Tech Science Press

# Class Imbalance Handling with Deep Learning Enabled IoT Healthcare Diagnosis Model

**T. Ragupathi[1,*], M. Govindarajan[1] and T. Priyaradhikadevi[2]**

[1]Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, 608002, Tamil Nadu, India
[2]Department of Computer Science and Engineering, Mailam Engineering College, Tindivanam, Tamil Nadu, 604304, India
*Corresponding Author: T. Ragupathi. Email: raguthamodar@gmail.com

**Abstract:** The rapid advancements in the field of big data, wearables, Internet of Things (IoT), connected devices, and cloud environment find useful to improve the quality of healthcare services. Medical data classification using the data collected by the wearables and IoT devices can be used to determine the presence or absence of disease. The recently developed deep learning (DL) models can be used for several processes such as classification, natural language processing, etc. This study presents a bacterial foraging optimization (BFO) based convolutional neural network-gated recurrent unit (CNN-GRU) with class imbalance handling (CIH) model, named BFO-CNN-GRU-CIH for medical data classification in IoT enabled cloud environment. The proposed BFO-CNN-GRU-CIH model initially enables the IoT devices to gather healthcare data and preprocess it for further processing. In addition, Lempel Ziv Markov chain Algorithm (LZMA) is employed for the compression of healthcare data to reduce the amount of data being communicated. Besides, Synthetic Minority Over-sampling Technique (SMOTE) is applied to handle class imbalance data problems. Moreover, BFO with CNN-GRU model is utilized to perform the classification process in which the hyperparameters of the CNN-GRU model are optimally adjusted by the use of BFO algorithm. In order to showcase the better performance of the BFO-CNN-GRU-CIH model, a wide range of simulations take place on three benchmark datasets and the results portrayed the betterment of the BFO-CNN-GRU-CIH model over the recent state of art approaches.

**Keywords:** IoT; cloud; healthcare; deep learning; parameter tuning; class imbalance handling

## 1 Introduction

The tremendous growth of Internet of Things (IoT), cloud computing (CC), and wearable devices has affected every aspect of people's lives and brought about significant development opportunities. In the medical industry, the adaption of IoT could bring us healthcare monitoring continuously [1]. With the popularity of smart bracelets, smartphones, etc., different kinds of sensors could monitor healthcare indicators accurately and timely [2]. In smart health care system, individuals are remotely monitored for

stopping the spread of disease and provide cost-effective and quick treatment. In this context, the incorporation of machine learning (ML) and IoT-assisted health care systems is considered an ideal solution. The ML and IoT-based solution is effective because of the developments in machine intelligence, sensing, processing, and spectrum utilization. This solution can be feasible because of the advancements in microelectronics which has offered cheap and tiny healthcare sensing devices that have revolutionized healthcare services [3]. Consequently, medical system classifies this solution as preventive treatment and symptomatic treatment. Hence, the development of national health care monitoring methods becoming an inevitable trend nowadays. In recent times, ML algorithms and IoT devices for remotely monitoring peoples for accurate detection have received considerable interest in telemedicine [4]. But, the usage of conventional wireless transmission techniques for health care systems suffers from higher costs and radiation. Deep learning (DL) and ML technologies in IoT assisted health care schemes are widely employed nowadays [5]. Also, both techniques are commonly used in sports for keeping track of the healthcare status of the athletes by means of medical image analysis, image interpretation, diagnosis of the athletes, and injury prediction.

In ML method, DL technique is considered as a novel domain and, as a result, crucial subsets of artificial intelligence (AI). Various descriptions are provided for DL methods, however, the subsequent description is the more wide-ranging of all. The DL method is a sequence of processes established on artificial neural network (ANN) having many layers [6]. Biological neural system stimulated the major concept of NN for processing data [7]. A biological neuron contains different entities, however, exploring the role of the succeeding elements is significant in examining the functionality of ANN. A deep neural network (DNN) is ANN with multiple layers. All the layers are accountable to extract certain data denoted by a score times weight and forward them to the succeeding layers [8].

This study presents a bacterial foraging optimization (BFO) based convolutional neural network-gated recurrent unit (CNN-GRU) with class imbalance handling (CIH) model, named BFO-CNN-GRU-CIH for medical data classification in IoT enabled cloud environment. Moreover, Lempel Ziv Markov chain Algorithm (LZMA) based compression and, Synthetic Minority Over-sampling Technique (SMOTE) based CIH is applied. Furthermore, BFO with CNN-GRU model is utilized to perform the classification process. For inspecting the enhanced outcomes of the BFO-CNN-GRU-CIH model, a series of experiments were carried out on three benchmark datasets and the results depicted the improvement of the BFO-CNN-GRU-CIH model over the recent state of art approaches.

## 2  Related Works

This section offers a detailed review of recent state of art approaches. He et al. [9] introduce a hybrid smart healthcare system called EdgeCNN which balances the capacity of edge CC for resolving the problems to agile learning of health care information from IoT devices. (2) proposed an efficient DL method electrocardiogram (ECG) inference, i.e., positioned for running on edge smart device for lower-latency detection. (3) develop a data augmentation methodology for ECG based deep convolution generative adversarial networks for expanding ECG data volume. Sharma et al. [10] presented a DL based Internet of Healthcare architecture to Support of Alzheimer's Patient (DeTrAs) technique. Recurrent neural network (RNN) based Alzheimer predictive technique was presented that employs sensory motion data, an ensemble model to track abnormality for Alzheimer patient, and also an IoT-based support system for the Alzheimer patient is proposed.

Yu et al. [11] designed a group of streamlined detection methods and learning models for edge computing based CNN, facilitating EdgeCNN to infer and identify ECG in real time, since certain health care applications utilize smart devices on the edge. This experiment result shows that under the principle of guaranteeing accuracy, EdgeCNN has considerable benefits in network I/O, diagnosis delay, resource

cost, and application usability, when compared to the framework based solely on the CC system. Veeramakali et al. [12] proposed an optimum DL based secured blockchain (ODLSB) assisted smart IoT and health care detection method. The presented method includes medical diagnosis, secure transaction, and hash value encrypted. The ODLSB method includes the orthogonal particle swarm optimization (OPSO) method for the secure sharing of healthcare images. Furthermore, the hash value encryption method is performed via neighbourhood indexing sequence (NIS) method. Finally, the optimum deep neural network (ODNN) is employed as a classification method for disease diagnosis. Tuli et al. [13] present a framework termed HealthFog for combining ensemble DL in Edge computing device and positioned it for a real-time application of automated Heart Disease detection. HealthFog provides health care as fog service with IoT devices and effectively handles the heart patient's information that comes as a user request.

## 3 The Proposed Model

In this study, a novel BFO-CNN-GRU-CIH model has been developed to diagnose the presence of disease in the IoT enabled cloud environment. The presented BFO-CNN-GRU-CIH model performs data acquisition, data pre-processing, LZMA based compression, SMOTE based CIH, CNN-GRU based classification, and BFO based parameter tuning. Fig. 1 displays the overall block diagram of proposed BFO-CNN-GRU-CIH technique. The detailed work of every module is elaborated in the succeeding sections.
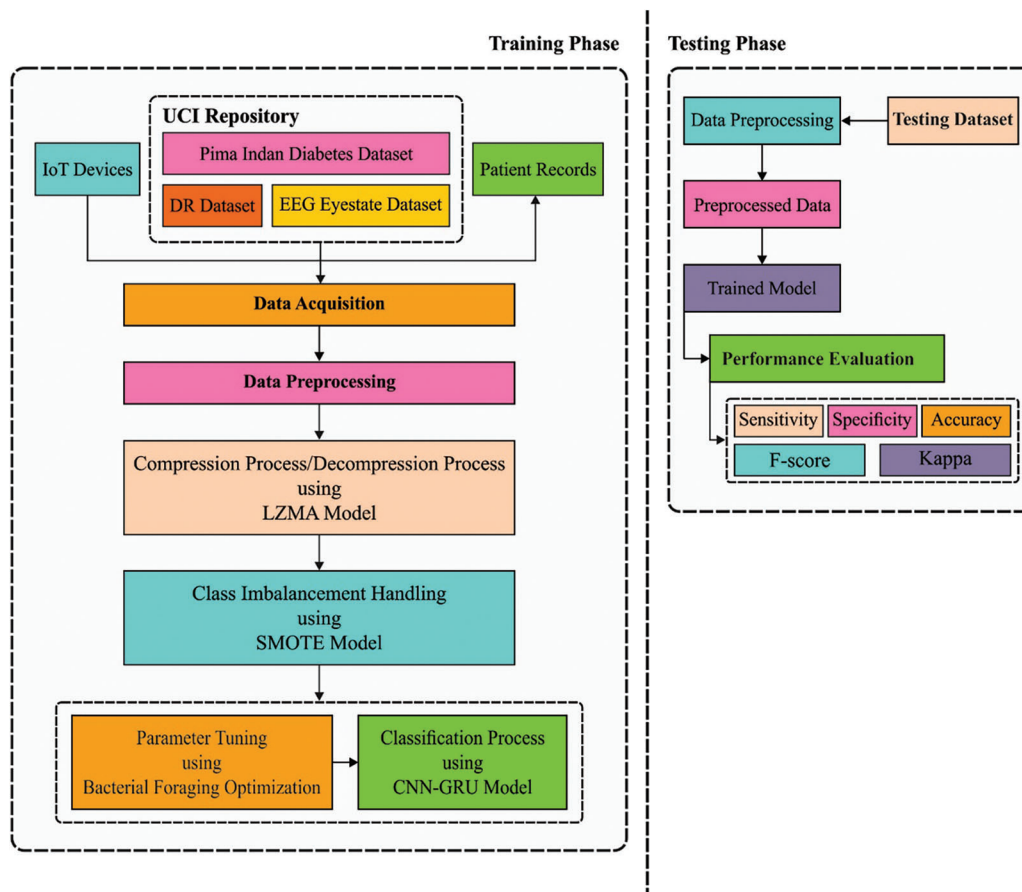


**Figure 1:** Block diagram of BFO-CNN-GRU-CIH technique

### 3.1 Data Pre-Processing

During this step, the IoT devices primarily captured the medicinal information of patients and pre-processing. Then the data pre-processing, input data from some format are changed as to the .arff format to generate it well-suited for further processing. Afterward, the outcome pre-processed data make sure classification utilizes the BFO-CNN-GRU-CIH manner.

### 3.2 LZMA Based Compression/Decompression Process

Next to data pre-processing, LZMA model is used to compress and decompress the healthcare data, in such a way as to lessen the quantity of data transmission or storage. The LZMA method was developed by Pavlov, and their fundamental is depending on the enhancement of LZ_77 compression model. LZMA employs sliding window based interval coding and dynamic dictionary compression algorithms that have the benefits of fast speed, higher compression rate, and smaller decompression space requirements. The LZMA assists a dictionary space of 4 KB to millions of MBs that rises the compression rate and cause its searching cache space to be larger [14]. To decrease the time needed for matching the longer string and rapidly seek matching character, in the execution of the LZMA, many feasible matches are saved in the Hash table, and the framework of the data of binary search tree/Hash linked list is utilized for searching information. The LZMA encoder set up dissimilar stages of hash function for 2, 3, and 4 nearby bytes to attain effective deployment respective to distinct dictionary sizes.

### 3.3 SMOTE Based CIH Process

In order to handle the class imbalance data, the SMOTE technique is applied to it. This over-sampling procedure can be attained by further synthetic information. As per, the original information attained by SMOTE is utilized for synthesizing novel minority information which is distinct from the original one, thus alleviating the effects of over-fitting on the minority class. The SMOTE is depending on the concept of the k-nearest neighbor (KNN) and considers a synthetic data instance which is interpolated among original and nearby neighbors. The SMOTE method estimates the neighboring environments of every data instance from the minority classes, arbitrarily chooses its neighbor, and makes synthetic information by using the interpolation of information among every instance and the nearby neighbor elected. Once the amount of synthetic data instance to be generated is small when compared to the size of original data sets, the process is arbitrarily chosen and an original data instance is employed for creating synthetic data instance. On the other hand, the amount of synthetic data instance to be generated is larger when compared to the size of original dataset, the process continuously generates synthetic data instances with predefined over-sampling ratio [15]. The SMOTE method requires input through the amount of minority data instance (T), the nearest neighbors (k), and oversampling ratio (N). The primary method is determining and searching the closest neighbor, followed by synthetic creation via data interpolation among all the nearest neighbors and minority instances.

---

**Algorithm 1:** SMOTE (T, N, k)

---

**Input:** Minority class samples count T, Quantity of SMOTE N%, Amount of nearest neighbor k

**Output:** $(N/100)^*T$ synthetic minority class instance

1. (*If $N < 100\%$, randomizes the minority class instance as an arbitrary percentage become SMOTE d.*)

2. if $N < 100$

3.   then Randomizes the $T$ minority class instances

4.     $T = (N/100) * T$

5.     $N = 100$

---

---

**Algorithm 1 (continued).**

---

6. end if

7. $N = (int)(N/I00)$ (*The quantity of SMOTE is consider to be in integral multiples of I 00. *)

8. k = Nearest neighbor count

9. *numattrs* = Feature count

10. *Sclmple*[][]: array for original minority class instances

11. *newindex*: keep an overall amount of synthetic instance made, initiated to zero

12. Synthetic[][]: array for synthetic instance

(*Calculate KNN for all minority class samples. *)

13. **for** $i \leftarrow 1$ **to** $T$

14.      Calculate KNN for $i$, and save *nnarray* indices

15.      Populate $(N, \quad i, \quad nn\text{array})$

16. **end for**

Populate $(N, \quad i, \quad nn\text{array})(*$Functions to create the synthetic instance. *)

17. **while** $N \neq 0$

18.      Select an arbitrary value among 1 and call it *nn*. this stage selects one of the KNN of $i$.

19.      for $attr \leftarrow 1$ to *numattrs*

20.              Calculate: $dif = Sample[nnarray[nn]][attr] - Sample[i][attr]$

21.              Calculate: gap = arbitrary numbers among [0, 1]

22.              $Synthetic[newindex][arrr] = Sample[i][arrr] + gap * dif$

23.      End for

24.      $newindex + +$

25.      $N = N - 1$

26. End while

27. Return (*End* of Populate.*)

End

---

### 3.4 CNN-GRU Based Classification

At this stage, the CNN-GRU model is applied to classify the presence or absence of disease. CNN is MLNN framework that is collected of several convolution layers, pooling layers, and fully connected (FC) layers, whereas the convolution as well as pooling layers are utilized for extracting features. Besides, the FC layer was utilized for classifying the medical data. All the layers are linked to the prior layer outcome, and their outcome permits to next layer. The parameter of the network is shared and trained by backpropagation (BP) technique. The LSTM is 3 gates containing forget, update, and input gates. In the meantime, the GRU has easy as that LSTM as it has only 2 gates that are reset as well as updated gates. These gates are utilized for determining whether the data was helpful or not. The helpful data was kept but useless data was forgotten. Therefore, at same CNN-FC layer, neural unit isn't linked to one another. The GRU network interchange CNN-FC layer for transforming the classification tasks as

to order tasks, but the classification outcomes of all feature maps are also the next feature map classification computation from similar hidden layer for improving the CNN detection efficiency [16]. The presented GRU framework is illustrated in Fig. 2. Besides, the dropout technique that is arbitrarily deleting few neural units under the hidden layer was utilized for avoiding the probable over-fit issue from the CNN-GRU technique.
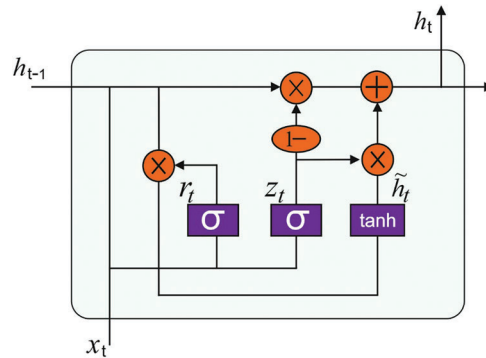


**Figure 2:** Structure of GRU model

In CNN-GRU framework, a novel CNN input, convolution, and pooling layers parameter with sizes are unaltered for implementing the feature extraction. All the resultant feature maps of $l^{th}$ convolution layer was calculated as:

$$O\_conv_j^l = sigmoid\left(\sum_{i=1}^{M} convn(a_i^{l-1*}k_{ij}^l) + b_j^l\right) \tag{1}$$

where $a_i^{l-1}$ refers the $i$th feature map from the entire feature maps $M$ of $l-1^{th}$ prior layer, and $k_{ij}^l$ stands for the $l^{th}$ layer kernels. The value at place of $(m, n)$ from the convolutional function Conv was determined as in Eq. (2).

$$convn\ (a_i^{l-1},\ k_{ij}^l)[m][n] = \sum_{w=0}^{k-1}\sum_{l=0}^{k-1} k_{ij}^l[w][l] * a_i^{l-1}[m+w][n+l] \tag{2}$$

where $k = 5$ has the convolution kernel width. During the pooling layer, all resultant feature maps value at place of $(m, n)$ was computed by utilizing average pooling technique demonstrated in Eq. (3), where $a_j$ is $j^{th}$ feature map of $l-1$ convolution layer, $K = 2$ is the pooling kernel widths.

$$aver\_pool_j^l[m][n] = \frac{1}{4}\sum_{l,w=0}^{K-1} a_j^{l-1}(m+l,\ n+w) \tag{3}$$

Next, every feature of all the feature maps from the last CNN pooling layer is linked to equivalent GRU manner. At last, the GRU outcomes are stimulated by softmax function determined as Eq. (4) for classifying the medical data. On the testing stages, every GRU output is voted for determining the last detection outcome.

$$y_k = \frac{\exp\ (w_k h)}{\sum_{k'} \exp\ (w_k \cdot h)} \tag{4}$$

### 3.5 BFO Based Hyperparameter Tuning Process

Finally, the BFO algorithm is executed to optimally choose the hyperparameter of the CNN-GRU model. BFO is stimulated by the foraging nature of E. coli. For determining the optimal food source and circumventing toxic regions, the E.coli performs swimming or tumbling process utilizing the flagella which are capable of rotating in the clockwise or anti-clockwise direction based on the atmosphere. Based on the natural selection process, the bacteria with minimum fitness get discarded whereas the ones with maximum fitness undergo reproduction process. Besides, the variations in the atmosphere also result in the disappearance or proliferation of bacteria in the region. Two of the major movements of E-coli are swimming and tumbling. Using the foraging nature of the E. coli, the BFO algorithm encompasses different processes as defined in the following:

Chemotaxis denotes the searching nature of the bacteria. If the bacteria searches for food, two types of motions are involved. Firstly, the swimming process enables the bacterium for moving in a particular direction; secondly, tumbling process denotes the direction of motion. During chemotaxis, the bacteria observes the nutrient content in the nearby atmosphere and therefore guides the chemotaxis nature. The chemotaxis operation of the $i$th bacterium at every level is defined as follows:

$$\theta^i(j+1,\ k,\ l) = \theta^i(j,\ k,\ l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{5}$$

where $\theta^i(j, k, l)$ determines the location of the ith bacterium, $C(i)$ indicates the moving step size of the $i$th bacterium and $\Delta$ signifies a unit vector from arbitrary direction.

Then, the bacteria interact with one another at the time of foraging. The focus of nutrients from a particular region is maximum, bacteria send out signals and produce bacteria for accumulating at this region [17]. Likewise, when the focus of nutrients from a specific region is minimum, signal is also broadcast that bacteria are kept at distance. This interface speeds up its search to regions with maximum focuses on nutrients.

In the whole chemotaxis procedure, various fitness values that regions are appropriate to the survival of bacteria were defined. Afterward, 1/2 of bacteria from the location in which the nutrient is comparatively worse is removed, and other 1/2 bacteria self-replicate. So, the amount of bacteria isn't modified under the procedure of reproductions. But, the convergence speed of technique is accelerated than the reproduction function.

Elimination-Dispersal could be risk of local optimum solution from the chemotaxis procedure. Thus, the elimination-dispersal function must follow the procedure of bacterial foraging. The bacteria go away at present place with probability of Ped and perform in an arbitrary position. This procedure is supports bacteria for foraging from a novel region and also used this technique for searching to the global optimum solutions.

The major intention of the BFO algorithm is to effectively adjust the hyperparameter for accomplishing maximum medical data classification performance. The attained optimal values are fed into the CNN-GRU model to carry out the classification process using the ten-fold cross validation. The overall classification error rate can be utilized as the fitness function, as defined in the following:

$$fitness = \frac{\left(\sum_{i=1}^{K} testError_i\right)}{k} \tag{6}$$

where test Error$_i$ denotes the average test error of the CNN-GRU classification model.

## 4  Experimental Validation

In this section, a detailed validation of the performance of the proposed BFO-CNN-GRU-CIH model takes place using three benchmark dataset namely PIMA Indians diabetes, diabetic retinopathy (DR), and Electroencephalography (EEG) EyeState. The details related to the dataset are provided in Tab. 1.

**Table 1:**  Dataset description

| Description | Pima indian diabetes | Diabetic retinopathy | EEG EyeState |
|---|---|---|---|
| No. of instances | 768 | 1151 | 7849 |
| No. of attributes | 8 | 19 | 14 |
| No. of class | 2 | 2 | 2 |
| % of Positive samples | 34.90% | 53.08 | 62.21 |
| % of Negative samples | 65.10% | 46.92 | 37.79 |
| Data sources | [18] | [19] | [20] |

Tab. 2 offers the compression results analysis of LZMA technique interms of compressed file size. The LZMA model has effectively compressed the medical datasets. For instance, on the test PIMA Indians diabetes dataset with 34 kB size, the LZMA model produces the compressed file of 8 kB size. Similarly, on the test DR dataset with 124 kB size, the LZMA method produces the compressed file of 41 kB size. Likewise, on the test EEG EyeState dataset with 869 kB size, the LZMA approach takes the compressed file of 173 kB size.

**Table 2:**  Results analysis of LZMA based compression process

| Dataset | Original size (KB) | Compressed size (KB) |
|---|---|---|
| Pima indian diabetes | 34 | 08 |
| Diabetic retinopathy | 124 | 41 |
| EEG EyeState | 869 | 173 |

Tab. 3 provides a brief results analysis of CIH process using SMOTE. The results show that the SMOTE proficiently balances the instances under all classes. For instance, with PIMA Indians diabetes dataset, the number of instances after CIH under negative and positive classes are 500 and 500 respectively. Eventually, with DR Dataset, the amount of instances after CIH under negative and positive classes are 611 and 611 correspondingly. Meanwhile, with EEG EyeState dataset, the count of instances after CIH under negative and positive classes are 4883 and 4883 correspondingly.
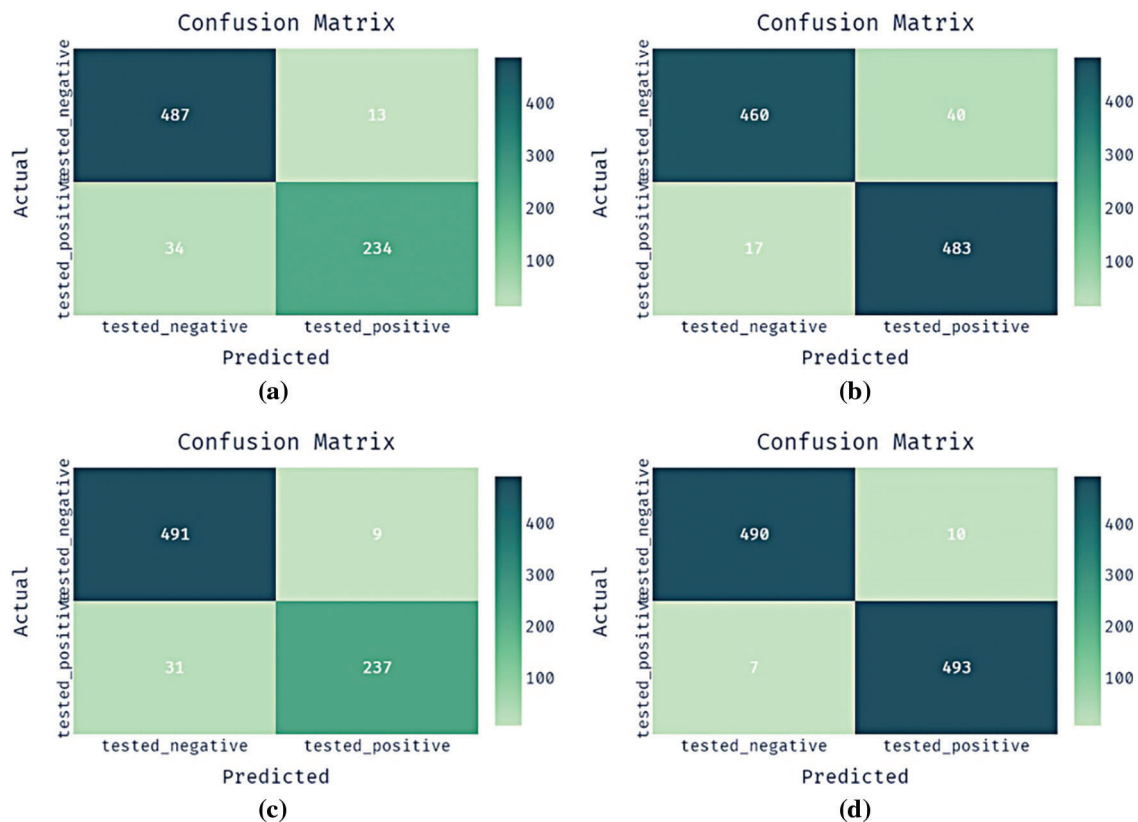
### 4.1  Results Analysis of BFO-CNN-GRU-CIH Model on PIMA Indians Diabetes Dataset

Fig. 3 demonstrates four confusion matrices offered by the proposed BFO-CNN-GRU-CIH model with other methods on the test PIMA Indians Diabetes dataset. The results show that the CNN-GRU model has classified 487 instances into negative class and 234 instances into positive class. In line with, the CNN-GRU-CIH technique has identified 460 instances into negative class and 483 instances into positive class. Along with that, the BFO-CNN-GRU model has categorized 491 instances into negative class and 237 instances into positive class. However, the proposed BFO-CNN-GRU-CIH model has reported effective outcomes by classifying 490 instances into negative classes and 193 instances into positive classes.

**Table 3:** Results analysis of SMOTE based outlier detection process

| Dataset | No. of classes | Before sampling | After sampling |
|---|---|---|---|
| Pima indian diabetes | Tested negative | 500 | 500 |
| | Tested positive | 268 | 500 |
| Diabetic retinopathy | No sign DR | 540 | 611 |
| | Sign DR | 611 | 611 |
| EEG EyeState | Eye open | 4883 | 4883 |
| | Eye closed | 2966 | 4883 |



**Figure 3:** Confusion matrix pima indian diabetes dataset a) CNN-GRU b) CNN-GRU-CIH c) BFO-CNN-GRU d) BFO-CNN-GRU-CIH

Tab. 4 provides a brief results analysis of the BFO-CNN-GRU-CIH model with other techniques on the PIMA Indians Diabetes dataset. The results represented that the CNN-GRU model has shown least performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 93.88%, 97.40%, 87.31%, 95.39%, and 86.28% respectively. At the same time, the CNN-GRU-CIH technique has gained slightly enhanced outcome with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 94.30%, 92%, 96.60%, 94.17%, and 88.60% respectively. In line with, the BFO-CNN-GRU model has accomplished moderate performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 94.79%, 98.20%, 88.43%, 96.09%, and 88.31% respectively. However, the BFO-CNN-GRU-CIH model has outperformed the other methods with the maximum $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 98.30%, 98.00%, 98.60%, 98.29%, and 96.60% respectively. From the

table, it can be noticeable that the proposed BFO-CNN-GRU-CIH model has obtained 0.20% reduced $sens_y$ compared to the BFO-CNN-GRU model due to the fact that the BFO-CNN-GRU-CIH model has classified 490 instances into negative class label whereas the BFO-CNN-GRU model has classified 491 instances into negative class label. However, the BFO-CNN-GRU-CIH model has obtained maximum $accu_y$ of 98.30%, which is higher than the accuracy obtained by the other methods, i.e., outperformed the BFO-CNN-GRU model with the accuracy of 3.51%.

**Table 4:** Results analysis of proposed methods on pima indian diabetes dataset

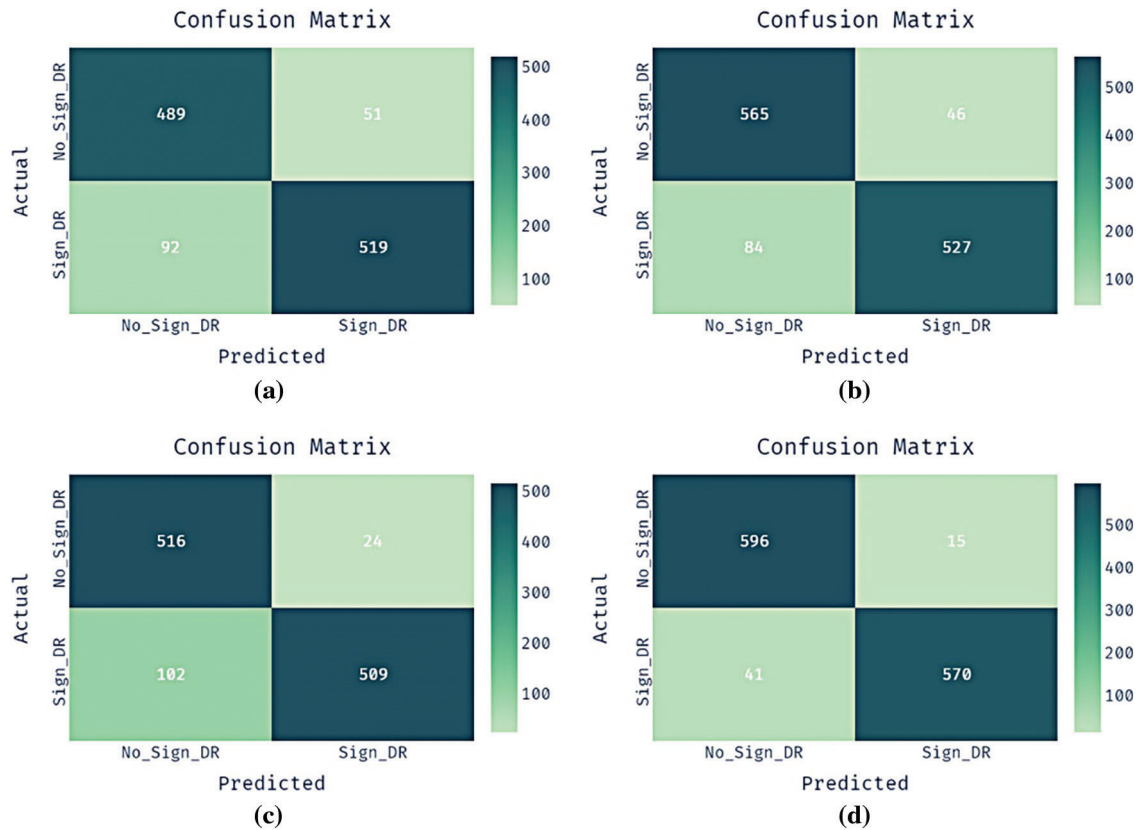| Methods | CNN-GRU | CNN-GRU-CIH | BFO-CNN-GRU | BFO-CNN-GRU-CIH |
|---|---|---|---|---|
| Accuracy | 93.88 | 94.30 | 94.79 | 98.30 |
| Sensitivity | 97.40 | 92.00 | 98.20 | 98.00 |
| Specificity | 87.31 | 96.60 | 88.43 | 98.60 |
| F-score | 95.39 | 94.17 | 96.09 | 98.29 |
| Kappa score | 86.28 | 88.60 | 88.31 | 96.60 |

### 4.2 Results Analysis of BFO-CNN-GRU-CIH Model on Diabetic Retinopathy Dataset

Fig. 4 illustrates four confusion matrices offered by the presented BFO-CNN-GRU-CIH technique with other approaches on the test DR dataset. The outcomes exhibited that the CNN-GRU system has classified 489 instances into negative class and 519 instances into positive class. Similarly, the CNN-GRU-CIH manner has identified 565 instances into negative class and 527 instances into positive class. Followed by, the BFO-CNN-GRU technique has categorized 516 instances into negative class and 509 instances into positive class. Finally, the proposed BFO-CNN-GRU-CIH methodology has reported effectual outcomes by classifying 596 instances into negative class and 570 instances into positive class.

Tab. 5 offers a brief outcomes analysis of the BFO-CNN-GRU-CIH method with other manners on the DR Dataset. The outcomes depicted that the CNN-GRU methodology has exhibited worse performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 87.58%, 90.56%, 84.94%, 87.25%, and 75.17% correspondingly. Simultaneously, the CNN-GRU-CIH method has achieved somewhat increased outcome with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 89.36%, 92.47%, 86.25%, 89.68%, and 78.72% correspondingly. At the same time, the BFO-CNN-GRU system has accomplished moderate performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 89.05%, 95.56%, 83.31%, 89.12%, and 78.21% respectively. At last, the BFO-CNN-GRU-CIH algorithm has demonstrated the other approaches with the maximal $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 95.42%, 97.55%, 93.29%, 95.51%, and 90.83% correspondingly.

### 4.3 Results Analysis of BFO-CNN-GRU-CIH Model on EEG EyeState Dataset

Fig. 5 showcases four confusion matrices existing by the projected BFO-CNN-GRU-CIH system with other techniques on the test EyeState dataset. The outcomes outperformed that the CNN-GRU approach has classified 4731 instances into negative class and 2848 instances into positive class. In addition, the CNN-GRU-CIH method has identified 4787 instances into negative class and 4840 instances into positive class. Besides, the BFO-CNN-GRU technique has categorized 4851 instances into negative class and 2935 instances into positive class. Eventually, the proposed BFO-CNN-GRU-CIH manner has reported effective results by classifying 4873 instances into negative class and 4875 instances into positive class.
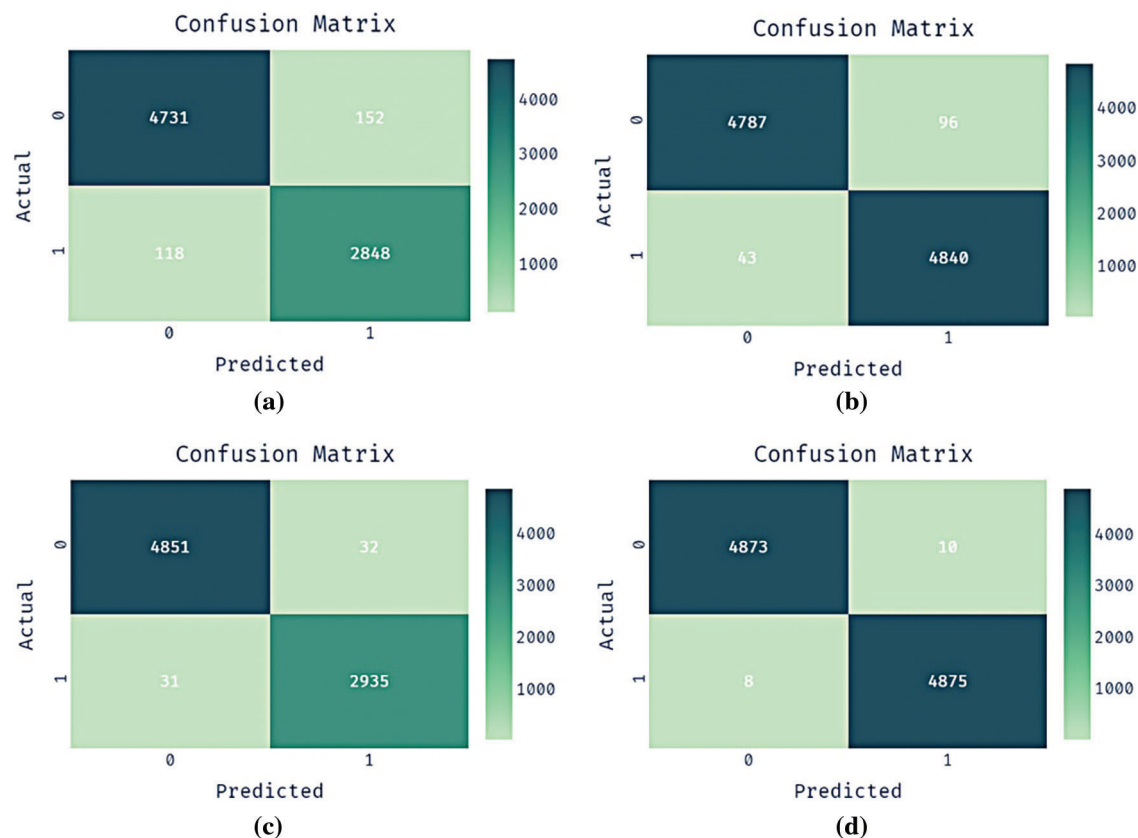
**Figure 4:** Confusion matrix diabetic retinopathy dataset a) CNN-GRU b) CNN-GRU-CIH c) BFO-CNN-GRU d) BFO-CNN-GRU-CIH

**Table 5:** Results analysis of proposed methods on diabetic retinopathy dataset

| Methods | CNN-GRU | CNN-GRU-CIH | BFO-CNN-GRU | BFO-CNN-GRU-CIH |
|---|---|---|---|---|
| Accuracy | 87.58 | 89.36 | 89.05 | 95.42 |
| Sensitivity | 90.56 | 92.47 | 95.56 | 97.55 |
| Specificity | 84.94 | 86.25 | 83.31 | 93.29 |
| F-score | 87.25 | 89.68 | 89.12 | 95.51 |
| Kappa score | 75.17 | 78.72 | 78.21 | 90.83 |

Tab. 6 offers a brief outcomes analysis of the BFO-CNN-GRU-CIH model with other methods on the EEG EyeState Dataset. The outcomes outperformed that the CNN-GRU system has demonstrated minimum performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 96.56%, 96.89%, 96.02%, 97.23%, and 92.70% correspondingly. Concurrently, the CNN-GRU-CIH technique has gained slightly higher outcome with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 98.58%, 98.03%, 99.12%, 98.57%, and 97.15% respectively. Afterward, the BFO-CNN-GRU approach has accomplished moderate performance with the $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 99.20%, 99.34%, 98.95%, 99.35%, and 98.29% correspondingly. Finally, the BFO-CNN-GRU-CIH methodology has depicted the other systems with the superior $accu_y$, $sens_y$, $spec_y$, F-score, and kappa of 99.82%, 99.80%, 99.84%, 99.82%, and 99.63% correspondingly.

**Figure 5:** Confusion matrix EyeState dataset a) CNN-GRU b) CNN-GRU-CIH c) BFO-CNN-GRU d) BFO-CNN-GRU-CIH

**Table 6:** Results analysis of proposed methods on EyeState dataset

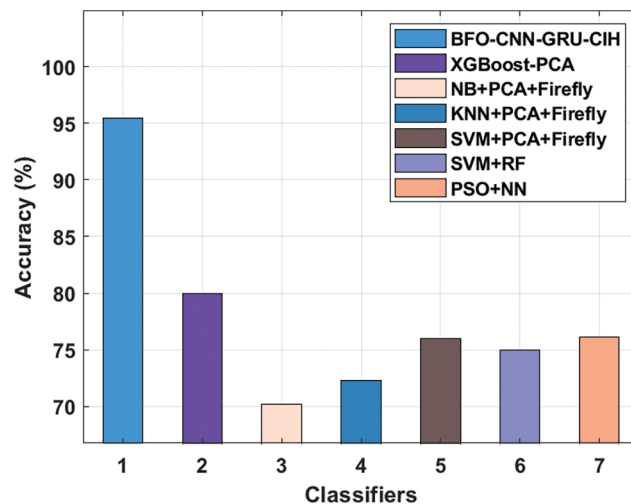| Methods | CNN-GRU | CNN-GRU-CIH | BFO-CNN-GRU | BFO-CNN-GRU-CIH |
|---|---|---|---|---|
| Accuracy | 96.56 | 98.58 | 99.20 | 99.82 |
| Sensitivity | 96.89 | 98.03 | 99.34 | 99.80 |
| Specificity | 96.02 | 99.12 | 98.95 | 99.84 |
| F-score | 97.23 | 98.57 | 99.35 | 99.82 |
| Kappa score | 92.70 | 97.15 | 98.29 | 99.63 |

### 4.4 Comparative Results Analysis of BFO-CNN-GRU-CIH Model

In order to ensure the enhanced performance of the BFO-CNN-GRU-CIH model, a comparative results analysis with recent methods [21,22] takes place on PIMA Indians Diabetes dataset in Fig. 6 From the experimental values, it is noticed that the teaching and learning based optimization (TLBO) + radial basis function network (RBFN), and Rotation Forest (ROF) methods have resulted in ineffective outcomes with the lower $accu_y$ of 66.66% and 67.20%. Followed by, the ROF + kernel extreme learning machine (KELM) and iTLBO + RBFN techniques have accomplished moderately reasonable $accu_y$ of 78.91% and 81.77% respectively. However, the presented BFO-CNN-GRU-CIH model has outperformed the other techniques by attaining maximum $accu_y$ of 98.30%.
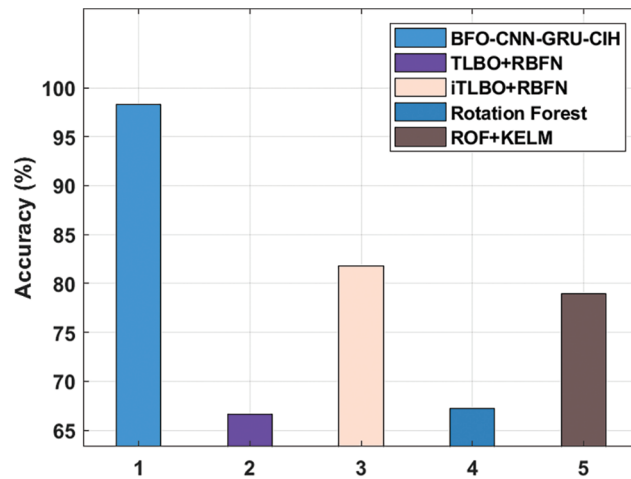
**Figure 6:** Accuracy analysis of BFO-CNN-GRU-CIH method on PIMA indians diabetes DATASET

To make sure the improved performance of the BFO-CNN-GRU-CIH method, a comparative outcomes analysis with recent approaches take place [23–25] on DR Dataset in Fig. 7. From the experimental values, it can be obvious that the Naïve Bayes (NB) + Principal Component Analysis (PCA) + Firefly, KNN + PCA + Firefly, and support vector machine (SVM) + random forest (RF) techniques have resulted in ineffective outcomes with the reduced $accu_y$ of 70.20%, 72.30%, and 75%. Next, the SVM + PCA + Firefly, particle swarm optimization (PSO) + neural network (NN), and XGBoost-PCA methods have accomplished moderately reasonable $accu_y$ of 76%, 76.11%, and 80% respectively. But, the presented BFO-CNN-GRU-CIH technique has demonstrated the other techniques by attaining maximal $accu_y$ of 95.42%.



**Figure 7:** Accuracy analysis of BFO-CNN-GRU-CIH method on DR dataset

For ensuring the increased performance of the BFO-CNN-GRU-CIH approach, a comparative results analysis with recent algorithms [26–29] occurs on EEG EyeState dataset in Fig. 8. From the experimental values, it can be stated that the incremental attribute learning with time-series classification (IAL-TSC) and simulated annealing (SA) with long short term memory (LSTM) manners have resulted in ineffective outcomes with the decreased $accu_y$ of 72.60% and 84.49%. Afterward, the Cosine + KNN and RBF +

improved PSO (IPSO) methods have accomplished moderately reasonable $accu_y$ of 93.50% and 95.19% correspondingly. Finally, the presented BFO-CNN-GRU-CIH algorithm has exhibited the other techniques by attaining higher $accu_y$ of 99.82%.
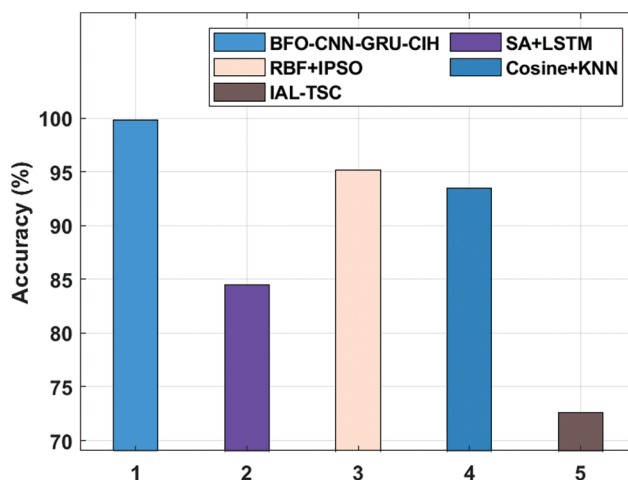


**Figure 8:** Accuracy analysis of BFO-CNN-GRU-CIH method on EEG EyeState DATASET

After examining the results and discussion, it is ensured that the BFO-CNN-GRU-CIH model has resulted in maximum classification performance on medical data in real time environment.

## 5 Conclusion

In this study, a novel BFO-CNN-GRU-CIH model has been developed to diagnose the presence of disease in the IoT enabled cloud environment. The presented BFO-CNN-GRU-CIH model performs data acquisition, data pre-processing, LZMA based compression, SMOTE based CIH, CNN-GRU based classification, and BFO based parameter tuning. In order to showcase the better performance of the BFO-CNN-GRU-CIH model, a wide range of simulations take place on three benchmark datasets and the results portrayed the betterment of the BFO-CNN-GRU-CIH model over the recent state of art approaches. Therefore, the BFO-CNN-GRU-CIH model can be utilized as an efficient method for healthcare diagnosis in real time environment. In future, outlier detection and clustering approaches can be developed to further boost the classification performance of the BFO-CNN-GRU-CIH model.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] B. Guo, Y. Ma, J. Yang and Z. Wang, "Smart healthcare system based on cloud-internet of things and deep learning," *Journal of Healthcare Engineering*, vol. 2021, pp. 1–10, 2021.

[2] M. Chen, W. Li, Y. Hao, Y. Qian and I. Humar, "Edge cognitive computing based smart healthcare system," *Future Generation Computer Systems*, vol. 86, pp. 403–411, 2018.

[3] M. Z. Uddin, "A wearable sensor-based activity prediction system to facilitate edge computing in smart healthcare system," *Journal of Parallel and Distributed Computing*, vol. 123, pp. 46–53, 2019.

[4]  B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Computer Methods and Programs in Biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.

[5]  Y. Kim and T. Moon, "Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 1, pp. 8–12, 2016.

[6]  H. Bolhasani, M. Mohseni and A. M. Rahmani, "Deep learning applications for IoT in health care: A systematic review," *Informatics in Medicine Unlocked*, vol. 23, pp. 100550, 2021.

[7]  L. Deng, "Deep learning: Methods and applications," *FNT in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[8]  K. Chung and H. Yoo, "Edge computing health model using P2P-based deep neural networks," *Peer-to-Peer Netorking and Applications*, vol. 13, no. 2, pp. 694–703, 2020.

[9]  Y. He, B. Fu, J. Yu, R. Li and R. Jiang, "Efficient learning of healthcare data from IoT devices by edge convolution neural networks," *Applied Sciences*, vol. 10, no. 24, pp. 8934, 2020.

[10]  S. Sharma, R. K. Dudeja, G. S. Aujla, R. S. Bali and N. Kumar, "Detras: Deep learning-based healthcare framework for IoT-based assistance of Alzheimer patients," *Neural Computing and Applications*, 2020. https://doi.org/10.1007/s00521-020-05327-2.

[11]  J. Yu, B. Fu, A. Cao, Z. He and D. Wu, "EdgeCNN: A hybrid architecture for agile learning of healthcare data from IoT devices," in *2018 IEEE 24th Int. Conf. on Parallel and Distributed Systems (ICPADS)*, Singapore, pp. 852–859, 2018.

[12]  T. Veeramakali, R. Siva, B. Sivakumar, P. C. Senthil Mahesh and N. Krishnaraj, "An intelligent internet of things-based secure healthcare framework using blockchain technology with an optimal deep learning model," *Journal of Supercomputing*, vol. 77, no. 9, pp. 9576–9596, 2021.

[13]  S. Tuli, N. Basumatary, S. S. Gill, M. Kahani, R. C. Arya *et al.,* "HealthFog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments," *Future Generation Computer Systems*, vol. 104, pp. 187–200, 2020.

[14]  B. Li, J. Xu and Z. Liu, "SW-LZMA: Parallel implementation of LZMA based on SW26010 many-core processor," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–10, 2021.

[15]  Asniar, N. U. Maulidevi and K. Surendro, "SMOTE-LOF for noise identification in imbalanced data classification," *Journal of King Saud University-Computer and Information Sciences*, pp. S1319157821000161, 2021. https://doi.org/10.1016/j.jksuci.2021.01.014.

[16]  V. Nguyen, J. Cai and J. Chu, "Hybrid CNN-GRU model for high efficient handwritten digit recognition," in *Proc. of the 2nd Int. Conf. on Artificial Intelligence and Pattern Recognition - AIPR '19*, Beijing, China, pp. 66–71, 2019.

[17]  H. Chen, Y. Zhu and K. Hu, "Adaptive Bacterial Foraging Optimization," *Abstract and Applied Analysis*, vol. 2011, pp. 1–27, 2011.

[18]  https://www.kaggle.com/uciml/pima-indians-diabetes-database, 2017.

[19]  A. Balint and A. Hajdu, 2014. https://archive.ics.uci.edu/ml/datasets/Diabetic+Retinopathy+Debrecen+Data+Set.

[20]  Oliver Roesler, 2013. https://archive.ics.uci.edu/ml/datasets/EEG+Eye+State.

[21]  H. Chen, Q. Zhang, J. Luo, Y. Xu and X. Zhang, "An enhanced bacterial foraging optimization and its application for training kernel extreme learning machine," *Applied Soft Computing*, vol. 86, pp. 105884, 2020.

[22]  F. Lv and M. Han, "Hyperspectral image classification based on improved rotation forest algorithm," *Sensors*, vol. 18, no. 11, pp. 3601, 2018.

[23]  T. R. Gadekallu, N. Khare, S. Bhattacharya, S. Singh, P. K. R. Maddikunta *et al.,* "Early detection of diabetic retinopathy using PCA-firefly based deep learning model," *Electronics*, vol. 9, no. 2, pp. 274, 2020.

[24]  M. Elnahas, M. Hussein and A. Keshk, "Artificial neural network as ensemble technique fuser for improving classification accuracy," in *2019 Ninth Int. Conf. on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, pp. 174–179, 2019.

[25] A. Herliana, T. Arifin, S. Susanti and A. B. Hikmah, "Feature selection of diabetic retinopathy disease using particle swarm optimization and neural network," in *2018 6th Int. Conf. on Cyber and IT Service Management (CITSM)*, Parapat, Indonesia, pp. 1–4, 2018.

[26] G. Kumar, S. M. Dr. and A. N. Dr., "An ensemble of feature subset selection with deep belief network based secure intrusion detection in big data environment," *Indian Journal of Computer Science and Engineering*, vol. 12, no. 2, pp. 409–420, 2021.

[27] S. K. Satapathy, S. Dehuri and A. K. Jagadev, "EEG signal classification using PSO trained RBF neural network for epilepsy identification," *Informatics in Medicine Unlocked*, vol. 6, pp. 1–11, 2017.

[28] C. M. Yilmaz, B. H. Yilmaz and C. Kose, "Prediction of eye states using k-NN algorithm: A comparison study for different distance metrics and number of neighbour parameters," in *2019 Medical Technologies Congress (TIPTEKNO)*, Izmir, Turkey, pp. 1–4, 2019.

[29] T. Wang, S. -U. Guan, K. L. Man and T. O. Ting, "EEG eye state identification using incremental attribute learning with time-series classification," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–9, 2014.