METAHEURISTICS WITH DEEP CONVOLUTIONAL NEURAL NETWORK FOR CLASS IMBALANCE HANDLING WITH ANOMALY DETECTION IN INDUSTRIAL IOT ENVIRONMENT

NENAVATH CHANDER[1], MUMMADI UPENDRA KUMAR[2]

[1]Research Scholar, Osmania University, Department of CSE, Hyderabad, India
[2]Professor, Muffakham Jah College of Engineering and Technology, Department of CSE, Hyderabad, IndiaE-mail: [1]eslavathravi@gmail.com, [2]upendra.kumar@mjcollege.ac.in

ABSTRACT

The advancements of industrial Internet of Things (IIoT) have brought substantial value and accessibility to the industry. At the same time, it is followed by various security risks involving anomalies in the gathered data. Anomalies could emerge in the system because of several reasons namely software and hardware malfunctions, or a cyber-attack. The major problems in designing an effectual anomaly detection system include complexity in different anomaly definitions in various domains, defining normal region, normal behavior variation over time, the noise presence in the datasets, and lack of suitable datasets. Furthermore, Class imbalance is the term utilized for data having minority and majority classes. The spectrum of class imbalance ranges from “slightly imbalanced” to “rarity”. In a majority–minority classification problem, class imbalance in the data can drastically skew the classifier performance, presenting a prediction bias for the majority class. This study develops an optimal Deep Convolutional Neural Network for Class Imbalance Handling Anomaly Detection (ODCNN-CIHAD) model. The proposed ODCNN-CIHAD technique majorly focuses on two major processes namely class imbalance data handling and anomaly detection. At the initial stage, the ODCNN-CIHAD technique follows min-max data normalization technique to convert the input data into compatible format. In addition, the ODCNN-CIHAD technique designs a group teaching optimization algorithm (GTOA) with SMOTE technique for handling class imbalance data. Also, the DCNN approach was applied for the recognition and classification of anomalies that exist in the IIoT data. Finally, the gorilla troops optimizer (GTRO) approach was exploited for optimum hyperparameter tuning of the DCNN approach. The experimental validation of the ODCNN-CIHAD technique is carried out utilizing benchmark dataset and the outcomes are inspected under various measures. The comparison study highlighted the improved performance of the ODCNN-CIHAD system on existing approaches.

Keywords: *Security; Anomaly detection; Industrial Internet of Things; Deep learning; SMOTE technique; Class imbalance data*

1. INTRODUCTION

With the tremendous growth of Industry 4.0, increasingly industrial applications, enabled by real-time and intelligent signal processing, are interactively connected by the increasing number of wireless network technologies with smart devices in industrial Internet of Things (IIoT) [1]. The intensification of applications and devices in IIoT result in a great scare of information with further complication across industrial cyber–physical system. It develops a crucial problem to secure the network and infrastructure security for fundamental tasks in IIoT [2]. It can be very difficult to protect IoT devices since they are heterogeneous, conventional security control is not real-world for the resource-constraint devices, and the distributed IoT network falls outside the scope of perimeter security, and present solutions for example the cloud suffers from high delay and centralization [3]. Additional reasons for these challenges are that IoT device vendors often overlook security requirements as a result of rush-to-market approach. In addition, the lack of privacy standards has added additional dimension to the complication of safeguarding IoT devices

[4]. The challenge and nature of IoT applications require a monitoring system namely anomaly detection at device and network level beyond the organizational boundary. Anomalies can be collective and contextual points based on the source of anomalies [5]. An anomalous event rarely occurs; but, the event causes adverse effects on government and businesses using IoT applications [6].

As a crucial technique in IIoT security, intrusion detection system (IDS) is deployed commonly as a software tool to detect and monitor anomalies or intrusion activities over the industrial network that is classified into the anomaly and signature- based IDS [7]. In recent times, anomaly detection has attracted considerable attention, because of its capability in identifying novel attacks from higher-dimension IIoD data across different IIoT sensors [8]. To improve the performance of anomaly detection while handling IIoT information, deep learning (DL) and machine learning (ML) algorithms are utilized in network and host-based systems. But still, it is challenging to perform the anomaly detection result from considerable amount of higher dimension datasets in IIoT [9]. It will be worse for traditional classification method for extracting significant features from the imbalanced input dataset, particularly if the positive sample becomes highly sparse in IIoT environment. To enhance the accuracy of categorizing imbalanced datasets, researchers developed a variety of methods involving active learning methods, resampling, and cost sensitive kernel modification methods [10].

This study focuses on the design of an optimal Deep Convolutional Neural Network for Class Imbalance Handling Anomaly Detection (ODCNN-CIHAD) model. The proposed ODCNN-CIHAD technique applies min-max data normalization technique to convert the input data into compatible format. For handling class imbalance data problems, the group teaching optimization algorithm (GTOA) with SMOTE technique for handling class imbalance data. Moreover, the DCNN method was exploited for anomaly detection and classification in the IIoT data. Finally, the gorilla troops optimizer (GTRO) approach was exploited for optimal hyperparameter tuning of the DCNN system. The experimental validation of the ODCNN-CIHAD technique is carried out using benchmark dataset and the outcomes are inspected under distinct measures. In short, the paper's contributions are summarized as follows.

- Develop a new ODCNN-CIHAD technique for anomaly detection and classification in the IIoT environment.

- Designs a GTOA with SMOTE technique for class imbalance data handling process where the GTOA is used to select optimally balanced subset.

- Employ GTRO with DCNN model for the classification process which recognizes the data samples as anomaly or normal.

2. RELATED WORKS

Liang et al. [11] suggest an optimized intra or inter-class-structure-related variational few-shot learning (OICS-VFSL) method to succeed in a detailed out-of-distribution issue from the imbalanced learning, and improvising microservice-based ID in dispersed IoT networks. Followed by, a new devised variational few-shot learning (VFSL) structure, an intra or inter-class optimizing method can be enhanced by utilizing reconstructed feature embedded, in that the intra-class distance can be optimizing on the basis of the estimate at the time of a variation Bayesian process. An IDS termed pre-training Wasserstein generative adversarial network IDS system (PWG-IDS) was suggested in [12]. This system has 2 key modules in first one, the researchers present the pre-training system in the Wasserstein generative adversarial network having gradient penalty (WGAN-GP) for the very first time, initially utilizing the normal network traffic for training the WGAN-GP, and inputting the imbalance data as pretrained WGAN-GP for retraining and generating the data which is needed finally. In the second module i.e. ID module, the authors employ LightGBM as the classifier system for detecting attack traffic in IIoT systems.

In [13], a two-stage training Deep Neural Network (DNN) depends upon focal loss and cross-entropy, known as CE-FL-NIDS, was suggested in managing the issues made to NIDS because of the uneven data distribution in the traffic datasets. Toldinas et al. [14] recommend a new technique for network ID by making use of multistage DL image detection. The features of the network were transmitted as four-channel (Alpha, Red, Blue, and Green) images. Afterward, the images were employed for categorization of tests and training the pretrained DL method ResNet50. Mokhtari et al. [15] suggested technique named measurement IDS

(MIDS) that allows the model to identify some abnormal activities in the system even if the attacker attempts to hide them in control layer of systems. A supervised ML method can be formulated for classifying abnormal and normal activity in an ICS for evaluating the performance of MIDS.

Abdel-Basset et al. [16] provide an innovative privacy-preserving federated semi-supervised class-rebalanced (Fed-SCR) structure for identifying anomalous power data from the fog-enabled smart grids. Fed-SCR presents a semi-supervised generation network for improvising the generated minority samples quality and devising the relation among unlabeled and labeled data. In [17], the researchers utilize the GAN capability to design complicated higher dimensionality image distribution and recommend a self-adaption AAE-GAN network dependent upon adaptive variations of input samples. This time series anomaly recognition technique transforms multidimensional time sequential data to two-dimensional matrixes, and merely normal specimens were required during the training that efficiently resolves the above-mentioned issues. The technique that authors suggested was using a decoder and encoder for constituting a discriminator and a generator. Chander, N et al. [32] used resnet-50, resnet-152v2 and inception v3 models for detection of anomalies and leaf disease prediction in cotton plant data.

3. THE PROPOSED MODEL

In this study, a novel ODCNN-CIHAD system was established for anomaly detection and classification in the IIoT environment. The ODCNN-CIHAD technique involves a series of subprocesses namely min-max data normalization, SMOTE based class imbalance data handling, GTOA based parameter optimization, DCNN classification, and GTRO based hyperparameter tuning. Fig. 1 illustrates the overall procedure of ODCNN-CIHAD algorithm.

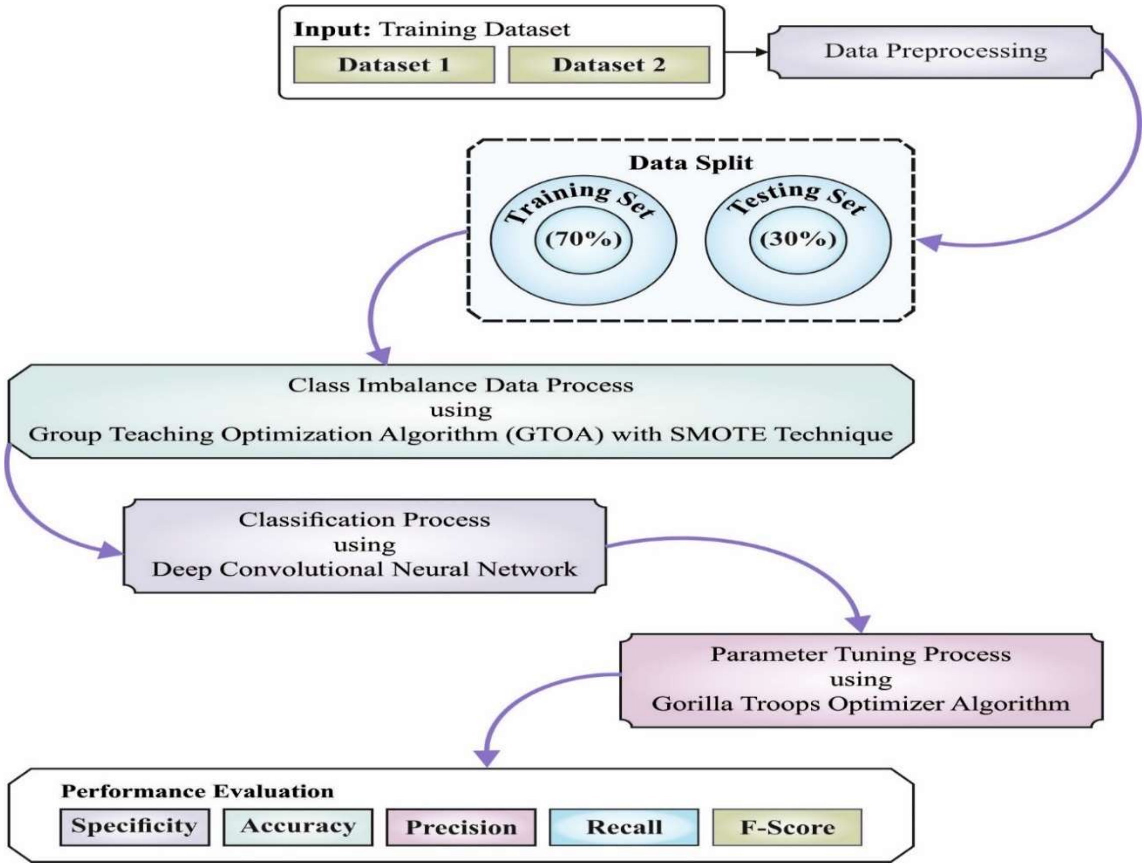


***Fig. 1.** Overall process of ODCNN-CIHAD approach*

3.1. Data pre-processing

At the initial stage, the ODCNN-CIHAD technique undergoes data pre-processing to normalize the input data into a compatible format. Min-max normalized approach to scale the feature from the range of zero and one with implementation in Eq. (1) [18].

$$v' = \frac{v - \min_A}{\max_A - \min_A} \qquad (1)$$

At this point, min_A and max_A represents the minimal and maximal values of features A. The original and normalizing value of attributes, A implied by ν and ν^' correspondingly. It could be detected in the above formula in which the maximal and minimal feature values were mapped to one and zero correspondingly.

3.2. Class imbalance data handling

For handling the class imbalance data problem, the GTOA with SMOTE technique is used. Chawla et al. [19] projected a SMOTE (synthetic minority oversampling method) as a common oversampling model. In the SMOTE, rather than mere data-oriented duplicating, the minority class is over-sampled by establishing synthetic instance in the feature space formed by the K-nearest neighbors and its instance that effectually prevent the over-fitting problems.

This process is defined in the following. Select two instances, x_1 and x_2, from the minority sample set in a random fashion, where every instance has n attributes. For x_1 and x_2, evaluated the difference on i-th attributes; viz., 〖diff〗 _i=x_2i-x_1i. Next, we attain the i-th attribute values of the new target samples as follows [20]:

$$x_{12i} = rand\,[0,1] * diff_i, \qquad (2)$$

In Eq. (2), rand[0,1] indicates a random integer lies within 0 and 1. Hence, the last synthetic samples of x_1 and x_2 are

$$x_{12} = rand\,[0,1] * diff, \qquad (3)$$

where $diff \ = (diff_1, diff_2, \dots, diff_n)$.

Based on the sampling rate, set the implementation time and repeat the abovementioned procedure. Incorporate the synthetic and the original samples, and the last minority sample set is attained.

$$\hat{\varepsilon} = \frac{1}{n}\sum_{i=1}^{f}\sum_{j=1}^{n/f} | y_j^{(i)} - p(t_j^{(i)} | \Theta^{(i)t} x_j^{(i)})| \qquad (4)$$

whereas f refers the amount of folds, n signifies the amount of instances, $y_j^{(i)}$ implies the offered class labels of sample j in the test fold i, $p\left(t_j^{(i)} | \Theta^{(i)t} x_j^{(i)}\right)$ defines the forecast of j^{th} instances in the test fold i, $x_j^{(i)}$ demonstrates the feature vector, $\Theta^{(i)}$ represents the model parameters learned at the time of training, and $t_j^{(i)}$denotes the predicted value of j^{th} instances. The error is f-fold cross validation error and subspace which decreases the error is the solution for optimizing problem. Determining the optimum sample subset including equivalent members of minority and majority samples is explained as optimized problem. An evolutionary technique was employed to optimize the sample subset to global lesser solution to the cost function. The globally optimum solution was targeted by utilizing the GTOA technique.

. The major optimized loop begins with the $MaxIt$ maximal iteration amount that must be fixed in the initialization stage and the existing iteration count T_{cur} need to be initialized $T_{cur}^0 = 0$. As well, the population X is produced on the basis of N population count shown as follows:

$$X_{i,j} = lb_j + rand \cdot \left(ub_j - lb_j\right) (i = 1, \dots, N; j = 1, \dots, D) \qquad (5)$$

In Eq. (5), $rand$ denotes a randomly generated number that lies between zero and one, lb_j and ub_j correspondingly refers to the lower and upper boundaries of j-th variable. Next, the teacher allocation stage is formulated as:

$$\mathrm{T}^t = \begin{cases} X_{first}^t & f\left(X_{first}^t\right) \leq f\left(\frac{X_{first}^t + X^t_{\text{second}} + X_{third}^t}{3}\right) \\ \frac{X_{first}^t + X^t_{\text{second}} + X_{third}^t}{3} & f\left(X_{first}^t\right) > f\left(\frac{X_{first}^t + X^t_{\text{second}} + X_{third}^t}{3}\right) \end{cases} \qquad (6)$$

Let T^t be the teacher selected at the existing iteration t. X_{first}^t, X_{second}^t and X_{third}^t, correspondingly, represent the 1st, 2nd, 3rd present finest students. $f(\cdot)$ represent the fitness function.

Next, the outstanding student at the teacher stage is determined in the following equations.

$$X_{teach}^t \;= X_i^t + a \times \left(\mathrm{T}^t - F \times (b \times \mathrm{M}^t + C \times X_i^t)\right) \left(j \;=\; 1, \dots, \frac{N}{2}\right) \quad (7)$$

$$\mathrm{M}^t = \frac{\sum_{i=1}^{N/2} X_i^t}{N/2} \quad (8)$$

$$b + c = 1 \quad (9)$$

$$X_{teach\quad ,i}^{t+1} = \begin{cases} X_{teacher,i}^{t+1}, f\left(X_{teacher,i}^{t+1}\right) \;<\; f(X_i^t) \\ \quad X_i^t, f\left(X_{teach\quad ,i}^{t+1}\right) \;\geq\; f(X_i^t) \end{cases} \left(j \;=\; 1, \dots, \frac{N}{2}\right) \quad (10)$$

Here, $X_{teach\quad ,i}^{t+1}$ implies the knowledge of i-th students in the outstanding student learning from the teacher at the existing iteration t, X_i^t represents the knowledge of i-th students in the outstanding student. a, b, and c indicate an arbitrary integer range from [0, 1].

Further, the average group in the teacher stage is determined by,

$$X_{teacher,i}^{t+1} = X_i^t + 2 \times d \times (\mathrm{T}^t - X_i^t) \left(j = \frac{N}{2} + 1, \dots, N\right) \quad (11)$$

$$X_{teacher,i}^{t+1} = \begin{cases} X_{teacher,i}^{t+1}, f\left(X_{teach\quad ,i}^{t+1}\right) < f(X_i^t) \\ \quad X_i^t, f\left(X_{teacher,i}^{t+1}\right) \;\geq\; f(X_i^t) \end{cases} \left(j \;=\; \frac{N}{2} + 1, \dots, N\right) \quad (12)$$

While $X_{teacher,i}^{t+1}$ indicates the knowledge of i-th student in the average student learned from teacher at t the existing iteration.

At last, the student stage is determined by.

$$X_{teach\quad ,i}^{t+1} = \begin{cases} X_{teacher,i}^{t+1} + e \times \left(X_{teacher,i}^{t+1} - X_{teacher,j}^{t+1}\right) + g \times \left(X_{teacher,i}^{t+1} - X_i^{t+1}\right), f\left(X_{teacher,i}^{t+1}\right) < f\left(X_{teach\quad ,j}^{t+1}\right) \\ X_{teach\quad ,i}^{t+1} - e \times \left(X_{teacher,i}^{t+1} - X_{teach\quad ,j}^{t+1}\right) + g \times \left(X_{teacher,i}^{t+1} - X_i^{t+1}\right), f\left(X_{teacher,i}^{t+1}\right) \geq f\left(X_{teacher,j}^{t+1}\right) \end{cases}$$

$$(i \;=\; 1, \dots, N, j \;=\; 1, \dots, N) \quad (13)$$

$$X_i^t = \begin{cases} X_{teach\quad ,i}^{t+1}, f\left(X_{teach\quad ,i}^{t+1}\right) \;<\; f\left(X_{student,t}^{t+1}\right) \\ stt + u1dent, tf\left(X_{teacher,i}^{t+1}\right) \geq f\left(X_{student,t}^{t+1}\right) \end{cases} (j \;=\; 1, \dots, N) \quad (14)$$

From the equation, e & g indicates the arbitrary number from the ranges from [0, 1], $X_{student,i}^{t+1}$ denotes the knowledge of i-th students learning from the student phase at $t+1$ iteration. The population reconstruction procedure is determined by.

$$\mathrm{X}^{t+1} = \left[\mathrm{X}_{out}^t; \mathrm{X}_{avg}^t\right] \quad (15)$$

In Eq. (15), X^{t+1} denotes the upgraded population, X_{out}^t and X_{avg}^t correspondingly indicates the upgrade outstanding and average student's then iteration. The optimized technique is end once the iteration amount T_{cur} exceeds the value of $N \times MaxIt$.

3.3. Anomaly Detection and Classification

In this study, the DCNN model is exploited for the identification and classification of anomalies that exist in the IIoT data. The DCNN model has two mechanisms: the lower subnetwork is for the character embedding and the upper subnetwork is for the word and concept embedding of short text. Both of them are CNN [22]. Using these models, we could learn rich features from the word and the character levels, correspondingly. The upper components consist of one input layer, two convolution, pooling, and hidden layers. Fig. 2 demonstrates the infrastructure of DCNN.

Input Layer. The input layer transforms the short texts into a matrix of embedding, represented as $W \in R^{(k+n)\times m}$ as the input of network, where n and k refer to the maximal amounts of words and concepts, correspondingly. Also, m denotes the dimension of word embedding. We get W through concatenating the embedding of word and concept: $W = W_w \oplus W_c$. Now, W_w and W_c indicates the embedding of word and concept, correspondingly. Also, $\oplus$ denotes the concatenation process. The way to build W_w is relatively simple: suppose the short text comprises n words, and $v_i^w \in R^m$ denotes an m-dimension vector of the i^{th} words in the short text. Then, obtain W_w through concatenating them:

$$W_w = v_1^w \oplus v_2^w \oplus \dots \oplus v_n^w \qquad (16)$$

In order to get the representation of W_c, we must assume the weight of concept simultaneously. For every embedding vector $v_i^c \in R^m$ of concept c_i, we multiply them with the constant w_i to represent the weight of a provided concept. Thus:

$$W_c = w_1 v_1^c \oplus w_2 v_2^c \oplus \dots \oplus w_k v_k^c \qquad (17)$$

Once the concept vector or short text is no longer, we apply zero as padding. We get the embedding v_i^w and v_i^c by the pretrained word embedding.

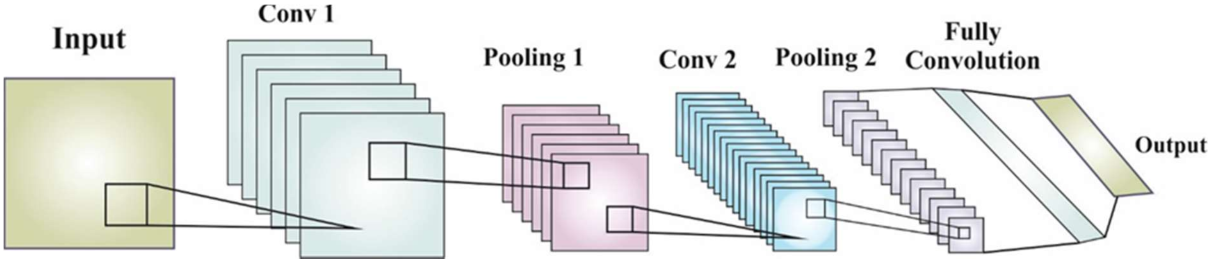


Fig. 2. Structure of DCNN

Convolution Layer. The aim is to extract high level features from the input matrixes. To get various types of features, we employ filters with varying sizes. Like previous work, we set the width of every filter as m and process the height h of it as a hyperparameter. Assumed a filter $\omega \in R^{h\times m}$, a feature s_i is produced from a window of word and concept $[v_i : v_{i+h-1}]$ as:

$$s_i = g(\omega \cdot [v_i : v_{i+h-1}] + b) \qquad (18)$$

Now $b \in R$ refers to a bias term and g represents a nonlinear function. In the study, we apply $ReLU$ as the nonlinear function for convolutional layer. The filter is employed for each potential window of word and concept in W for producing a feature map $s \in R^{n+k-h+1}$. This procedure is repetitive for different filters with various heights for increasing the feature coverage.

Pooling Layer. The aim is to further abstract the feature produced from convolutional layer by aggregating the score for all the filters. In the study, we employ a max-over-time pooling function over every feature map. The concept is to select the maximum value on every dimension of vector for capturing the primary feature. Using pooling layer, we induce a fixed-length vector from feature map.

Hidden Layer. To utilize rich features attained from the pooling layer, we utilize nonlinear hidden layers for combining distinct pooling features. Then, apply tanh as the activation function. Likewise, the lower sub-network comprises one input layer, two convolution, pooling, and hidden layers. The input of sub-network is a series of encoded characters. The encoding can be performed by initially generating an alphabet of each character in the data and later randomly initializing the embedding of every character with m_c dimension. Next, the series of characters is converted into a matrix $W_c \in R^{L\times m_c}$. Now, L is a hyperparameter that limits the maximal size of the sequence. The character which exceeds length L is disregarded. Hence, we set the value of L to 256.

At last, we integrate the output vector of the two sub-networks by concatenating them. Next, we employ an output layer on the joint vector for converting the output number into probability.

3.4. Hyperparameter Tuning

For optimally adjusting the hyperparameters related to the DCNN approach, the GTRO technique was exploited in such a way that the classification performance gets boosted to a maximum extent. GTRO is a novel optimization technique which simulates the movements and social behavior of gorillas [23]. The GTRO method is comparable to other optimization methods based on exploitation and exploration stages. Exploration in GTRO encompasses three approaches: the initial depends on the gorilla's movement towards unidentified location, while the next and last are relies on the moving of a gorilla towards other gorillas or an identified sites. The exploitation stage encompasses 2 techniques: the initial one relies on movement with the silverback whereas the next one defines the moving of adult females. Now, the position of gorillas is represented by X, whereas the position of silverback can be indicated by GX. Imagine a gorilla trying to discover best food sources. Therefore, in the iteration technique, GX is produced in every iteration and replaced if other solutions with the best value are attained. From the above mentioned, the exploration stage of the GTRO depends on three approaches that are modelled arithmetically in the following:

$$GX(t+1) = \begin{cases} (UB-LB)\times R_1 + LB, rand < p \\ (R_2 - C)\times X_r(t) + L\times H, rand \ \geq 0.5 \\ X(i) - L\times\left(L\times\left(X(t)-GX_r(t)\right) + R_3\times\left(X(t)-GX_r(t)\right)\right), rand \ < 0.5 \end{cases} \quad (19)$$

In Eq. (19), LB and UB represent the lower and upper limits, R_1, R_2, and R_3 indicates arbitrary parameter lies in [0,1], GX denotes a solution candidate that upgraded, t denotes the existing iteration, rand indicates a randomly generated number lies in [0,1], p is a predetermined value lies in zero and one, GX_r and X_r denotes solutions that are arbitrarily designated as follows:

$$C = F\times\left(1-\frac{t}{\mathrm{Max}It}\right) \quad (20)$$

$$F = \cos\left(2\times R_4\right) + 1 \quad (21)$$

$$L = C\times l \quad (22)$$

$$H = Z\times X(t) \quad (23)$$

$$Z = [-C, C] \quad (24)$$

Let $MaxIt$ be the maximal iteration count and R_4 denotes an arbitrary integer within the range of zero and one. The value of l is transformed from -1 to 1. The exploitation stage in GTRO is relying on two techniques: the initial one depends on the troop movement, that is, following the silver back; the next one is relying on competition for adult females, in which the male in the group fight one another if the silver back become old or weak. The transition among the two motions depends on C, as determined in Eq. (24), W is a predefined value. Where $C \geq W$, the gorillas upgrade the position by following the silver back:

$$GX(t+1) = L\times M\times\left(X(t) - X_{silverback}\right) + X(t) \quad (25)$$

$$M = \left(\left|\frac{1}{N}\sum_{i=1}^{N} G\,X_i(t)\right|^{g}\right)^{\frac{1}{8}} \quad (26)$$

$$g = 2^{L} \quad (27)$$

Given that $X_{silverback}$ indicates the position of the silver back. Where $C < W$, the location of the gorillas are upgraded according to the competition for adult female that is formulated by:

$$GX(i) = X_{silverback} - \left(X_{silverback}\times Q - X(t)\times Q\right)\times A \quad (28)$$

$$Q = 2\times r_5 - 1 \quad (29)$$

$$A = \beta\times E \quad (30)$$

$$E = \begin{cases} N_1, & rand \ \geq 0.5 \\ N_2, & rand \ < 0.5 \end{cases} \quad (31)$$

From the expression, Q stimulates the impact force, r_5 denotes an arbitrary number ranges from zero to one, and β is a predetermined variable.

The GTRO system resolves a fitness function

(FF) for obtaining maximal classifier efficiency. It resolves a positive integer for portraying the best efficiency of candidate results. During this case, the minimization classifier error rate was supposed that FF is providing in Eq. (32).

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \quad (32)$$

4. Experimental Validation

In this section, the experimental result analysis of the ODCNN-CIHAD model is tested using two datasets. The first UNSW-NB15 [24] is created by an Australian security laboratory using IXIA Perfect Strom tool and the details are given in Table 1. It has 42 features (excluding the labels) and ten classes (9 attacks and 1 normal). The number of samples in the dataset before class imblancing handling is 82332 and it becomes 82300 after class balanced data as illustrated in Table 1.

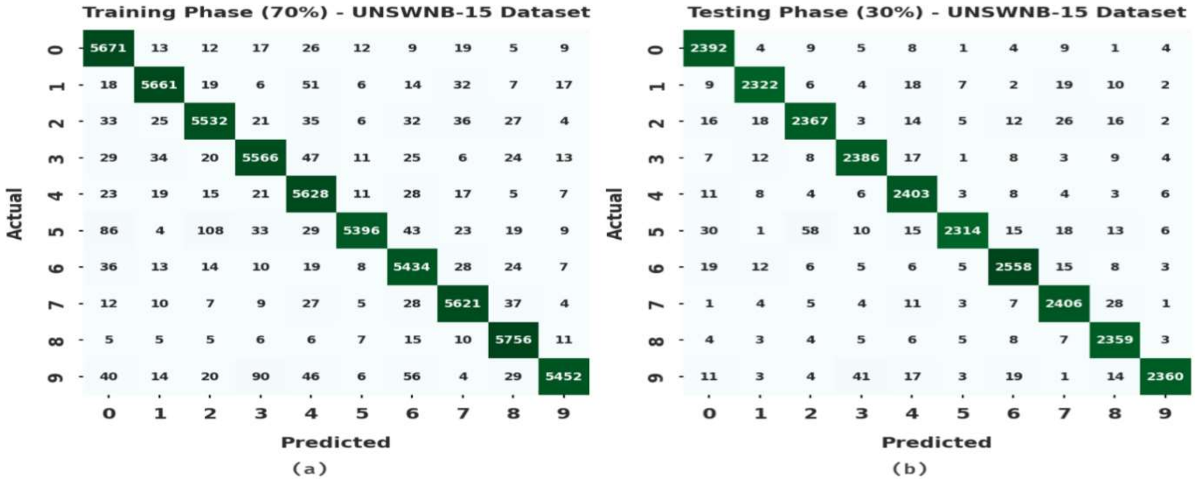
Table 1 UNSW-NB15 dataset details

Label	Class Name	Before	After
0	Normal	37000	8230
1	Generic	18871	8230
2	Exploits	11132	8230
3	Fuzzers	6062	8230
4	DoS	4089	8230
5	Reconnaissance	3496	8230
6	Analysis	677	8230
7	Backdoor	583	8230
8	Shellcode	378	8230
9	Worms	44	8230
Total Number of Samples		**82332**	**82300**

Next, the results are inspected using Kaggle dataset [25], comprising samples under two classes. Before class imbalance handling, the number of instances in the dataset is 1567 and it becomes 2800 samples after class balance as depicted in Table 2.

Table 2 Kaggle dataset details

Class Name	Before	After
Pass	1463	1400
Fail	104	1400
Total	**1567**	**2800**



Training Phase (70%) - UNSWNB-15 Dataset

Actual \ Predicted	0	1	2	3	4	5	6	7	8	9
0	5671	13	12	17	26	12	9	19	5	9
1	18	5661	19	6	51	6	14	32	7	17
2	33	25	5532	21	35	6	32	36	27	4
3	29	34	20	5566	47	11	25	6	24	13
4	23	19	15	21	5628	11	28	17	5	7
5	86	4	108	33	29	5396	43	23	19	9
6	36	13	14	10	19	8	5434	28	24	7
7	12	10	7	9	27	5	28	5621	37	4
8	5	5	5	6	6	7	15	10	5756	11
9	40	14	20	90	46	6	56	4	29	5452

(a)

Testing Phase (30%) - UNSWNB-15 Dataset

Actual \ Predicted	0	1	2	3	4	5	6	7	8	9
0	2392	4	9	5	8	1	4	9	1	4
1	9	2322	6	4	18	7	2	19	10	2
2	16	18	2367	3	14	5	12	26	16	2
3	7	12	8	2386	17	1	8	3	9	4
4	11	8	4	6	2403	3	8	4	3	6
5	30	1	58	10	15	2314	15	18	13	6
6	19	12	6	5	6	5	2558	15	8	3
7	1	4	5	4	11	3	7	2406	28	1
8	4	3	4	5	6	5	8	7	2359	3
9	11	3	4	41	17	3	19	1	14	2360

(b)

Fig. 3. Confusion matrices of ODCNN-CIHAD approach under UNSW-NB15 dataset (a) 70% of TR data and (b) 30% of TS data

Table 3 Result analysis of ODCNN-CIHAD approach with various measures under UNSW-NB15 dataset

Class Labels	**Accuracy**	**Precision**	**Sensitivity**	**Specificity**	**F-Score**	**MCC**
Training Set (70%)						
Normal	99.30	95.26	97.89	99.46	96.56	96.18
Generic	99.47	97.64	97.08	99.74	97.36	97.06
Exploits	99.24	96.18	96.19	99.58	96.18	95.76
Fuzzers	99.27	96.31	96.38	99.59	96.35	95.94
DoS	99.25	95.16	97.47	99.45	96.30	95.90
Reconnaissance	99.26	98.68	93.84	99.86	96.20	95.83
Analysis	99.29	95.60	97.16	99.52	96.37	95.98
Backdoor	99.45	96.98	97.59	99.66	97.28	96.98
Shellcode	99.57	97.02	98.80	99.66	97.90	97.67
Worms	99.33	98.54	94.70	99.84	96.58	96.23
Average	99.34	96.74	96.71	99.63	96.71	96.35
Testing Set (30%)						
Normal	99.38	95.68	98.15	99.51	96.90	96.57
Generic	99.42	97.28	96.79	99.71	97.03	96.71
Exploits	99.13	95.79	95.48	99.53	95.64	95.15
Fuzzers	99.38	96.64	97.19	99.63	96.91	96.57
DoS	99.33	95.55	97.84	99.50	96.68	96.32
Reconnaissance	99.19	98.59	93.31	99.85	95.88	95.48
Analysis	99.34	96.86	97.00	99.62	96.93	96.56
Backdoor	99.33	95.93	97.41	99.54	96.67	96.29
Shellcode	99.40	95.86	98.13	99.54	96.98	96.66
Worms	99.42	98.70	95.43	99.86	97.04	96.73

Average	99.33	96.69	96.67	99.63	96.67	96.30

Fig. 3 demonstrates the confusion matrices accomplished by the ODCNN-CIHAD model on UNSW-NB15 dataset. The figure shows that the ODCNN-CIHAD model has proficiently categorized all the ten classes on 70% of TR data and 30% of TS data.

Table 3 provides an overall anomaly classification performance of the ODCNN-CIHAD system on 70% of TR data and 30% of TS data under UNSW-NB15 dataset. The results highlighted that the ODCNN-CIHAD model has shown enhanced performance in both aspects. For instance, with 70% of TR data, the ODCNN-CIHAD model has attained maximum average $accu_y$ of 99.34%, $prec_n$ of 96.74%, $sens_y$ of 96.71%, $spec_y$ of 99.63%, F_{score} of 96.71%, and MCC of 96.35%. Additionally, with 30% of TS data, the ODCNN-CIHAD approach has obtained maximal average $accu_y$ of 99.33%, $prec_n$ of 96.69%, $sens_y$ of 96.67%, $spec_y$ of 99.63%, F_{score} of 96.67%, and MCC of 96.30%.

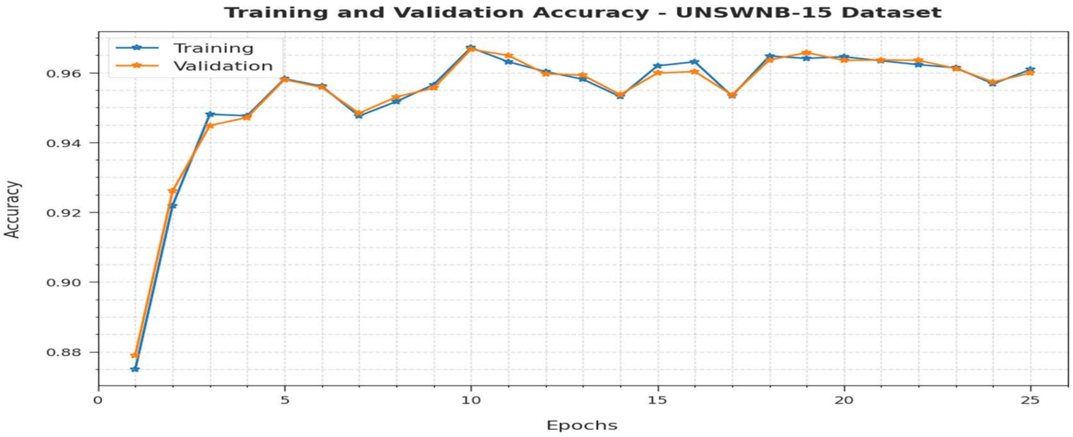


Fig. 4. TA and VA analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset

The training accuracy (TA) and validation accuracy (VA) obtained by the ODCNN-CIHAD system on UNSW-NB15 dataset is depicted in Fig. 4. The experimental outcome revealed that the ODCNN-CIHAD technique has achieved maximal values of TA and VA. Particularly the VA appeared superior to TA.

The training loss (TL) and validation loss (VL) gained by the ODCNN-CIHAD approach on UNSW-NB15 dataset are established in Fig. 5. The experimental outcome exposed that the ODCNN-CIHAD algorithm has been able to minimal values of TL and VL. In specific, the VL is lesser than TL.

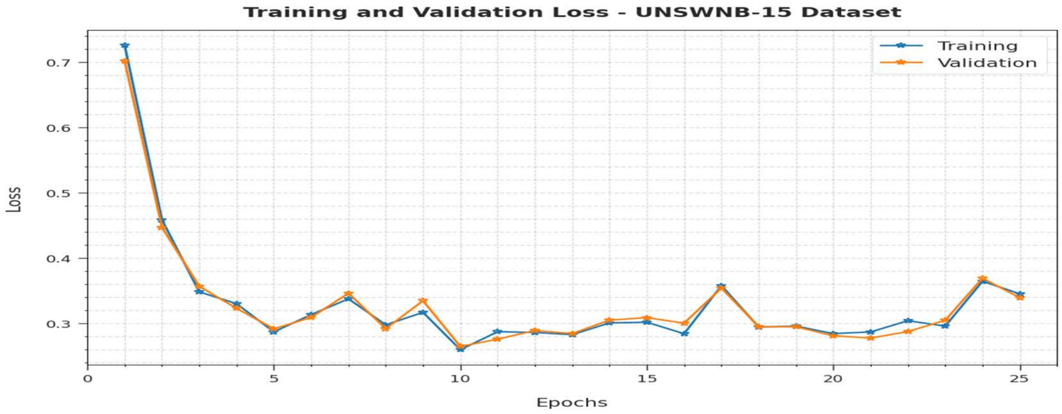


Fig. 5. TL and VL analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset

A clear precision-recall examination of the ODCNN-CIHAD method on UNSW-NB15 dataset is represented in Fig. 6. The figure indicated that the ODCNN-CIHAD method has resulted in enhanced values of precision-recall values under all classes.

A brief ROC analysis of the ODCNN-CIHAD method on UNSW-NB15 dataset is portrayed in Fig. 7. The outcomes exposed the ODCNN-CIHAD approach has demonstrated its ability in categorizing distinct classes on the UNSW-NB15 dataset.

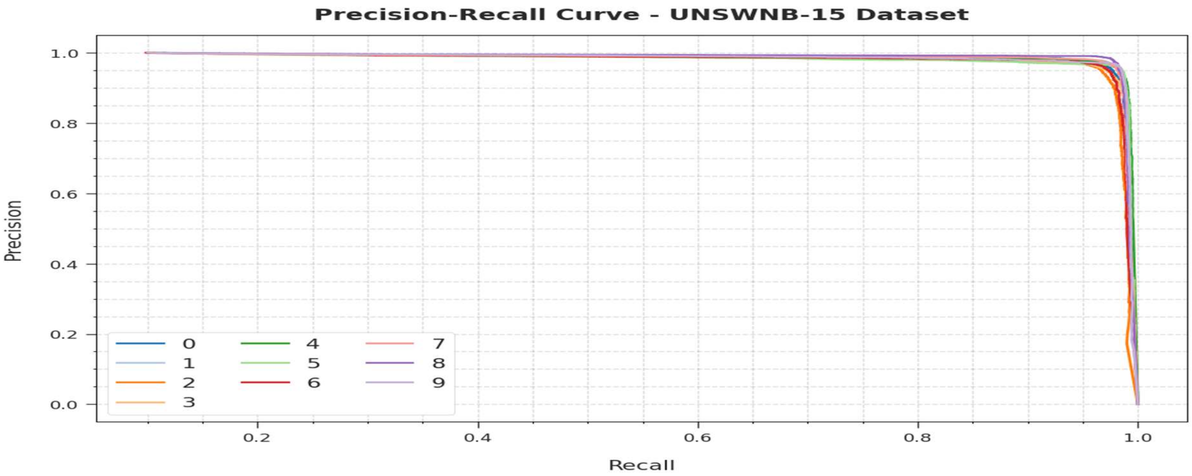


Fig. 6. *Precision-recall analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset*

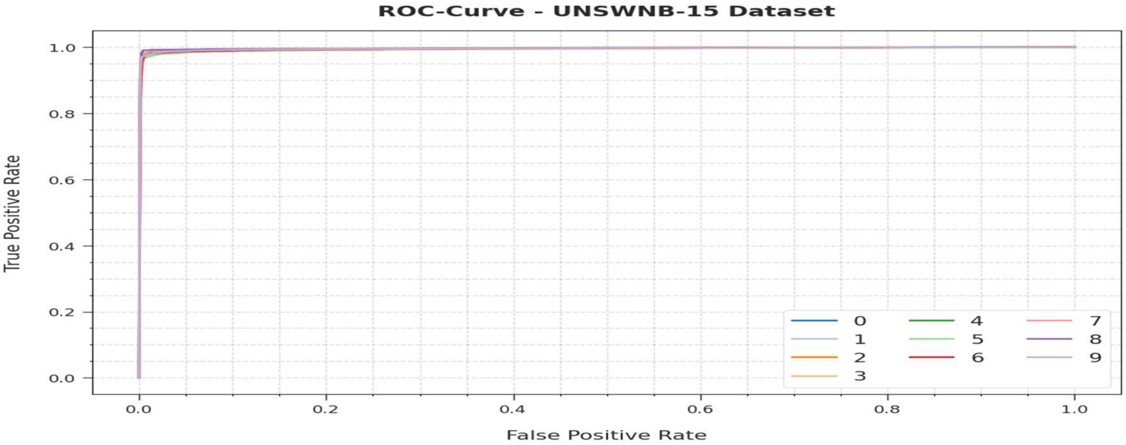


Fig. 7. *ROC analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset*

Table 4 reports a detailed comparative study of the ODCNN-CIHAD methodology with existing models on the test UNSW-NB15 dataset [26-28]. Fig. 8 highlights the comparative study of the ODCNN-CIHAD system with existing techniques interms of $prec_n$, $sens_y$, and F_{score} on the UNSW-NB15 dataset. The figure implied that the ODCNN-CIHAD approach has gained maximal values of $prec_n$, $sens_y$, and F_{score} over other models. For instance, based on $prec_n$, the ODCNN-CIHAD model has obtained higher $prec_n$ of 96.69% whereas the ANN, LR, KNN, SVM, DT, SSAE, NB, VLSTM, and SSA-CRNN models have attained lower $prec_n$ values of 57.41%, 66.93%, 62.94%, 50.52%, 63.85%, 59.30%, 64.76%, 66.75%, and 67.01% respectively. In addition, according to F_{score}, the ODCNN-CIHAD approach has reached superior F_{score} of 96.30% whereas the ANN, LR, KNN, SVM, DT, SSAE, NB, VLSTM, and SSA-CRNN systems have attained lesser F_{score} values of 55.12%, 57.33%, 53.07%, 53.37%, 48.86%, 51.41%, 52.49%, 58.55%, and 59.73% correspondingly.

Fig. 9 demonstrates the comparison analysis of the ODCNN-CIHAD technique with existing approaches with respect to $accu_y$ on the UNSW-NB15 dataset. The figure revealed that the ODCNN-CIHAD approach has reached maximal values of $accu_y$ over other models. For sample, based on $prec_n$, the ODCNN-CIHAD system has obtained higher $accu_y$ of 99.33% whereas the ANN, LR, KNN, SVM, DT, SSAE, NB, VLSTM, and SSA-CRNN methodologies have obtained reduced $accu_y$ values of 79.33%, 63.03%, 71.09%, 64.58%, 70.53%, 88.96%, 93.94%, 95.91%, and 98.39% correspondingly.

***Table 4** Comparative analysis of ODCNN-CIHAD approach with existing methodologies under UNSW-NB15 dataset*

Methods	Accuracy	Precision	Sensitivity	F-Score
ANN Algorithm	79.33	57.41	58.45	55.12
LR Algorithm	63.03	66.93	55.08	57.33
kNN Algorithm	71.09	62.94	53.38	53.07
SVM Algorithm	64.58	50.52	53.20	53.37
DT Algorithm	70.53	63.85	53.19	48.86
SSAE Algorithm	88.96	59.30	58.99	51.41
NB Algorithm	93.94	64.76	57.73	52.49
VLSTM Algorithm	95.91	66.75	52.90	58.55
SSA-CRNN	98.39	67.01	59.14	59.73
ODCNN-CIHAD	99.33	96.69	96.67	96.30

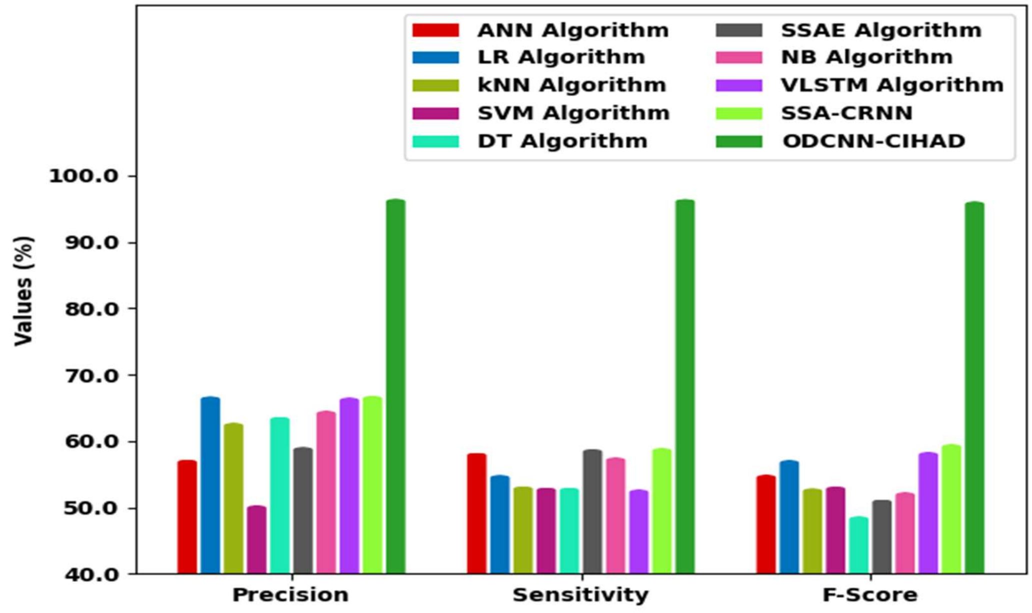


Fig. 8. Comparative analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset

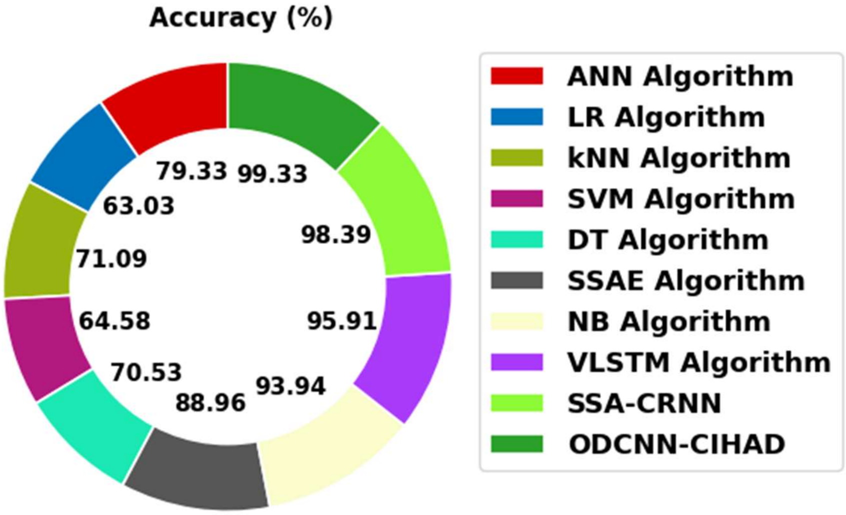


Fig. 9. $Accu_y$ analysis of ODCNN-CIHAD approach under UNSW-NB15 dataset

Fig. 10 depicts the confusion matrices accomplished by the ODCNN-CIHAD methodology on Kaggle dataset. The figure demonstrated that the ODCNN-CIHAD system has proficiently categorized all the ten classes on 70% of TR data and 30% of TS data.

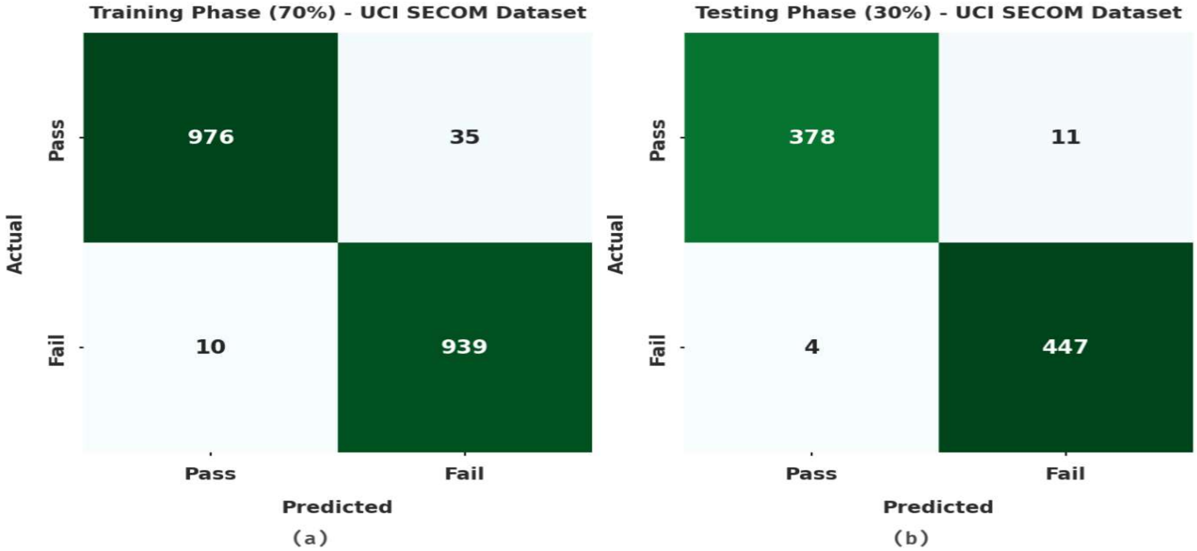


Fig. 10. Confusion matrices of ODCNN-CIHAD approach under Kaggle dataset (a) 70% of TR data and (b) 30% of TS data

Table 5 offers an overall anomaly classification performance of the ODCNN-CIHAD system on 70% of TR data and 30% of TS data under Kaggle dataset. The outcomes displayed that the ODCNN-CIHAD system has outperformed higher performance in both aspects. For instance, with 70% of TR data, the ODCNN-CIHAD approach has reached higher average $accu_y$ of 97.70%, $prec_n$ of 97.70%, $sens_y$ of 97.74%, $spec_y$ of 97.74%, and F_{score} of 97.70%. Moreover, with 30% of TS data, the ODCNN-CIHAD algorithm has attained maximal average $accu_y$ of 98.21%, $prec_n$ of 98.28%, $sens_y$ of 98.14%, $spec_y$ of 98.14%, and F_{score} of 98.20%.

Table 5 Result analysis of ODCNN-CIHAD approach with various measures under Kaggle dataset

UCI SECOM Dataset					
Class Labels	**Accuracy**	**Precision**	**Sensitivity**	**Specificity**	**F-Score**
Training Set (70%)					
Pass	97.70	98.99	96.54	98.95	97.75
Fail	97.70	96.41	98.95	96.54	97.66
Average	**97.70**	**97.70**	**97.74**	**97.74**	**97.70**
Testing Set (30%)					
Pass	98.21	98.95	97.17	99.11	98.05
Fail	98.21	97.60	99.11	97.17	98.35
Average	**98.21**	**98.28**	**98.14**	**98.14**	**98.20**

The TA and VA gained by the ODCNN-CIHAD methodology on Kaggle dataset are illustrated in Fig. 11. The experimental outcome exposed that the ODCNN-CIHAD technique has achieved maximal values of TA and VA. Particularly the VA performed superior to TA.

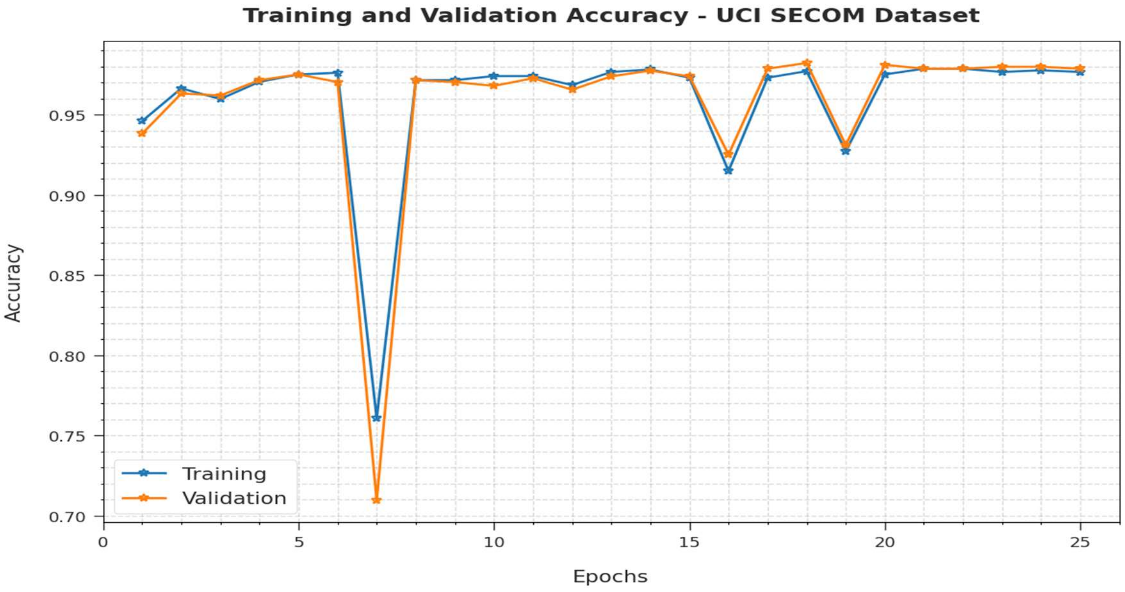


Fig. 11. TA and VA analysis of ODCNN-CIHAD approach under Kaggle dataset

The TL and VL achieved by the ODCNN-CIHAD approach on Kaggle dataset are depicted in Fig. 12. The experimental outcome revealed that the ODCNN-CIHAD algorithm has been able to minimal values of TL and VL. In specific, the VL is lesser than TL.

The TL and VL achieved by the ODCNN-CIHAD approach on Kaggle dataset are depicted in Fig. 12. The experimental outcome revealed that the ODCNN-CIHAD algorithm has been able to minimal values of TL and VL. In specific, the VL is lesser than TL.

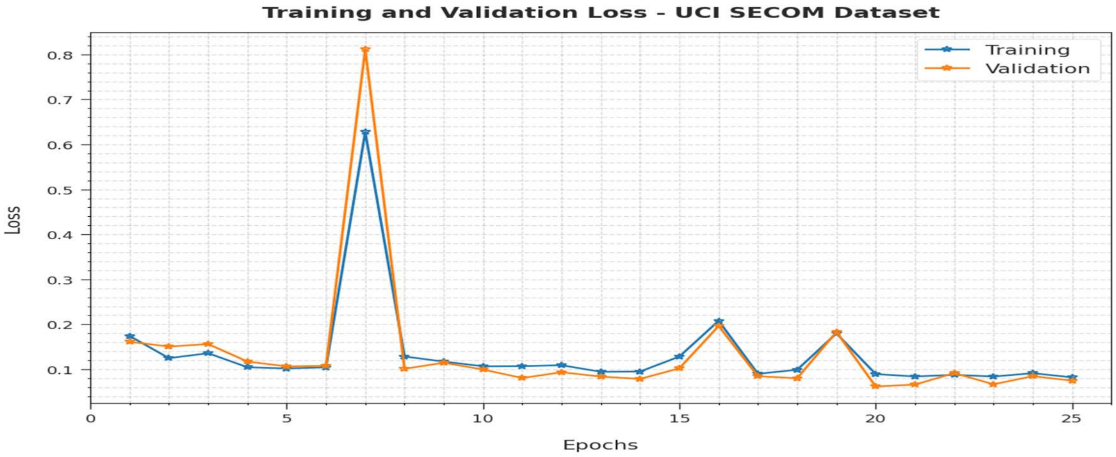


Fig. 12. TL and VL analysis of ODCNN-CIHAD approach under Kaggle dataset

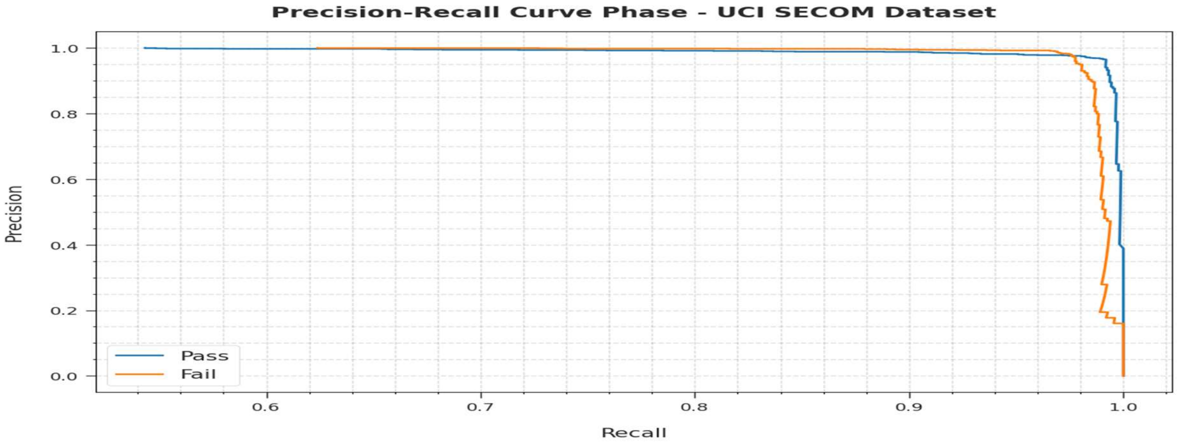


Fig. 13. Precision-recall analysis of ODCNN-CIHAD approach under Kaggle dataset

A clear precision-recall examination of the ODCNN-CIHAD approach on Kaggle dataset is represented in Fig. 13. The figure exposed that the ODCNN-CIHAD methodology has resulted in higher values of precision-recall values under all classes.

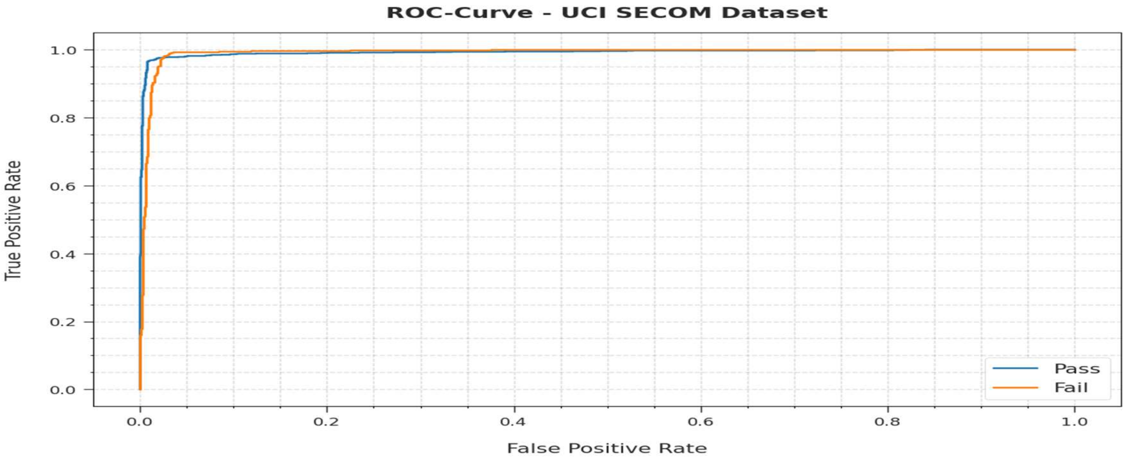


Fig. 14. ROC analysis of ODCNN-CIHAD approach under Kaggle dataset

A brief ROC investigation of the ODCNN-CIHAD method on Kaggle dataset is displayed in Fig. 14. The outcomes referred to as the ODCNN-CIHAD approach have outperformed its capability in categorizing distinct classes on the Kaggle dataset.

Table 6 demonstrates detailed comparative analysis of the ODCNN-CIHAD approach with existing methodologies on the test Kaggle dataset [29]. Comparative analysis of the ODCNN-CIHAD approach with existing methodologies taken from metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment [30].

Fig. 15 depicts the comparison examination of the ODCNN-CIHAD method with existing methodologies with respect to $prec_n$, $sens_y$, and F_{score} on the Kaggle dataset. The figure implied that the ODCNN-CIHAD system has reached maximal values of $prec_n$, $sens_y$, and F_{score} over other techniques. For sample, according to $prec_n$, the ODCNN-CIHAD approach has obtained improved $prec_n$ of 98.28% whereas the DNN layer1, DNN layer2, DNN layer3, DNN layer4, ensemble, PSO ensemble, and SSA-CRNN techniques have attained lesser $prec_n$ values of 90.36%, 89.72%, 90.48%, 90.87%, 90.08%, 91.10%, and 92.03% correspondingly. Besides, based on F_{score}, the ODCNN-CIHAD algorithm has achieved maximal F_{score} of 98.20% whereas the DNN layer1, DNN layer2, DNN layer3, DNN layer4, ensemble, PSO ensemble, and SSA-CRNN systems have obtained minimal F_{score} values of 90.04%, 90.65%, 86.36%, 90.60%, 90.79%, 90.61%, and 91.08% correspondingly.

Table 6 Comparative analysis of ODCNN-CIHAD approach with existing methodologies under Kaggle dataset

Methods	Accuracy	Precision	Sensitivity	F-Score
DNN Layer-1	92.04	90.36	83.83	90.04
DNN Layer-2	92.85	89.72	86.61	90.65
DNN Layer-3	83.29	90.48	87.47	86.36
DNN Layer-4	93.27	90.87	87.98	90.60
Ensemble Model	91.15	90.08	88.23	90.79
PSO Ensemble Model	93.65	91.10	86.58	90.61
SSA-CRNN	96.84	92.03	89.38	91.08
ODCNN-CIHAD	98.21	98.28	98.14	98.20

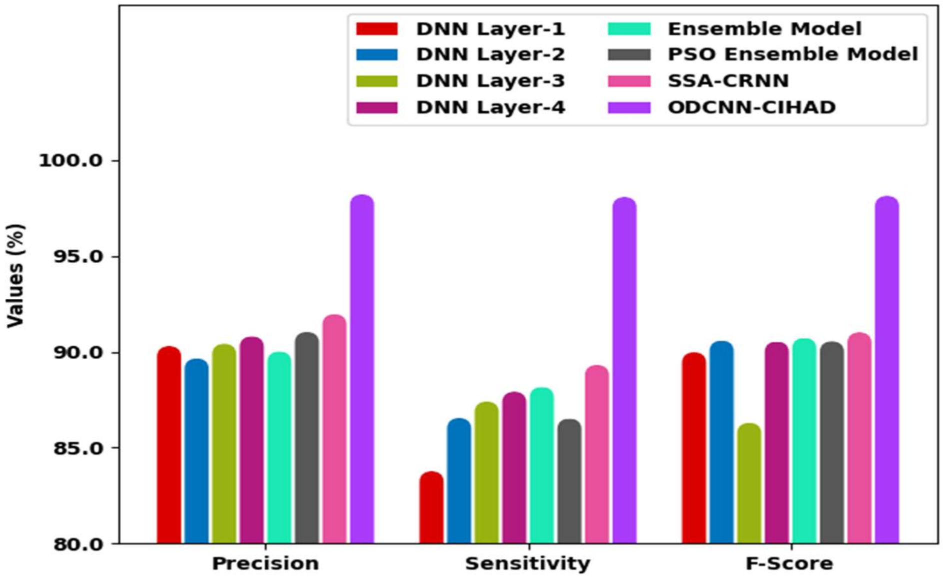


Fig. 15. Comparative analysis of ODCNN-CIHAD approach under Kaggle dataset

Fig. 16 examines the comparison investigation of the ODCNN-CIHAD approach with existing systems in terms of $accu_y$ on the Kaggle dataset. The figure referred that the ODCNN-CIHAD system has attained superior values of $accu_y$ over other models. For instance, based on $accu_y$, the ODCNN-CIHAD method has obtained maximal $accu_y$ of 98.21% whereas the DNN layer1, DNN layer2, DNN layer3, DNN layer4, ensemble, PSO ensemble, and SSA-CRNN approaches have gained reduced $accu_y$ values of 92.04%, 92.85%, 83.29%, 93.27%, 91.15%, 93.65%, and 96.84% correspondingly.

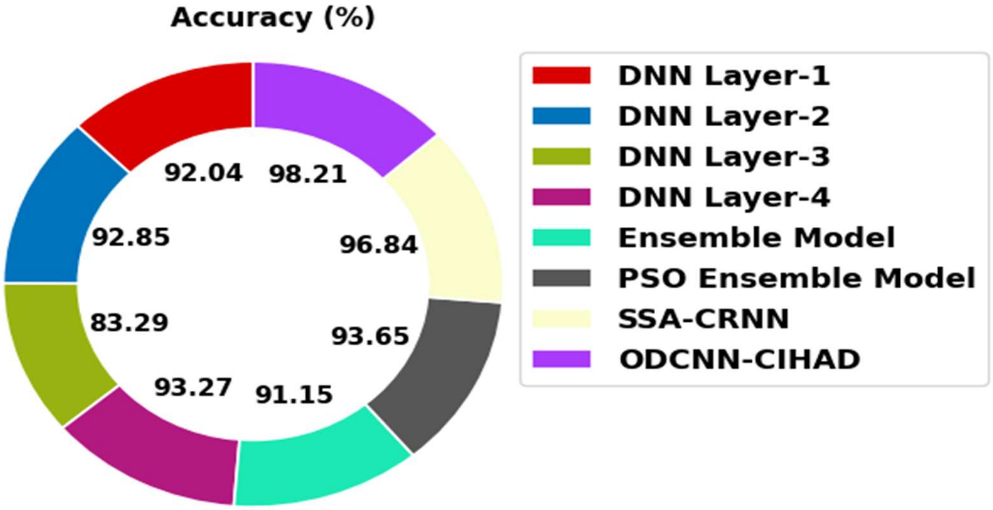


Fig. 16. $Accu_y$ analysis of ODCNN-CIHAD approach under Kaggle dataset

From the detailed results and discussion, it is assumed that the ODCNN-CIHAD model has gained effectual performance over other models on both datasets.

5. CONCLUSION

In this study, a novel ODCNN-CIHAD system was established for anomaly detection and classification in the IIoT environment. The ODCNN-CIHAD system involves a series of subprocesses namely min-max data normalization, SMOTE based class imbalance data handling, GTOA based parameter optimization, DCNN classification, and GTRO based hyperparameter tuning. The experimental validation of the ODCNN-CIHAD technique is carried out utilizing benchmark dataset and the outcomes are investigated under distinct measures. The comparison study highlighted the enhanced performance of the ODCNN-CIHAD technique on existing approaches. Therefore, the ODCNN-CIHAD model is appeared as an effective solution to accomplish security in the IIoT environment. In future, an ensemble of DL classification approaches are integrated into the ODCNN-CIHAD technique to improve the overall classification performance.

Declarations

Funding (information that explains whether and by whom the research was supported)

No funding

Conflicts of interest/Competing interests (include appropriate disclosures)

There is no conflict of interest

Availability of data and material (data transparency)

Data will be made available on reasonable request

Code availability (software application or custom code)

Available on reasonable request

Authors' contributions (optional: please review the submission guidelines from the journal whether statements are mandatory)

The authors confirm contribution to the paper as follows:
Nenavath Chander- study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.
Dr. M. Upendra Kumar reviewed the results and approved the final version of the manuscript.

REFERENCES

[1] Adnan, A., Muhammed, A., Abd Ghani, A.A., Abdullah, A. and Hakim, F., 2021. An Intrusion Detection System for the Internet of Things Based on Machine Learning: Review and Challenges. *Symmetry*, *13*(6), p.1011.

[2] Belenko, V., Chernenko, V., Kalinin, M. and Krundyshev, V., 2018, September. Evaluation of GAN applicability for intrusion detection in self-organizing networks of cyber physical systems. In *2018 International Russian Automation Conference (RusAutoCon)* (pp. 1-7). IEEE.

[3] Panigrahi, R. and Borah, S., 2018. A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems. *International Journal of Engineering & Technology*, *7*(3.24), pp.479-482.

[4] Ahmad, Rasheed, et al. "Towards building data analytics benchmarks for IoT intrusion detection." *Cluster Computing* 25.3 (2022): 2125-2141.

[5] Binbusayyis, A., Alaskar, H., Vaiyapuri, T. and Dinesh, M., 2022. An investigation and comparison of machine learning approaches for intrusion detection in IoMT network. *The Journal of Supercomputing*, pp.1-20.

[6] Ferrag, M.A., Maglaras, L., Ahmim, A., Derdour, M. and Janicke, H., 2020. Rdtids: Rules and decision tree-based intrusion detection system for internet-of-things networks. *Future internet*, *12*(3), p.44.

[7] Li, R., Zhou, Z., Liu, X., Li, D., Yang, W., Li, S. and Liu, Q., 2021. GTF: An Adaptive Network Anomaly Detection Method at the Network Edge. *Security and Communication Networks*, *2021*.

[8] Mokhtari, S., Abbaspour, A., Yen, K. and Sargolzaei, A., 2021. A machine learning approach for anomaly detection in industrial control systems based on measurement data. electronics 2021 10 407.

[9] Himeur, Y., Ghanem, K., Alsalemi, A., Bensaali, F. and Amira, A., 2021. Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. *Applied Energy*, *287*, p.116601.

[10] Ding, H., Chen, L., Dong, L., Fu, Z. and Cui, X., 2022. Imbalanced data classification: A KNN and generative adversarial networks-based hybrid approach for intrusion detection. *Future Generation Computer Systems*, *131*, pp.240-254.

[11] Liang, Wei, et al. "Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial IoT." IEEE Transactions on Industrial Informatics 18.8 (2021): 5087-5095.

[12] Zhang, L., Jiang, S., Shen, X., Gupta, B.B. and Tian, Z., 2021. PWG-IDS: An Intrusion Detection Model for Solving Class Imbalance in IIoT Networks Using Generative Adversarial Networks. *arXiv preprint arXiv:2110.03445*

[13] Luo, S., Zhao, Z. and Hu, Q., 2021, November. Focal loss based two-stage training for class imbalance network intrusion detection. In *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)* (pp. 687-693). IEEE

[14] Toldinas, J., Venčkauskas, A., Damaševičius, R., Grigaliūnas, Š., Morkevičius, N. and Baranauskas, E., 2021. A novel approach for network intrusion detection using multistage deep learning image recognition. *Electronics*, *10*(15), p.1854

[15] Mokhtari, S., Abbaspour, A., Yen, K.K. and Sargolzaei, A., 2021. A machine learning approach for anomaly detection in industrial control systems based on measurement data. *Electronics*, *10*(4), p.407

[16] Abdel-Basset, M., Moustafa, N. and Hawash, H., 2022. Privacy-Preserved Generative Network for Trustworthy Anomaly Detection in Smart Grids: A Federated Semi-Supervised Approach. *IEEE Transactions on Industrial Informatics*

[17] Cao, D., Liu, D., Ren, X. and Ma, N., 2021. Self-Adaption AAE-GAN for Aluminum Electrolytic Cell Anomaly Detection. *IEEE Access*, *9*, pp.100991-101002.

[18] Singh, Bikesh Kumar, Kesari Verma, and A. S. Thoke. "Investigations on impact of feature normalization techniques on classifier's performance in breast tumor classification." International Journal of Computer Applications 116.19 (2015).

[19] Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, pp.321-357.

[20] Shen, F., Zhao, X., Kou, G. and Alsaadi, F.E., 2021. A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique. *Applied Soft Computing*, *98*, p.106852.

[21] Zhang, Y. and Jin, Z., 2020. Group teaching optimization algorithm: A novel metaheuristic method for solving global optimization problems. *Expert Systems with Applications*, *148*, p.113246.

[22] Jin, K.H., McCann, M.T., Froustey, E. and Unser, M., 2017. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, *26*(9), pp.4509-4522.

[23] Abdollahzadeh, B., Soleimanian Gharehchopogh, F. and Mirjalili, S., 2021. Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *International Journal of Intelligent Systems*, *36*(10), pp.5887-5958.

[24] https://www.kaggle.com/mrwellsdavid/unsw-nb15

[25] https://www.kaggle.com/paresh2047/uci-semcom

[26] Kasongo, S.M. and Sun, Y., 2020. Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data*, *7*(1), pp.1-20

[27] Kotecha, K.; Verma, R.; Rao, P.V.; Prasad, P.; Mishra, V.K.; Badal, T.; Jain, D.; Garg, D.; Sharma, S. Enhanced Network Intrusion Detection System. Sensors 2021, 21, 7835. https://doi.org/10.3390/s21237835

[28] Zhou, X., Hu, Y., Liang, W., Ma, J. and Jin, Q., 2020. Variational LSTM enhanced anomaly detection for industrial big data. *IEEE Transactions on Industrial Informatics*, *17*(5), pp.3469-3477

[29] Moldovan, D., Anghel, I., Cioara, T. and Salomie, I., 2020, September. Particle Swarm Optimization Based Deep Learning Ensemble for Manufacturing Processes. In *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)* (pp. 563-570). IEEE.

[30] Chander, N., Upendra Kumar, M. Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment. Cluster Comput (2022). https://doi.org/10.1007/s10586-022-03719-8.

[31] Chander, N., Upendra Kumar, M. (2023). Comparative Analysis on Deep Learning Models for Detection of Anomalies and Leaf Disease Prediction in Cotton Plant Data. In: Kumar, S., Sharma, H., Balachandran, K., Kim, J.H., Bansal, J.C. (eds) Third Congress on Intelligent Systems. CIS 2022. Lecture Notes in Networks and Systems, vol 608. Springer, Singapore. https://doi.org/10.1007/978-981-19-9225-4_20