A Novel Classifier for Cyber Attack Detection System in Industrial Internet of Things

Fathe Jeribi
College of Computer Science and Information Technology, Jazan University, Jazan, Saudi Arabia - Email: fjeribi@jazanu.edu.sa

Abstract

The usage of the Internet of Things (IoT) conception in the industrial sector along with applications is referred to as the Industrial Internet of Things (IIoT). Various applications have been subsumed in the IIoT. Nevertheless, cybercriminals mostly target these systems. Thus, here, a novel methodology of Cyber Attack Detection (CAD) system has been proposed in IIoT to overcome the aforementioned issue. UNSW-NB2015 and DS2OS are the two IIoT datasets utilized in this work. Initially, in both datasets, the missing values are replaced; subsequently, the feature extraction is performed. Next, by utilizing Poisson Distribution-based Naked Mole Rat Optimization Algorithm (PD-NMROA), the significant features are selected as of both datasets. After that, by employing MaHalanobis distance-based K-Means (MaH-KMeans) algorithm, the features extracted as of the datasets are normalized along with clustered. Eventually, to classify the data, the clustered features are inputted to the TanSwish - Restricted Boltzmann Dense Machines (TS-RBDMs). The experiential outcomes displayed that the proposed methodology obtained higher efficacy in contrast to the prevailing systems.

Keywords:
Poisson Distribution-based Naked Mole Rat Optimization Algorithm (PD-NMROA, MaHalanobis distance-based K-Means algorithm (MaH-KMeans), TanSwish - Restricted Boltzmann Dense Machines (TS-RBDMs), feature scaling, deep learning.

Introduction

Recently, the establishment of the IIoT has been brought about by the development in the industrial field by amalgamating the IoT, industrial system, along with cloud computing [1]. Acquiring the benefits of IoT technology in Industrial Control Systems (ICSs) is the major idea behind IIoT [2]. In the industrial process, to abate the human factor encumbrance and deal with the complicated industrial system process along with communications amongst them effectively, the ICS is utilized [3]. From several sensors, larger amounts of data can be gathered via IIoT for utilization all over the world. Retail, healthcare, transport, and automotive are several industries in which these applications are employed [4]. IIoT increases productivity, effectiveness, and operational efficiency significantly in numerous industries [5]. In general, cyber operations along with their effects are constrained to the cyber dimension in the conventional Information and Technology (IT) systems; however, special effects of normal operations outdo the limitations of the physical realm in IIoT [6]. To store along with to analyze big data engendered by the

IoT as well as IIoT systems, numerous data management, and security tools have been deployed in the cloud [7, 8]. The IIoT permits higher productivity; nevertheless, an attack on the infrastructure might be disastrous if it is not secured; thus, leading to an immense loss [9]. The development of IDS along with its security solutions brought about IIoT; however, to verify IIoT system requirements, these solutions have to be analyzed, checked, along with tuned utilizing labelled datasets; thus, espousing it in a real-world environment is highly challenging [10]. Thus, for the CAD system, a novel TS-RBDMS classifier is proposed in IIoT.

Application of Artificial Intelligence/Machine Learning in Cyber Security

Artificial Intelligence can be applied to security systems as a way to reduce cyber security threats. Here, a machine learns from the input data and makes a future prediction. It is utilized in email filters to sort out spam, banking software for detecting unusual transactions, internet search engines, websites for making personalized recommendations, and numerous apps on our phones like voice recognition. For cybersecurity, ML has become a significant technology. With ML, the patterns can be analyzed by cybersecurity, and learn from them to help in preventing similar attacks and respond to changing behaviour.

In the process of detecting cyber-attacks in IIoT, several benefits have been provided by the prevailing models; even then, there are certain uncertainties in those models; the drawbacks in the existing methodologies are enlisted below.

- There occurs exponential progression in computing times along with other complexities owing to the number of nodes and layers that augment the network structure.
- The huge cyber-attack classification problem, which evolved in the face of a real network application environment, is not addressed effectually by the prevailing system. Numerous classification tasks would result in lesser accuracy owing to the dynamic growth of datasets.
- Owing to higher energy consumption, time complexity, along with deprived algorithm design, there is a deficiency in QoS with energy efficiency.

Thus, for detecting cyber-attacks in IIoT, a novel TS-RBDMS classifier is proposed in this work. The proposed technique's major contributions and their significant are enlisted further:

- PD-NMROA is utilized for selecting the optimal features. This overcomes the problem of generating the same probability values.
- MaH-KMeans is proposed for clustering the features with non-convex shapes.
- TS-RBDMs are proposed to overcome the overfitting problem and reduce computation time.

The data are collected as of the datasets initially; then, they are pre-processed for replacing the missing values. After that, the features are extracted from the pre-processed data. Now, by utilizing PD-NMROA, the optimal features are selected.

Then, the selected features are scaled and then clustered by utilizing MaH-KMeans. Lastly, for classifying whether the data is attacked or non-attacked, TS-RBDMs are utilized.

The rest of the paper is organized as follows: the related works regarding the proposed model are reviewed in section 2; the proposed methodology is explicated in section 3; the results and discussion is demonstrated in section 4; lastly, section 5 offers conclusions and future work.

Literature Review

Zil e. Huma et al. [11] presented a Hybrid Deep Random Neural Network (HDRaNN) aimed atCADin the IIoT. The applications of DRaNN, as well as Multilayer Perceptron (MLP), were utilized by the HDRaNN. The experimental outcomes displayed the presented model's accuracy. Nevertheless, owing to Deep Learning (DL) ability, the developed model's computation time is high.

Shahid latif et al. [12] developed a lightweight Random Neural Network (RaNN)-centric prediction model. Attacks had been detected precisely by the presented RaNN model. The experiential outcomes demonstrated that the model attained a higher accuracy. However, merely limited attacks were deemed by this system.

Shahid Latif et al. [13] illustrated a DRaNN-centric scheme intended for intrusion detection in IIoT. For classifying the varied sorts of attacks, the DRaNN was employed. The evaluation outcomes exhibited that the presented methodology possessed a higher attack detection rate. Nevertheless, the system had a higher complexity.

Muna AL-Hawawreh et al. [14] suggested an anomaly detection mechanism meant for Internet ICSs (IICSs) grounded on DL models. The execution of a consecutive training process utilizing a deep auto-encoder was enclosed in this model. The experiential outcomes displayed that when analogized with the prevailing methodologies, the presented one achieved a higher detection rate along with a lower False Positive Rate (FPR). Nevertheless, owing to the NN's narrow waist structure, the model had a higher training time.

Radhakrishna Vangipuram et al. [15] developed a machine learning strategy aimed at imputation as well as anomaly detection in an IoT environment. The imputed datasets acquired by utilizing K-Means, F-Kmeans, and developed imputation methodologies were considered to perform classification. The experiential outcomes displayed that in contrast to the conventional classifiers, the presented model's performance was far better. However, the system had a higher computation cost.

Di Wu et al [16] recommended a Long Short-Term Memory (LSTM)-Gauss-NBayes model, which was a synergy of the LSTM Neural Network (LSTM-NN) and the Gaussian Bayes model for outlier detection in IIoT. In this, to detect the prediction error, the presented LSTM model was utilized. The experimental results demonstrated that optimistic results were obtained by this model. Nevertheless, more memory was utilized by this model

to train.

Tran Viet Khoa et al. [17] developed a collaborative learning-centric Intrusion Detection System (IDS). To classify the packets into normal and abnormal behaviors, the Deep Belief Network (DBN) was utilized. The experiential outcomes displayed that when analogized with traditional machine learning methodologies, the presented model attained a better performance. However, for a smaller number of data, the DBN was not appropriate.

Faezeh Farivar et al. [18] recommended a model to determine along with to reimburse for attacks hurled in the forward link of nonlinear Cyber-Physical Systems (CPSs) utilizing the intelligent variable structure control. For estimating the attack, Neural Network (NN) estimator was utilized. The simulation outcomes proved the developed system's efficacy. Nevertheless, the system had higher training time owing to NN's narrow waist structure.

Yanmiao Li et al. [19] illustrated a DL model for intrusion detection utilizing a multi-Convolutional Neural Network (multi-CNN) fusion methodology. For classification, the CNN was presented into the IDS by utilizing the flow data visualization model. The experimental outcomes that the presented system possessed a higher accuracy of multi-CNN. However, owing to the existence of a vanishing gradient problem in CNN, the data was learned gradually by the developed methodology.

Muna AL-Hawawreh and Elena Sitnikova [20] presented a detection system grounded on the stacked Variational Auto-Encoder (VAE) with a fully connected NN. The latent structure of system activities was learned by the VAE with a fully connected NN; in addition, it exposed the ransomware behavior. The outcomes displayed that a superior detection rate was attained by the presented model in contrast to the prevailing methodologies. However, as a result of the auto encoder's blurry characteristics, an accurate output was not provided by the system.

Proposed Cyber Attack Detection Method

For effective detection along with classification of attack or non-attack, a novel TS-RBDM Classifier has been proposed in this paper. Here, initially, the features are extracted. Next, as of extracted features, the significant features are selected. After that, for the classification of attacks or non-attacks, the features being selected are inputted into the TS-RBDM Classifier. Figure 1 exhibits the block diagram of the proposed methodology.

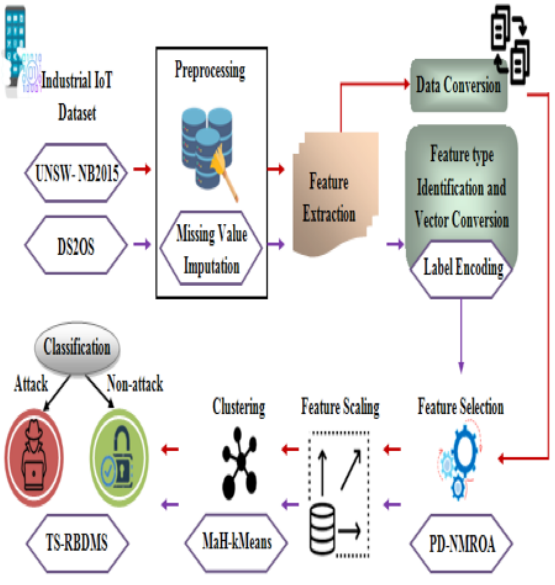

Fig. 1. Block Diagram of the Proposed Methodology

UNSW-NB2015 and DS2OS are the datasets utilized by the proposed CAD system. Here, owing to the non-existence of values in those datasets, the missing value imputation is executed. To retain most data of the dataset, the missing data is replaced with certain substitute values by performing the imputation process. Let the UNSW-NB2015 data set be U . In this, the missing value is substituted with the same attribute values that are signified in the dataset. It is formulated as:

$$\hbar_i^{mi} \in U = \hbar_i \leftrightarrow U_{sameatt} \quad (1)$$

Where, the missing value and output of the missing value are specified as $\hbar_i$ and $\hbar_i^{mi}$, the same attribute value in the dataset is notated as $U_{sameatt}$.

Let the DS2OS dataset be D . Here, a few values are not assigned. Additionally, these columns are substituted with certain meaningful value λ_i^{mi} , which is expressed as:

$$\lambda_i^{mi} \in D \to \delta(\lambda_i) \quad (2)$$

Where, the replacement function is symbolized as δ , and the column that represents the True, False, Twenty, and None are substituted with 1.0, 0.0, 20.0, and 0.0, correspondingly. The data as of both datasets U_{pre} and D_{pre} are obtained following the completion of pre-processing.

Feature extraction

More information about the dataset can be obtained swiftly with the aid of feature extraction (attributes extraction). Therefore, from U_{pre} , protocol, service state, standard mean, deviation mean, et cetera are the key attributes being extracted. The extracted attributes $f_n \in U_{pre}$ are expressed as:

$$f_n^{U_{pre}} = \{f_1^{U_{pre}}, f_2^{U_{pre}}, f_3^{U_{pre}}, \ldots\ldots f_N^{U_{pre}}\} \quad (3)$$

In this process, from D_{pre} , the attributes like address, source ID, destination, type, et cetera are extracted; eventually, the output $f_n^{D_{pre}}$ is attained.

Data conversion

In this, as the strings are extant in the dataset, the extracted attributes $f_n^{U_{pre}}$ are transmuted into numbers. Moreover, those strings are not processed in the classifier. Thus, the strings are converted into numbers. In the dataset, the numbers are assigned for every single string to perform this conversion $f_{i(con)}^{U_{pre}}$. It is modelled as:

$$f_{i(con)}^{U_{pre}} = \Delta\left(f_i^{U_{pre}}\right) \quad (4)$$

Where, the conversion function is represented as Δ .

Feature type identification and vector conversion

Here, the feature type is detected as whether it is a string or vector in $f_n^{D_{pre}}$. If the feature is detected as a string then the string features are partitioned; in addition, they are transmuted into the vector format by encoding. The process of transmuting the labels into numeric by assigning the numeral values to strings in alphabetical order is termed label encoding. It is formulated as:

$$f_{n(con)}^{D_{pre}} = S\left(f_i^{D_{pre}}\right) \quad (5)$$

Where, the vector conversion's output is specified as $f_{n(con)}^{D_{pre}}$, the state is signified as S, which illustrates the numerals.

Feature selection

In this, by utilizing the PD-NMROA, the features are selected as of $f_n^{U_{pre}}$. Naked mole rats' behavioral characteristics are the major concept behind the NMROA. Regarding the breeding probability, the breeder group is selected; here, the uniform distribution random process is utilized to perform initialization. Betwixt the ranges with the same probability, the population is created. In the initialization step, the Poisson Distribution model is replaced to overcome the problem of such generation of the same probability values in the prevailing algorithm.

(a) Population initialization

Firstly, the NMR's population is engendered randomly in *d* dimensional vector where the features being extracted are regarded as a number of NMR; furthermore, by utilizing the Poisson Distribution system, every single NMR is initialized as:

$$f_{uv}^{U_{pre}} = \frac{e^{-\ell} * \ell^{n\left(f_n^{U_{pre}}\right)}}{n\left(f_n^{U_{pre}}\right)} \tag{6}$$

Where, the u^{th} NMR in v^{th} dimension is specified as $f_{uv}^{U_{pre}}$, the number of NMRs is signified as $n\left(f_n^{U_{pre}}\right)$, and the average number of $f_n^{U_{pre}}$ occurrences of is notated as $\ell^{n\left(f_n^{U_{pre}}\right)}$.

(b) Calculating fitness value

Regarding the classifier's accuracy, the objective function along with its fitness value is computed after initializing the population. It is measured as:

$$\wp_n = \Gamma\left(f_{(uv)}^{U_{pre}}\right) \tag{7}$$

Where, the output of the n^{th} fitness function of th number of NMR is symbolized as $\wp_n$, and the fitness function is represented as Γ. The population is further partitioned into breeder and worker concerning the fitness value; moreover, the queen (q) is also estimated.

(c) Worker group

Here, by enhancing their fitness, NMR workers attempt to turn into breeders to mate with the queen. Subsequently, regarding its own experience along with local information, the NMR's new solution is generated; in addition, for the new solution, the fitness value is computed. Next, the new solution is forwarded to the breeder group. The new solution will be accepted if it is better than the preceding solution. Or else, it will be continued with the previous solution. Here, the new solution is spawned as:

$$\omega_u(I+1) f_{uv}^{U_{pre}} = \omega_u(I) + \alpha\left(\omega_x(I) - \omega_y(I)\right) \tag{8}$$

Where, the u^{th} worker in $(I+1)^{th}$ iteration is specified as $\omega_u(I+1)$, the u^{th} worker in I^{th} iteration is indicated as $\omega_u(I)$, the uniform distribution in the range of $[0,1]$ is denoted as a, and the random solutions from the worker's group are represented as $\omega_x(I)$ and $\omega_y(I)$.

(d) Breeder group

Every single breeder NMR in this breeder

group attempts to update its position with an intention to stay as a breeder, additionally, to be selected as the breeder for mating. Regarding the breeding probability, the breeder NMRs are updated in terms of the overall best in the range of $[0,1]$. The breeder will be sent to the worker's group if its NMR is not capable to ameliorate its fitness. The breeders update their position as:

$$B_u(I+1)f_{uv}^{U_{pre}} = (1-\alpha)B_u(I) + \alpha(q - B_u(I)) \quad (9)$$

Where, the u^{th} breeder in $(I+1)^{th}$ iteration is notated as $B_u(I+1)$, and the u^{th} breeder in I^{th} iteration is illustrated as $B_u(I)$.
Until satisfying the termination condition, the whole search procedure will be continued iteratively. Next, the significant features are selected just like the best breeder selected utilizing the PD-NMROA. It is modelled as:

$$f_{n(sel)}^{U_{pre}} = \{f_{1(sel)}^{U_{pre}}, f_{2(sel)}^{U_{pre}}, f_{3(sel)}^{U_{pre}}, \ldots\ldots f_{N(sel)}^{U_{pre}}\} \quad (10)$$

Where, the number of selected features is specified as $f_{n(sel)}^{U_{pre}}$. In the same manner, by utilizing the same algorithm that is utilized for the feature selection in the UNSW-NB2015 dataset, the features are extracted $f_n^{D_{pre}}$; consequently, the selected features' output in the DS2OS dataset $f_{n(sel)}^{D_{pre}}$ is obtained.

Feature scaling

The range of variables in the selected features is extremely varied; so to unify feature ranges in data, a mechanism is utilized, which is termed the feature-scaling model. Therefore, the proposed model in which the features within the range are normalized utilizing robust scaling for the UNSW-NB2015 dataset $f_{n(nor)}^{U_{pre}}$ is formulated as:

$$f_{n(nor)}^{U_{pre}} = \frac{f_{i(sel)}^{U_{pre}} - \left(f_{i(sel)}^{U_{pre}}\right)^*}{\Psi} \quad (11)$$

Where, the median of $f_{n(nor)}^{U_{pre}}$ is defined as $\left(f_{i(sel)}^{U_{pre}}\right)^*$, and the Inter Quartile Range is notated as Ψ. Similarly, the features are normalized for $f_{n(sel)}^{D_{pre}}$ and the output $f_{n(nor)}^{D_{pre}}$ is attained. 80% of the normalized features are utilized for training whereas the remaining 20% are utilized for testing.

Clustering

By utilizing the MaH-KMeans, the features $f_{n(nor)}^{U_{pre}}$ are clustered with regard to protocol, state, id, et cetera following the normalization process. The K Means segmentation is the technique of vector quantization; the major intention of this model is to partition the number of features into ϕ clusters where every single feature corresponds to the cluster with the nearest mean.
(i) Selecting the number of clusters, (ii) Initializing centroids, (iii) Assigning features to the nearest value, and (iv) Reinitializing centroids are the steps undergone by the algorithm for segmenting the scaled features. Generally, the basic Euclidean distance is utilized for the partitioning of features in clustering. Nevertheless, for the detection of clusters with non-convex shapes, this model is not appropriate. Here, the model is replaced with the MaHalanobis distance technique. The steps in MaH-KMeans are:

- The number of clusters, which is estimated by their centroids, is selected.

The centroid is the cluster's center. However, primarily, the feature's exact center is not known. Thus, to define every single cluster, the centroids C_ϕ can be selected randomly as:

$$C_\phi = \{C_1, C_2, C_3, \ldots\ldots C_N\} \quad (12)$$

- The feature $f_{i(nor)}^{U_{pre}}$ is assigned to the closest centroid.
- The distance betwixt the assigned feature and centroid is computed utilizing the MaHalanobis distance strategy. It is expressed as,

$$\Phi^2 = \left(f_{i(nor)}^{U_{pre}} - C_\phi\right)^T * m^{-1} * \left(f_{i(nor)}^{U_{pre}} - C_\phi\right) \quad (13)$$

Where, the MaHalanobis distance technique's output is specified as Φ^2, and the inverse covariance matrix of C_ϕ is symbolized as m^{-1}.

- A cluster is chosen for features where the distance betwixt the feature and centroid is minimum.
- By computing the average of all the data points of that cluster, the centroids are reinitialized.

$$C_\phi = \frac{1}{n\left(f_{n(nor)}^{U_{pre}}\right)} \sum f_{n(nor)}^{U_{pre}} \quad (14)$$

Where, the number of features is denoted as $n\left(f_{n(nor)}^{U_{pre}}\right)$.

This process is repeated until no alterations occur in clusters. The clustered output $\aleph_n^{U_{pre}} \in f_{n(nor)}^{U_{pre}}$ is attained via this process. Likewise, based on source id, type, address, et cetera, the features presented in $f_{n(nor)}^{D_{pre}}$ are clustered by employing the same process; furthermore, the output $\aleph_n^{D_{pre}} \in f_{n(nor)}^{D_{pre}}$ is acquired. The pseudo-code of MaH-KMeans is:

Input: Normalized features $f_{n(nor)}^{U_{pre}}$

Output: clustered features $\aleph_n^{U_{pre}} \in f_{n(nor)}^{U_{pre}}$

Begin

Initialize C_ϕ, Φ^2 and $n\left(f_{n(nor)}^{U_{pre}}\right)$

While $(\phi = 1)$

Select number of centroids

Assign feature to the closest centroid

For each feature, **do**

Compute distance Φ^2

End for

Reinitialize centroids

End while

Return $\aleph_n^{U_{pre}} \in f_{n(nor)}^{U_{pre}}$

Classification

In this, to classify whether the data is attacked or non-attacked, the clustered features $\aleph_n^{U_{pre}}$ are inputted into the TS-RBDMs. In the context of unsupervised learning, the Restricted Boltzmann Machine (RBM), a latent-variable generative model, is utilized most frequently. It comprises hidden H_g as well as visible units ε_k and contains a weight matrix in the size of $l \times z$, which is associated betwixt visible and hidden units. It has no output layer. However, the prevailing methodologies are integrated with some additional layers like MLP, drop layer, and so on; thus, resulting in a higher computation time along with an overfitting problem. Thus, a dense layer is proposed in this work to address this issue; this layer compensates for all the characteristics of the aforementioned layers. Figure 2 exhibits the architecture of TS-RBDMs.

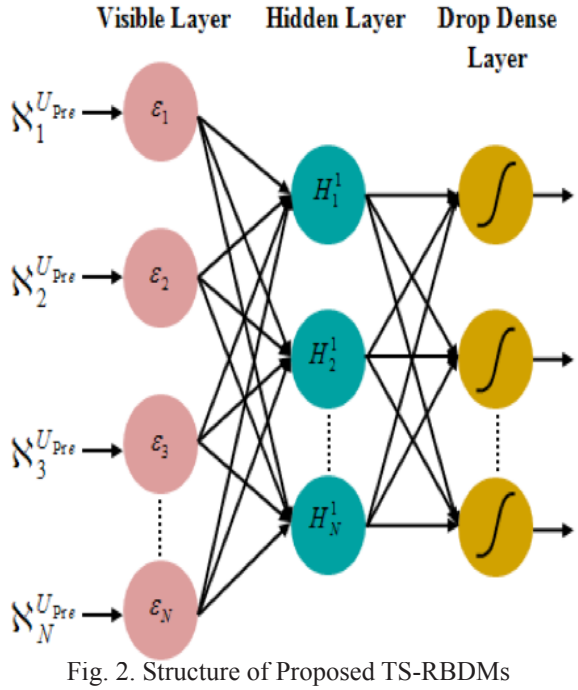

Fig. 2. Structure of Proposed TS-RBDMs

Primarily, with the input features $\aleph_n^{U_{pre}}$, the TS-RBDMs' first layer is pre-trained. By means of the energy function, the TS-RBDMs' learning process is performed. The energy function $E(\varepsilon, H)$ is proffered as:

$$E(\varepsilon,H)=-\sum_k a_k\varepsilon_k-\sum_g b_g H_g-\sum_k\sum_g \varepsilon_k W_{k,g}H_g \qquad (15)$$

Where, the bias values are represented as a_k and b_g, the element weight is specified as $w_{k,g}$,and the number of units is notated as k,g.

The $E(\varepsilon, H)$ is formulated in the matrix representation as:

$$E(\varepsilon,H)=a^T\varepsilon-b^T H-\varepsilon^T WH \qquad (16)$$

The first hidden layer's output is inputted into the subsequent hidden layer after obtaining all the parameters of the first hidden layer. Next, the '2' hidden layers are deemed as new TS-RBDMs. Similarly, by updating the bias along with weight values continuously, TS-RBDMs' every single layer is trained separately. The weight and bias values of the first hidden layer's visible unit are updated as:

$$H^1(\varepsilon_k)=\chi(a_k+\sum_{k,g} W_{k,g}\varepsilon_k(\aleph_n^{U_{pre}})) \qquad (17)$$

Where, the TanSwish activation function in the drop dense layer is specified as χ ; here, every single neuron gets input as of all the neurons of the preceding layer; moreover, they are changed into a single output. Therefore, in this work, the overfitting problem is prevented. The TanSwish activation function is expressed as:

$$\chi=\frac{\aleph_n^{U_{pre}}\left(e^{\aleph_n^{U_{pre}}}-e^{-\aleph_n^{U_{pre}}}\right)}{1+e^{-\aleph_n^{U_{pre}}}\left(e^{\aleph_n^{U_{pre}}}+e^{-\aleph_n^{U_{pre}}}\right)} \qquad (18)$$

After that, the subsequent hidden layer's visible unit is fed with the output being computed. The output is achieved by the continuous updation along with training till the last layer of TS-RBDMs; subsequently, the attacked or non-attacked data in the IIoT system is retrieved. Furthermore, to predict whether the data is attacked or non-attacked, the same process is proceeded for $\aleph_n^{D_{pre}}$.

Result and Discussion

Here, to analyze the proposed methodology's performance, various experiments were performed.

The data used in the proposed work is obtained from UNSW-NB15 and DS2OS datasets. The proposed model is executed in PYTHON.

Dataset description

- UNSW-NB15
  It is a network intrusion dataset. Information pertinent to Denial of Service (DoS), raw network packets, worms, Backdoors, and Fuzzers attacks is included in this dataset. With multiple attack records, it is separated into training and testing datasets. The number of records in the training set is 175,341 records, whereas in the testing set are 82,332 records from the different types, attack and normal. Argus and Bro-IDS tools extracted a total of 49 features comprising packet-centric and flow-centric features from the raw network packets [22]. Packet-based features are extracted from the packet header along with its payload. Conversely, flow-centered features are generated utilizing the sequencing of packets, from a source to a destination, traveling in the network.
- DS2OS
  Information attained as of network traces is included in this dataset. This data is employed for the evaluation of different anomalies in the network. Here, from numerous organizations conducting varying services, the information is obtained. The dataset encompasses a total of 357952 samples with 10017 anomalous and 347935 normal values [23]. It contains 13 features and '7' various sorts of attacks like malicious operations, wrong setup, scan, denial of service, malicious control, spying, along with data type probing attacks.

Performance analysis for UNSW-NB15 dataset

Here, regarding feature selection, classification accuracy, along with clustering time, the proposed CAD model's performance is assessed.

Performance evaluation of proposed PD-NMROA

Naked Mole Rat Optimization Algorithm (NMROA), Whale Optimization Algorithm (WOA), Crow Search Algorithm (CSA), and Fish Swarm Optimization (FSO) Algorithm are the prevailing methodologies with which the proposed PD-NMROA is analogized regarding fitness vs.iteration.

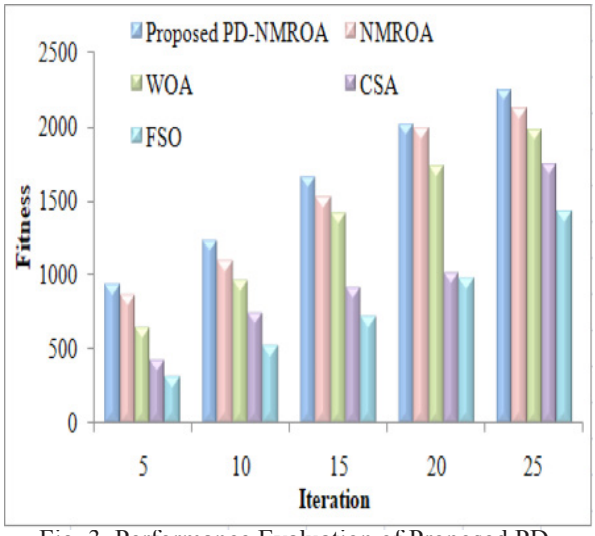

Fig. 3. Performance Evaluation of Proposed PD-NMROA

As per figure 3, it is evident that the proposed model's fitness value increases with the increase in the number of iterations. In the proposed model, for the varying number of iterations like 5, 10, and 25, the fitness values obtained are 948, 1235, and 2245, respectively; however, the reduced fitness values attained by the prevailing WOA are 653 (5), 970 (10), and so on. In

the same manner, only lower range values are obtained by the other prevailing NMROA, CSA, and FSO methodologies. Thus, it is proved that in contrast to the prevailing methodologies, the proposed one attained a higher performance.

Performance evaluation of proposed MaH-Kmeans

Here, regarding clustering time, the proposed model's performance is analogized with the prevailing KMeans (KM), Birch, Fuzzy C Means (FCM), and Mean Shift (MS) methodologies.

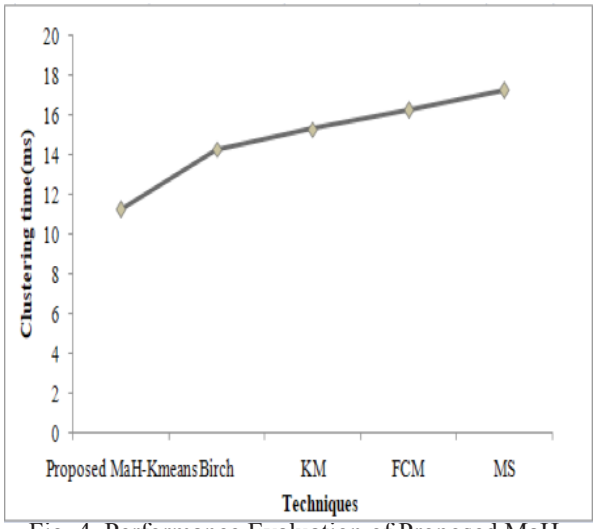

Fig. 4. Performance Evaluation of Proposed MaH-KMeans

Regarding clustering time, the proposed model's performance is evaluated in figure 4. It is evident that a lower clustering time of 11.2365ms was attained by the proposed MaH-Kmeans whereas the clustering time obtained by the prevailing Birch, KM, FCM, and MS methodologies are 14.2563ms, 15.2896ms, 16.2358ms, and 17.2356ms, correspondingly, which are higher than that of the proposed model. Consequently, it is concluded that when analogized with the prevailing methodologies, the proposed model is highly secure as well as faster.

Superiority measure of proposed TS-RBDMS

Here, DRaNN [13], DAE-DFFNN (Deep Auto-Encoder-Deep Feed Forward Neural Network) [14], and HDRaNN [11] are conventional methodologies with which the proposed TS-RBDMS is analogized regarding the metrics like Accuracy.

Table 1: Comparative analysis of proposed TS-RBDMS

Techniques	Accuracy (%)
Proposed TS-RBDMS	99.68
DRaNN [13]	99.54
DAE-DFFNN [14]	92.48
HDRaNN [11]	90.21
DAE-DFFNN [21]	98.9

With regard to the accuracy, the proposed TS-RBDMS is analogized with other prevailing methodologies and is tabulated in table1. The proposed model attained the highest accuracy of 99.68% whereas the least accuracy of 90.21% was obtained by the conventional HDRaNN mechanism. Similarly, the performance metrics differ for other classifiers also. Thus, it is evident that better performance was achieved by the proposed model than the prevailing methodologies.

Table 2 depicts the comparative analysis of the proposed and the conventional systems regarding precision, recall, and f-measure. The precision, recall, and f-measure attained by the proposed approach are 99.86%, 99.55%, and 99.54%,

Table 2: Comparative analysis of the proposed model in terms of precision, recall, and f-measure

Techniques/ Metrics	Precision (%)	Recall (%)	F-Measure (%)
Proposed TS-RBDMs	99.86	99.55	99.54
HDRaNN [11]	99.07	98.98	99.02
DAE-DFFNN [21]	99.8	99.6	96.7

correspondingly, which are higher than the prevailing approaches, namely HDRaNN and DAE-DFFNN. Thus, it is concluded that the proposed model is more efficient in attack detection in IIoT.

Performance analysis for DS2OS dataset

In this section, the proposed methodology's performance is assessed concerning feature selection, clustering time, along with classification accuracy.

Performance evaluation of proposed PD-NMROA

The proposed model is analogized with the prevailing methodologies regarding fitness vs. iteration in figure 5.

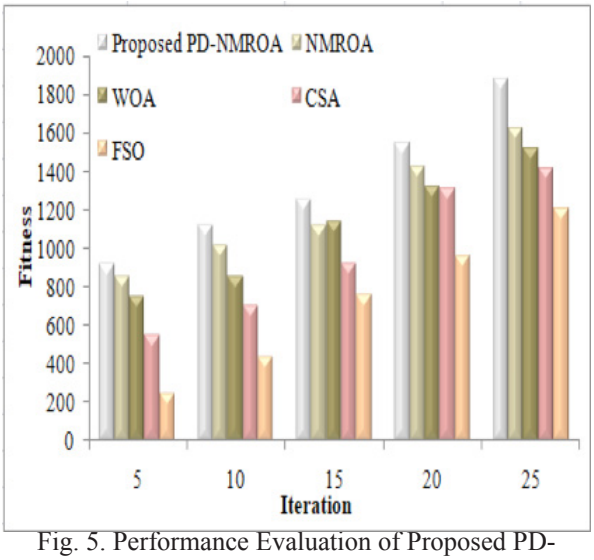

Fig. 5. Performance Evaluation of Proposed PD-NMROA

Figure 5 demonstrates that a higher fitness value was obtained by the proposed model in contrast to existing methodologies. For 5 iterations, a fitness value of 920 is acquired by the proposed PD-NMROA; conversely, for the same number of iterations, the conventional WOA obtained 850 fitness values. Similarly, the fitness values differ for the other conventional models also. Therefore, the proposed model outshines the existing methodologies.

Performance evaluation of proposed MaH-Kmeans

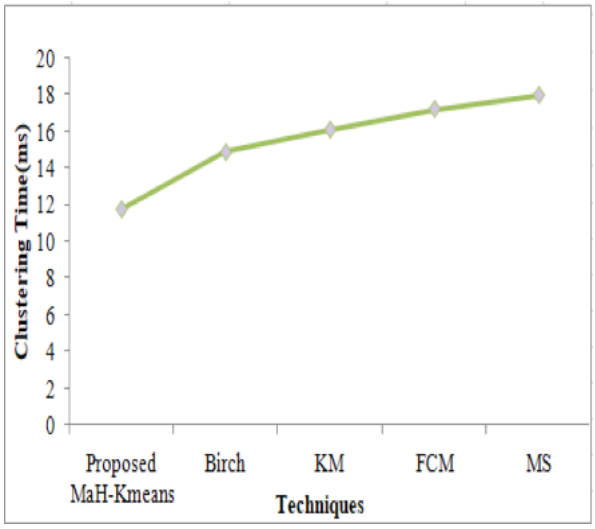

Fig. 6. Performance Evaluation of Proposed MaH-Kmeans

Figure 6 exhibits the superiority measure of the proposed model with regard to clustering time. Here, the proposed MaH-Kmeans attained a clustering time of 11.7892ms, which is lower than the clustering time of 14.8914ms obtained by the prevailing Kmeans model. In the same manner, the clustering time varies for the other methodologies also. Therefore, it is

evident that when analogized with the prevailing methodologies, the proposed model achieved better performance.

Superiority Measure of proposed TS-RBDMS

Here, concerning the accuracy metric, the proposed TS-RBDMS's performance is compared with other prevailing algorithms like RaNN [12], and HDRaNN [11].

Table 3: Comparative analysis of proposed TS-RBDMS based on the accuracy (%)

Techniques	Accuracy (%)
Proposed TS-RBDMS	99.70
HDRaNN [11]	98
RaNN [12]	99.20

Table 3 compares the accuracy of the proposed TS-RBDMS with existing works like HDRaNN and RaNN. The model having higher accuracy will be the best model. In accordance with this, the accuracy achieved by the proposed model was 99.70% whereas the accuracy values attained by the prevailing models are HDRaNN (98%), and RaNN (99.20%). Therefore, in contrast to the traditional models, the proposed one achieved better performance.

Regarding precision, recall, and f-measure, the performance analysis of the proposed and the prevailing models are represented in table 4. The proposed model attains a precision of 99.74%, recall of 99.66%, and f-measure of 99.67%, which are higher when analogized with the prevailing techniques like HDRaNN and RaNN. The outcomes exhibited that the proposed mechanism displays better performance than the conventional frameworks in attack detection.

Table 4: Comparative analysis of the proposed model in terms of precision, recall, and f-measure

Techniques/ Metrics	Precision	Recall	F-Measure
Proposed TS-RBDMs	99.74	99.66	99.67
HDRaNN [11]	98.25	98.36	98.3
RaNN [12]	99.08	99.16	99.04

Figure 7 displays the computational complexity of the proposed TS-RBDMs. The best training and testing accuracies of the proposed model are achieved at 99.85% and 99.70%, correspondingly. Similarly, the best training and testing accuracies of DNNBoT are achieved at 90.71% and 90.54%, respectively [24]. Likewise, the best training and testing accuracies of PCCNN are 99.34% and 98.64%, respectively [25]. On comparing these values, the proposed model shows better performance in the detection of attacks in IIoT.

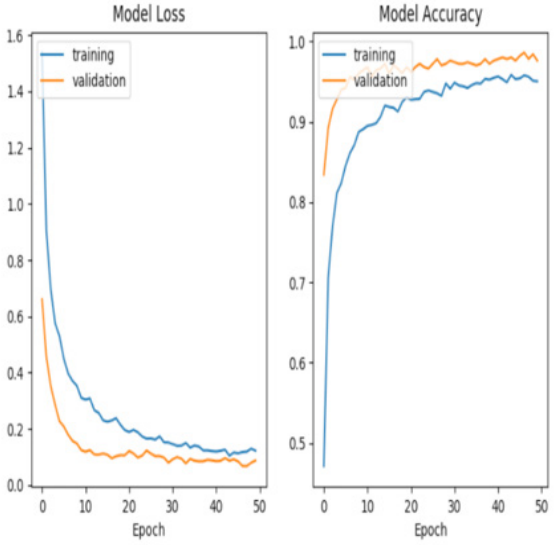

Fig. 7. Computational Complexity of Proposed TS-RBDMs

Conclusion

To detect attacks in IIoT, a novel TS-RBDMS model has been proposed in this work. (i) feature selection, (ii) clustering, and (iii) classification are the operations undergone by the system. After that, the experimental evaluation is performed; here, to validate the proposed model's efficacy, the performance along with a comparative analysis of the proposed is done in comparison with the prevailing methodologies regarding certain performance metrics. Several uncertainties along with attacks are recognized accurately by the proposed model. For the evaluation, UNSW-NB15 and DS2OS datasets are utilized. In this, the proposed TS-RBDMS attained an accuracy of 99.68% for UNSW-NB15 and 99.70% accuracy for DS2OS datasets, in that order. Therefore, to detect cyber-attacks in IIoT, major support was provided by the proposed framework. But the model shows low energy efficiency in real-time data sensing time. So, the work may concentrate on the data security process for non-attacked data, and energy efficiency will be concentrated on real-time data sensing time in the future.

Conflict of Interest

None

References

1. Xinghua Li, Mengfan Xu, Pandi Vijayakumar, Neeraj Kumar and Ximeng Liu, "Detection of low-frequency and multi-stage attacks in industrial internet", IEEE Transactions on Vehicular Technology, vol. 69, no. 8, pp. 8820-8831, 2020.
2. Maede Zolanvari, Marcio A Teixeira, Lav Gupta, Khaled M Khan and Raj Jain, "Machine learning based network vulnerability analysis of industrial internet of things", EEE Internet of Things Journal, vol. 6, no. 4, pp. 6822-6834, 2018.
3. Gamal Eldin I Selim, EZZ El-Din Hemdan, Ahmed M Shehata, Nawal A El-Fishawy, "Anomaly events classification and detection system in critical industrial internet of things infrastructure using machine learning algorithms", Multimedia Tools and Applications, vol. 80, no. 8, pp. 12619-12640, 2021.
4. Muna Al-Hawawreh, Elena Sitnikova and Neda Aboutorab, "X-IIoTID a connectivity- and device-agnostic intrusion dataset for industrial internet of things", IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3962-3977, 2021.
5. Sharmistha Nayak, Nurzaman Ahmed and Sudip Misra, "Deep learning-based reliable routing attack detection mechanism for industrial internet of things", Ad Hoc Networks, vol. 123, pp. 1-11, 2021.
6. Abdulrahman Al-Abassi, Hadis Karimipour, Ali Dehghantanha and Reza M Parizi, "An ensemble deep learning-based cyber-attack detection in industrial control system", IEEE Access, vol. 8, pp. 83965-83973, 2017.
7. Bela Genge, Piroska Haller and Calin

Enachescu, "Anomaly detection in aging industrial internet of things", IEEE Access, vol. 4, pp. 1-14, 2016.

8. Truong Thu Huong, Ta Phuong Bacb, Dao Minh Longa, Tran Duc Luonga, Nguyen Minh Dana, Le Anh Quanga, Le Thanh Conga, Bui Doan Thanga and Kim Phuc Tran, "Detecting cyberattacks using anomaly detection in industrial control systems: A Federated Learning approach", Computers in Industry, vol. 132, no. 7, pp. 1-16, 2021.
9. Mohamed Abdel-Basset, Victor Chang, Hossam Hawash, Ripon K Chakrabortty and Michael Ryanmn, "Deep-IFS intrusion detection approach for IIoT traffic in fog environment", IEEE Transactions on Industrial Informatics, vol. 17, no. 11, pp. 7704-7715, 2020.
10. Yash Shah and Shamik Sengupta, "A survey on classification of cyber-attacks on IoT and IIoT devices", 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference, 28-31 October 2020, New York, NY, USA, 2020.
11. Zil E Huma, Shahid Latif, Jawad Ahmad, Zeba Idrees, Anas Ibrar, Zhuo Zou, Fehaid Alqahtani and Fatmah Baothman, "A hybrid deep random neural network for cyberattack detection in the industrial internet of things", IEEE Access, vol. 9, pp. 55595-55605, 2021.
12. Shahid Latif, Zhuo Zou, Zeba Idrees and Jawad Ahmad, "A novel attack detection scheme for the industrial internet of things using a lightweight random neural network", IEEE Access, vol. 8, pp. 89337- 89351, 2020.
13. Shahid Latif, Zeba Idrees, Zhuo Zou and Jawad Ahmad, "DRaNN a deep random neural network model for intrusion detection in industrial IoT", International Conference on UK-China Emerging Technologies, IEEE, 20-21 August 2020, Glasgow, UK, 2020.
14. Muna AL-Hawawreh, Nour Moustafa and Elena 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.
15. Radhakrishna Vangipuram, Rajesh Kumar Gunupudi, Veereswara Kumar Puligadda and Janaki Vinjamuri, "A machine learning approach for imputation and anomaly detection in IoT environment", Expert Systems, vol. 37, no. 5, pp. 1-16, 2020.
16. Di Wu, Zhongkai Jiang, Xiaofeng Xie, Xuetao Wei, Weiren Yu and Renfa Li, "LSTM learning with Bayesian and Gaussian processing for anomaly detection in industrial IoT", IEEE Transactions on Industrial Informatics, vol. 16, no. 8, pp. 5244-5253, 2019.
17. Tran Viet Khoa, Yuris Mulya Saputra, Dinh Thai Hoang, Nguyen Linh Trung, Diep N Nguyen, Nguyen Viet Ha and Eryk Dutkiewicz "Collaborative learning model for cyberattack detection systems in IoT industry 4.0", Wire-

less Communications and Networking Conference, 25-28 May 2020, Seoul, Korea, 2020.

18. Faezeh Farivar, Mohammad Sayad Haghighi, Alireza Jolfaei and Mamoun Alazab, "Artificial intelligence for detection, estimation, and compensation of malicious attacks in nonlinear cyber physical systems and industrial IoT", IEEE Transactions on Industrial Informatics, vol. 16, no. 4, pp. 2716-2725, 2019.
19. Yanmiao Li, Yingying Xu, Zhi Liu, Haixia Hou, Yushuo Zheng and Yang Xin, Yuefeng Zhao and Lizhen Cui, "Robust detection for network intrusion of industrial IoT based on multi-CNN fusion", Measurement, vol. 154, no. 2, pp. 1-27, 2019.
20. Muna AL-Hawawreh and Elena Sitnikova, "Industrial internet of things based Ransomware detection using stacked variational neural network", 3rd International Conference on Big Data and Internet of Things, 22-24 August 2019, Melbourn, Australia, 2019.
21. Joseph Bamidele Awotunde, Chinmay Chakraborty and Abidemi Emmanuel Adeniyi, "Intrusion Detection in Industrial Internet of Things Network-Based on Deep Learning Model with Rule-Based Feature Selection" Wirel. Commun. Mob. Comput., vol. 2021, 2021, doi: 10.1155/2021/7154587.
22. Nour Moustafa and Jill Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)" 2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc., 2015, doi: 10.1109/MilCIS.2015.7348942.
23. Zil E. Huma, Shahid Latif, Jawad Ahmad, Zeba Idrees, Anas Ibrar, Zhuo Zou, Fehaid Alqahtani and Fatmah Baothman "A Hybrid Deep Random Neural Network for Cyberattack Detection in the Industrial Internet of Things" IEEE Access, vol. 9, pp. 55595–55605, 2021, doi: 10.1109/ACCESS.2021.3071766.
24. Mohd Anul Haq, Mohd Abdul Rahim Khan, "Dnnbot: Deep neural network-based botnet detection and classification" Comput. Mater. Contin., vol. 71, no. 1, pp. 1729–1750, 2022, doi: 10.32604/cmc.2022.020938.
25. Mohd Anul Haq, Mohd Abdul Rahim Khan and Talal AL-Harbi, "Development of pccnn-based network intrusion detection system for edge computing" Comput. Mater. Contin., vol. 71, no. 1, pp. 1769–1788, 2022, doi: 10.32604/cmc.2022.018708.