Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 1523 – 1533
© TÜBİTAK
doi:10.3906/elk-1802-40

A novel hybrid teaching-learning-based optimization algorithm for the classification of data by using extreme learning machines

Ender SEVİNÇ[1,*], **Tansel DÖKEROĞLU**[2]

[1]Department of Computer Engineering, Faculty of Engineering, Turkish Aeronautical Association University, Ankara, Turkey

[2]Department of Computer Engineering, Faculty of Engineering, TED University, Ankara, Turkey

Received: 04.02.2018 • **Accepted/Published Online:** 11.02.2019 • **Final Version:** 22.03.2019

Abstract: Data classification is the process of organizing data by relevant categories. In this way, the data can be understood and used more efficiently by scientists. Numerous studies have been proposed in the literature for the problem of data classification. However, with recently introduced metaheuristics, it has continued to be riveting to revisit this classical problem and investigate the efficiency of new techniques. Teaching-learning-based optimization (TLBO) is a recent metaheuristic that has been reported to be very effective for combinatorial optimization problems. In this study, we propose a novel hybrid TLBO algorithm with extreme learning machines (ELM) for the solution of data classification problems. The proposed algorithm (TLBO-ELM) is tested on a set of UCI benchmark datasets. The performance of TLBO-ELM is observed to be competitive for both binary and multiclass data classification problems compared with state-of-the-art algorithms.

Key words: Teaching-learning-based optimization, extreme learning machines, metaheuristic, classification, feature selection

1. Introduction

Data classification is a big challenge for scientists when they need to extract any useful knowledge contained by the data and answer some important questions related to the patterns of data [1, 2]. Many data-mining and machine learning techniques have been proposed for the solution of this important problem. With the advent of the big data era, the problem has gained more importance due to the dirty and redundant data features that negatively impact the performance of decision systems (see Figure 1). Raw data (not preprocessed) may harm the accuracy level of data classification significantly [3]. Feature selection is a promising technique to make use of selected data where there exist large amounts of useless features [4]. Since the feature selection process is an NP-hard problem, it becomes an intractable process for datasets with many features. Therefore, metaheuristic approaches like evolutionary computation can be used as efficient tools to deal with this important problem [5].

A recent metaheuristic, teaching-learning-based optimization (TLBO), has been reported to be an efficient optimization tool that is inspired by the knowledge passing mechanisms of teachers and learners in a classroom [6]. It has been applied to several well-known combinatorial optimization problems, producing good results [7]. Hybrid algorithms that use a heuristic approach and a machine learning technique are efficient tools for the classification of data. However, due to the huge number of fitness calculations, the optimization process

*Correspondence: esevinc@thk.edu.tr

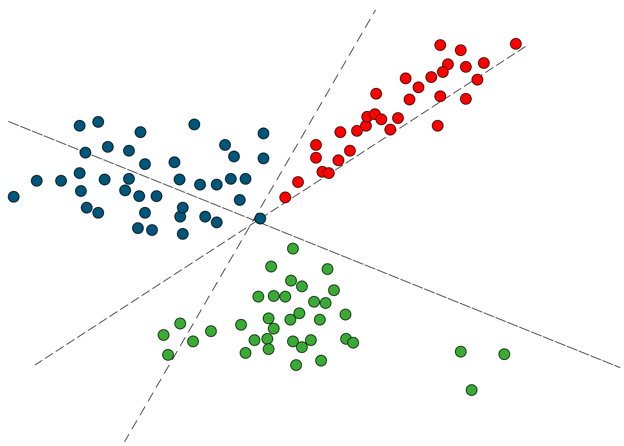

Figure 1. A picture of multiclass classification of data instances in a two-dimensional space.

of the classification can take a very long time with machine learning techniques that progress slowly. Hence, we propose a new hybrid TLBO algorithm that uses extreme learning machines (ELM), one of the fastest and most successful machine learning techniques in the literature [8–10]. ELM is an outstanding machine learning technique with a fast and accurate evaluation process. Therefore, ELM becomes a very suitable tool in a hybrid optimization processes that need to calculate several fitness values during the data classification process. With this property of ELM, the accuracy of the classification process can be improved significantly as is observed in our experiments while making comparisons with state-of-the-art algorithms in the literature.

We have carried out comprehensive experiments on UCI benchmark datasets to show the effectiveness of our proposed algorithm. During the experiments, we tune the parameters (number of hidden nodes) of the ELM and use these well-tuned parameters in all of our subsequent experiments, which is a significant issue that greatly affects the performance of our proposed algorithm. To the best of our knowledge, our study is the first single-objective TLBO algorithm that uses ELM to solve binary and multiclass data classification problems.

The rest our paper is organized as follows. Section 2 discusses the previous studies with TLBO, ELM, and data classification. In Section 3 we introduce our proposed algorithm, TLBO-ELM. Section 4 reports the observed results of our experiments. Finally, conclusions and future work are discussed in the last section.

2. Related work

In this section, we give information about the previous studies related to the data classification problem, TLBO, and ELM. There have been many studies in the feature selection area to date [9–11]. In a study by Kohavi and John, the relevance of features were supervised at the beginning and weak/strong relevant features were concluded in order to capture the intuition better [10]. Algorithms proposed for data classification can be categorized into two main types: filtering wrapper algorithms. The main distinction between these two types of algorithms is that filtering algorithms select the feature subset before the application of any classification process. By using statistical properties, the filtering approach eliminates less important features as it is applied in our algorithm.

The methods used are mainly beneficial with respect to an optimality rule and features are selected with respect to the specific learning algorithm. In a study by Kohavi and John, compound operators were used to apply a backward search, starting with the full set of features. Best-first search with compound operators was chosen to improve the methods, ID3, C4.5, and naive Bayes, in terms of accuracy and comprehensibility [10]. The study by Zexuan and Dash has fundamentally the same basis [11]. A filter method was developed and reported to be computationally more intensive than that of a wrapper, but wrapper methods generally outperform filter methods in terms of prediction accuracy. Stochastic, multiple-solution, optimal, and node pruning techniques are the most discussed techniques for feature selection problems [9]. In particular, this study used land images for the purpose of classification. Zexuan and Dash proposed a hybrid wrapper/filter feature selection algorithm by using a memetic framework [11]. Solution representation for a candidate feature subset was encoded as a chromosome. Results showed that the proposed method performs its search more efficiently and is capable of producing good classification accuracy with a small number of features simultaneously.

The method of Kohavi and John involves hill-climbing and a best-first search engine [10]. The problem solution technique is claimed to be a simulated annealing approach. Xue et al. presented a novel wrapper feature selection algorithm for classification problems [12]. The algorithm is a hybrid genetic algorithm with ELM (HGEFS). It uses evolutionary methods to wrap ELM to explore the optimum set of features to improve the final prediction accuracy. Kashef and Nezamabadi-Pour proposed a novel feature selection algorithm based on ant colony optimization (ABACO) [13]. The performance of the proposed ABACO was compared with the performance of the binary genetic algorithm (BGA), binary particle swarm optimization (BPSO), catfish BPSO, improved binary gravitational search algorithm (IBGSA), and some ACO-based algorithms for feature selection. Experiments yielded good accuracy results on UCI Machine Learning Repository datasets [13]. Unler and Murat developed a discrete particle swarm optimization (PSO) algorithm for the feature subset selection problem [14].

TLBO is a competitive metaheuristic with outstanding performance. It is reported to outperform some of the well-known metaheuristics regarding constrained benchmark functions, constrained mechanical design, and continuous nonlinear numerical optimization problems [15]. It was also applied to discrete optimization problems successfully in previous studies and therefore it attracted our interest and we decided to evaluate its performance on the feature selection problem in this study.

3. The proposed algorithm, TLBO-ELM

In this section of our study, we give details of our proposed algorithm, TLBO-ELM. The algorithm has two phases, the TLBO feature selection phase and the data classification phase with ELM technique (using the selected features). The TLBO-ELM algorithm uses the ELM technique for data classification. TLBO-ELM is a member of the class of filtering algorithms [8]. Initially, it selects subsets of features randomly, constructs learner individuals, and then calculates the fitness value of each individual in the classroom. New solutions are generated by using the classical crossover and mutation operators and this process continues until the termination condition is met. The flowchart of the TLBO-ELM algorithm is presented in Figure 2. The ELM phase classifies the data instances of the set with selected features that are sent by the TLBO process of the TLBO-ELM algorithm.

The TLBO phase uses the best individual of the classroom (population) as the teacher of students/learners. The teacher trains the learners and then the learners interact with each other to share the information they gain. This process goes on until the termination criterion is satisfied. A representation of a learner (individual in the classroom) structure of the TLBO-ELM algorithm is given in Figure 3. Crossover and mutation operators of the TLBO-ELM algorithm are presented in Figures 4 and 5, respectively.

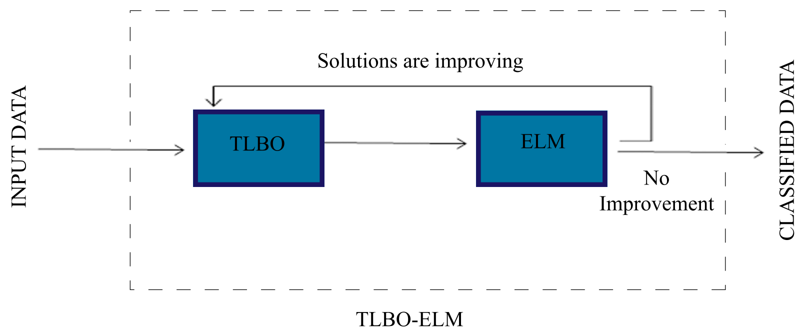

Figure 2. Flowchart of the TLBO-ELM algorithm.

1	1	0	0	0	1	0	0

Figure 3. Representation of a learner (individual in a classroom) of the TLBO-ELM algorithm. Selected features are represented with value one whereas the others are zero.

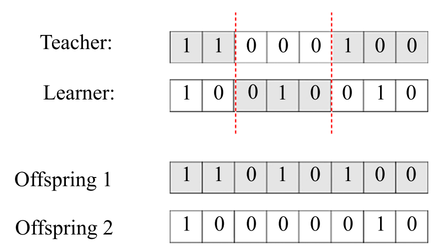

Figure 4. Crossover operator that produces two offspring from two selected individuals.

For the ELM phase of the algorithm, the output of a single-hidden-layer feedforward neural network (SLFN) having L hidden nodes is represented by Eq. (1). The learning parameters of hidden nodes are a_i and b_i and the weight connecting the ith hidden node to the output node is β_i. Function G(a_i, b_i, x) is the output of the ith hidden node with respect to input x.

$$f_L(x) = \sum_{i=1}^{L} \beta_i . G(a_i, b_i, x) \qquad x \epsilon \mathbf{R}^n, a_i, b_i \epsilon \mathbf{R} \tag{1}$$

Eq. (2) describes the activation function $G(a_i, b_i, x)$.

$$G(a_i, b_i, x_j) = g(a_i . x_j + b_i) = \mathbf{o}_j \qquad b_i \epsilon \mathbf{R}, j = 1, ..., N \tag{2}$$

In Eq. (2), $a_i.x$ denotes the inner product of vectors a_i and x, where both are elements of **R**. It is inferred that activation function G(x) can approximate these L samples with zero error, equal to $\sum_{j=1}^{L} \|\mathbf{o}_j - \mathbf{t}_j\| = 0$. This means that there exist β_i, a_i, and b_i in Eq. (3) such that:

$$\sum_{i=1}^{L} \beta_i . G(a_i . x_j + b_i) = \mathbf{t}_j \qquad j = 1, ..., N \tag{3}$$

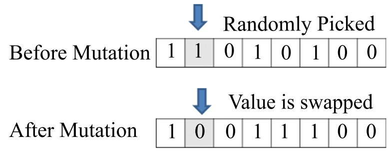

Figure 5. Mutation operator that swaps the genes of a chromosome.

N is the number of samples, i.e. inputs. We can rewrite this equation in another way, as shown in Eq. (4), for better understanding.

$$\mathbf{H}\beta = \mathbf{T} \tag{4}$$

Here,

$$\mathbf{H}(a_1, ..., a_L, b_1, ..., b_L, \mathbf{x}_1, ..., \mathbf{x}_N) = \begin{bmatrix} g(a_1.\mathbf{x}_1 + b_1) & \cdots & g(a_L.\mathbf{x}_1 + b_L) \\ \vdots & \ldots & \vdots \\ g(a_1.\mathbf{x}_N + b_1) & \cdots & g(a_L.\mathbf{x}_N + b_L) \end{bmatrix}_{NxL} \tag{5}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{Lxm} \quad and \quad \mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix}_{Nxm} \tag{6}$$

In Eq. (5), $\mathbf{H}$ is the output of the hidden layer matrix of the neural network. β^T is the transpose of a matrix or vector β in Eq. (6). $\mathbf{H}$ is called the hidden layer output matrix of the network, the ith column of $\mathbf{H}$ is the ith hidden node's output vector with respect to inputs $x_1, x_2, ..., x_N$, and the jth row of $\mathbf{H}$ is the output vector of the hidden layer with respect to input x_j. The number of hidden nodes is commonly less than the number of training data, which causes the aggravation of the error ratio. Under the constraint of minimum norm least squares, i.e. $min\|\beta\|$ and $min\|\mathbf{H}\beta - \mathbf{T}|$, a simple representation of Eq. (4) that was proven in studies [16–18] is presented in Eq. (7).

$$\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{7}$$

Here, $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse [19] of the hidden layer output matrix $\mathbf{H}$. The work in [18] further showed that $\mathbf{H}$ is full column rank with probability one when $L \leq N$ if the N training data are distinct. In real applications, the number of hidden nodes is usually less than the number of training data, $L < N$. Thus, $\hat{\beta}$ can be written as $\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T$, which is clearly presented in [3, 16–18].

The pseudocode of the TLBO-ELM algorithm is presented in Algorithm 1.

4. Experimental setup and evaluation of the results

All our experiments are carried out on a PC having i5 1.60 GHz 64-Bit CPU with 8 GB of RAM. The TLBO-ELM algorithm is developed by using the Java programming language and MATLAB (version 2015b). The parameter settings of the TLBO-ELM algorithm are presented in Table 1. All the results are the average values of 10 runs of tenfold cross-validation to lessen the impact of random factors. In the experiments, the dataset

Algorithm 1: TLBO-ELM algorithm.

Input: n is the number learners in the classroom, r is the rate of elitism
Output: solution instance X

//Initialization
Generate n members of the classroom randomly;
for $\boldsymbol{i}$ = *1 to* $\boldsymbol{n}$ **do**
    Calculate hidden-layer output matrix $\mathbf{H}$;
    Calculate (β and $\mathbf{T}$);
    Evaluate $\mathbf{H}^{\dagger}$, i.e. the Moore–Penrose generalized inverse of matrix H;
    Evaluate the fitness of each instance i;
Sort the learners (individuals) in the classroom w.r.t. the fitness values;
while *(termination criterion is not met)* **do**
    Train learners with teacher (the best individual in the classroom);
    Train individuals with the other learners in the classroom;
    Generate new individuals using crossover and mutation operators;
    for $\boldsymbol{i}$ = *1 to* $\boldsymbol{n/2}$ **do**
        Calculate hidden-layer output matrix $\mathbf{H}$;
        Calculate (β and $\mathbf{T}$);
        Evaluate $\mathbf{H}^{\dagger}$, i.e. the Moore–Penrose generalized inverse of matrix $\mathbf{H}$;
        Evaluate the fitness of new instance $\mathbf{i}$;
    Truncate the worst $\boldsymbol{n/2}$ individuals and add the newly found $\boldsymbol{n/2}$ instances;
    Sort the solutions in the pool w.r.t. fitness values;
X = select the best learner in the classroom;
return X;

is first partitioned into 10 equal size sets and 9 out of 10 subsets are used for training while the final one is in turn used as the test dataset. The mean of the test dataset is used as our accuracy value.

Table 1. Parameters of the TLBO-ELM algorithm.

Parameter	Value
# learners	50
Convergence ratio	95%
Crossover type	Truncate, 2-point
Truncate ratio	50%
Crossover ratio	0.6 (60%)
Mutation ratio	0.02 (2%)

Datasets are selected to provide a fair comparison with other studies in the literature [10, 12, 13]. There are 11 datasets (see Table 2 for details). The datasets have a wide range of features ranging from 4 to 279.

The number of hidden neurons of the SLFN is decided after some optimization experiments. This value is selected to be between 30 and 60. Table 3 gives the values of hidden neurons used for the datasets. We need to use different numbers of hidden neurons to obtain better accuracy levels for each dataset.

Table 2. Descriptive statistics of the selected datasets.

Dataset	ID	# instances	# features	# classes
Vehicle	VEH	846	18	4
WDBC	WDB	569	32	2
Ionosphere	ION	351	34	2
Sonar	SON	208	60	2
Musk	MUS	476	168	2
Iris	IRI	150	4	3
Spambase	SPM	4601	57	2
Waveform	WAV	5000	21	3
Wisconsin B.C.(Or.)	WIS	699	10	2
Pima-Indian Diabetes	PID	768	8	2
Arrythmia	ART	452	279	16

Table 3. Optimized number of hidden neurons for each dataset.

ID	Instance no.	# hidden neurons
VEH	846	60
WDB	569	38
ION	351	30
SON	208	30
MUS	168	30
IRI	150	30
SPM	4601	60
WAV	5000	38
WIS	699	47
PID	768	51
ART	452	44

In the first step of our experiments, we decide whether we are going to have any improvement in the accuracy of results with selected features instead of working with all the features of a dataset. Table 4 gives details of our experiments in terms of accuracy improvements when a subset of features is selected intelligently instead of working with all features. Improvements ranging from 3.82% to 41.6% are observed during the experiments. This shows us that working with related features can significantly improve the accuracy level of the data classification results. It is observed that it gets harder to achieve a higher fitness value as the number of attributes increases. Datasets with small numbers of attributes produce higher accuracy levels with selected features.

Table 5 shows the features that are selected by the TLBO-ELM algorithm for some of the datasets. The last column shows the selected features of the input data. For the PID dataset, if we consider the selected 3 attributes as reported in Table 5 rather than the whole set, we reach 76.30% accuracy level. If we check Table 4, where the same process is performed by using all features, we can observe that the accuracy value is only 52.98%. It is possible to improve the accuracy level by 23.32% by selecting the features given in Table 5.

Table 4. Comparison of accuracy values with all feature sets and selected features.

ID	All features	Selected features	Improvement(%)
VEH	25.45	66.20	40.75
WDB	55.24	96.84	41.6
ION	79.06	92.60	13.54
SON	67.03	82.35	15.32
MUS	55.81	66.38	10.57
IRI	96.00	98.67	2.67
SPM	59.66	90.56	30.9
WAV	54.17	80.66	26.49
WIS	93.84	97.66	3.82
PID	52.98	76.30	23.32
ART	60.44	68.10	12.6

Table 5. The selected features by the TLBO-ELM algorithm. The reported set of features provides the best results obtained by the TLBO-ELM algorithm.

ID	Accuracy	Selected features
VEH	66.20	0-1-0-0-0-1-0-0-1-0-0-0-0-0-0-0-0-0
WDB	96.84	1-0-0-0-1-1-0-1-1-1-1-0-0-0-1-1-0-1-0-1-0-0-0-0-1-0-1-1-1-0
ION	92.60	0-0-0-0-0-0-1-1-0-1-0-0-1-0-0-0-0-0-0-0-0-0-1-0-0-0-0-0-0-0-0-0-0-0
SON	82.35	1-0-0-1-0-0-1-0-1-1-1-0-0-0-1-0-0-1-1-1-1-0-1-0-0-1-0-1-1-1-0-0-1-0-0-0-1-1-0-1-0-1-0-0-0-1-1-0-1-1-1-0-0-0-0-1-0-1-0-0
MUS	66.38	1-0-1-1-0-1-1-1-1-1-0-1-0-1-0-0-1-1-1-1-0-0-1-0-0-1-1-0-0-0-0-0-0-0-1-0-1-0-0-0-0-0-1-1-1-0-0-0-1-1-1-1-0-1-1-0-1-1- 0-0-0-1-1-1-0-0-0-0-1-0-0-1-1-0-0-1-0-1-1-0-1-0-0-0-1-0-1-0-0-0-0-1-0-0-1-1-1-0-0-1-1-1-0-0-0-0-1-0-1-0-1-1-1-0-1-0-0-0- 1-1-0-1-1-1-0-0-0-1-0-1-0-0-1-0-1-0-0-0-1-1-0-0-1-0-1-0-0-1-1-0-0-1-0-1-0-0-0-0-0-1-0-1-1-0-0
IRI	98.67	1-1-1-1
SPM	90.56	0-0-0-0-1-0-1-0-1-0-0-0-0-0-0-1-0-0-0-0-0-0-1-1-1-1-1-0-1-1-1-0-0-1-0-0-0-0-1-0-0-1-0-1-1-1-1-0-0-0-1-1-1-0-0-0-0
WAV	80.66	0-0-0-0-0-1-0-1-0-1-1-0-0-0-1-1-0-0-0-0-0
WIS	97.66	1-1-1-1-0-1-1-0-0
PID	76.30	1-1-0-0-0-0-1-0

4.1. Comparison with state-of-the-art algorithms

In this section of our study, we compare our TLBO-ELM algorithm with state-of-the-art algorithms in literature (namely the HGEFS [12], ABACO [13], ACOFS [13], attribute bagging (AB) [20], multiview AdaBoost (MVA) [21], random subspacing ensemble (RSE) [22], CFS-SFS [23], and C4.5 [24] algorithms). Table 6 gives the comparison of our study with state-of-the-art algorithms in terms of accuracy (these were the only results we could obtain from the previous studies related to our proposed algorithm). The bold values are the best results obtained by the algorithms. The TLBO-ELM algorithm is competitive with other wrapper studies. It produces the best results for five of the compared 11 datasets. Two datasets are observed to be very close to the reported best values of related state-of-the-art algorithms [12, 13].

4.2. Execution times of the TLBO-ELM algorithm

The execution time of the TLBO-ELM algorithm is compared with the same kind of population-based approaches. It is observed from the results of the previous studies [16–18] that the TLBO-ELM is a fast polynomial time algorithm (see Table 7). HGEFS [12], PSO-SVM [25], and GA-ELM [26] are the algorithms that we make comparisons to on the Arrhythmia dataset (ART) with 279 features and 452 instances. Since we integrate TLBO with ELM, our algorithm reaches/converges the results quickly (the accuracy result of the ART dataset is 68.10%). The only parameter that can affect our execution time is the stopping criterion, which can be decided by the user. The parameter settings that we provide for the TLBO-ELM algorithm work well for the classification problem (see Table 1). The TLBO-ELM algorithm can be reported to be one of the fastest algorithms in its category due to the quick evaluation process of the ELM technique.

Table 6. Comparison of accuracy values with state-of-the-art algorithms.

ID	AB [20]	MVA [21]	RSE [22]	CFS-SFS [23]	C4.5 [24]	HGEFS [12]	ABACO [13]	ACOFS [13]	TLBO-ELM
VEH	80.95	81.20	77.32	69.17	73.64	**82.02**	75.3	74.9	66.20
WDB	95.14	95.73	94.84	95.80	93.14	**97.10**	-	-	96.84
ION	89.54	90.14	89.01	89.06	91.16	91.33	-	-	**92.60**
SON	80.83	80.17	79.50	78.75	71.15	**83.00**	-	-	82.35
MUS	85.27	85.63	84.73	79.55	84.87	**88.13**	-	-	66.38
IRI	-	-	-	-	-	-	97.4	97.7	**98.67**
SPM	-	-	-	-	-	-	92.1	**92.2**	90.56
WAV	-	-	-	-	-	-	79.5	79.7	**80.66**
WIS	-	-	-	-	-	-	97.6	97.4	**97.66**
PID	-	-	-	-	-	-	-	-	**76.30**
ART	63.75	64.58	63.11	63.00	63.27	**68.30**	-	-	68.10

Table 7. Comparison of execution times (s).

Algorithm	Execution (s)
HGEFS [12]	4936.7
PSO-SVM [25]	50493.1
GA-ELM [26]	4373.8
TLBO-ELM	4448.4

5. Conclusions and future work

In this study, we propose a novel hybrid algorithm for the data classification problem. To the best of our knowledge, the TLBO-ELM is the first algorithm designed by combining these two techniques for the data classification problem. We combine the fast behavior of ELM for the first time with a recent metaheuristic algorithm, TLBO, and produce a robust hybrid metaheuristic algorithm. Even with datasets that have a large number of features (like the SON dataset), the prediction accuracy level of the TLBO-ELM algorithm is among the best-performing algorithms with a prediction accuracy rate that is above 82.0%. With the datasets that we perform our experiments on, the TLBO-ELM algorithm can be reported as one of the top two algorithms in the literature. The TLBO-ELM uses (near-)optimal parameter settings for ELM. These parameter settings are reported in our study. The prediction accuracy performance of the TLBO-ELM algorithm is promising and competitive with state-of-the-art algorithms. It leads to the conclusion that, in the future, more specialized data classification problems can be solved by using this new algorithm. The multiobjective version of this algorithm might be another subject of research. Parallel execution of TLBO-ELM with advanced GPU architectures can reduce the execution time and increase the exploration/exploitation capability of the algorithm significantly.

References

[1] Mirhosseini M. A clustering approach using a combination of gravitational search algorithm and k-harmonic means and its application in text document clustering. Turkish Journal of Electrical Engineering & Computer Sciences 2017; 25 (2): 1251-1262. doi: 10.3906/elk-1508-31

[2] Nazar NB, Senthilkumar R. An online approach for feature selection for classification in big data. Turkish Journal of Electrical Engineering & Computer Sciences 2017; 25 (1): 163-171. doi:10.3906/elk-1501-98

[3] Feng G, Qian Z, Zhang X. Evolutionary selection extreme learning machine optimization for regression. Soft Computing 2012; 16 (9): 1485-1491. doi:10.1007/s00500-012-0823-7

[4] Guyon I, Elisseeff A. An introduction to variable and feature selection. Journal of Machine Learning Research 2003; 3: 1157-1182. doi: 10.1162/153244303322753616

[5] Deniz A, Kiziloz HE, Dokeroglu T, Cosar A. Robust multiobjective evolutionary feature subset selection algorithm for binary classification using machine learning techniques. Neurocomputing 2017; 241: 128-146. doi: 10.1016/j.neucom.2017.02.033

[6] Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Computer-Aided Design 2011; 43 (3): 303-315. doi: 10.1016/j.cad.2010.12.015

[7] Dokeroglu T. Hybrid teaching–learning-based optimization algorithms for the quadratic assignment problem. Computers & Industrial Engineering 2015; 85: 86-101. doi: 10.1016/j.cie.2015.03.001

[8] Lan Y, Soh YC, Huang GB. Two-stage extreme learning machine for regression. Neurocomputing 2010; 73: 3028–3038. doi: 10.1016/j.neucom.2010.07.012

[9] Jain A, Zongker D. Feature selection: evaluation, application, and small sample performance. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997; 19: 153–158. doi: 10.1109/34.574797

[10] Kohavi R, John GH. Wrappers for feature subset selection. Artificial Intelligence 1997; 1-2: 273–324. doi: 10.1016/S0004-3702(97)00043-X

[11] Zhu Z, Ong YS, Dash M. Wrapper-filter feature selection algorithm using a memetic framework. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 2007; 37: 70–76. doi: 10.1109/TSMCB.2006.883267

[12] Xue X, Yao M, Wu Z. A novel ensemble-based wrapper method for feature selection using extreme learning machine and genetic algorithm. Knowledge and Information Systems 2018; 57 (2): 389-412. doi: 10.1007/s10115-017-1131-4

[13] Kashef S, Nezamabadi-Pour H. An advanced ACO algorithm for feature subset selection. Neurocomputing 2015; 147: 271-279. doi: 10.1016/j.neucom.2014.06.067

[14] Unler A, Murat A. A discrete particle swarm optimization method for feature selection in binary classification problems. European Journal of Operational Research 2010; 206 (3): 528-539. doi: 10.1016/j.ejor.2010.02.032

[15] Crepinsek M, Liu SH, Mernik L. A note on teaching–learning-based optimization algorithm. Information Sciences 2012; 212: 79-93. doi: 10.1016/j.ins.2012.05.009

[16] Huang GB, Zhou H, Ding X. Extreme learning machine for regression and multiclass classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 2012; 42: 513–529. doi: 10.1109/TSMCB.2011.2168604

[17] Huang GB, Ding X, Zhou H. Optimization method based extreme learning machine for classification. Neurocomputing 2010; 74: 757-754. doi: 10.1016/j.neucom.2010.02.019

[18] Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing 2006; 70 (1): 489–501. 10.1016/j.neucom.2005.12.126

[19] Penrose R. A generalized inverse for matrices. Mathematical Proceedings of the Cambridge Philosophical Society 1955; 51: 406-413. doi: 10.1017/S0305004100030401

[20] Bryll R, Gutierrez-Osuna R, Quek F. Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. Pattern Recognition 2003; 36 (6): 1291–1302. doi: 10.1016/S0031-3203(02)00121-8

[21] Sun S. A survey of multi-view machine learning. Neural Computing and Applications 2013; 23 (7-8): 2031-2038. doi: 10.1007/s00521-013-1362-6

[22] Ho TK. The random subspace method for constructing decision forests. IEEE Transactions Pattern Analysis and Machine Intelligence 1998; 20 (8): 832–844. doi: 10.1109/34.709601

[23] Hall MA. Correlation-based feature selection for machine learning. PhD, University of Waikato, Hamilton, New Zealand, 1999.

[24] Quinlan JR. Improved use of continuous attributes in C4.5. Journal of Artificial Intelligence Research 1996; 4: 77-90. doi: 10.1613/jair.279

[25] Garcia-Nieto J, Alba E, Jourdan L, Talbi E. Sensitivity and specificity based multiobjective approach for feature selection: application to cancer diagnosis. Information Processing Letters 2009; 109 (16): 887-896. doi: 10.1016/j.ipl.2009.03.029

[26] Alexandre E, Cuadra L, Salcedo-Sanz S, Pastor-Sánchez A, Casanova-Mateo C. Hybridizing extreme learning machines and genetic algorithms to select acoustic features in vehicle classification applications. Neurocomputing 2015; 152: 58–68. doi: 10.1016/j.neucom.2014.11.019