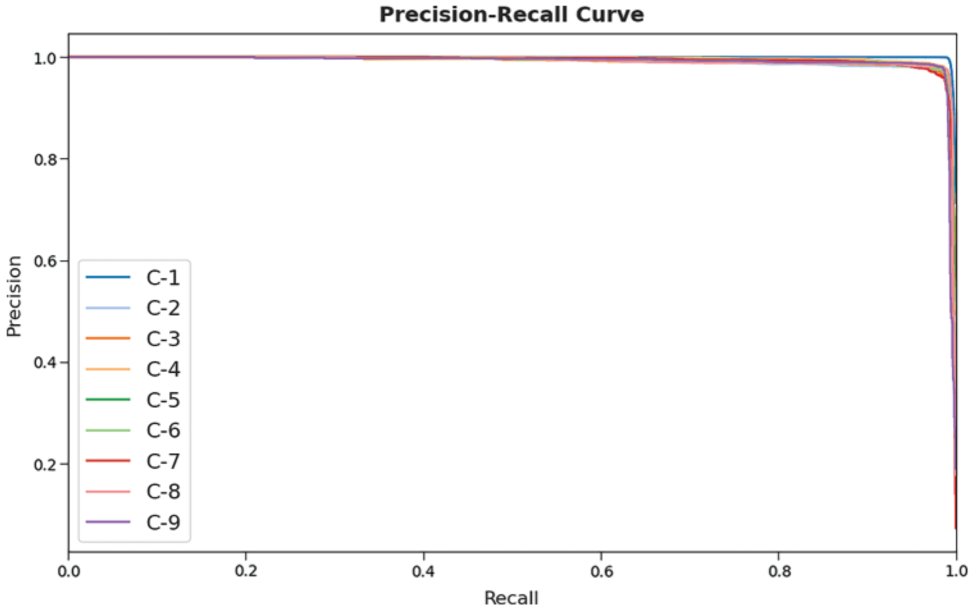


Figure 5: Precision-recall curve analysis results of AITDC-IIoT technique

Training Accuracy (TA) and Validation Accuracy (VA) values, attained by AITDC-IIoT model on test dataset, are demonstrated in Fig. 6. The experimental outcome imply that AITDC-IIoT model gained the maximum TA and VA values. To be specific, VA seemed to be higher than TA.

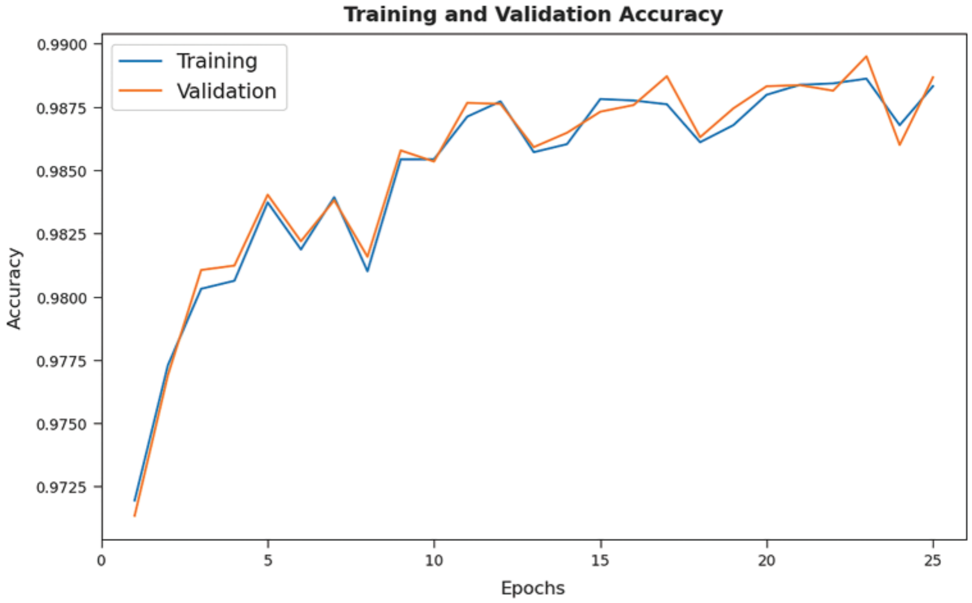


Figure 6: TA and VA graph analyses results of AITDC-IIoT technique

Training Loss (TL) and Validation Loss (VL) values, achieved by the proposed AITDC-IIoT technique on test dataset, are portrayed in Fig. 7. The experimental outcomes infer that AITDC-IIoT model achieved the least TL and VL values. To be specific, VL seemed to be lower than TL.

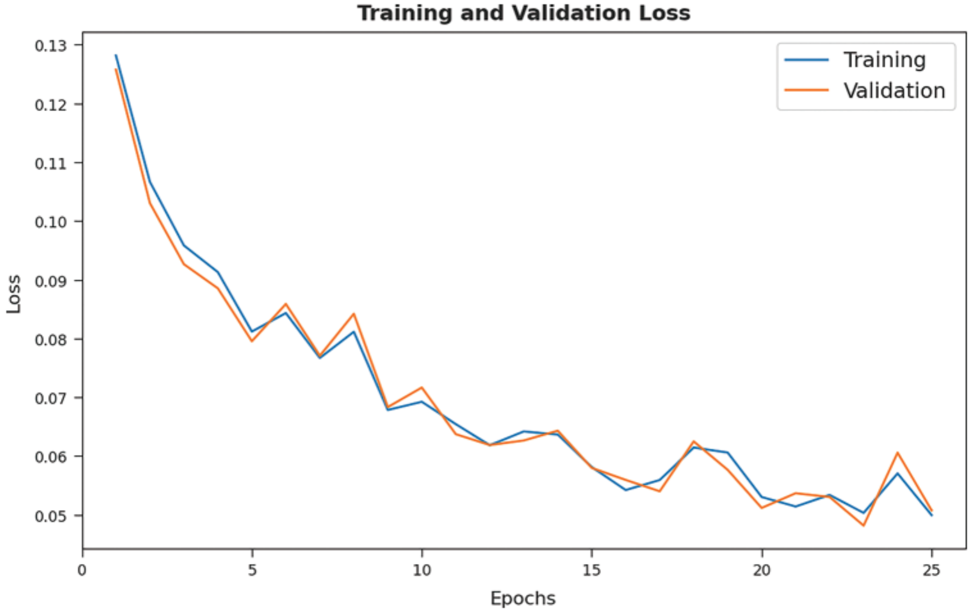


Figure 7: TL and VL graph analyses results of AITDC-IIoT technique

In order to validate the supremacy of the proposed AITDC-IIoT model, a detailed comparative analysis was performed against existing models and the results are shown in Tab. 3 [28].

Table 3: Comparative analysis results of AITDC-IIoT technique and other existing approaches

Methods	Precision	Recall	Accuracy	F-Score
GRU-RNN	96.75	94.40	96.87	97.88
AutoEncoders-EDSA	96.36	95.59	97.24	97.41
Multi-CNN Model	96.79	97.65	99.11	96.81
Cu-LSTMGRU-Cu-BLSTM	96.99	98.12	99.47	97.95
Cu-DNN-LSTM Model	94.91	97.70	98.86	97.51
Cu-DNN-GRU Model	96.11	97.01	99.16	97.57
AITDC-IIoT	97.58	98.53	99.83	98.05

Fig. 8 illustrates the comparative examination results of AITDC-IIoT model and other existing methods in terms of $prec_n$. The experimental values indicate that Cu-DNN-long Short Term Memory (LSTM) model achieved ineffectual outcome with the least $prec_n$ of 94.91%. Followed by, Gated Recurrent Unit (GRU)-Recurrent Neural Network (RNN), AutoEncoders-EDSA, Multi-CNN, Cu-LSTMGRU-Cu-BLSTM, and Cu-DNN-GRU models produced reasonably closer $prec_n$ values such as 96.75%, 96.36%, 96.79%, 96.99%, and 96.11% respectively. However, the proposed AITDC-IIoT model accomplished an enhanced performance with a maximum $prec_n$ of 97.58%.

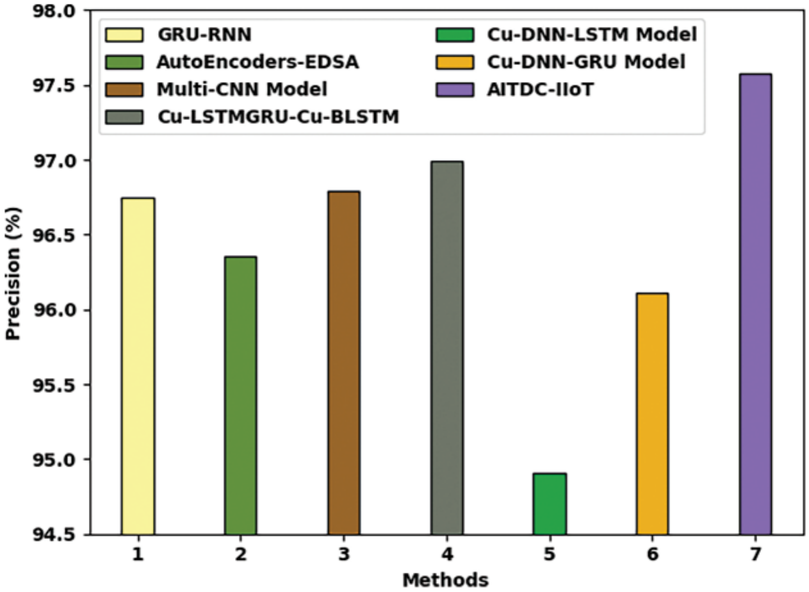


Figure 8: *Prec$_n$* analysis results of AITDC-IIoT technique and other recent algorithms

Fig. 9 showcases the comparative analysis results achieved by the proposed AITDC-IIoT model and other existing methods in terms of *reca$_l$*. The experimental values indicate that Cu-DNN-LSTM model showcased ineffectual outcomes with a minimal *reca$_l$* of 97.70%. Next, GRU-RNN, AutoEncoders-EDSA, Multi-CNN, Cu-LSTMGRU-Cu-BLSTM, and Cu-DNN-GRU models produced reasonably closer *reca$_l$* values such as 94.40%, 95.59%, 97.65%, 98.12%, and 97.01% correspondingly. But, the proposed AITDC-IIoT model accomplished an enhanced performance with a maximum *reca$_l$* of 97.58%.

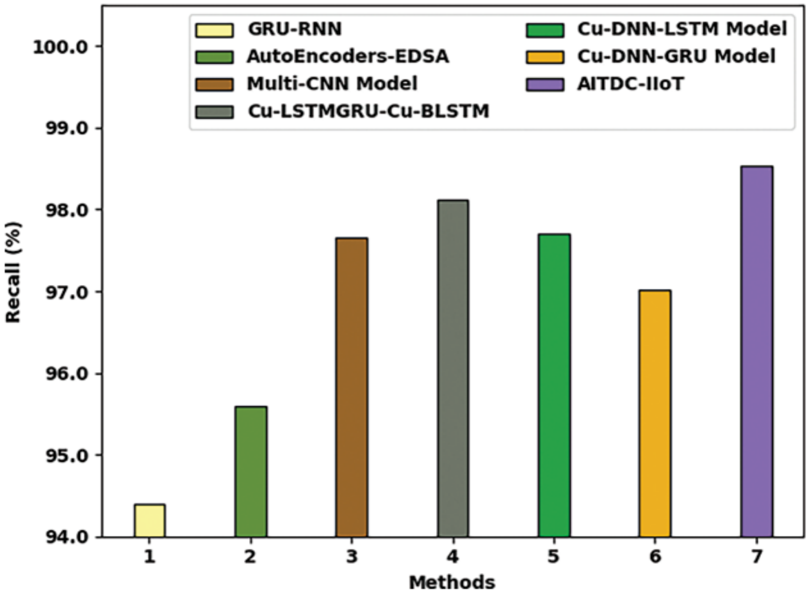


Figure 9: *Reca$_l$* analysis results of AITDC-IIoT technique and other recent algorithms

Fig. 10 depicts the comparative investigation results attained by the proposed AITDC-IIoT approach and other existing methods in terms of $accu_y$. The experimental values infer that Cu-DNN-LSTM model achieved ineffectual outcome with the least $accu_y$ of 98.86%. Likewise, GRU-RNN, AutoEncoders-EDSA, Multi-CNN, Cu-LSTMGRU-Cu-BLSTM, and Cu-DNN-GRU models produced reasonably closer $accu_y$ values such as 96.87%, 97.24%, 99.11%, 99.47%, and 99.16% correspondingly. However, the proposed AITDC-IIoT model accomplished enhanced performance with a maximum $accu_y$ of 99.83%.

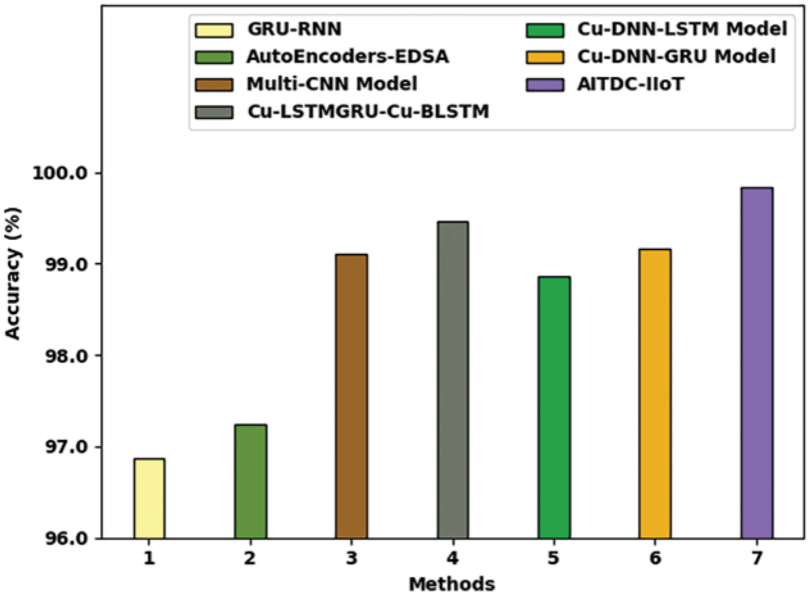


Figure 10: $Accu_y$ analysis results of AITDC-IIoT technique and other recent algorithms

Fig. 11 demonstrates the comparative analysis results achieved by AITDC-IIoT system and other existing systems in terms of F_{score}. The experimental values imply that Cu-DNN-LSTM algorithm attained ineffectual outcome with a minimal F_{score} of 97.51%. Along with that, GRU-RNN, AutoEncoders-EDSA, Multi-CNN, Cu-LSTMGRU-Cu-BLSTM, and Cu-DNN-GRU techniques produced reasonably closer F_{score} values such as 97.88%, 97.41%, 96.81%, 97.95%, and 97.57% respectively. At last, the proposed AITDC-IIoT methodology accomplished an enhanced performance with a maximum F_{score} of 98.05%.

Based on the results and discussion made above, it is apparent that the proposed AITDC-IIoT model is an excellent performer in terms of threat detection and classification compared to the existing techniques.

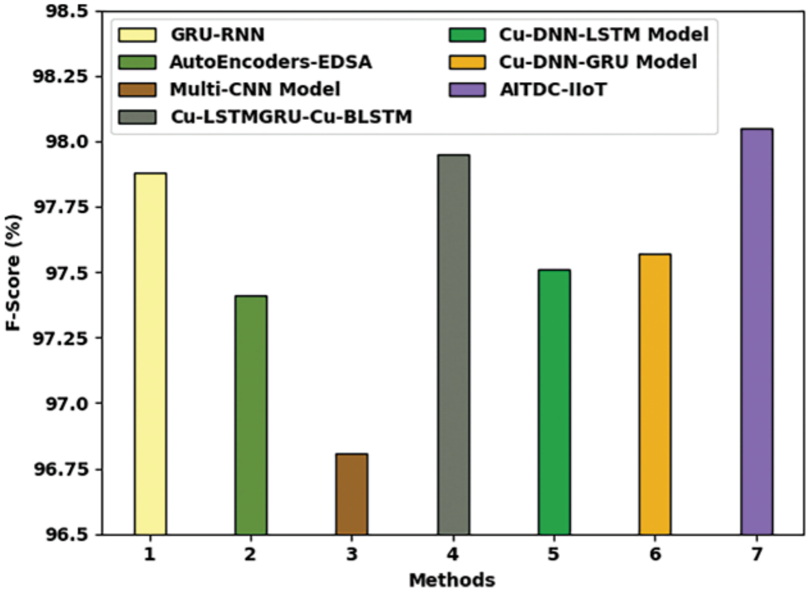


Figure 11: F_{score} analysis results of AITDC-IIoT technique with other recent algorithms

4 Conclusion

In this study, a new AITDC-IIoT model has been developed for proficient threat detection and classification. The presented AITDC-IIoT model initially pre-processes the input data so as to convert it to a compatible format. Followed by, WOA-FS model is involved to elect the subset of features. Moreover, CSO is employed with RVFL model for threat classification. Finally, CSO algorithm is applied to appropriately adjust the parameters related to RVFL model. The performance of the proposed AITDC-IIoT model was validated under benchmark datasets. The experimental results established the supremacy of the proposed AITDC-IIoT technique over recent approaches. Thus, AITDC-IIoT model can be employed for effectual threat detection and classification in IIoT environment. In future, the performance of the model can be enhanced by including outlier detection and clustering processes.

Funding Statement: The author received no specific funding for this study.

Conflicts of Interest: The author declares that he has no conflicts of interest to report regarding the present study.

References

[1] L. L. Dhirani, E. Armstrong and T. Newe, “Industrial IoT, cyber threats, and standards landscape: Evaluation and roadmap,” *Sensors*, vol. 21, no. 11, pp. 3901, 2021.

[2] K. Tsiknas, D. Taketzis, K. Demertzis and C. Skianis, “Cyber threats to industrial iot: A survey on attacks and countermeasures,” *IoT*, vol. 2, no. 1, pp. 163–186, 2021.

[3] J. E. Rubio, R. Roman and J. Lopez, “Integration of a threat traceability solution in the industrial internet of things,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6575–6583, 2020.

[4] H. Naeem, F. Ullah, M. R. Naeem, S. Khalid, D. Vasan *et al.,* "Malware detection in industrial internet of things based on hybrid image visualization and deep learning model," *Ad Hoc Networks*, vol. 105, pp. 102154, 2020.

[5] S. Sharmeen, S. Huda, J. H. Abawajy, W. N. Ismail and M. M. Hassan, "Malware threats and detection for industrial mobile-iot networks," *IEEE Access*, vol. 6, pp. 15941–15957, 2018.

[6] K. Yu, L. Tan, S. Mumtaz, S. A. Rubaye, A. A. Dulaimi *et al.,* "Securing critical infrastructures: Deep-learning-based threat detection in iiot," *IEEE Communications Magazine*, vol. 59, no. 10, pp. 76–82, 2021.

[7] P. Trakadas, P. Simoens, P. Gkonis, L. Sarakis, A. Angelopoulos *et al.,* "An artificial intelligence-based collaboration approach in industrial iot manufacturing: Key concepts, architectural extensions and potential applications," *Sensors*, vol. 20, no. 19, pp. 5480, 2020.

[8] M. A. Hawawreh, F. d. Hartog and E. Sitnikova, "Targeted ransomware: A new cyber threat to edge system of brownfield industrial internet of things," *IEEE Internet Things Journal*, vol. 6, no. 4, pp. 7137–7151, 2019.

[9] G. D. L. T. Parra, P. Rad and K. K. R. Choo, "Implementation of deep packet inspection in smart grids and industrial internet of things: Challenges and opportunities," *Journal of Network and Computer Applications*, vol. 135, pp. 32–46, 2019.

[10] A. H. Muna, N. Moustafa and E. Sitnikova, "Identification of malicious activities in industrial internet of things based on deep learning models," *Journal of Information Security and Applications*, vol. 41, pp. 1–11, 2018.

[11] M. M. N. Aboelwafa, K. G. Seddik, M. H. Eldefrawy, Y. Gadallah and M. Gidlund, "A Machine-learning-based technique for false data injection attacks detection in industrial IoT," *IEEE Internet Things Journal*, vol. 7, no. 9, pp. 8462–8471, 2020.

[12] M. M. Hassan, M. R. Hassan, S. Huda and V. H. C. d. Albuquerque, "A robust deep-learning-enabled trust-boundary protection for adversarial industrial iot environment," *IEEE Internet Things Journal*, vol. 8, no. 12, pp. 9611–9621, 2021.

[13] K. N. Qureshi, S. S. Rana, A. Ahmed and G. Jeon, "A novel and secure attacks detection framework for smart cities industrial internet of things," *Sustainable Cities and Society*, vol. 61, pp. 102343, 2020.

[14] M. M. Hassan, A. Gumaei, S. Huda and A. Almogren, "Increasing the trustworthiness in the industrial iot networks through a reliable cyberattack detection model," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6154–6162, 2020.

[15] M. A. Hawawreh and E. Sitnikova, "Industrial internet of things based ransomware detection using stacked variational neural network," in *Proc. of the 3rd Int. Conf. on Big Data and Internet of Things*, Melbourn VIC Australia, pp. 126–130, 2019.

[16] A. M. Hilal, J. S. Alzahrani, I. Abunadi, N. Nemri, F. N. Al-Wesabi *et al.,* "Intelligent deep learning model for privacy preserving iiot on 6 g environment," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 333–348, 2022.

[17] M. A. Alohali, F. N. Al-Wesabi, A. M. Hilal, S. Goel, D. Gupta and A. Khanna, "Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment," *Cognitive Neurodynamics*, 2022, https://doi.org/10.1007/s11571-022-09780-8.

[18] A. M. Hilal, M. A. Alohali, F. N. Al-Wesabi, N. Nemri, J. Hasan *et al.,* "Enhancing quality of experience in mobile edge computing using deep learning based data offloading and cyberattack detection technique," *Cluster Computing*, 2021, https://doi.org/10.1007/s10586-021-03401-5.

[19] A. A. Albraikan, S. B. Haj Hassine, S. M. Fati, F. N. Al-Wesabi, A. M. Hilal *et al.,* "Optimal deep learning-based cyberattack detection and classification technique on social networks," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 907–923, 2022.

[20] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2015.

[21] F. Essa, M. A. Elaziz and A. Elsheikh, "Prediction of power consumption and water productivity of seawater greenhouse system using random vector functional link network integrated with artificial ecosystem-based optimization," *Process Safety and Environmental Protection*, vol. 144, pp. 322–329, 2020.

[22] A. Muthumari, J. Banumathi, S. Rajasekaran, P. Vijayakarthik, K. Shankar *et al.,* "High security for deduplicated big data using optimal simon cipher," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1863–1879, 2021.

[23] R. Gopi, P. Muthusamy, P. Suresh, C. G. G. S. Kumar, I. V. Pustokhina *et al.,* "Optimal confidential mechanisms in smart city healthcare," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 4883–4896, 2022.

[24] D. A. Pustokhin, I. V. Pustokhin, P. Rani, V. Kansal, M. Elhoseny *et al.,* "Optimal deep learning approaches and healthcare big data analytics for mobile networks toward 5G," *Computers & Electrical Engineering*, vol. 95, pp. 107376, 2021.

[25] J. A. Alzubi, O. A. Alzubi, M. Beseiso, A. K. Budati and K. Shankar, "Optimal multiple key-based homomorphic encryption with deep neural networks to secure medical data transmission and diagnosis," *Expert Systems*, vol. 39, no. 4, pp. e12879, 2022.

[26] J. Kwiecień and M. Pasieka, "Cockroach swarm optimization algorithm for travel planning," *Entropy*, vol. 19, no. 5, pp. 213, 2017.

[27] A. Abeshu and N. Chilamkurti, "Deep learning: The frontier for distributed attack detection in fog-to-things computing," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 169–175, 2018.

[28] D. Javeed, T. Gao, M. Khan and D. Shoukat, "A hybrid intelligent framework to combat sophisticated threats in secure industries," *Sensors*, vol. 22, no. 4, pp. 1582, 2022.