The current issue and full text archive of this journal is available on Emerald Insight at:
https://www.emerald.com/insight/2210-8327.htm

Received 2 January 2019
Revised 5 March 2019
Accepted 16 March 2019

Building a novel classifier based on teaching learning based optimization and radial basis function neural networks for non-imputed database with irrelevant features

Ch. Sanjeev Kumar Dash and Ajit Kumar Behera
Silicon Institute of Technology, Bhubaneswar, India
Satchidananda Dehuri
Department of Information and Communication Technology, Fakir Mohan University, Balasore, India, and
Sung-Bae Cho
Soft Computing Laboratory, Department of Computer Science, Yonsei University, Seoul, Republic of Korea

Abstract

This work presents a novel approach by considering teaching learning based optimization (TLBO) and radial basis function neural networks (RBFNs) for building a classifier for the databases with missing values and irrelevant features. The least square estimator and relief algorithm have been used for imputing the database and evaluating the relevance of features, respectively. The preprocessed dataset is used for developing a classifier based on TLBO trained RBFNs for generating a concise and meaningful description for each class that can be used to classify subsequent instances with no known class label. The method is evaluated extensively through a few bench-mark datasets obtained from UCI repository. The experimental results confirm that our approach can be a promising tool towards constructing a classifier from the databases with missing values and irrelevant attributes.

Keywords Pattern recognition, Imputation, Classification, Radial basis function neural networks, Teaching learning based optimization, k-Nearest neighbor

Paper type Original Article

Publishers note: The publisher wishes to inform readers that the article “Building a novel classifier based on teaching learning based optimization and radial basis function neural networks for non-imputed database with irrelevant features” was originally published by the previous publisher of Applied Computing and Informatics and the pagination of this article has been subsequently changed. There has been no change to the content of the article. This change was necessary for the journal to transition from the previous publisher to the new one. The publisher sincerely apologises for any inconvenience caused. To access and cite this article, please use Kumar Dash, Ch. K., Kumar Behera, A., Dehuri, S., Cho, S. B. (2019), “Building a novel classifier based on teaching learning based optimization and radial basis function neural networks for non-imputed database with irrelevant features” Applied Computing and Informatics, Vol. ahead-of-print No. ahead-of-print. https://10.1016/j.aci.2019.03.001. The original publication date for this paper was 18/03/2019.

1. Introduction

The occurrence of missing values and irrelevant features in real data are not uncommon, whereas data mining algorithms are designed for quality data [1]. Hence, building a classifier for the dataset consist of missing values and many irrelevant attributes leads to non-useful results [2]. Therefore, to derive novel and useful results for the decision maker, the process of imputing and identifying missing values and relevant features, respectively are highly recommended. Since decades ago these two problems are treated as the problem of importance in object detection & recognition (pattern recognition)[11] and data mining [3] in general and ECG signals diagnosis [13], power flow calculation [14], simulation and control of dynamic system [15], magnetic modeling [16], identification and classification of plant leaf diseases [17], discrimination of low and full fat Yogurts [19,20,22] in specific.

There are several approaches to impute missing values of which we concentrate on least square estimation method [2,3]. A large variety of feature selection techniques have been developed under the umbrella of filter, wrapper, and embedded methods with a goal to select relevant subset of features [4]. In this work a filter style approach known as "Relief" method is used for selecting a subset of attributes that preserves the relevant information found in the entire set of attributes [5]. After the task of imputation of missing values and selection of the relevant set of features, we develop a classifier based on TLBO and RBFNs by inheriting their best features [6,7,8]. RBFN one of the members of artificial neural networks (ANNs) [21,22] has good generalization, simple structure and strong tolerance to noise which ignited us to consider here as a suitable method of classification. Many methods have been developed for training RBFNs [12,17,18], however, to the best of our knowledge, training RBFNs using TLBO is new. TLBO is a population based optimization algorithm motivated by a teacher on the output of learners within a classroom environment, where learners first obtain knowledge from teacher and subsequently from classmates. Moreover, a new improved TLBO (iTLBO) has been proposed to train the RBFNs.

In a nutshell, this work undergoes three different phases like imputation of missing values by least square estimation approach, feature selection through Relief, and classification by iTLBO trained RBFNs in pipeline.

2. Background

The background of this research work like missing values imputation, feature selection, RBFNs, and TLBO are discussed here.

2.1 Imputation of missing values and feature selection

The problem of classification is basically the foundation of dividing the feature space into sections, one section for each category of inputs. Classifiers are usually, designed with labeled data, which is sometime referred to as supervised classification. In general, classification with missing data and irrelevant features focuses on three distinct tasks: handling missing values [1] (i.e., imputing values), feature selection, and pattern classification. Let $D = [x_{ij}]_{Nxd}$, where $i = 1, 2, \ldots, N$, and $j = 1, 2, \ldots, d$, is the dataset containing N samples and d features. In D, each sample is assigned a class label from the set $C = \{c_1, c_2, .., c_M\}$, where $|C| = M$. Let each x_{ij}, be represented as a tuple (x_{ij}, y_{ij}), in which y_{ij} can take only two values either 0 or 1. If the value of $v_{ij} = 0$, then its associated x_{ij} value is missing, otherwise present. Input data has quantitative and qualitative variables. Quantitative or continuous data is measured on a numerical scale. Non-numerical (i.e., colors, names, opinions) is called qualitative data, which can be discrete or categorical. The overall goal of handling missing value is to map the value of y_{ij} from 0 to 1 by substituting an appropriate value of x_{ij} with less bias.

Alongside feature selection problem is defined as to select a subset of features from the given set of features, thereby the dataset is mapped from $(x_{ij})_{Nxd}$ to $(x_{ij})_{Nxk}$, where $k \ll d$. With this intention, filter method is selecting the most relevant features, however, a predefined quality measure is necessary to establish the level of relevance of the features. Filter method is not able to identify correlation among the features simultaneously. Unlike filter, wrapper is able to address correlation among features because it uses the performance of the classifier to optimize the subset. This also led towards problem of intractability. Moreover, this method has the additional cost of reconstructing the classifier with modified feature subset. Hence to avoid these issues, a filter like algorithm known as Relief method is employed here.

2.2 Radial basis function networks

The RBF network [8] is a topology having three layers: an input, a hidden, and a linear output layer (see Figure 1). The input can be modeled as *n*-dimensional input vector. The hidden layer implements a radial activation function and that carry out a non-linear transformation from the input space to the hidden space. The center and width are two parameters associated for each hidden node. Usually, a nonlinear transformation from input to the hidden space is made based on Gaussian kernel as described in Eq. (1).

$$\varphi_i(x) = \exp\left(-\frac{\|x-\mu_i\|^2}{2\sigma_i^2}\right), \tag{1}$$

where $||\ldots||$ represents Euclidean norm, μ_i, σ_i, and φ_i are center, spread, and the output of ith hidden unit, respectively. The interconnection between the hidden and output layer are made through a weighted connections w_i. The output layer, a summation unit, supplies the response of the network to the outside world.

The radial basis function is so named because the value of the function is same for all points which are at the same distance from the center.

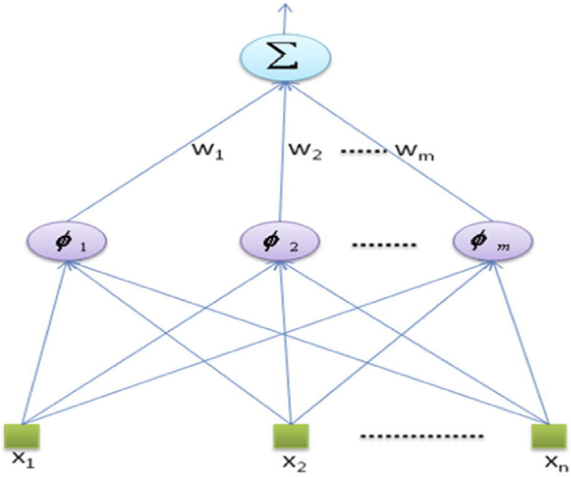


Figure 1.
Architecture of RBFN.

In literature, radial basis function networks [6] have many extensive uses, including classification, time series prediction, function approximation, etc. Training RBF networks is normally faster than training multi-layer perceptron (MLP) networks. Training of RBF network [9,11] involves two steps: (1) the kernel parameters of the hidden neurons are determined by an unsupervised method or heuristic method; (2) The weights of the output-layer are determined by pseudo-inverse method.

2.3 Teaching learning based optimization

Teaching learning based optimization is one of the population based nature inspired algorithms introduced by Rao et al. [6,9]. This is inspired purely from the natural phenomena of teaching-learning process that motivated by a teacher on the output of learners within a classroom environment, where learners first obtain knowledge from teacher and subsequently from classmates. In the first phase, a teacher imparts knowledge directly to his/her students. In practice, the possibility of a teacher's teaching being successful, is distributed under Gaussian law. Overall, how much knowledge is transferred to a student depends not only on his/her teacher but also interactions among the students through peer learning. A basic algorithm of TLBO is presented below.

3. Proposed method

Our integrated approach is undergoing three phases in pipeline. In first phase, the missing values are imputed by least square estimator, in second phase the relief algorithm is used for feature selection and finally our improved TLBO based RBFN is used for building the classifiers for the preprocessed database. Figure 2 is illustrating our approach.

3.1 Missing value imputation using least-square estimator

In this phase, we estimate the missing value from D by formulating a matrix A, where all the attribute values are known. In the least-square problem, the output of a model is given by the linearly parameterized expression,

$$y = \phi_1 f_1(u) + \phi_2 f_2(u) + \ldots + \phi_n f_n(u). \tag{2}$$

If the target system has q outputs, expressed as $y = [y_1, \ldots y_q]^T$ with q > 1 then we have a set of linear equations in matrix form, AΘ + E = Y, where A is an mxn matrix as given below:

$$A = \begin{bmatrix} f_1(u_1)\ldots f_n(u_1) \\ . \\ f_1(u_m)\ldots f_n(u_m) \end{bmatrix}$$

and Θ is an $n \times q$ unknown parameter matrix:

$$\Theta = \begin{bmatrix} \theta_{11}\ldots\theta_{1q} \\ . \\ \theta_{n1}\ldots\theta_{nq} \end{bmatrix} \quad Y = \begin{bmatrix} y_{11}\ldots y_{1q} \\ . \\ y_{m1}\ldots y_{mq} \end{bmatrix}$$

is an $m \times q$ output matrix with y_{ij} denoting the jth output value in the ith data pairs.

$$\Theta = \left(A^T A\right)^{-1} A^T Y \tag{3}$$

After getting the value of Θ, we will continue imputing all the values in the data set D.

Algorithm TLBO

```
BEGIN
    Set generation = 0;
    Initialize Population (P);   // |P| is the size of the population
    Evaluate (P)
    DO
        BS = Select Best Solution from (P); // Archive of best solution.
        FOR $i$ = 1 to |P|
          $T_F$= Round (1+r);                                      Teacher Phase
          // $T_F$ & $r$ are teaching factor and random number, respectively.
          Xmean= Mean Vector from (P);
          Xteacher=Select Best Solution from (P);
          Xnew=$X_i$ + r. (Xteacher-($T_F$.Xmean));
          IF
          Xnew is better than $X_i$ then $X_i$ = Xnew;
          END IF
          $X_j$= random (P); $X_i$ <> $X_j$;                          Learner Phase
          IF
          Xi is better than Xj then Xinew= Xi+ r. (Xi-Xj)
          ELSE
          Xinew = Xi + r. (Xj - Xi);
          END IF
          Evaluate (Xinew);
          IF
          Xinew is better than Xi then Xi=Xinew;
          END IF
        END FOR
        P=Replace worst solution of (P) by (BS);
        P=Remove duplicate individuals from (P);
        generation++;
    WHILE (generation <> Max-generation);
    PRINT best result of P;
END
```

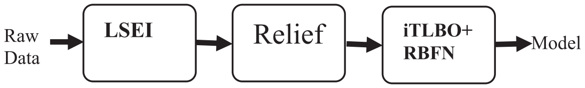
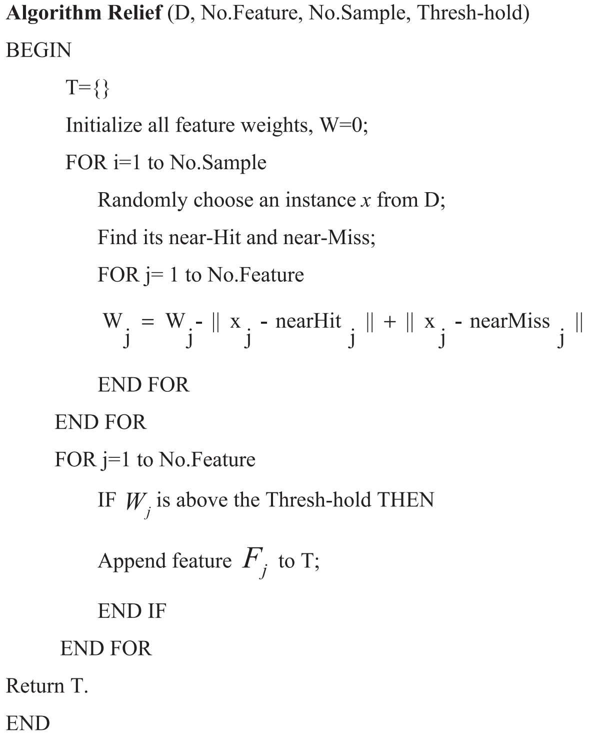


Figure 2. Pictorial Representation of our Approach (LSEI: Least Square Estimation for Imputation).

3.2 Relief algorithm for feature selection

In this second phase of our work, we discuss Relief algorithm inspired by instance based learning. It is an filter method algorithm for individual feature selection. It calculates a proxy statistics for each feature that can be used to estimate the feature quality or relevance to the target concept. The pseudocode of this method is given below.

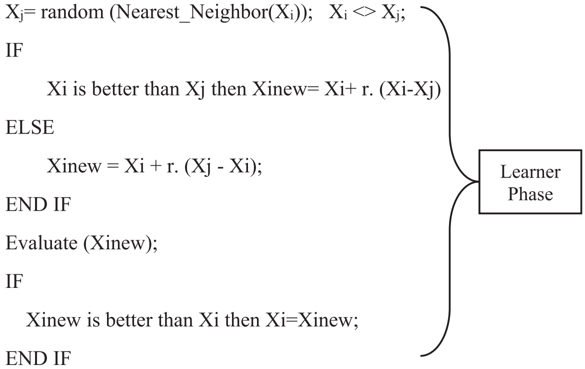


```
Algorithm Relief (D, No.Feature, No.Sample, Thresh-hold)
BEGIN
    T={}
    Initialize all feature weights, W=0;
    FOR i=1 to No.Sample
        Randomly choose an instance $x$ from D;
        Find its near-Hit and near-Miss;
        FOR j= 1 to No.Feature
            $W_j = W_j - \| x_j - nearHit_j \| + \| x_j - nearMiss_j \|$
        END FOR
    END FOR
    FOR j=1 to No.Feature
        IF $W_j$ is above the Thresh-hold THEN
        Append feature $F_j$ to T;
        END IF
    END FOR
Return T.
END
```

3.3 Improved TLBO based RBFN

In third phase, we are building a RBFN classifier which is trained by TLBO and improved TLBO. First we will provide a detailed introduction to the improved TLBO and then the improved TLBO + RBFN network is developed with the aim of achieving better classification accuracy.

3.3.1 Improved TLBO (iTLBO). In the canonical TLBO, during the learning phase the learner is exposed to the entire population of the class. However, it has been realized that if the learner is restricted with a peer team instead of all individuals of the population then he/she can raise his/her level of acquiring knowledge. With this idea, we are introducing a neighborhood structure of learners as peer learners group for making a learner to learn. Hence, in the learner phase, we have adopted a square topology as peer learners group for a learner. That means a student will not only acquire knowledge from the best of all individuals (i.e., teacher) but also he/she improves his/her standard from his neighborhood of fellow learners. In that context, the learner phase of TLBO has been modified as given below.



```
Xj= random (Nearest_Neighbor(Xi));   Xi <> Xj;
IF
      Xi is better than Xj then Xinew= Xi+ r. (Xi-Xj)
ELSE
      Xinew = Xi + r. (Xj - Xi);
END IF
Evaluate (Xinew);
IF
   Xinew is better than Xi then Xi=Xinew;
END IF
```

Here the nearest_neighbor( ) will find out a group of peer learners for a learner. The size of the neighborhood can be treated as a parameter for learner phase. Alongside, we have also made the teaching parameter (T_F) adaptive by considering the individual fitness value and population diversity. Recall that the teaching factor decides the value of mean to be changed. In the canonical TLBO, the value of T_F is either 1 or 2 thereby learners learn nothing from the teacher or learn all the things from the leaner. But in real practice, the value of T_F may be between 1 and 2 include both. Hence to make this idea fruitful, the fitness variable is selected as inputs to choose T_F. BS is containing the global best solution denoted as X_g^k found so far i.e. up to *k*th iteration, which is just a position for one individual, corresponding to the best fitness F_g^k. So the global best solution fitness differentials between *k*th and k-1th can be defined as:

$$\Delta F^k = \left|F_{SB}^k - F_{SB}^{k-1}\right|. \tag{4}$$

Now, we can give definition for function of convergence speed as follows:

$$C_s = \Delta F^k / \Delta, \tag{5}$$

where $\Delta_1 = \max\{\Delta F^1,\ \Delta F^2,\ \ldots \Delta F^k\}$ Eq. (5) can calculate the convergence speed, which is less than or equal to 1.

In evolution process of TLBO, population diversity is a major factor. For computing the diversity of the population, standard deviation of the individual fitness values of population can be used. In this paper, we present a new strategy for calculating population position diversity by fitness value. The population position diversity can be obtained by using deviation ideology approach defined in Eqs. (6)–(8).

$$F_{avg}^k = 1/|P| \sum_{i=1}^{|P|} F^k(i), \tag{6}$$

$$\Delta_2 = \max\left\{\left|F^k(1) - F_{avg}^k\right|,\ \ldots,\ \left|F^k(|P|) - F_{avg}^k\right\}\right|, \tag{7}$$

$$\sigma^2 = 1/|P| \sum_{i=1}^{|P|} \left(\frac{F^k(i) - F_{avg}^k}{\Delta_2}\right)^2, \tag{8}$$

where F_{avg}^{k} is the average population fitness for current kth iteration; $F^{k}(i)$ stands for ith individual fitness; Δ_2 stands for normalization factor; |P| is the population size, σ^2 represents the population diversity. It is evident that larger the σ^2, the larger is the population diversity.

To improve adaptive teaching factor T_F, we use the index C_S for representing the convergence speed with respect to the best solution fitness found so far in current iteration, and the index σ^2 to represent diversity with regards to the population fitness deviation. Hence we can compute T_F by C_S and σ^2 adaptively, as follows:

$$T_F = \alpha.C_S + \beta.\sigma^2 + 1, \tag{9}$$

where α, β are factors; σ^2 and C_S are less than or equal to 1 and greater than 0, so $1 \leq T_F \leq \alpha + \beta + 1$, Rao et al. [6] suggested that the value of T_F can be either 1 or 2. Hence, we set $\alpha + \beta + 1 \leq 2$. The proposed method of adaptive teaching factor (T_F) is applied for better local searching ability that improves the accuracy and convergence speed.

3.3.2 iTLBO + RBFN. This section describes the iTLBO + RBFN which can adjust the network parameters during the training process. In the initialization stage, let the position of the ith individual be represented as shown in Figure 3. RBFNs mainly depend on center and width of the kernel in addition to weights and bias. However, here, we just encode the centers, widths, and bias into an individual for stochastic search using iTLBO.

Suppose the maximum number of kernel nodes is set to K_{max}, then the structure of the individual is represented as follows (c.f., Figure 3):

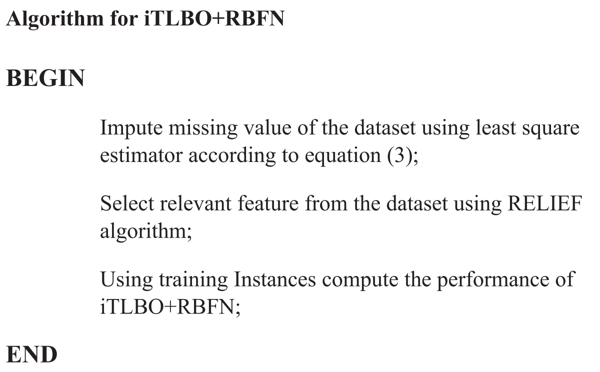
In other words, each individual has three constituent parts such as center, width, and bias. The length of the individual is $2K_{max} + 1$.

The fitness function which is used to guide the search process is defined in equation (10).

$$f(x) = \frac{1}{N}\sum_{i=1}^{N}\left(t_i - \widehat{\Phi}(\overrightarrow{x_i})\right)^2 \tag{10}$$

where, N is the total number of training instances, t_i is the actual output and $\widehat{\Phi}(\overrightarrow{x_i})$ is the estimated output of RBFNs. Initially, the centers, widths, and bias are computed using training vectors, the weight is computed using pseudo-inverse method.

$$\begin{aligned} &Y = W\Phi \\ &\Rightarrow W = \left(\Phi^T\Phi\right)^{-1}\Phi^T Y \end{aligned} \tag{11}$$



```
Algorithm for iTLBO+RBFN

BEGIN

        Impute missing value of the dataset using least square
        estimator according to equation (3);

        Select relevant feature from the dataset using RELIEF
        algorithm;

        Using training Instances compute the performance of
        iTLBO+RBFN;

END
```

4. Experimental study

In the experimental study, we start with a brief description of the datasets, their characteristics about missing information and parameters used for simulation. Then we display results obtained by two different methods like TLBO + RBFN and iTLBO + RBFN along with detail analysis.

4.1 Description of datasets and parameters

The datasets used in this work were obtained from the UCI machine learning repository [10]. Seven datasets have been chosen to validate the proposed method i.e., iTLBO + RBFN. The details about the seven datasets are given in Table 1. The algorithmic parameters like population size, number of iterations, etc are fixed based on empirical analysis as follows.

The size of the population is equal to 100, number of iterations fixed at 300, size of the neighborhood is restricted with 10% of population size, and the value of T_F has been adapted as per suggestions given in sub-section 3.3.1 with α value from (0, 1) and β value from (0, 1). The parameters of multi-layer perceptron (MLP) along with training algorithms and Simple Logistic are defined as prescribed in [3].

4.2 Results and analysis

The average results of the experiment obtained from 10 fold cross validation of 30 independent runs are given in Tables 2–7.

From Table 2 it is found that for 7 different datasets iTLBO + RBFN gives better accuracy than TLBO + RBFN, MLP, and Simple Logistic. To support the above results of TLBO + RBFN, statistical analysis based on the measures derived through confusion matrix is presented in Tables 3 and 4.

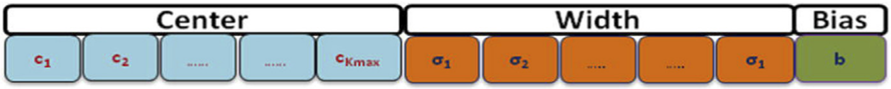


Figure 3. Structure of the Individual.

Table 1. Description of Datasets.

Dataset	# Instances	# Attributes	#Classes
Hepatitis	155	19	2
Housevotes	435	16	2
Mammographic	961	6	2
Horse Colic	368	27	2
Wisconsin	699	9	2
Diabetes	768	8	2
Post-operative	90	8	2

Table 2. Classification Accuracy of iTLBO + RBFN Using Least Square Imputation without Feature Selection.

Dataset	TLBO + RBFN	MLP	Simple Logistic	iTLBO + RBFN
Hepatitis	88.4615	88.4615	85.8974	92.3077
House-votes	96.789	94.4954	94.0367	98.1651
Mammographic	81.4969	76.5073	82.5364	83.9917
Horse Colic	81.345	80.4348	82.0652	82.0652
Wisconsin	95.1429	94.5714	92.8571	96.2857
Diabetes	71.0938	70.5729	70.8333	74.2188
Post-operative	66.6667	64.4444	64.4444	75.5556

From the Statistical analysis it can be observed that the calculated Kappa-values for TLBO + RBFN with feature selection are much better than TLBO + RBFN without feature selection.

Table 3. Further Details Analysis of TLBO + RBFN (Table 2).

Dataset	TP Rate	FP Rate	Precision	Recall	F-Measure	Kappa Statistics
Hepatitis	88.5	19	88.3	88.5	88.2	0.7194
House-votes	96.8	2	96.9	96.8	96.68	0.9318
Mammographic	81.5	19	81.6	81.5	81.4	0.6272
Horse Colic	82.6	23	82.5	82.6	82.3	0.6151
Wisconsin	95.1	4	95.2	95.1	95.2	0.9015
Diabetes	71.1	37.6	70.4	71.1	0.7	0.3525
Post-operative	66.7	66.7	44.4	66.7	53.3	0

Table 4. Further Detail Analysis of iTLBO + RBFN (Table 2).

Dataset	TP Rate	FP Rate	Precision	Recall	F-Measure	Kappa Statistics
Hepatitis	91.0	65.3	89.4	91.0	89.9	0.684
House-votes	98.6	1.7	98.6	98.6	98.6	0.9714
Mammographic	83.8	16.8	83.9	83.8	83.7	0.6728
Horse Colic	82.1	20.2	82.1	82.1	82.1	0.6163
Wisconsin	97.7	3.7	97.7	97.7	97.7	0.939
Diabetes	74.5	33.7	74.1	74.5	73.6	0.4292
Post-operative	75.6	75.6	57.1	75.6	65.0	0

Table 5. Classification Accuracy of iTLBO + RBFN Using Least Square Imputation with Feature Selection.

Dataset	No of Feature Removed	TLBO + RBFN	MLP	Simple Logistic	iTLBO + RBFN
Hepatitis	6	89.4615	89.7436	82.0513	93.5897
House-votes	5	98.6239	96.789	93.0876	99.0826
Mammographic	2	78.1705	81.4969	81.3546	83.7838
Horse Colic	7	85.8696	80.4348	81.0562	87.5
Wisconsin	3	95.1429	93.7104	92.8571	99.1429
Diabetes	2	73.4375	69.7129	70.7291	81.7708
Post-operative	2	66.6667	63.4144	62.5434	75.5556

Table 6. Further Detail Analysis of TLBO + RBFN (Table 5).

Dataset	TP Rate	FP Rate	Precision	Recall	F-Measure	Kappa Statistics
Hepatitis	89.7	78.3	86.7	89.7	87.8	0.1545
House-votes	98.6	13	98.6	98.6	98.6	0.9705
Mammographic	78.3	27.3	77.9	78.3	77.9	0.5218
Horse Colic	85.9	18	85.8	85.9	85.7	0.6917
Wisconsin	95.1	4	95.2	95.1	95.2	90.15
Diabetes	73.4	34.5	72.8	73.4	72.7	0.405
Post-operative	66.7	56.7	62.8	66.7	61.7	0.1176

Table 7. Further Detail Analysis of iTLBO + RBFN (Table 5).

Dataset	TP Rate	FP Rate	Precision	Recall	F-Measure	Kappa Statistics
Hepatitis	92.3	15.0	92.5	92.5	92.1	0.81
Housevotes	99.1	1.0	99.1	99.1	99.1	0.98
Mammographic	83.8	16.0	83.9	83.8	83.8	0.67
Horse Colic	87.5	17.1	87.5	87.5	87.3	0.72
Wisconsin	99.1	0.003	99.2	99.1	99.1	0.98
Diabetes	81.8	25.8	81.5	81.8	81.6	0.57
Post-operative	75.6	75.6	57.1	75.6	65.0	0.00

5. Conclusions

An integrated approach of iTLBO and RBFN has been proposed for making a classifier to classify unseen data by carefully considering the issues like missing values and dimensionality reduction. The approach undergoes three different phases before drawing any conclusions. In first phase, preprocessing task like missing value imputation is carried out by least square estimator. In second phase by Relief the relevant attributes are selected. Finally in the third phase a classifier is built by integrating iTLBO and RBFN. Determining the optimum key parametric values of RBFN, iTLBO is adopted. After careful training, the model was tested and it was noticed that in all datasets, iTLBO + RBFN is performing better than TLBO + RBFN in the case of complete dataset. Our bag of future research includes applications in big data and more parametric analysis of iTLBO in correspondence with the natural teaching-learning process.

References

[1] B. Twala, M.C. Jones, D.J. Hand, Good methods for coping with missing data in decision trees, Pattern Recogn. Lett. 29 (7) (2008) 950–956.

[2] T.H. Bø, B. Dysvik, I. Jonassen, LSimpute: accurate estimation of missing values in microarray data with least squares, Methods. Nucleic Acids Res. 32 (3), 2004.

[3] C.S.K. Dash, A. Saran, P. Sahoo, S. Dehuri, S.B. Cho, Design of self-adaptive and equilibrium differential evolution optimized radial basis function neural network classifier for imputed database, Pattern Recogn. Lett. 80 (2016) 76–83.

[4] K. Kira, L.A. Rendell, The feature selection problem: traditional methods and a new algorithm, AAAI 2 (1992) 129–134.

[5] R.J. Urbanowicz, M. Meeker, W. Lacava, R.S. Olson, J.H. Moore, Relief-based feature selection: introduction and review, J. Biomed. Inf., 2018.

[6] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (3) (2011) 303–315.

[7] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems, Inf. Sci. 183 (1) (2012) 1–15.

[8] C.S.K. Dash, A.K. Behera, S. Dehuri, S.B. Cho, Radial basis function neural networks: a topical state-of-the-art survey, Open Comput. Sci. 6 (2016) 33–63.

[9] R.V. Rao, V.J. Savsani, J. Balic, Teaching–learning-based optimization algorithm for unconstrained and constrained real-parameter, Optim. Prob. Eng. Optim. 44 (12) (2012) 1447–1462.

[10] A. Frank, A. Asuncion, UCI Machine Learning Repository, University of California, School of Information and Computer Science, Irvine, CA, 2010 http://archive.ics.uci.edu/ml.

[11] M. Woźniak, D. Połap, Object detection and recognition via clustered features, Neurocomputing 320 (2018) 76–84.

[12] N. Jankowski, Prototype-based kernels for extreme learning machines and radial basis function networks, International Conference on Artificial Intelligence and Soft Computing, 2018, pp. 70–75.

[13] F. Beritelli, G. Capizzi, G.L. Sciuto, C. Napoli, M. Woźniak, A novel training method to preserve generalization of RBPNN classifiers applied to ECG signals diagnosis, Neural Networks 108 (2018) 331–338.

[14] H.R. Baghaee, M. Mirsalim, G.B. Gharehpetan, H.A. Talebi, Nonlinear load sharing and voltage compensation of microgrids based on harmonic power-flow calculations using radial basis function neural networks, IEEE Syst. J. 12 (3) (2018) 2749–2759.

[15] M. Woźniak, D. Połap, Hybrid neuro-heuristic methodology for simulation and control of dynamic systems over time interval, Neural Networks 93 (2017) 45–56.

[16] L. Ortombina, F. Tinazzi, M. Zigliotto, Magnetic modeling of synchronous reluctance and internal permanent magnet motors using radial basis function networks, IEEE Trans. Ind. Electron. 65 (2) (2018) 1140–1148.

[17] S.S. Chouhan, A. Kaul, U.P. Singh, S. Jain, Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: an automatic approach towards plant pathology, IEEE Access 6 (2018) 8852–8863.

[18] H. de Leon-Delgado, R.J. Praga-Alejo, D.S. Gonzalez-Gonzalez, M. Cantú-Sifuentes, Multivariate statistical inference in a radial basis function neural network, Expert Syst. Appl. 93 (2018) 313–321.

[19] D. Granato, P. Putnik, D.B. Kovačević, J.S. Santos, V. Calado, R.S. Rocha, A. Pomerantsev, Trends in chemometrics: food authentication, microbiology, and effects of processing, Compr. Rev. Food Sci. Food Saf. 17 (3) (2018) 663–677.

[20] A.G.D. Cruz, R.S. Cadena, M.B.V.B. Alvaro, A.D.S. Sant'Ana, C.A.F.D. Oliveira, J.D.A.F. Faria, M.M.C. Ferreira, Assessing the use of different chemometric techniques to discriminate low-fat and full-fat yogurts, LWT-Food Sci. Technol. 50 (1) (2013) 210–214.

[21] J.A. Matera, A.G. Cruz, R.S.L. Raices, M.C. Silva, L.C. Nogueira, S.L. Quitério, C.C. Júnior, Discrimination of Brazilian artisanal and inspected pork sausages: application of unsupervised, linear and non-linear supervised chemometric methods, Food Res. Int. 64 (2014) 380–386.

[22] A.G. Da Cruz, E.H.M. Walter, R.S. Cadena, J.A.F. Faria, H.M.A. Bolini, A.F. Fileti, Monitoring the authenticity of low-fat yogurts by an artificial neural network, J. Dairy Sci. 92 (10) (2009) 4797–4804.

Corresponding author

Satchidananda Dehuri can be contacted at: satchi.lapa@gmail.com