Optimization Driven Adam-Cuckoo Search-Based Deep Belief Network Classifier for Data Classification

MOHAMMED MOHSIN[1,2], **HONG LI**[1], **AND HEMN BARZAN ABDALLA**[3]

[1]Department of Mathematics and Statistics, Huazhong University of Science and Technology, Wuhan 430074, China
[2]Department of Statistics, University of Basra, Basra 61014, Iraq
[3]Department of Computer Science, Wenzhou-Kean University, Wenzhou 325060, China

Corresponding author: Mohammed Mohsin (mohammed_78@hust.edu.cn)

ABSTRACT Data classification effectively classifies the data based on the labeled class distribution. To classify the data using the imbalanced distribution poses a significant challenge in the class inequity problem. Various data classification methods are developed in the learning framework, but proving better classification accuracy is a significant challenge in the application domain. Therefore, an effective classification method named Adam-Cuckoo search based Deep Belief Network (Adam-CS based DBN) is proposed to perform the classification process. At first, the input data is forwarded to the pre-processing stage, and then the feature selection stage. The wrapper-based feature selection model conducts the search in space with the possible parameters. The operators specify the connectivity between the states and select the features based on their state. The classification is performed using the Deep Belief Network (DBN) classifier such that the multilayer perceptron (MLP) layer of Deep Belief Network (DBN) is trained using the proposed Adam based Cuckoo search (Adam-CS) algorithm. The breeding behavior of cuckoos is integrated with the step size parameter to enhance the accuracy of the classification process. The adaptive learning rate parameter effectively estimates the moments using a sparse gradient. The proposed Adam based Cuckoo search (Adam-CS) algorithm attained better performance using the metrics, such as accuracy, specificity, and sensitivity, with 90% training data.

INDEX TERMS Cuckoo search algorithm, Adam optimization algorithm, deep belief network (DBN), deep learning approach, data classification.

I. INTRODUCTION

Classification is a significant process in the pattern recognition framework. Various learning models, like backpropagation neural network, support vector machines, Bayesian network, decision tree, newly formed associative classification, and nearest neighbor are introduced in the classification scheme, which are effectively applied in different application domains [1]. In general, classification is a complex issue in the data analysis framework, because it is required to access a huge amount of data to train the classifier. The classifier is modeled using the training data, and further, the training data is utilized to perform the classification process in the future data. Mostly, the classifier uses the structure format, which will be further used to classify the future data but

The associate editor coordinating the review of this manuscript and approving it for publication was Liangxiu Han.

not the noise that exists in the training information [2]. Moreover, the class distribution with the imbalanced data set encountered a complex issue in most of the classifier learning techniques [1]. Classifying the data using the imbalanced distribution of the class poses a significant challenge in the performance obtained by the classifier algorithms, which ensure the balance between the misclassification costs and the class distribution. Moreover, the significant complexity and the frequent occurrence of the class inequity problem signify the requirement of extra efforts in the research field. Classification is a significant process in the data mining and knowledge discovery in databases (KDD) scheme. The classification model is used to learn the function from the training data that makes few errors when applied to the previously unseen data. Moreover, the report generated from both the industry and the academia specifies that the imbalanced distribution of the class data set poses a major issue

in most of the learning models in the classifier approaches, which further used to generate a moderately imbalanced distribution [3].

In the state of art machine learning approach, the deep learning technique is recently gained major attention in the data classification. The traditional training approach has the drawbacks of the need for maximum iteration and local minima [36]. The major reason behind the deep learning model is that it automatically generates valuable features in the problem domain by eliminating the handcrafted and the complicated engineering process. By designing the deep learning infrastructure as special, the Convolutional neural network (CNNs) is widely applicable in image segmentation and image recognition, as it focuses on the temporal and the spatial correlation between pixels [4]. The deep learning approach (DL) is a field of the machine learning algorithm, which functions based on the learning representations of multiple levels to make the features hierarchy as effective. The higher levels in the class distribution are obtained from the lower level, whereas the features available at the same level helps to define various higher level class features [5]. The structure of DL is used to extend the CNN by including additional hidden layers in the network infrastructure between the output and the input layers for modeling highly nonlinear and complex relationships [31]. Hence, the DL gained more interest among the researchers recently based on its performance to attain better solutions in various medical application-oriented problems, 3D Sensed Data Classification [38], and image analysis, like image classification, segmentation, denoising, and registration [6]. The DL method consumes less time than traditional methods [37]. Recently, the DT techniques are triggered by various learning models or traditional representations, respectively. DL uses the information data set and assimilates the complex behavior for selecting the effective characteristic features automatically with the profound layers using the neural network structure [7].

The DL method attained significantly better performance recently in the classification of Hyperspectral images (HSI). In [8], the DL technique, like DBN is explored in the classification of HIS for the first time. The spatial correlation is incorporated into the DBN to attain better classification. In [11], the DBN performance is enhanced by promoting diversity prior to the latent factors. CNN is the most significant deep learning model, as most of the reports based on CNN is generated in recent years [12]. The spectral based joint features are used in the DL model to minimize computational complexity [13]. Due to the availability of various hidden layers in the non-linear functions, the DL is more effective in different tasks, like machine translation, image classification, network optimization [15], and automatic speech recognition [14], which efficiently enables the higher-level data in the logistic functions [16]. In DL, the Long Short Term Memory (LSTM) is used to improve the classification results [17].

A. LITERATURE SURVEY

Various existing classification methods are surveyed: Hassan and Mahmood [20] developed a joint CNN and RNN model for extracting the higher-level features. Due to the pooling and the Convolutional layer, the long term locality dependencies were effectively captured. It attained better performance in the sentiment benchmark analysis framework. Moreover, it failed to use the model in the machine translation and information retrieval process. Perera and Patel [21] introduced a deep learning approach to learn the feature from one-class classification. It operates under CNN and generates the descriptive features for each class in the feature space. It attained better performance in the state of art mechanism but failed to generate precise values. Zhong *et al.* [22] developed a deep learning approach for the remotely sensed data. It captures the temporal variations and focuses on seasonal patterns. It used the feature extractor for extracting the features based on the time-series data. It offered an efficient and effective representation of data in the classification tasks. It does not provide a provision to integrate the patterns with human knowledge. Wang *et al.* [23] introduced a deep CNN based approach to extract the hidden patterns automatically from the smart meter and massive data. It extracts the features from the load files and effectively identifies the demographic information. It enhances the accuracy of the learning model, but the granularity of the data might be affected. Zhang *et al.* [24] introduced an adaptive dropout deep computation model that was developed for the data features in the industrial IoT framework. It prevents overfitting in the distribution function by setting the dropout rate at each hidden layer. Some other optimal techniques were not considered in the initialization process. Hossain and Muhammad [25] modeled an urban environment classification framework for communication purposes. It utilized mobile edge technology and offered efficient transmission with low latency. It effectively classified the noise and was robust against sound classification. It failed to use the noise cancellation algorithm. Zhai *et al.* [26] developed an ensemble learning model to train the hidden layer in the feed-forward network. It enhanced prediction stability but failed to consider the relationship between testing and dropout probability. Chambon *et al.* [27] introduced a deep learning approach to learn the end to end computing spectrograms. It learns the spatial filters for exploiting the array of sensors in the softmax classifier. It attained better classification performance with minimum computational cost but the drawback of this method was the limitation of the spatial context.

The main intention of this work is to design and develop an effective data classification method using the proposed Adam-CS algorithm. The proposed algorithm involves three stages which are depicted as follows: Pre-processing, feature selection, and classification. Initially, the input data is subjected to the pre-processing stage, where the missing value imputation and log transformation are performed

to remove the noise and artifacts present in the data. The resulted pre-processed data is forwarded to the feature selection stage, where the essential features are effectively selected using the wrapper-based feature selection method. Finally, the data classification is carried out with the selected features using the DBN classifier, which is trained by the proposed Adam-CS algorithm.

The major contribution of this work is explained as follows:

●The missing value imputation and log transformation are performed in the pre-processing stage to remove the noise and artifacts present in the data. The features are selected by the wrapper-based model, which finds the state has the maximum evaluation such that it uses accuracy estimation for both the evaluation function and heuristic function.

●The selected features provide an intuitive and effective abstraction, which facilitates the data classification. Moreover, the data classification is performed using the DBN classifier, which is trained by the proposed Adam-CS algorithm. The breeding behavior of cuckoos is integrated with the step size parameter to enhance the accuracy of the classification process. The adaptive learning rate parameter effectively estimates the moments using a sparse gradient.

The rest of this paper is organized as follows: Section 2 elaborates on the proposed algorithm, and section 3 discussed the results and discussion along with the performance evaluation. Finally, section 4 concludes the paper.

II. PROPOSED ALGORITHM DATA CLASSIFICATION
This section describes the overall details about the proposed Adam-CS based DBN.

A. PROBLEM FORMALIZATION
Data classification is a method of organizing the data into various categories, which is useful in different filed for future usage. Some of the challenges in the data classifications are:

Due to the increasing population in urban areas, it is very difficult for the service provider for managing the big data coming from various devices and users, which leads to degrade performance [25]. The training data may overfit in the deep learning model, as there exists a small dataset in the medical image classification. Hence it required more processing functions to acquire image data and annotation, which is a major challenge in the medical image analysis [18]. CNN is the significant architecture used to perform the critical classification operations using the convolution filter. Training the CNN using the large labeled data set is a complex and time-consuming issue in the classification model [6]. The DL model performs the classification through the Hyperspectral data. The feature representation is dependent on the training samples in the deep approach, which is a major challenge associated with the classification framework. Moreover, the manual annotation of the Hyperspectral data is significantly complex [19]. The extreme learning machine model is quite simple and efficient to train the hidden layer, but selecting the architectural features and the prediction instability was a major problem in the feed forward

network [23]. The ability to quickly generate large sets of data by validating and evaluating the data poses a major challenge in data classification.

These are the drawbacks found in the existing methods for data classification. The proposed Adam-CS based DBN is used to overcome those drawbacks of the existing methods.

B. ARCHITECTURE OF THE PROPOSED SYSTEM
The proposed approach involves three stages, which are depicted as follows: Pre-processing, feature selection, and classification. Initially, the input data is passed into the pre-processing phase, where the missing value imputation and log transformation is performed to remove the noise and artifacts present in the data. The resulted pre-processed data is subjected to the feature selection phase. Moreover, the feature selection is carried out using the wrapper selection method. The features selected from the data provide an intuitive and effective abstraction, which is used to facilitate the data classification. Finally, the selected features are subjected to the classification phase. In the classification stage, the data classification is carried out using the DBN classifier, such that training of the DBN classifier is performed using the proposed Adam-CS optimization algorithm. The proposed algorithm is the integration of the Adam optimization algorithm [28] and the Cuckoo search (CS) algorithm [29]. Figure 1 shows the architecture of the proposed Adam-CS based DBN for data classification.

1) PRE-PROCESSING
At first, the input data k with the dimension $[X \times Y]$ is fed to the pre-processing stage, where the log transformation and the missing data imputation are effectively carried out to remove the noise and artifacts present in the data. The input data k is represented as,

$$k = k_{ij}; \quad 1 \le i \le X; \ 1 \le j \le Y \qquad (1)$$

where, k_{ij} denotes i^{th} data at j^{th} attribute, X is the total number of data, and Y denotes the total number of attributes. Missing data imputation is a statistical model, which replaces the missing data points with the substituted values. The log transformation process is used to minimize the variability of data by transforming the skewed data to approximately conform to normality. The log transformation process is denoted as,

$$L = \log_{10} [k] \qquad (2)$$

where L denotes the log transformation process, and K is the pre-processed data. The resulted pre-processed data is denoted as L with the dimension of $X \times Y$.

2) WRAPPER-BASED FEATURE SELECTION MODEL
The pre-processed data L is subjected to the feature section stage, where the essential features X is effectively selected using the wrapper-based feature selection model. The wrapper model performs the search in the space using the possible parameters [30]. The search in the space requires a search
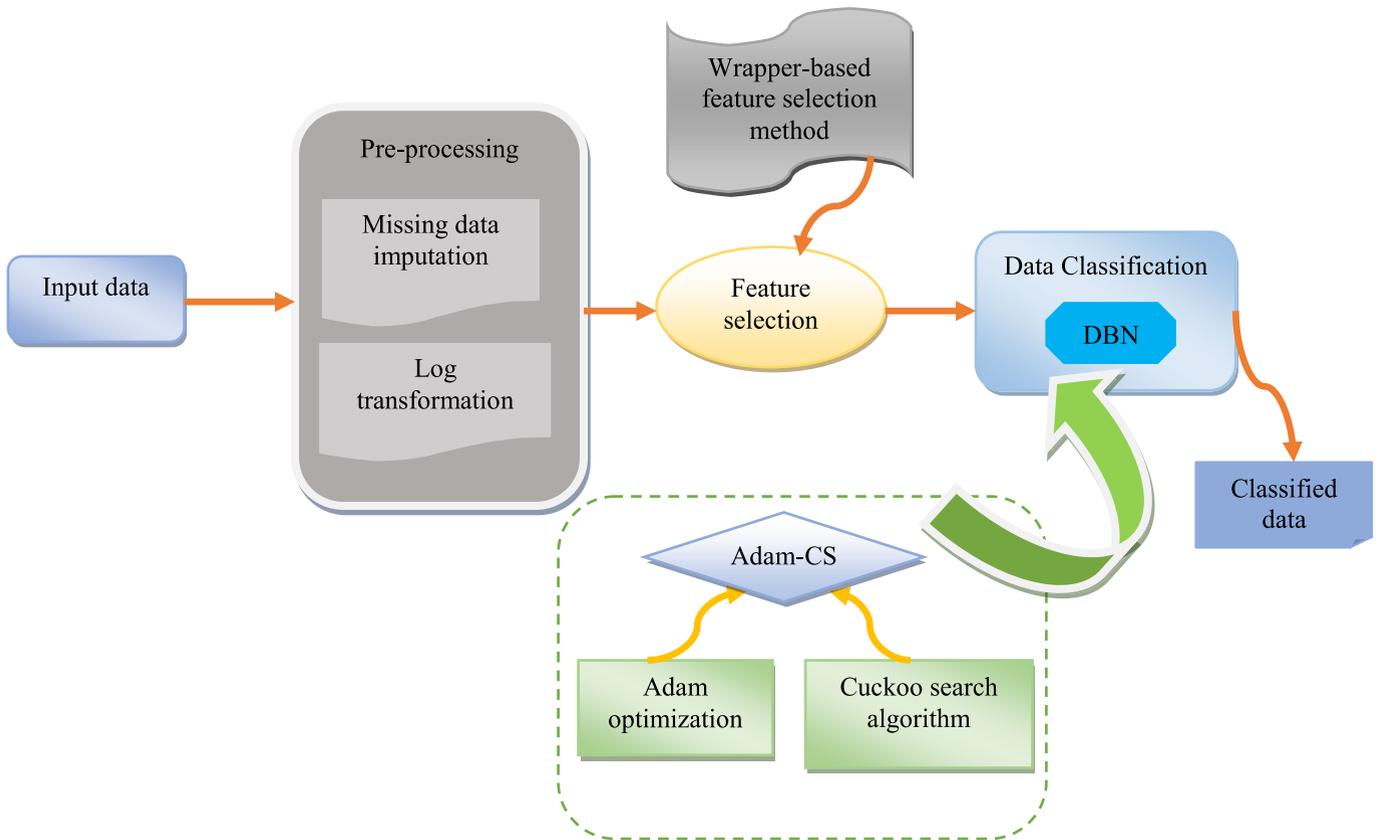
FIGURE 1. Block diagram of the proposed Adam-CS based Deep Belief Network for data classification.

engine, a termination condition, an initial state, and state-space, respectively. Each state in the search space denotes a feature subset. For a number of features there exists a bits in the state space, and each bit determines '1' for the presence of features and '0' for the absence of features. The operators specify the connections between the states and the operators are selected to use for adding or deleting the features from the state space. The heuristic function is used to find the state space with maximum evaluation such that the accuracy estimation is used in both the evaluation function and the heuristic function.

The accuracy estimation effectively executes the cross-validation in the small dataset as the small dataset uses less time to estimate and learn the function. The wrapper approach uses the feed forward and the backward elimination model. In the forward model, the searching process begins with the empty feature set, while the backward approach starts the search using a full feature set. Most of the state uses the empty feature set as it is faster in building the classifier. The complexity penalty is included in the evaluation function to penalize the subset of features. The penalty value is set as 0.1%, which is small when compared to the accuracy estimation. Moreover, the feature subset is randomly selected based on the estimated accuracy. In the feature selection process, the dimension of the pre-processed data may change

based on the usage of the dataset. The features selected using the wrapper approach is denoted as, Z has the dimension of $[303 \times 13]$ using heart disease dataset, the dimension of $[286 \times 9]$ using breast cancer dataset, the dimension of $[32 \times 56]$ using lung cancer dataset, and the dimension of $[768 \times 8]$ using diabetes dataset. The resulted pre-processed data Z is further send to the feature selection module.

3) PROPOSED ADAM-CUCKOO SEARCH-BASED DEEP BELIEF NEURAL NETWORK

DBN classifier is used to perform the data classification using the multiple layers of hidden units. The RBM layers use unsupervised learning, whereas the MLP layer undergoes the supervised learning based on the new optimization algorithm. The optimization is based on the characteristics of adam optimization and cuckoo search algorithm. The breeding behavior of cuckoo species is effectively used to compute the weight and bias for training the classifier. In order to compute the weight and bias optimally, the Adam features are used with the cuckoo species to attain better performance in performing the data classification.

a: ARCHITECTURE OF DEEP BELIEF NEURAL NETWORK
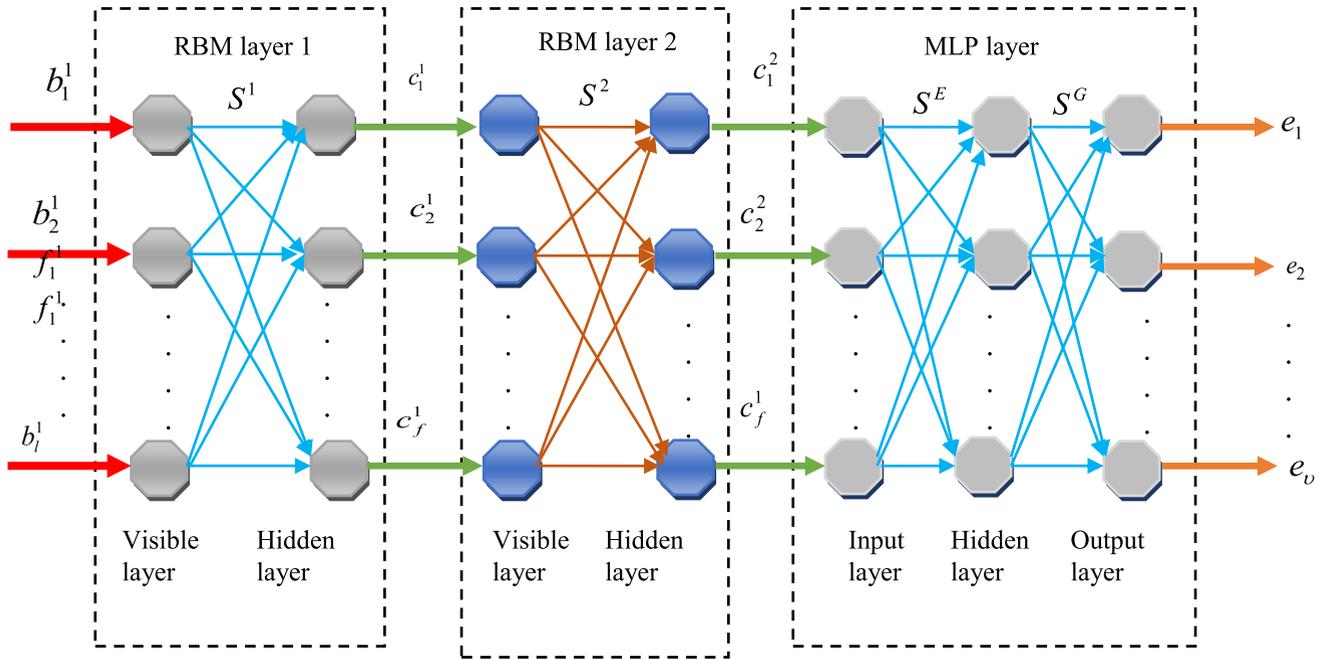The proposed DBN classifier is developed using two restricted Boltzmann machine (RBM) layers and one

FIGURE 2. Architecture of Deep Belief Network classifier.

MLP layer. In DBN, there are no connections among the hidden neurons as well as among the visible neurons. The feature vector F is passed as the input to the visible layer of RBM layer 1. The resultant output from hidden layers of RBM layer 1 forms the input for RBM layer 2, and the output of RBM layer 2 is the input to the MLP layer. Figure 2 shows the architecture of the DBN classifier.

The feature vector F is passed as input to the visible layer of RBM layer 1 such that the visible and the hidden layer of RBM layer 1 is expressed as:

$$b^1 = \left\{ b_1^1, b_2^1, K, b_m^1, K, b_l^1 \right\}; \quad 1 \leq m \leq l; \quad (3)$$

$$c^1 = \left\{ c_1^1, c_2^1, K, c_n^1, K, c_f^1 \right\}; \quad 1 \leq n \leq f \quad (4)$$

where, f denotes the hidden neurons, b_m^1 indicates the mth visible neuron at RBM layer 1, and c_n^1 denotes the nth hidden neuron and each neuron presented in the hidden and the visible layer has a bias.

Let x and y represent the biases of the visible and hidden layers. Therefore, the two biases that correspond to the neurons at both the layers are represented as,

$$x^1 = \left\{ x_1^1, x_2^1, K, x_m^1, K, x_l^1 \right\} \quad (5)$$

$$y^1 = \left\{ y_1^1, y_2^1, K, y_n^1, K, y_f^1 \right\} \quad (6)$$

where, y_n^1 indicates the bias that corresponds to n^{th} hidden neuron, and x_m^1 denotes the bias that corresponds to m^{th} visible neuron. The weight applied to the RBM layer 1 is expressed as,

$$S^1 = \left\{ S_{mn}^1 \right\}; \quad 1 \leq m \leq l; \ 1 \leq n \leq f \quad (7)$$

where, S_{mn}^1 denotes the weight between the mth visible and the nth hidden with the weight vector dimension as, $l \times f$. Hence, the output of the hidden layer in RBM layer 1 is calculated using the weight and bias associated with each neuron, which is expressed as,

$$c_n^1 = \lambda \left[y_n^1 + \sum_m b_m^1 S_{mn}^1 \right] \quad (8)$$

where, λ denotes the activation function. Moreover, the output generated from RBM layer 1 is expressed as,

$$c^1 = \left\{ c_n^1 \right\}; \quad 1 \leq n \leq f \quad (9)$$

Moreover, the learning process of RBM layer 2 is processed using the output obtained from the hidden layer of RBM layer 1. The output of RBM layer 1 is specified in Eq. (9), which is fed as the input to the visible layer of RBM layer 2. Hence, the number of visible neurons in RBM layer 2 is equivalent to the number of hidden neurons of RBM layer 1, which is represented as,

$$b^2 = \left\{ b_1^2, b_2^2, \ldots, b_f^2 \right\} = \left\{ c_n^1 \right\}; \quad 1 \leq n \leq f \quad (10)$$

where, $\left\{ c_n^1 \right\}$ denotes the output vector of RBM layer 1. Thereby, the hidden layer at RBM layer 2 is represented as,

$$c^2 = \left\{ c_1^2, c_2^2, \ldots, c_n^2, \ldots, c_f^2 \right\}; \quad 1 \leq n \leq f \quad (11)$$

The bias of the hidden and the visible layer has a similar representation as specified in Eq. (5) and (6), but they are represented as x^2 and y^2. The weight vector applied to the RBM layer 2 is denoted as,

$$S^2 = \left\{ S_{nn}^2 \right\}; \quad 1 \leq n \leq f \quad (12)$$

where, S_{nn}^2 denotes the weight between the nth visible and n^{th} hidden neuron at RBM layer 2 with the weight vector dimension as $f \times f$. Hence, the output of nth hidden neuron is computed as,

$$c_n^2 = \lambda \left[y_n^2 + \sum_m b_m^2 S_{nn}^2 \right], \quad \forall b_m^2 = c_n^1 \quad (13)$$

where, y_n^2 denotes the bias linked with n^{th} hidden neurons. Hence, the output generated by the hidden layer is expressed as,

$$c^2 = \left\{ c_n^2 \right\}; \quad 1 \le n \le f \quad (14)$$

The output of RBM layer 2 is given as the input of the MLP layer such that the neurons present in the input layer are denoted as, f. The input of the MLP layer is expressed as,

$$p = \{ p_1, p_2, K, p_n, K, p_f \} = \left\{ c_n^2 \right\}; \quad 1 \le n \le f \quad (15)$$

where, q denotes the neurons of the input layer which is passed by the output of the hidden layer at RBM layer 2 $\left\{ c_n^2 \right\}$. Therefore, the MLP hidden layer is expressed as,

$$d = \{ d_1, d_2, K, d_A, K, d_C \}; \quad 1 \le A \le C \quad (16)$$

where, C denotes the total number of hidden neurons. Let us consider, D_A the bias of Ath hidden neuron with the values of $A = 1, 2, K, C$. The output of MLP layer is expressed as,

$$e = \{ e_1, e_2, K, e_\omega, K, e_\upsilon \}; \quad 1 \le \omega \le \upsilon \quad (17)$$

where, υ denotes the number of neurons present at the output layer. MLP contains two weight vectors, where one weight is applied between input and hidden layer while the other is applied between the hidden and the output layer, respectively. S^E is the weight vector used between the input and hidden layers, which is expressed as,

$$S^E = \left\{ S_{nA}^E \right\}; \quad 1 \le n \le f; \ 1 \le A \le C \quad (18)$$

where, S_{nA}^E denotes the weight between n^{th} input neuron and A^{th} hidden neuron with the size of S^E is $[f \times C]$. Based on the weight and bias of the neurons, the output of the hidden layer is represented as,

$$d_A = \left[\sum_{n=1}^f S_{nA}^E * p_n \right] D_A \quad \forall p_n = c_n^2 \quad (19)$$

where, D_A denotes the bias of the hidden neuron such that $p_n = c_n^2$, as the output of RBM layer 2 is the input of MLP. Moreover, the weight between hidden and the output layer is indicated as S^G and is represented as,

$$S^G = \left\{ S_{A\omega}^G \right\}; 1 \le A \le C; \quad 1 \le \omega \le \upsilon \quad (20)$$

Based on the output of the hidden layer and weight S^G, the output vector is computed as,

$$e_\omega = \sum_{A=1}^C S_{A\omega}^G * d_A \quad (21)$$

where, $S_{A\omega}^G$ denotes the weight between A^{th} hidden and ω^{th} output neuron, and d_A denoted as the hidden layer output.

Training of RBM Layer 1: The training sample F is subjected to the input to RBM layer 1. It calculates the probability distribution for the hidden neurons and determines the positive gradient using the visible vector. Moreover, the probability of each visible neuron is identified, and the negative gradient function is calculated.

Training of RBM Layer 2: The output attained from the hidden layer of RBM layer 1 is passed as the input to the visible layer of RBM layer 2. It computes the probability distribution, and calculates the energy based on the weight and selects the weight having the lowest energy.

Training of MLP Layer: The MLP layer of the DBN classifier is trained using the proposed Adam-CS optimization algorithm. The training data obtained from the RBM layer 2 is passed as the input to the MLP layer. By analyzing the data, the network is modified iteratively to select the best weight. Moreover, the proposed Adam-CS is adopted to calculate the weight based on the error function. Adam-CS is developed by integrating the Adam optimization algorithm [28] with the Cuckoo search (CS) algorithm [29]. Adam optimization is a stochastic gradient descent algorithm used to efficiently compute the gradients in the data classification model. The cuckoo search is a metaheuristic algorithm, which aimed to perform the global search by imitating the best feature in nature.

b: PROPOSED ADAM-CS ALGORITHM FOR TRAINING THE DBN

The proposed algorithm is effectively used to train the MLP layer of the DBN classifier to enhance the performance of data classification. Adam-CS is the integration of the Adam algorithm and CS algorithm. The breeding behavior of cuckoos is integrated with the step size parameter to enhance the accuracy of the classification process. The adaptive learning rate parameter effectively estimates the moments using a sparse gradient. Cuckoos are the fascinating birds because of their effective reproduction, and beautiful sound-making strategy. Certain species, like Guin and ani, place the eggs in the common nest by removing other species eggs in the nest in order to enhance the hatching probability of the eggs. Moreover, various species lay their eggs in other species nest, which violates the brood parasitism. Three different kinds of brood parasitism are: nest takeover, cooperative breeding, and intraspecific brood parasitism. Some of the host birds have a direct interaction between other intruding cuckoos. When a host bird discovers an egg, and it came to know that it was not their own egg then, either it abandons the nest or removes the alien eggs away and designs a new nest somewhere. In the Tapera cuckoo species, the female parasitic is specialized in a patter of laying eggs, and colour, which minimizes the probability of eggs to be abandoned and maximize the reproductivity rate.

The proposed Adam-CS algorithm incorporates the metaheuristic algorithms with the stochastic gradient descent

algorithm to enhance the performance for providing better classification accuracy. The breeding behavior of cuckoos is integrated with the step size parameter to enhance the accuracy of the classification process. The adaptive learning rate parameter effectively estimates the moments using a sparse gradient. The data classification is effective even using a single optimization algorithm, but to enhance the accuracy of the classification performance, the breeding parameter of the CS algorithm is integrated with the gradient function. Hence, the proposed Adam-CS algorithm effectively performs the data classification process and increases the accuracy rate of the classification.

Levy Flight: Furthermore, the time to lay the egg by some species is also fabulous. Parasitic cuckoos select the nest, where the other species bird laid their eggs. Moreover, the cuckoo species eggs will hatch earlier than the host eggs. When the first cuckoo species chick is hatched at once the cuckoo throws the host eggs away from the nest by slightly moving the eggs out of the nest, which maximizes the food sharing among the cuckoos chick. In general, the animals search their food in a quasi or random manner in nature. Moreover, the levy behavior of the animals is determined using the characteristic features of levy flight.

Cuckoo Search: The cuckoo search model uses the following rules for effectively describing the proposed algorithm.

- Each and every cuckoo lay a single egg and place the egg with the randomly selected nest.
- The nest has a better solution (quality of eggs) will be taking over to the subsequent generations.
- The host species either abandon the nest or throw away the alien eggs from the nest and build a new nest at some other location.

The algorithmic steps involved in the proposed algorithm are discussed as follows:

Population Initialization: The objective function of the cuckoo species is defined as, $F(P); P = (P_1, \ldots \ldots P_v)^M$, and the population initialization of the k host nest is specified as, $P_z(z = 1, 2, \ldots, k)$, respectively. The new solution for the cuckoo species is computed using the below equation as,

$$P_z^{u+1} = P_z^u + \beta \oplus Levy(\gamma) \qquad (22)$$

where, P^{u+1} denotes the new solution, z represents the cuckoo, $\oplus$ denotes the entry-wise multiplications, $Levy(\gamma)$ denotes the levy flight, and β indicates the step-size, which has the values of $\beta > 0$.

Fitness Evaluation: The fitness is evaluated by considering the cuckoo egg when the cuckoo's egg is similar to the host egg, then egg of cuckoo species is likely to be discovered. The solution with the minimum error value is taken as the best fitness solution.

$$R_z = \frac{1}{g} \sum_{w=1}^{g} (e_w - q_t) \qquad (23)$$

where, e_w is the output of the classifier, and q_t is the estimated output.

Update the Nest: The update equation of the Adam optimization algorithm by considering the step-size β is expressed as,

$$\varphi_u = \varphi_{u-1} - \beta . \frac{\hat{f}_u}{\left(\sqrt{\hat{s}_u} + K\right)} \qquad (24)$$

where, $\hat{f}_u$ denotes the average moving gradient, $\hat{s}_u$ denotes the squared gradient, and β denotes the step size, respectively. Let us assume, $u = u + 1$, and $u - 1 = u$. By substituting the value for u and $u - 1$ in the above equation results,

$$\frac{\beta . \hat{f}_{u+1}}{\sqrt{\hat{s}_{u+1}} + K} = \phi_u - \phi_{u+1} \qquad (25)$$

$$\beta = \frac{(\phi_u - \phi_{u+1})\left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}} \qquad (26)$$

By substituting the term β of Eq. (26) in Eq. (22), by assuming $\varphi = P$, the resulted equation is expressed as,

$$P_z^{u+1}$$
$$= P_z^u + \frac{(P_z^u - P_z^{u+1})\left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}} \oplus Levy(\gamma) \qquad (27)$$

$$P_z^{u+1}$$
$$= P_z^u + \frac{P_z^u\left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}} - \frac{P_z^{u+1}\left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}$$
$$\oplus Levy(\gamma) \qquad (28)$$

$$P_z^{u+1} + \frac{P_z^{u+1}\left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}$$
$$= P_z^u \left(\frac{\hat{f}_{u+1} + \left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}\right) \oplus Levy(\gamma) \qquad (29)$$

$$P_z^{u+1}\left(\frac{\hat{f}_{u+1} + \left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}\right)$$
$$= P_z^u \left(\frac{\hat{f}_{u+1} + \left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}\right) \oplus Levy(\gamma) \qquad (30)$$

$$P_z^{u+1}$$
$$= \frac{\hat{f}_{u+1}}{\hat{f}_{u+1} + \left(\sqrt{\hat{s}_{u+1}} + K\right)}$$
$$\times \left\{ P_z^u \left(\frac{\hat{f}_{u+1} + \left(\sqrt{\hat{s}_{u+1}} + K\right)}{\hat{f}_{u+1}}\right) \oplus Levy(\gamma) \right\} \qquad (31)$$

Here, the term f_u and s_u is computed as,

$$f_{u+1} = \kappa_1 f_u + (1 - \kappa_1) h_u \qquad (32)$$
$$s_{u+1} = \kappa_2 . s_{u-1} + (1 - \kappa_2) h_u^2 \qquad (33)$$

where, $K = 10^{-8}$, and f_u denotes the moving average gradient, and s_u denotes the squared gradient, respectively. Hence,

TABLE 1. Algorithm 1. Pseudo code of the proposed algorithm.

Sl. No	Pseudo code of the proposed Adam-CS algorithm
1	Input: Z
2	Output: classified output
3	Begin
4	Objective function, $F(P); P = (P_1,......P_v)^M$
5	Population Initialization $P_z(z=1,2,...,k)$
6	While $(t < \max Gen)$
7	Randomly get a cuckoo Z, by using levy flight
8	Fitness evaluation using equ. (23)
9	Randomly Select the nest r among n
10	If$(R_z > R_r)$
11	Replace r as the new solution
12	end
13	Abandon the worst nest
14	Determine the best solution
15	Find the best solution by ranking each solution
16	End while

Eq. (30) is the updated equation of the proposed algorithm. Here, $\hat{f}_{u+1}$ denotes the average moving gradient, $\hat{s}_{u+1}$ denotes the squared gradient, and p_z^{u+1} specifies the new updated equation of cuckoo species.

Determine the Feasibility: The feasibility of the solution is determined to generate the best optimal solution. The fitness of the new solution is compared with the fitness of the previous solution, when the fitness of the new solution is higher than the old solution, then the old solution is replaced with the new solution as the optimal solution.

Termination: The above steps are repeated until the optimal solution is attained. Algorithm 1 shows the pseudo-code of the proposed Adam-CS algorithm for data classification.

III. RESULTS AND DISCUSSION
This section describes results and discussion obtained using the proposed Adam-CS algorithm along with the performance analysis. The performance of the classification model is evaluated and the analysis is carried out using the metrics, like accuracy, sensitivity, and specificity, respectively.

A. EXPERIMENTAL SETUP
The implementation of the proposed algorithm is carried out by the MATLAB tool using the Breast Cancer Data Set, Diabetes Data Set, Heart Disease Data Set, Lung Cancer Data

Set, and Cardiotocography Data Set. The breast cancer data set contains 201 instances of one class and 85 instances of other classes. Here, the instances are specified using nine attributes. The diabetes data set contains numerous files, and each file holds four fields. Heart disease data set consists of 76 attributes, where the goal field determines the presence of heart disease. The lung cancer dataset describes three different pathological lung cancers, where the attributes contain a class label. The Cardiotocography Data Set contains 2126 instances, 23 attributes, and 162219 number of web hits. This is a multivariate dataset with real attribute characteristics.

B. EVALUATION METRICS
The performance of the proposed algorithm is evaluated using the metrics, such as accuracy, sensitivity, and specificity.

Accuracy: It is defined as the ratio of true measure with the total number of observations, which is expressed as,

$$D = \frac{I+J}{total} \tag{34}$$

where, I denotes the true positive, J indicates the true negative, and D specifies the accuracy.

Sensitivity: It is defined as the ratio of true positive rate to the predicted observation rate, which is represented as,

$$H = \frac{I}{I+V} \tag{35}$$

where, V denotes the false negative, and H is the sensitivity.

Specificity: It is termed as the ratio of true negative measure with that of the unpredicted observations, which is expressed as,

$$L = \frac{J}{J+U} \tag{36}$$

where, U denotes the false positive rate, and L is the specificity.

C. COMPARATIVE METHODS
The performance of the proposed Adam-CS based DBN is analyzed and is compared with the existing methods, such as Deep Artificial Neural network (Deep ANN) [31], Convolutional Recurrent Neural Network (Convolutional RNN) [20], Convolutional Neural Network (CNN) [21], Firefly and Particle Swarm Optimization with Levenberg Marquardt Neural Network (FFPSO+LM NN) [39], Particle Swarm Optimization-Genetic Algorithm in Artificial Neural Network (PSO-GA ANN) [40], respectively.

D. COMPARATIVE ANALYSIS
This section describes the comparative analysis of the proposed data classification approach made using different datasets are elaborated as follows:
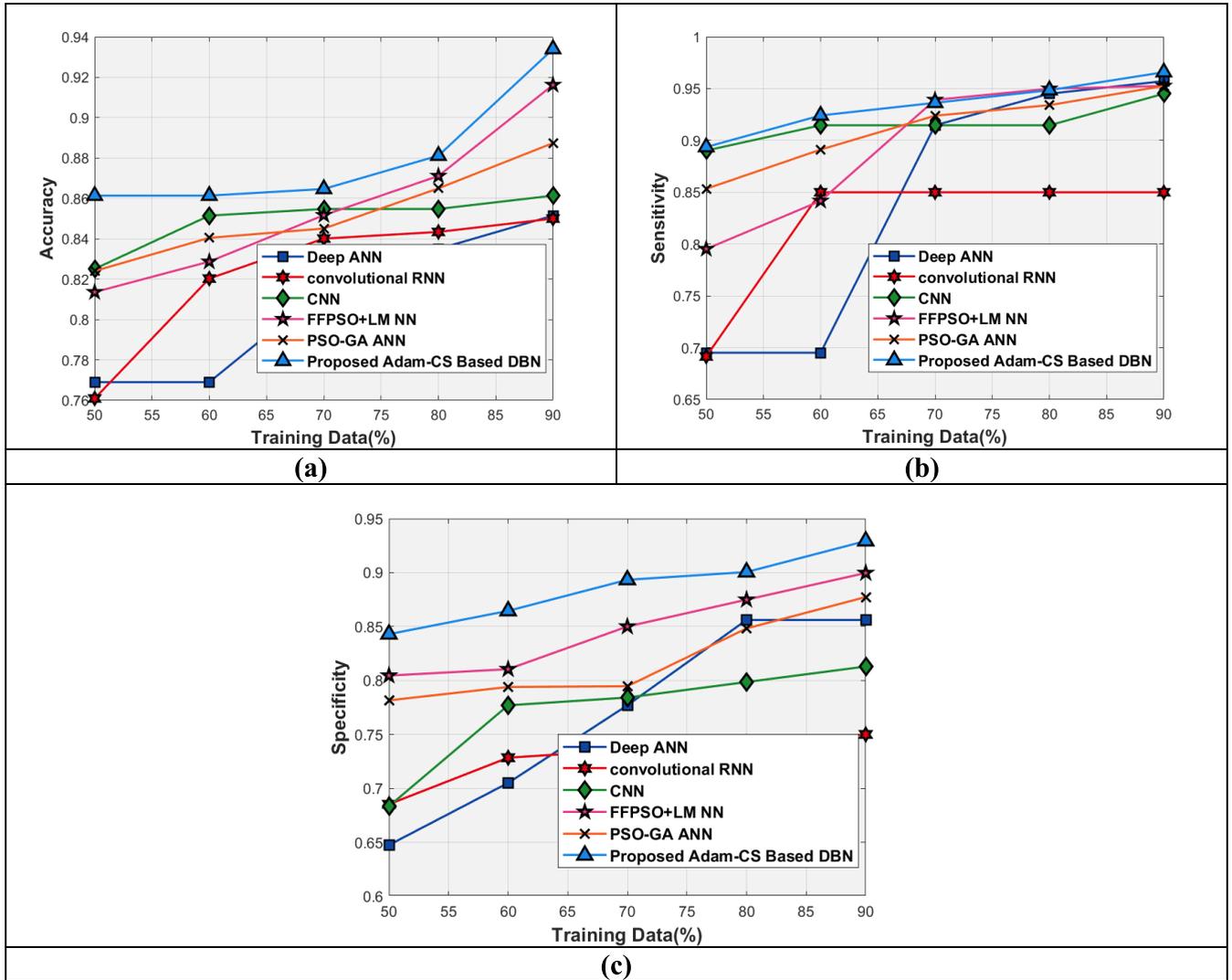
FIGURE 3. Comparative analysis using heart disease dataset, a) accuracy, b) sensitivity, c) specificity.

1) COMPARATIVE ANALYSIS USING HEART DISEASE DATASET

Figure 3 shows the comparative analysis of the data classification methods when the heart disease dataset is used. Figure 3 a) shows the analysis of the accuracy of the methods based on the training data. When training data = 50%, the accuracy obtained by the proposed Adam-CS based DBN is 0.8614. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 12%, 13%, 4%, 5.54%, and 4.33%, respectively. When training data = 70%, the accuracy obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.8152, 0.8401, 0.8548, 0.8517, and 0.8451, respectively while the proposed Adam-CS based DBN obtained an accuracy of 0.8647, respectively.

Figure 3 b) shows the analysis of sensitivity based on training data. When training data = 50%, the sensitivity obtained by the proposed Adam-CS based DBN is 0.8937. Hence, the percentages of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 26%, 29%, 0.4%, 11.02%, and 4.52%, respectively. When training data = 80%, the sensitivity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.9451, 0.8500, 0.9146, 0.95, and 0.9341, respectively, whereas the proposed Adam-CS based DBN obtained a sensitivity of 0.9485 respectively.

Figure 3 c) shows the analysis of specificity with respect to the training data. When training data = 50%, the specificity obtained by the proposed Adam-CS based DBN is 0.8429. Hence, the percentage of improvement reported by
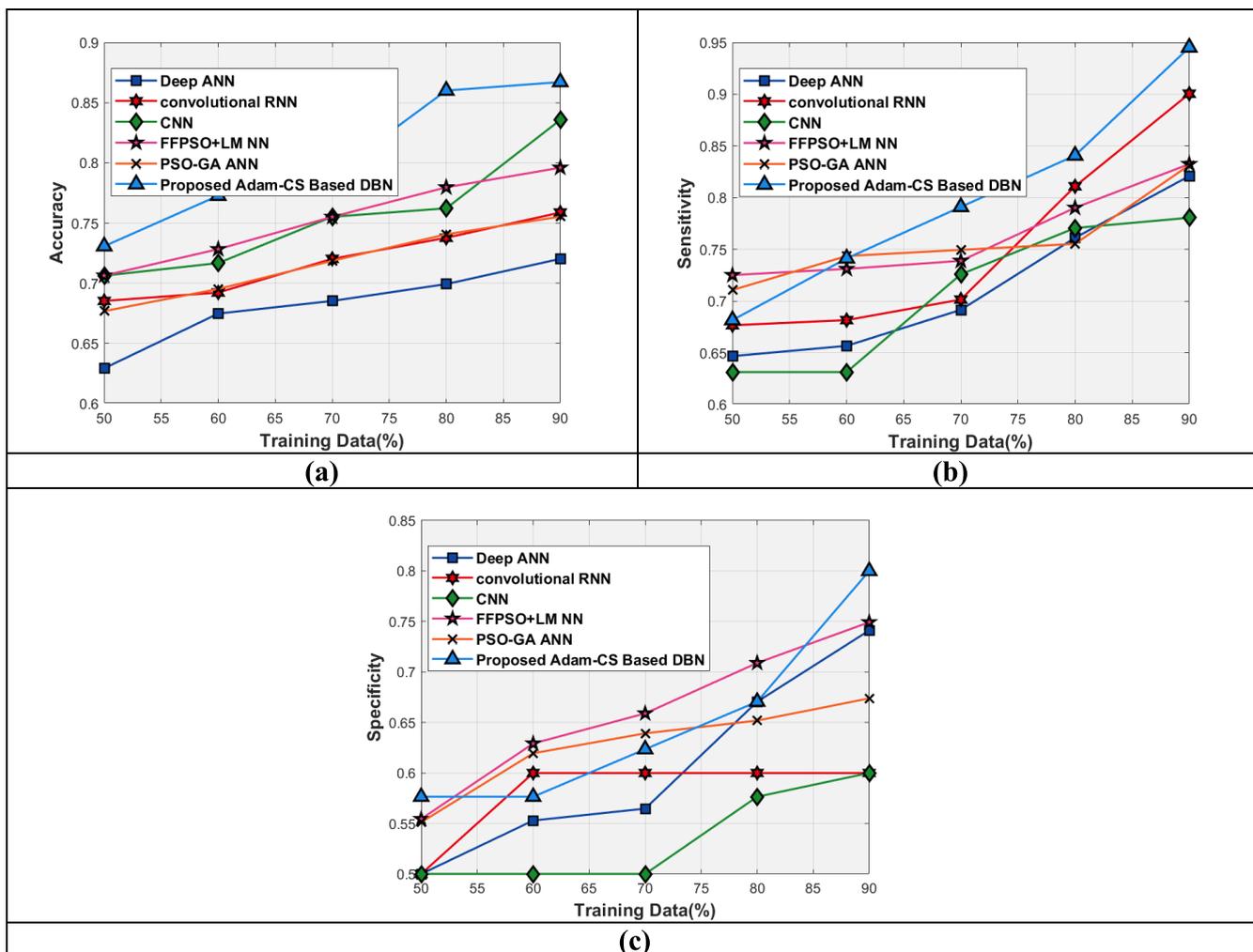
FIGURE 4. Comparative analysis using breast cancer dataset, a) accuracy, b) sensitivity, c) specificity.

the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 30%, 23%, 23%, 4.56%, and 7.27%, respectively. When training data = 60%, the specificity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.7050, 0.7284, 0.7770, 0.8105, and 0.7934, respectively, whereas the proposed Adam-CS based DBN obtained specificity of 0.8645.

2) COMPARATIVE ANALYSIS USING BREAST CANCER DATASET

Figure 4 shows the comparative analysis of the proposed algorithm using the breast cancer dataset. Figure 4 a) shows the analysis of accuracy based on the training data. When training data= 50%, the accuracy obtained by the proposed Adam-CS based DBN is 0.7308. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep

ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 16%, 6%, 3.5%, 3.34%, and 7.39%, respectively. When training data = 70%, the accuracy obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.6853, 0.7203, 0.7552, 0.7551, and 0.7186, respectively, whereas the proposed Adam-CS based DBN obtained an accuracy of 0.7902.

Figure 4 b) shows the analysis of sensitivity based on training data. When training data = 70%, the sensitivity obtained by the proposed Adam-CS based DBN is 0.791. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 12.57%, 11.31%, 8.23%, 6.59%, and 5.2% respectively. When training data = 80%, the sensitivity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.7612, 0.8109, 0.7706, 0.7902, and 0.7553, respectively, whereas
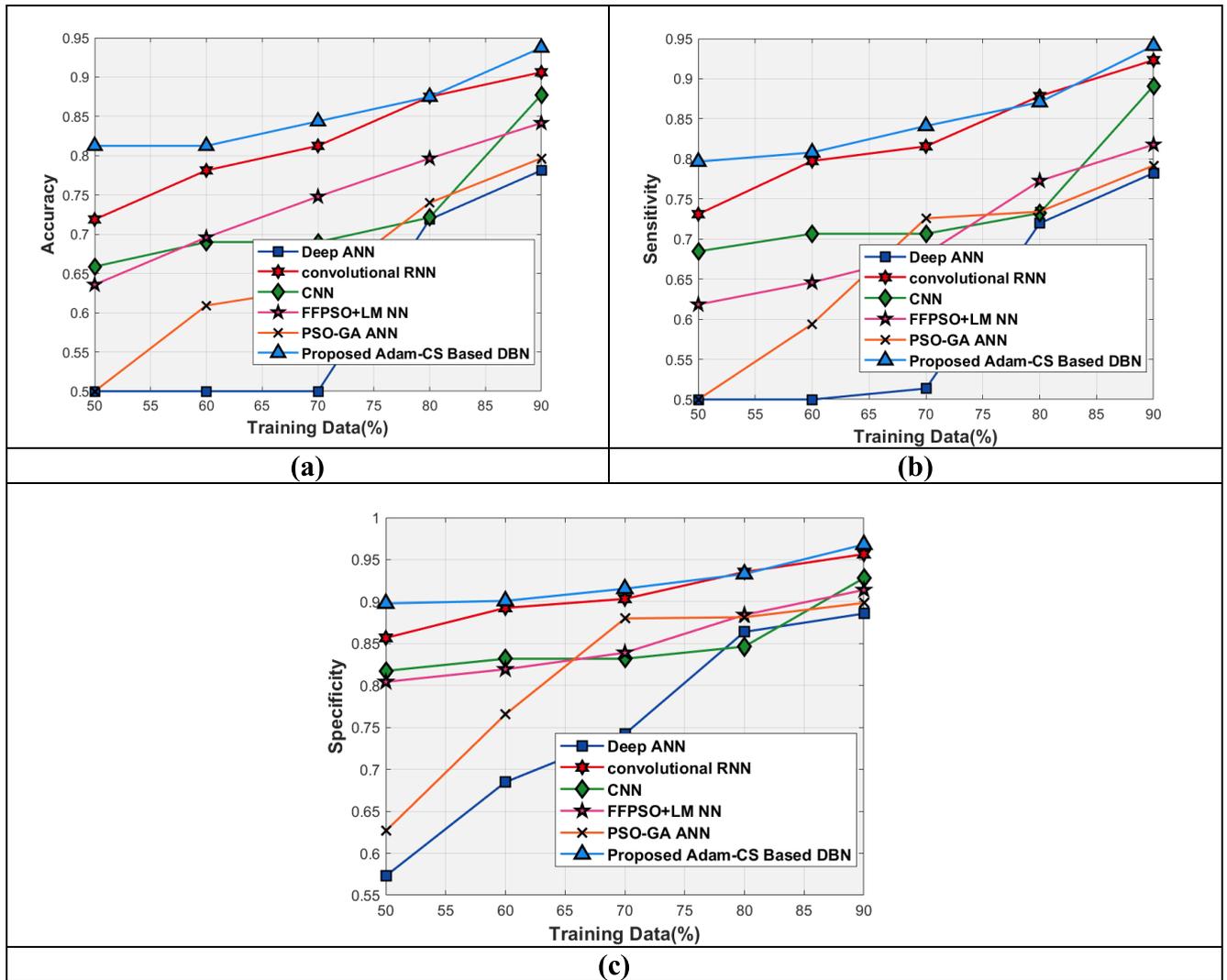
FIGURE 5. Comparative analysis using lung cancer dataset, a) accuracy, b) sensitivity, c) specificity.

the proposed Adam-CS based DBN obtained a sensitivity of 0.8408.

Figure 4 c) shows the analysis of specificity with respect to the training data. When training data = 50%, the specificity obtained by the proposed Adam-CS based DBN is 0.5764. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 15%, 15%, 15%, 3.82%, and 4.37%, respectively. When training data = 60%, the specificity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.5529, 0.60, 0.50, 0.6292, and 0.6195, respectively, whereas the proposed Adam-CS based DBN obtained specificity of 0.5765.

3) COMPARATIVE ANALYSIS USING LUNG CANCER DATASET
Figure 5 shows the comparative analysis of the proposed algorithm using a lung cancer dataset. Figure 5 a) shows

the analysis of accuracy based on the training data. When training data = 50%, the accuracy obtained by the proposed Adam-CS based DBN is 0.8125. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 63%, 13%, 23%, 21.74%, and 38.46%, respectively. When training data = 70%, the accuracy obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.5, 0.8125, 0.69, 0.7478, and 0.6317, respectively, whereas the proposed Adam-CS based DBN obtained an accuracy of 0.8437.

Figure 5 b) shows the analysis of sensitivity based on training data. When training data = 50%, the sensitivity obtained by the proposed Adam-CS based DBN is 0.7965. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 59%, 8%, 16%, 22.37%, and 37.24%,
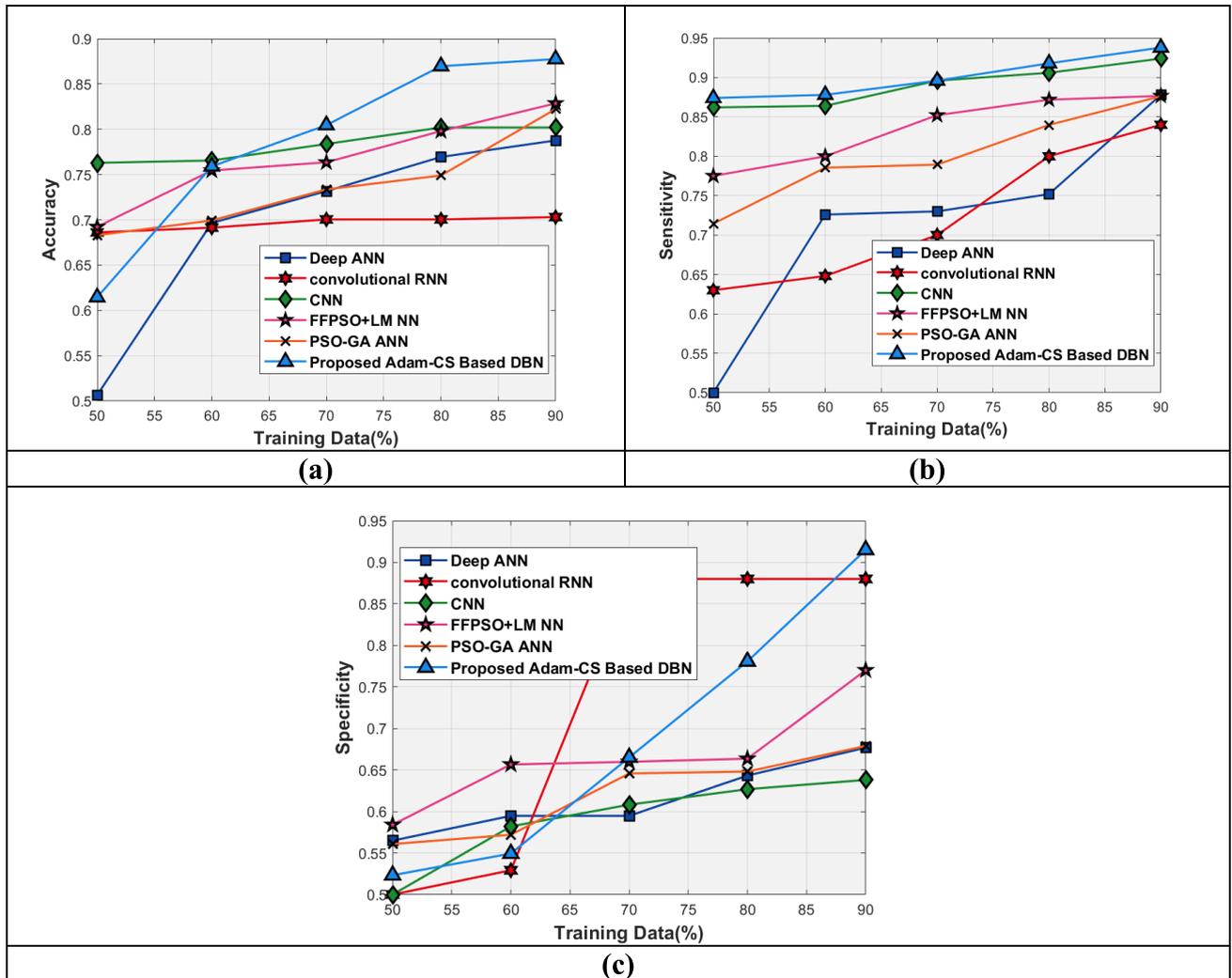
FIGURE 6. Comparative analysis using diabetes dataset, a) accuracy, b) sensitivity, c) specificity.

respectively. When training data = 80%, the sensitivity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.7197, 0.8783, 0.7323, 0.7726, and 0.7342, respectively, whereas the proposed Adam-CS based DBN obtained a sensitivity of 0.8707.

Figure 5 c) shows the analysis of specificity with respect to the training data. When training data = 50%, the specificity obtained by the proposed Adam-CS based DBN is 0.8978. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 56%, 4%, 9%, 10.41%, and 30.21%, respectively. When training data = 60%, the specificity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.6847, 0.8924, 0.8319, 0.8192, and 0.7655, respectively, whereas the proposed Adam-CS based DBN obtained specificity of 0.9008.

4) COMPARATIVE ANALYSIS USING DIABETES DATASET

Figure 6 shows the comparative analysis of the proposed algorithm using the diabetes dataset. Figure 6 a) shows the analysis of accuracy based on the training data. When training data = 70%, the accuracy obtained by the proposed Adam-CS based DBN is 0.7591. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 9%, 14%, 2%, 5.11%, and 8.82%, respectively. When training data = 80%, the accuracy obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.7695, 0.7005, 0.8021, 0.7981, and 0.75, respectively, whereas the proposed Adam-CS based DBN obtained better accuracy of 0.8698.

Figure 6 b) shows the analysis of sensitivity based on training data. When training data = 50%, the sensitivity obtained by the proposed Adam-CS based DBN is 0.874. Hence, the percentage of improvement reported by the
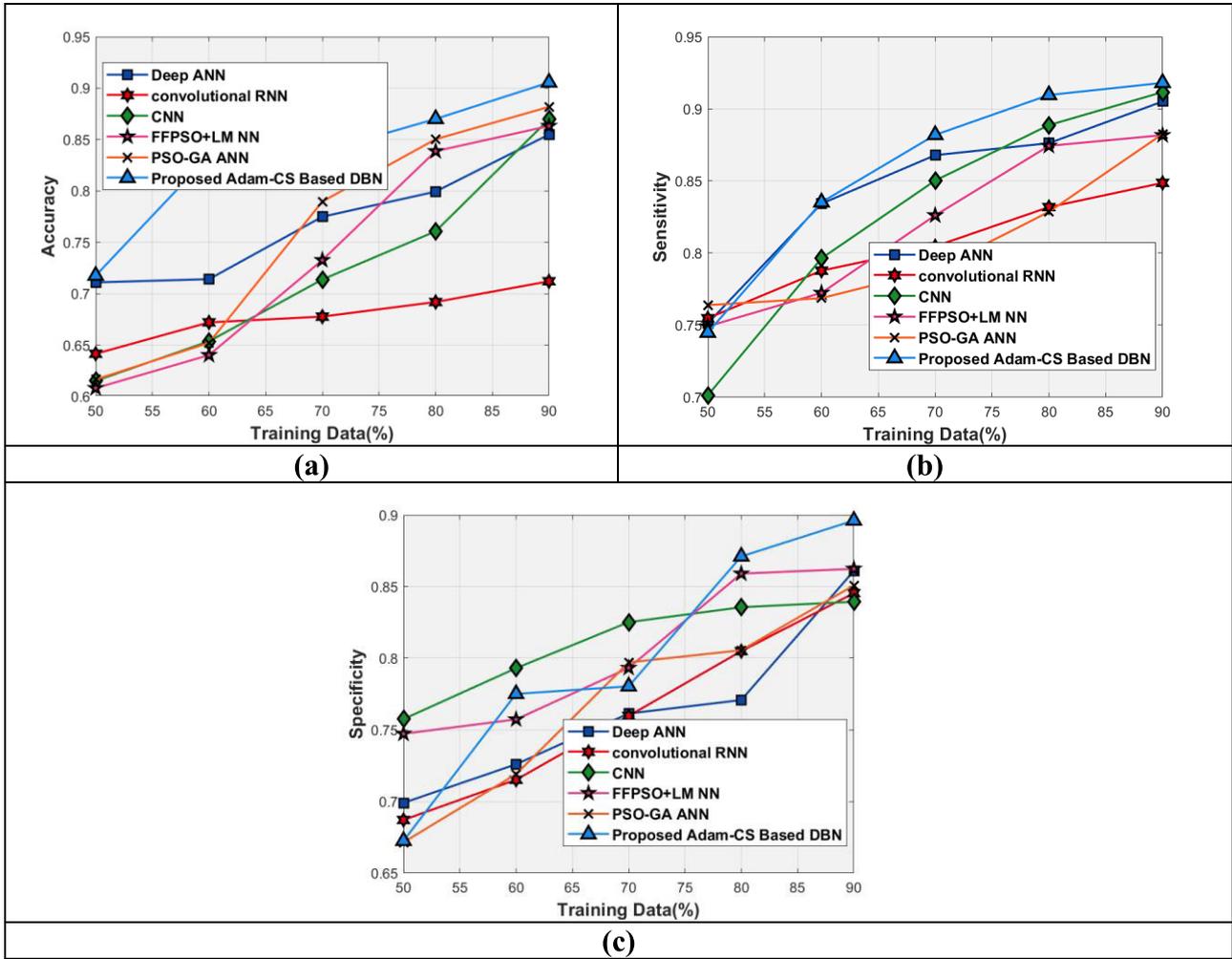
FIGURE 7. Comparative analysis using Cardiotocography Dataset, a) accuracy, b) sensitivity, c) specificity.

proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 79%, 39%, 1%, 11.32%, and 18.25%, respectively. When training data = 80%, the sensitivity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.752, 0.8, 0.906, 0.8719, and 0.8398, respectively, while the proposed Adam-CS based DBN obtained a sensitivity of 0.918.

Figure 6 c) shows the analysis of specificity with respect to the training data. When training data = 70%, the specificity obtained by the proposed Adam-CS based DBN is 0.665. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 7%, 4%, 4%, 10.54%, and 2.89%, respectively. When training data = 80%, the specificity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.6433, 0.88, 0.6269, 0.6636, and 0.649, respectively, whereas

the proposed Adam-CS based DBN obtained specificity of 0.79.

5) COMPARATIVE ANALYSIS USING CARDIOTOCOGRAPHY DATASET

Figure 7 shows the comparative analysis of the proposed algorithm using the Cardiotocography Dataset. Figure 7 a) shows the analysis of accuracy based on the training data. When training data = 50%, the accuracy obtained by the proposed Adam-CS based DBN is 0.7175. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 16%, 6%, 3.5%, 15.26%, and 14.09%, respectively. When training data = 70%, the accuracy obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.6853, 0.7203, 0.7552, 0.7326, and 0.7897, whereas the proposed Adam-CS based DBN obtained an accuracy of 0.7902, respectively.
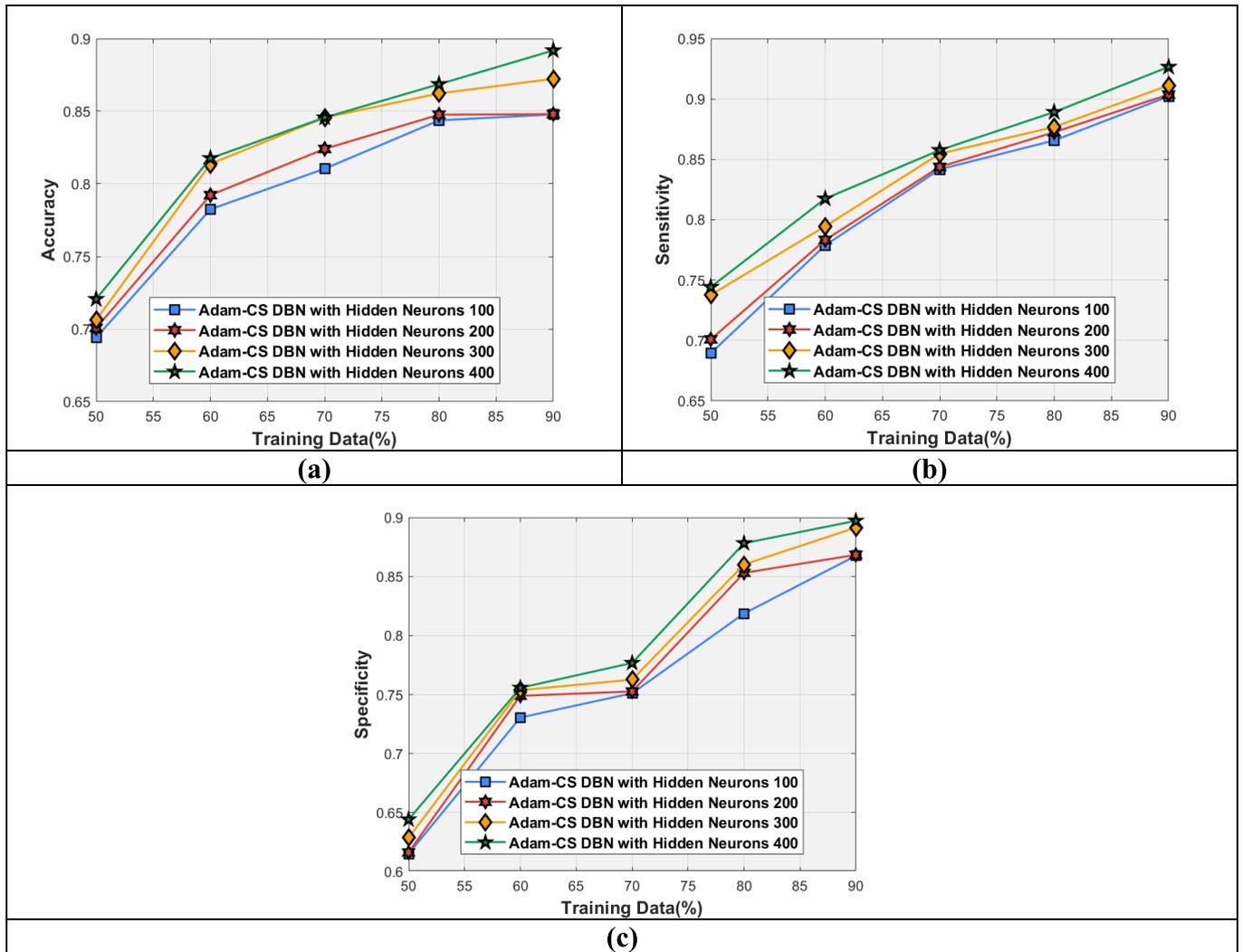
FIGURE 8. Sensitivity parameter analysis based on hidden neurons using Cardiotocography Dataset, a) accuracy, b) sensitivity, c) specificity.

Figure 7 b) shows the analysis of sensitivity based on training data. When training data = 60%, the sensitivity obtained by the proposed Adam-CS based DBN is 0.7448. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.1%, 5.68%, 4.65%, 7.52%, and 7.96%, respectively. When training data = 80%, the sensitivity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0..8762, 0.8319, 0.8887, 0.8745, and 0.8286, whereas the proposed Adam-CS based DBN obtained a sensitivity of 0.9096.

Figure 7 c) shows the analysis of specificity with respect to the training data. When training data = 50%, the specificity obtained by the proposed Adam-CS based DBN is 0.7751. Hence, the percentage of improvement reported by the proposed algorithm with respect to the existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 6.33%, 7.72%, 2.71%, 2.3%, and

7.24%, respectively. When training data = 80%, the specificity obtained by existing methods, like Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN is 0.7708, 0.805, 0.8356, 0.8591, and 0.8056, whereas the proposed Adam-CS based DBN obtained specificity of 0.871.

6) ANALYSIS BASED ON SENSITIVITY PARAMETERS

This section discusses the analysis based on the sensitivity parameters in the proposed Adam-CS based DBN algorithm, for Cardiotocography Dataset. Here, the sensitivity parameters considered are a) Hidden Neurons, b) Epochs, and c) Batch size.

a: ANALYSIS BASED ON VARYING THE HIDDEN NEURONS

The sensitivity analysis based on varying the hidden neurons is shown in Figure 8. The accuracy of varying the hidden neurons is shown in Figure 8 a) When the training data is 60%, the accuracy of the proposed Adam-CS based DBN algorithm
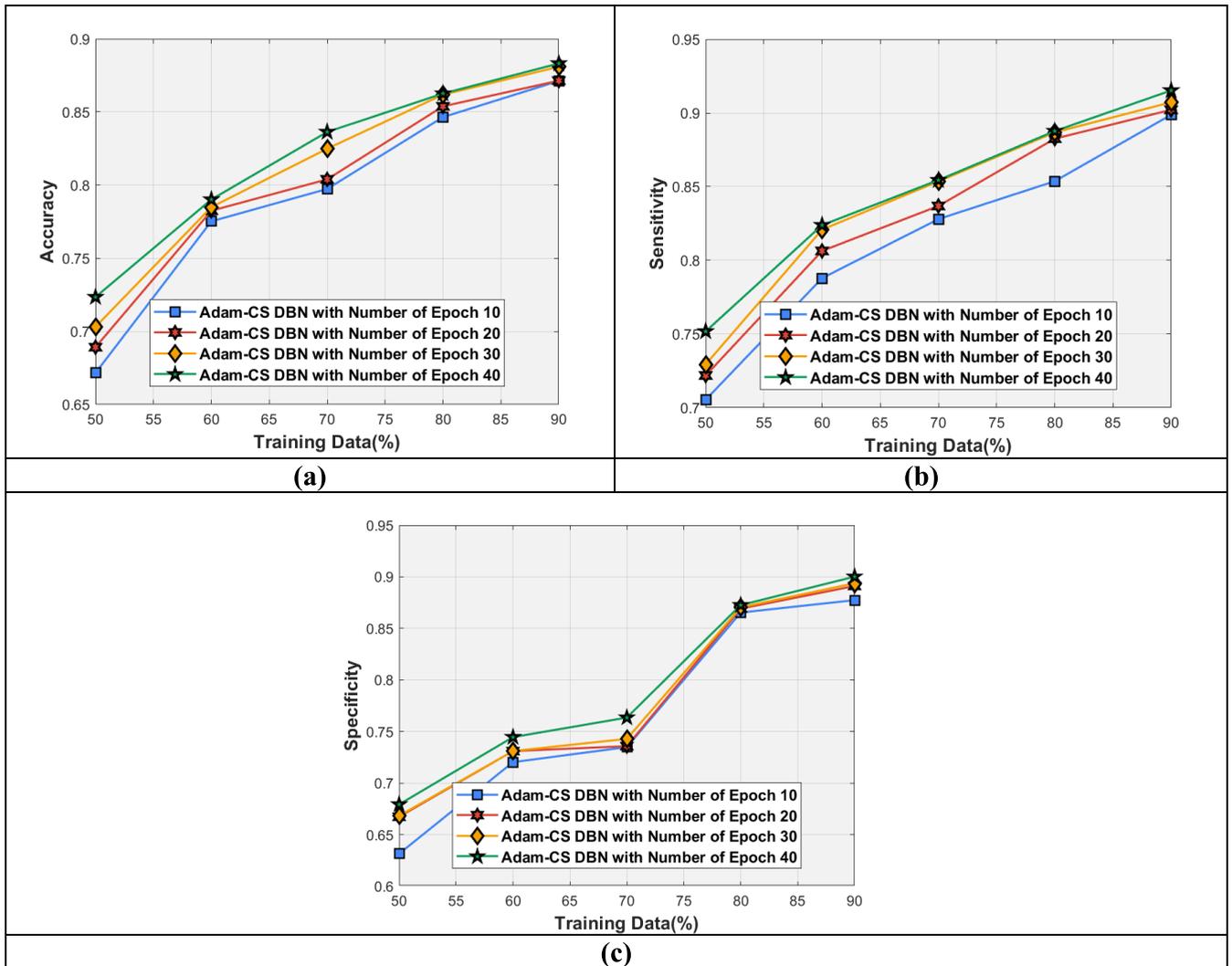
FIGURE 9. Sensitivity parameter analysis based on epochs using Cardiotocography Dataset, a) accuracy, b) sensitivity, c) specificity.

is 0.7824, 0.7922, 0.8138, and 0.8137, for the hidden neurons, such as 100, 200, 300, and 400, respectively. Figure 8 b) shows the sensitivity of varying the hidden neurons as 100, 200, 300, and 400. For the same training percentage, the sensitivity of the proposed Adam-CS based DBN algorithm is 0.7787, 0.7832, 0.7945, and 0.8175, for the hidden neurons, such as 100, 200, 300, and 400, respectively. Figure 8 c) depicts the specificity of the proposed system by varying the hidden neurons. The specificity of the proposed Adam-CS based DBN is 0.7305, 0.7489, 0.7537, and 0.7557, for the hidden neurons, such as 100, 200, 300, and 400, respectively in training data is 60%.

b: ANALYSIS BASED ON VARYING THE EPOCHS
The sensitivity analysis based on varying epochs is shown in Figure 9. Figure 9 a) shows the accuracy of the proposed system by varying the epochs. The proposed Adam-CS based DBN algorithm has the accuracy of 0.6715, 0.6892, 0.7031, and 0.7235, for varying the epochs 10, 20, 30, and 40,

respectively, for training data 50%. The sensitivity of varying the epochs for the proposed Adam-CS based DBN algorithm is given in Figure 9 b. The sensitivity is 0.7054, 0.7218, 0.7218, and 0.7518, for the proposed Adam-CS based DBN algorithm for the epochs 10, 20, 30, and 40, respectively, for the training data is 50%. The specificity of varying the epochs for the proposed method is shown in Figure 9 c. When the training data 50 %, the specificity of the proposed Adam-CS based DBN algorithm based on varying the epochs is 0.6315, 0.6673, 0.6683, and 0.6793, for the epochs 10, 20, 30, and 40, respectively.

c: ANALYSIS BASED ON VARYING THE BATCH SIZES
Figure 10 shows the sensitivity parameter analysis based on varying the batch size for the proposed Adam-CS based DBN algorithm. The accuracy based on varying the batch size of the proposed Adam-CS based DBN algorithm is depicted in Figure 10 a. For the training data is 70%, the accuracy of the
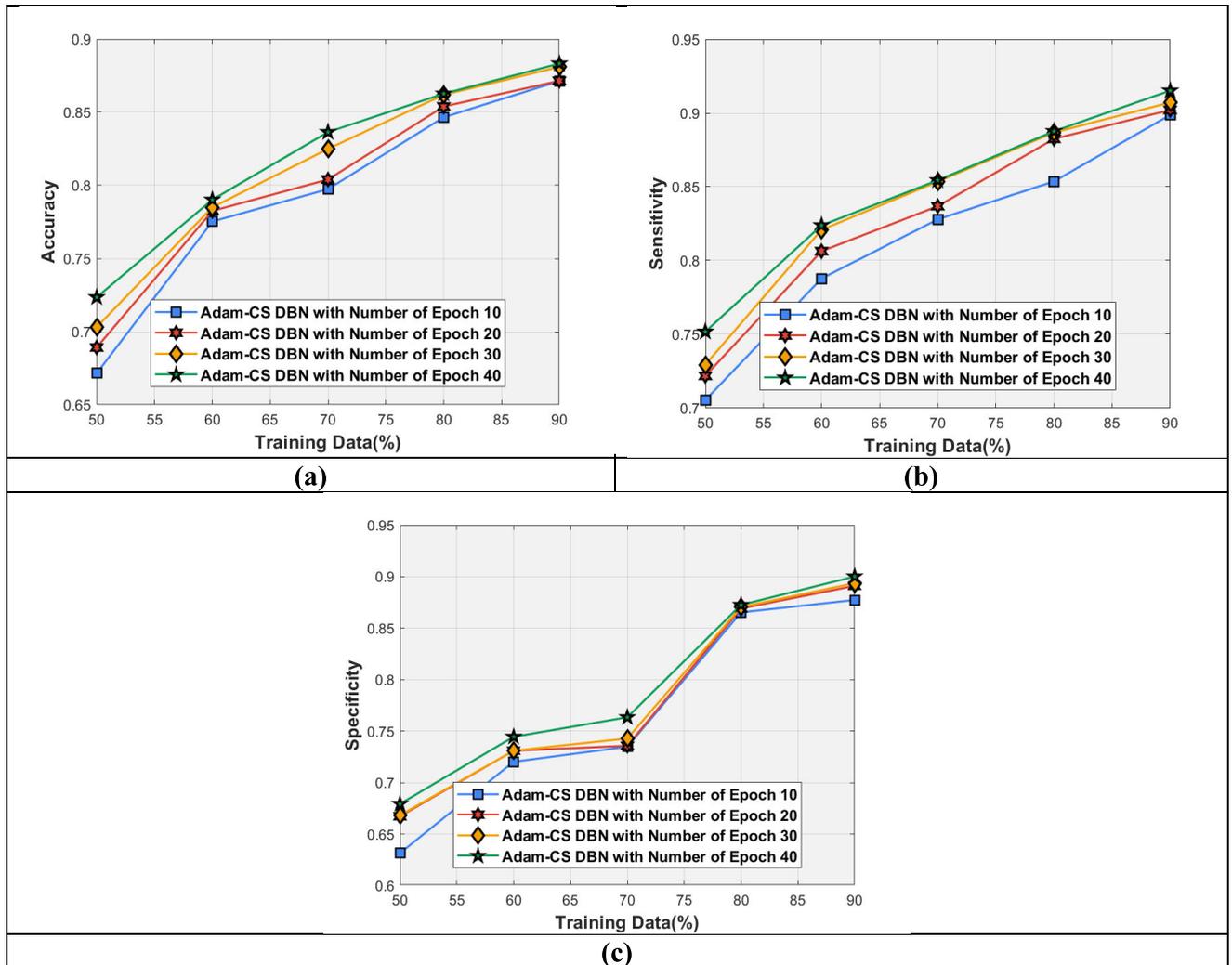
FIGURE 10. Sensitivity parameter analysis based on batch sizes using Cardiotocography Dataset, a) accuracy, b) sensitivity, c) specificity.

proposed Adam-CS based DBN algorithm based on varying the batch size is 0.8156, 0.7903, 0.7956, and 0.7656, for the batch sizes 50, 100, 150, and 200, respectively. Figure 10 b) shows the sensitivity of the proposed Adam-CS based DBN algorithm based on the batch size. The sensitivity of the proposed Adam-CS based DBN algorithm based on varying the batch size as 50, 100, 150, and 200 is 0.837, 0.8331, 0.8509, and 0.8029, respectively. The specificity analysis based on the batch size of the proposed Adam-CS based DBN algorithm is given in Figure 10 c. When the batch size is 50, 100, 150, and 200, of the proposed Adam-CS based DBN algorithm is have the specificity of 0.739, 0.7695, 0.7595, and 0.7026, respectively, for the training data is 70%.

7) COMPARATIVE DISCUSSION
This section elaborates on the comparative discussion made using the proposed Adam-CS based DBN algorithm for data classification with the existing methods, such as Deep ANN, Convolutional RNN, CNN, FFPSO+LM NN, and PSO-GA ANN. Analysis using various datasets for the metrics, like

accuracy, sensitivity, and specificity by varying the training data is discussed in Table 2. By analyzing five different datasets the metrics, like accuracy, sensitivity, and specificity, obtained the maximum value with the proposed algorithm is using training data = 90%. The reason for the performance improvement of the proposed system is the classifier to enhance the better accuracy of the classification, to increase the accuracy rate of the classification, to improve the quality of parameter size of classification, also to find the optimal solution.

8) SENSITIVITY PARAMETERS DISCUSSION
The comparative discussion made by the proposed Adam-CS based DBN algorithm for data classification using various sensitivity parameters, such as hidden layers, epochs, and batch sizes are elaborated in this section. Table 3 depicts the analysis based on the accuracy, sensitivity, and specificity of the Cardiotocography Dataset. The maximum accuracy and sensitivity occur when varying the hidden neurons of the proposed Adam-CS based DBN algorithm.

TABLE 2. Comparative discussion.

Datasets	Metrics	Deep ANN	Convolutional RNN	CNN	FFPSO +LM NN	PSO-GA ANN	Proposed Adam-CS based DBN
Heart disease dataset	Accuracy	0.8515	0.85	0.8614	0.9162	0.8874	0.934
	Sensitivity	0.9573	0.85	0.9451	0.9526	0.9526	0.9659
	Specificity	0.8561	0.75	0.8129	0.8997	0.8773	0.9293
Breast cancer dataset	Accuracy	0.7203	0.7587	0.8357	0.7960	0.7553	0.8671
	Sensitivity	0.8209	0.9005	0.7806	0.8325	0.8315	0.9453
	Specificity	0.7411	0.6	0.6	0.7493	0.6739	0.8
Lung cancer dataset	Accuracy	0.7813	0.9063	0.8775	0.8417	0.7967	0.9375
	Sensitivity	0.7823	0.9231	0.891	0.8177	0.7917	0.941
	Specificity	0.8857	0.9565	0.928	0.9138	0.8984	0.968
Diabetes dataset	Accuracy	0.7878	0.7031	0.8021	0.8287	0.8284	0.8776
	Sensitivity	0.878	0.84	0.924	0.8766	0.8762	0.938
	Specificity	0.6769	0.88	0.6381	0.7702	0.6689	0.9151
Cardiotocography Dataset	Accuracy	0.8549	0.7120	0.8698	0.8632	0.8818	0.9056
	Sensitivity	0.9051	0.8487	0.9117	0.8818	0.8826	0.9180
	Specificity	0.861	0.8454	0.8395	0.8624	0.8509	0.8962

TABLE 3. Sensitivity parameters discussion.

Dataset	Sensitivity Parameters	Accuracy	Sensitivity	Specificity
Cardiotocography Dataset	Hidden Neurons	0.8919	0.9265	0.8972
	Epochs	0.8832	0.9152	0.9002
	Batch size	0.8872	0.9156	0.8618

IV. CONCLUSION

An effective data classification method named Adam based Cuckoo search is proposed to perform the classification process. At first, the input data is subjected to the pre-processing stage, where the log transformation and missing value imputation is carried out for eliminating the noise and artifacts present in the data. Then, the wrapper-based approach is used to select the suitable features using the heuristic function. The data classification is achieved using the Deep Belief classifier, in which the MLP layer is trained by the proposed Adam based Cuckoo search algorithm. The breeding behavior of cuckoos is integrated with the step size parameter to enhance the accuracy of the classification process. The key idea of cuckoo species is that they walk in a random manner in a biased way using some random step sizes. Moreover, the proposed Adam-Cuckoo search based DBN attained better performance with respect to the metrics, such as accuracy, specificity, and sensitivity, with 90% of training data. In the future, the classification accuracy of the data classification approach will be enhanced using some other optimization algorithm.

REFERENCES

[1] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 23, no. 4, pp. 687–719, 2009.

[2] B. C. M. Fung, K. Wang, and P. S. Yu, "Anonymizing classification data for privacy preservation," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 5, pp. 711–725, May 2007.

[3] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognit.*, vol. 40, no. 12, pp. 3358–3378, Dec. 2007.

[4] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang, and X. Huang, "Hyperspectral image classification with deep learning models," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5408–5423, Sep. 2018.

[5] S. Tharani and C. Yamini, "Classification using convolutional neural network for heart and diabetics datasets," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 5, no. 12, pp. 417–422, Dec. 2016.

[6] H. Mohsen, E.-S. A. El-Dahshan, E.-S. M. El-Horbaty, and A.-B. M. Salem, "Classification using deep learning neural networks for brain tumors," *Future Comput. Informat. J.*, vol. 3, no. 1, pp. 68–71, 2018.

[7] C. Affonso, A. L. D. Rossi, F. H. A. Vieira, and A. C. P. de Leon Ferreira, "Deep learning for biological image classification," *Expert Syst. Appl.*, vol. 85, pp. 114–122, Nov. 2017.

[8] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[9] W. Zhao, Z. Guo, J. Yue, X. Zhang, and L. Luo, "On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery," *Int. J. Remote Sens.*, vol. 36, no. 13, pp. 3368–3379, Jul. 2015.

[10] Y. Liu, G. Cao, Q. Sun, and M. Siegel, "Hyperspectral classification via deep networks and superpixel segmentation," *Int. J. Remote Sens.*, vol. 36, no. 13, pp. 3459–3482, Jul. 2015.

[11] P. Zhong, Z. Gong, S. Li, and C. B. Schonlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Rem. Sens*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

[12] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, pp. 1–12, Jul. 2015.

[13] B. Pan, Z. Shi, N. Zhang, and S. Xie, "Hyperspectral image classification based on nonlinear spectral-spatial network," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1782–1786, Dec. 2016.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[15] M. Zorzi, A. Zanella, A. Testolin, M. De Filippo De Grazia, and M. Zorzi, "Cognition-based networks: A new perspective on network optimization using learning and distributed intelligence," *IEEE Access*, vol. 3, pp. 1512–1530, 2015.

[16] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. Neural Netw.* Springer, 2016, pp. 213–226.

[17] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Mar. 2017, pp. 1–6.

[18] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, "Medical image classification using synergic deep learning," *Med. Image Anal.*, vol. 54, pp. 10–19, May 2019.

[19] B. Pan, Z. Shi, and X. Xu, "MugNet: Deep learning for hyperspectral image classification using limited samples," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 108–119, Nov. 2018.

[20] A. Hassan and A. Mahmood, "Convolutional recurrent deep learning model for sentence classification," *IEEE Access*, vol. 6, pp. 13949–13957, 2018.

[21] P. Perera and V. M. Patel, "Learning deep features for one-class classification," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5450–5463, Nov. 2019.

[22] L. Zhong, L. Hu, and H. Zhou, "Deep learning based multi-temporal crop classification," *Remote Sens. Environ.*, vol. 221, pp. 430–443, Feb. 2019.

[23] Y. Wang, Q. Chen, D. Gan, J. Yang, D. S. Kirschen, and C. Kang, "Deep learning-based socio-demographic information identification from smart meter data," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2593–2602, May 2019.

[24] Q. Zhang, L. T. Yang, Z. Chen, P. Li, and F. Bu, "An adaptive dropout deep computation model for industrial IoT big data learning with crowdsourcing to cloud computing," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2330–2337, Apr. 2019.

[25] M. S. Hossain and G. Muhammad, "Environment classification for urban big data using deep learning," *IEEE Commun. Mag.*, vol. 56, no. 11, pp. 44–50, Nov. 2018.

[26] J. Zhai, L. Zang, and Z. Zhou, "Ensemble dropout extreme learning machine via fuzzy integral for data classification," *Neurocomputing*, vol. 275, pp. 1043–1052, Jan. 2018.

[27] S. Chambon, M. N. Galtier, P. J. Arnal, G. Wainrib, and A. Gramfort, "A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 4, pp. 758–769, Apr. 2018.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[29] X.-S. Yang and S. Deb, "Engineering optimisation by cuckoo search," 2010, *arXiv:1005.2908*. [Online]. Available: http://arxiv.org/abs/1005.2908

[30] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, Dec. 1997.

[31] X. Yao, "Evolving artificial neural networks," *Proc. IEEE*, vol. 87, no. 9, pp. 1423–1447, Sep. 1999.

[32] N. Gomathi and N. P. Karlekar, "Ontology and hybrid optimization based SVNN for privacy preserved medical data classification in cloud," *Int. J. Artif. Intell. Tools*, vol. 28, no. 3, May 2019, Art. no. 1950009.

[33] R. P. Singh and P. Yadav, "Intelligent information retrieval in data mining," *Int. J. Sci. Eng. Res.*, vol. 3, no. 5, p. 1, 2012.

[34] A. N. Jadhav and N. Gomathi, "DIGWO: Hybridization of dragonfly algorithm with Improvedc grey wolf optimization algorithm for data clustering," *Multimedia Res.*, vol. 2, no. 3, pp. 1–11, 2019.

[35] L. Pastur-Romay, F. Cedrón, A. Pazos, and A. Porto-Pazos, "Deep artificial neural networks and neuromorphic chips for big data analysis: Pharmaceutical and bioinformatics applications," *Int. J. Mol. Sci.*, vol. 17, no. 8, p. 1313, Aug. 2016.

[36] W. H. Bangyal, J. Ahmad, and H. T. Rauf, "Optimization of neural network using improved bat algorithm for data classification," *J. Med. Imag. Health Informat.*, vol. 9, no. 4, pp. 670–681, May 2019.

[37] B. A. Kumar and A. Neelima, "Machine learning techniques in biological data classification and clustering: Initiation of a scientific voyage," *J. PeerScientist*, vol. 2, no. 2, 2020, Art. no. e1000011.

[38] D. Griffiths and J. Boehm, "A review on deep learning techniques for 3D sensed data classification," *Remote Sens.*, vol. 11, no. 12, p. 1499, Jun. 2019.

[39] *Breast Cancer Data Set*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/breast+cancer

[40] *Diabetes Data Set*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/diabetes

[41] *Heart Disease Data Set*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Heart+Disease

[42] *Lung Cancer Data Set*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Lung+Cancer

[43] *Cardiotocography Data Set*. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Cardiotocography

[44] P. Kora and K. S. Rama Krishna, "Hybrid firefly and particle swarm optimization algorithm for the detection of bundle branch block," *Int. J. Cardiovascular Acad.*, vol. 2, no. 1, pp. 44–48, Mar. 2016.

[45] N. Y. Moteghaed, K. Maghooli, S. Pirhadi, and M. Garshasbi, "Biomarker discovery based on hybrid optimization algorithm and artificial neural networks on microarray data for cancer classification," *J. Med. Signals Sensors*, vol. 5, no. 2, pp. 88–96, 2015.

MOHAMMED MOHSIN received the B.Sc. degree from Basra University, Basra, Iraq, and the M.Sc. degree from the Sam Higginbottom Institute of Agriculture, Technology and Sciences, Allahabad, India. He is currently pursuing the Ph.D. degree with the School of Mathematics and Statistics, Huazhong University of Science and Technology. He is also working as a Lecturer with the Department of Statistics, Basra University. He has six years of teaching experience. His research interests include optimization, approximation, and deep learning.

HONG LI received the M.Sc. degree in mathematics and the Ph.D. degree in pattern recognition and intelligence control from the Huazhong University of Science and Technology, Wuhan, China, in 1986 and 1999, respectively. She is currently a Professor with the School of Mathematics and Statistics, Huazhong University of Science and Technology. Her research interests include approximation theory, wavelet analysis, learning theory, neural networks, signal processing, and pattern recognition.

HEMN BARZAN ABDALLA received the Ph.D. degree in communication and information engineering. He worked as a Project Assistant in various higher education places. He is currently working as a Lecturer with Wenzhou-Kean University with a member of the Institute of Training and Development in Sulaimani (KRG). He possesses one decade of experience in teaching. He has more than 100 project systems for several places. His research interests include big data, data security, NoSQL, and application. He is an editorial board member/Reviewer of international/national journals and conferences.

● ● ●