Indonesian Journal of Electrical Engineering and Computer Science
Vol. 15, No. 3, September 2019, pp. 1382~1391
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v15.i3.pp1382-1391

Dimentionality reduction based on binary cooperative particle swarm optimization

Sharifah Sakinah Syed Ahmad[1], Ezzatul Farhain Azmi[2], Fauziah Kasmin[3], Zuraini Othman[4]
[1,3,4]Department of Intelligent Computing and Analytics, Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia
[2]Department of Mechanical Engineering Technology, Faculty of Mechanical and Manufacturing Engineering Technology, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia

Article Info

Article history:

Received Dec 9, 2018
Revised Mar 6, 2019
Accepted Apr 22, 2019

Keywords:

Binary cooperative particle swarm optimization
Feature selection
Instances selection
K-nearest neighbor

ABSTRACT

Even though there are numerous classifiers algorithms that are more complex, k-Nearest Neighbour (k-NN) is regarded as one amongst the most successful approaches to solve real-world issues. The classification process's effectiveness relies on the training set's data. However, when k-NN classifier is applied to a real world, various issues could arise; for instance, they are considered to be computationally expensive as the complete training set needs to be stored in the computer for classification of the unseen data. Also, intolerance of k-NN classifier towards irrelevant features can be seen. Conversely, imbalance in the training data could occur wherein considerably larger numbers of data could be seen with some classes versus other classes. Thus, selected training data are employed to improve the effectiveness of k-NN classifier when dealing with large datasets. In this research work, a substitute method is present to enhance data selection by simultaneously clubbing the feature selection as well as instances selection pertaining to k-NN classifier by employing Cooperative Binary Particle Swarm Optimisation (CBPSO). This method can also address the constraint of employing the k-nearest neighbour classifier, particularly when handling high dimensional and imbalance data. A comparison study was performed to demonstrate the performance of our approach by employing 20 real world datasets taken from the UCI Machine Learning Repository. The corresponding table of the classification rate demonstrates the algorithm's performance. The experimental outcomes exhibit the efficacy of our proposed approach.

Corresponding Author:

Sharifah Sakinah Syed Ahmad,
Department of Intelligent Computing and Analytics,
Faculty of Information & Communication Technology,
Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
Email: sakinah@utem.edu.my

1. INTRODUCTION

The k-Nearest Neighbour (k-NN) classifier is regarded as a popular non-parametric classification method. It employs a simple supervised learning concept to categorise unseen instances by identifying previous observed instance that is closest, and takes a note of its class to predict the class for unseen data [1-2]. This approach has been used in numerous application domains like classification and machine learning [3-8]. Its popularity stems from its easily implementation, as well as being conceptually straightforward versus other supervised learning methods. Not many models concentrated on the training phase pertaining to non-parametric classifier. The classification process's effectiveness relies on the instances pertaining to the training set. Thus, the quality of the training set determines the classification rate [9-10]. Furthermore, using the k-NN classifier method to solve real world problem is associated with various problems like being regarded as computationally expensive classifiers as the complete training set needs to be stored in the computer for classification of unseen data. Also, k-NN classifier seemed to be intolerant towards irrelevant features. Conversely, the training data could be imbalanced in numerous real-world classification problems, in which extremely large number of data could be found in some classes versus classes [11-12]. Limitation in data collection process as well as rarity of cases or privacy issues could cause imbalance in data. This issue has a major impact on the current classification method's performance, particularly k-NN classifier.

When dealing with massive data amounts, use of certain selection mechanism is necessary. Computational complexity and classification rate pertaining to the classifier can be enhanced by choosing proper feature as well as instances in the training dataset. Furthermore in some cases, noisy or redundant data could be associated with the training dataset [13]. Thus, by employing the features as well as instances selection, the training data form the subset pertaining to the most useful set of input data. As per Brighton and Mellish, the instances that form the structure for classes could vary considerably, and thus the instances selection algorithm may show great performance only for one problem and not for other cases [14]. Thus, it was suggested that some insight regarding the structure of the classes is required for instances selection. However, acquiring such insight was very difficult or structure could become unavailable, particularly for the real-world problems that include numerous variables along with complex class boundaries. In this, employing the bio-inspired optimisation method offers advantages in decreasing the limitation pertaining to the existing instances selection algorithm. Any form of search space, classes and boundaries amongst classes is not considered for the applications pertaining to bio-inspired algorithm. The selection of instances is steered by each solution's ability to resolve a given problem [15-16].

In the literature, the subset of instances is chosen for most of the work related to enhancing the k-NN classifier. For instance, in [17], an instances selection algorithm was put forward by the author with regards to the nearest neighbour rule by integrating three strategies to select evolutionary instance. This method was aimed at identifying a subset in a way that each member pertaining to the original dataset would be closer to a subset member for the same class when compared with the subset member of a different class. In [18], an enhancement to this method was put forward, also known as the Selective Sampling for Nearest Neighbour, wherein each member pertaining to the original dataset needs to be nearer to a dataset's member for the same class versus any other member of the original dataset for a different class. Another well-known technique using the same approach is presented in [19] and is referred as a divide-and-conquer recursive approach.

Cano et al. wrote a comprehensive review regarding the main instance selection algorithm that was derived from the evolutionary algorithm. In this paper, an empirical performance study has been conducted to make a comparison of the four key evolutionary methods along with classical instances selection [20]. Based on the findings, these evolutionary methods were seen to give better performance than classical algorithms in terms of both data reduction and accuracy. However, a major issue of scaling up the algorithm is faced when employing an evolutionary algorithm for instances as well as for feature selection algorithm. With rise in the number of instances and feature, there is a considerable increase in the time required for the evolutionary algorithm to get a good solution. Cano et al. [20] have put forward a stratified approach to enhance the limitation of employing standard evolutionary algorithm. Also, Czarnowski [21] put forward an agent-based population that can be applied for distributed prototype selection. A divide-and-conquer method was proposed by Andrews and Fox [22] to deal with large datasets, which follow the same philosophy of stratification, as well as replacing the random sampling with the help of a clustering approach. A different approach to enhance evolutionary strategy in instances selection has been described in [23-24].

In this research work, we put forward an alternative method for performance enhancement of the k-NN classifier by concurrently choosing the best subset of feature as well as instances pertaining to the training data. Cooperative Binary PSO was employed to guide the combination of feature and data selection process. Though it is simple to use the cooperative version of binary PSO, it is also an efficient method simultaneously search for the best subset of feature as well as instances. Here, the candidate solution (particles) is segmented to various sub-components known as sub-swarms. The first component involves handling feature selection, while the rest handle instances selection. The selection process pertaining to both feature and instances is enhanced with the cooperative behaviours amongst sub-swarms. The feature selection is done directly from the candidate solution's first sub-swarm. The search space is divided by classification label presented by the original dataset for the selection of instances. Thus, each classification group is guided by one sub-swarm. Finally, to identify the best solution, all sub-swarms work in a cooperative manner with each other.

The classifier's performance is dependent on the classification rate. Thus, our performance index has been derived from the classification rate of k-NN classifier. Our framework also concentrates on addressing the limitation pertaining to the k-NN classifier when handling massive and imbalanced dataset. Similarly, Binary Cooperative PSO is used to identify the best data subset so that it can be employed in training the k-NN classifier.

This paper is structured as follows: Section 2 introduces the recommended method for assimilation of feature and instances selection based on Binary Coperative PSO: Section 3 presents the experimental results by using 20 datasets containing different domain problems from the Machine Learning Database Repository and StaLIB. Section 5 offers the conclusion and future recommendations.

2. THE PROPOSED METHODOLOGY

In this section, a detailed description of our proposed framework, Binary Cooperative PSO Data Selection, has been provided, which is derived from the Binary PSO and cooperative approach. Our key contribution includes enhancement of data selection by employing the binary cooperative PSO approach [25]. This made the selection more efficient and robust. Moreover, the framework helps overcome the limitation associated with instances selection algorithm, particularly when handling the imbalanced dataset.

In this research work, we used the Binary Cooperative Particle Swarm Optimisation (BCPSO) method to look out for the best subset of instances and feature. The search space pertaining to the BCPSO method has been derived from the classification label provided in the dataset. For instance, in the Iris dataset, three classification groups are present; thus, the total number of sub-swarms to identify the best subset is four. In this framework, four different search spaces were involved and were solved individually by employing the BCPSO method. The key goal of employing cooperative version of PSO is to efficiently handle the dimensionality pertaining to the search space, which becomes a grave issue when there is a large dimensionality pertaining to the feature space and instance. This dimensionality poses as a significant impediment that negatively affects the standard PSO's effectiveness. The PSO's cooperative version can be regarded as a parallel search pertaining to the optimal subset of instances and feature. The candidate solution vector is segmented into components known as sub-swarm to achieve the cooperative strategy, where each sub-swarm signifies a small portion of the overall optimisation processes. With this, the concept of divide and conquer was employed to address the optimisation issue to make the process faster and more efficient. The key attributes of the proposed approach will be presented in the following sub section.

2.1. The PSO–based Representation of the Search Space

A data subset and a subset of the features need to be chosen for reduction of data as well as feature spaces. Thus, the problem can be said to be combinatorial with regards to its nature. Here, PSO is employed for the formation of a subset of integers that are indexes pertaining to the data or features that allow formation of F' D'. For example, D' is characterised by a set of indexes {i1, i2, …, iM'} that are a subset of integers {1, 2, …, M}. From the standpoint of PSO, the formation of the particle is in the form of a string of binary numbers with length n+M, effectively, while a hypercube {0,1}n+M is the search space. The features are denoted by the first substring of length n, while the second one (possessing M entries) is employed for optimisation of the subset of instances. Decoding of the particle is done as follows. A binary number is associated with each element's value in the particle, which could be 0 and 1. To establish a relation between the particle representation and feature and instances selection, a value of 1 represents that features or instances have been selected and vice versa.

For instance, when the first sub-swarm is presented with a list of features:

F= {F1, F2, F3, F4, F5} and n = 5, a sub-swarm may look like:

X(1)= { 0, 0, 1, 1, 1 },
X(2)= { 1, 0, 1, 0, 1 },
X(3)= { 0, 0, 1, 0, 1 }

In the cooperative PSO, a more sophisticated approach is used to realise the formation of the search space. The cooperative facet mostly includes exchanging information regarding the best positions identified by the various sub-swarms. Next, we recommend a new cooperative PSO (CPSO) algorithm for the data and feature selection procedure. The choice of the number of cooperating swarms is vital as it would impact the cooperative PSO model's performance. Sub-swarm 1 signifies the features' column, and sub-swarm 2 onwards depicts the instances' row of the specific data set based on the class stated.

Figure 1 shows the depiction of every sub-swarm. The particle's length in the first sub-swarm equals the total quantity of features in the dataset. Then for the remaining sub-swarms, the length is dependent on the classification label's total number of sample. The information regarding the global best position (PGB) is shared across all sub-swarms to achieve cooperative search between one sub-swarm and another. In this, with this algorithm, two steps forward can be taken since the candidate solution comes via the best position pertaining to all sub-swarm with the exception of the current sub-swarm that is being assessed. Thus, the algorithm will spend only limited time looking for the best subset of instances or feature which have little impact on the solution in general. At a considerably faster rate, converging of each swarm to the solution is achieved when compared with the standard PSO's rate of convergence.

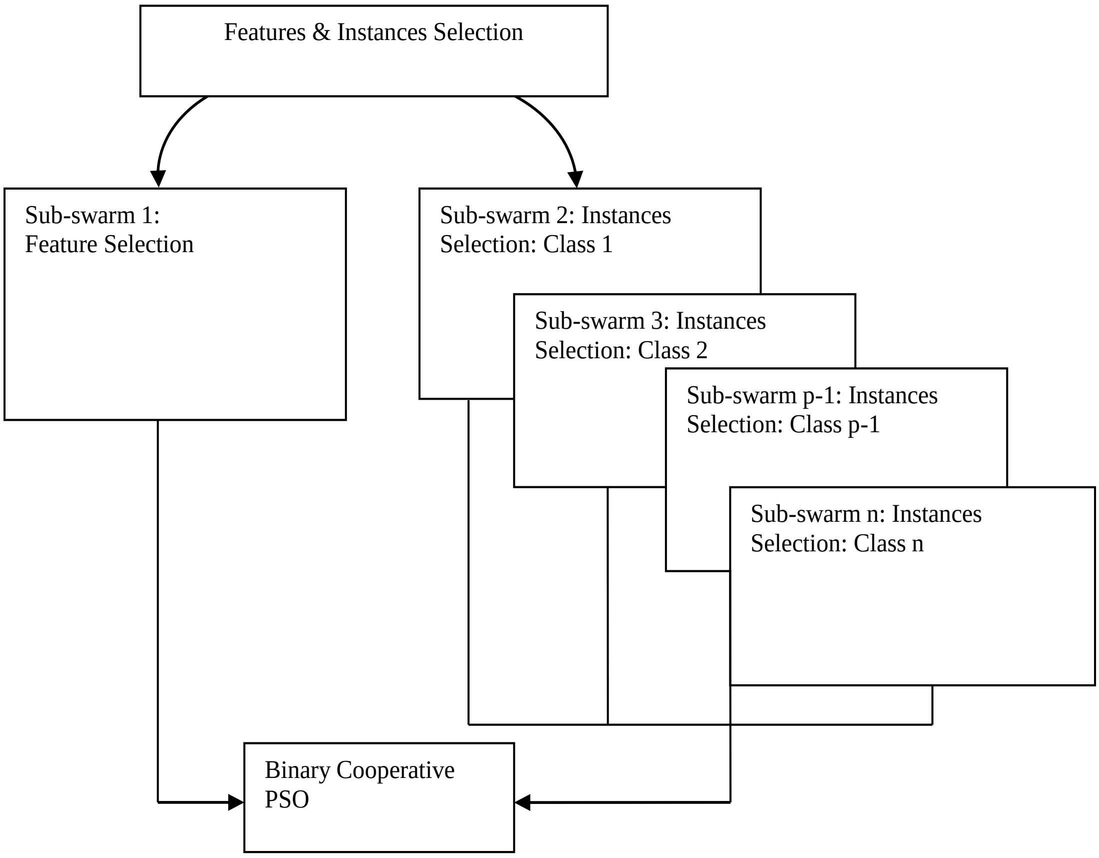


Figure 1. The particle scheme of BCPSO for feature and instances selection

Figure 2 depicts the Binary Cooperative PSO (BCPSO) pseudo code which we execute in the optimisation procedure. First, the BCPSO is split into m subspaces known as sub-swarms. The first sub-swarm signifies the features search space, and the remaining are for instances search space. Pj (xi) indicates the position of particle i of sub-swarms j. Each sub-swarm's global best can be represented as Pj(GB), while the local best as Pj(LBi). The cooperation amongst the sub-swarms is used in the function C(j,k), which returns the formed m-dimensional vector via concantenating of the entire global best vector across all sub-swarms, excluding the current position j. Here, the jth component is referred as k, which signifies any particle's position from sub-swarm Pj.

Initialize m one-dimensional Binary Cooperative
PSO: P_j, $j \in [1, \ldots, m]$
Create
$\mathbf{C}(j,k)$=[$P_1(_{GB})$,$P_2(_{GB})$,..., $P_{j-1}(_{GB})$, k , $P_{j+1}(_{GB})$, ..., $P_m(_{GB})$]
While stop criteria not met **do**
 for each sub-swarm $j \in [1, \ldots, m]$ **do**
 for each particle $i \in [1, \ldots, s]$ **do**
  if fitness($\mathbf{C}$(j, $P_j(x_i)$)) > fitness($\mathbf{C}$(j, $P_j(_{LBi})$))
  then $P_j(_{LBi}) = P_j(x_i)$
  if fitness($\mathbf{C}$(j, $P_j(_{LBi})$)) > fitness($\mathbf{C}$(j, $P_j(_{GB})$))
  then $P_j(_{GB}) = P_j(_{lbi})$
 end for
 for each P_j do
 $v_{i,j}(t+1) = w.v_{i,j}(t)+c_1.r_{1,i}(t)[P_{LBi,j}(t) - x_{i,j}(t)]$
  $+c_2.r_{2,i}(t)[P_{BGi,j}(t) - x_{i,j}(t)]$
 If rand() < $S(v_{i,j}(t+1))$ Then $x_{i,j}(t+1) = 1$
 Else $x_{i,j}(t+1) = 0$
 end for
 end for
end while

Figure 2. Pseudo code for Cooperative PSO

2.2. Proposed Integration of Feature and Instances Selection Via Binary Cooperative PSO

Figure 3 depicts the framework of the instances and feature selection by deploying the Binary Cooperative PSO (FIS_BCPSO). The framework can be split into three key parts and can be elucidated:

Part 1: A key tool to look out for the best instances to solve the classification issue is the reduction process via BCPSO. By deploying the cooperative technique, we can decrease the search space's size for the PSO methodology. Here, each class is placed in one sub-swarm. The total number of classification label defines the total number of sub-swarms. For instance, the number of sub-swarm pertaining to iris dataset is 3. Thus, by splitting the search space into its own classification group, we can decrease the algorithm's complexity and enhance the computational time, particularly for the large dataset.

Part 2: k-Nearest Neighbour Classifier is a much simpler and efficient technique in comparison to other learning methods. Then, as training data for k-NN classifier, the selected data are employed, while the data that were not selected are used as test data.

Part 3: The generation of new subset of instances is halted until stop condition is met by the algorithm. In the end, the subset of instances with the greatest precision will be chosen as the most excellent subset of instances.

The FISBCPSO employs the fitness function that concentrated on the key objective of enhancing the classification rate. There are two criteria to be taken into account in the optimisation procedure. The first pertains to the choice of the features, and the second is regarding the choice of the instances. F represents the sub-swarm pertaining to feature selection, I is the sub-swarms signifying instances selection and N denotes the training set's number of instances.

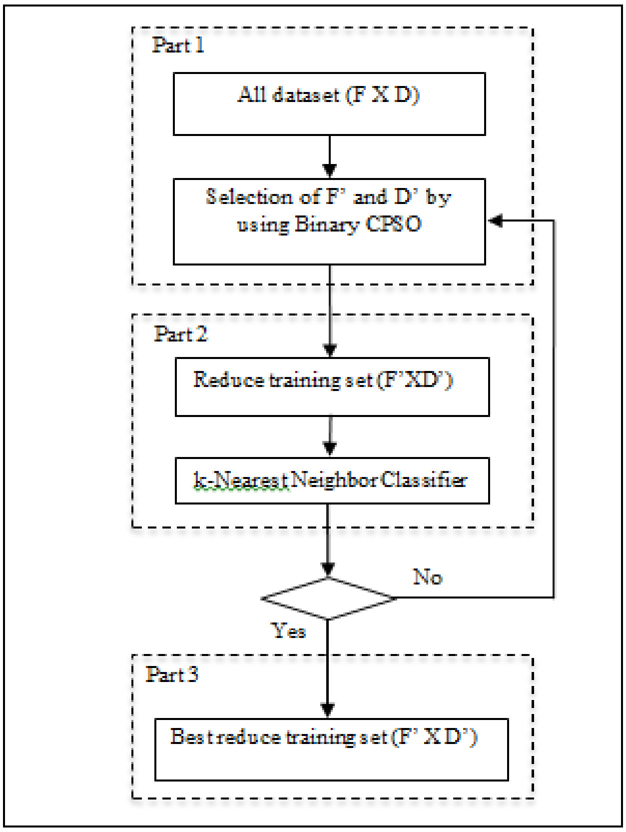


Figure 3. General framework of integration FS and IS

3. RESULTS AND ANALYSIS

In this section, a set of experiments has been described by employing various regression data sets from machine learning repository. The key aim of these experiments is to demonstrate the put forward method's abilities as well as quantify the performance pertaining to the selected instances and features. A short summary of the data sets utilised in the experiment is depicted in Table 1.

Table 1. Data Description

Data set	Number of features	Number of data	Number of Classes
Bupa	7	645	2
Ionosphere	34	351	2
Mammography	6	961	2
Pima	7	768	2
Wisconsin (Wis)	32	699	2
Image	19	210	7
Iris	4	150	3
Glass	9	214	7
Zoo	17	101	7
Ecoli	8	336	8
Yeast	8	1484	10
Segmentation	20	2310	7
Heart	13	270	2
Magic	11	19020	2
Wine	13	178	3
Libras	90	360	15
Spambase	57	4597	2
Satimage	36	6435	7
Sonar	60	208	2
Vehicle	19	846	4

3.1. Parameter setup

A standard form is employed to set the values of the BCPSO parameters as follows. Over the course of optimisation, the linear values of the inertia weight, w, ranged from 1 to 0. The values pertaining to the cognitive factor, c1, as well as social factor, c2, were set at 0.5 and 1.5, respectively. The number of sub-swarms used in the BCPSO is according to the number of classes pertaining to the dataset employed in the experiments. Next, the search space is segmented into several sub-swarms allowing cooperation with each other and wherein individuals are employed within the sub-swarms to denote a portion of the search space. As per Shi and Eberhert, the size of the population did not have much impact on the PSO method's performance [26]. In the experiment, a smaller size of generation was employed versus particles size because of large search space pertaining to instances selection. Due to this, the best solution could be identified quicker when compared to employing a smaller particles size

3.2. Experimental results

In this segment, we experimentally assess the framework recommended in Section 3. Firstly, these experiments make a comparison of the performance pertaining to the standard k-NN classifier as well as the put forward method, which has been employed as baseline result. The overall result obtained by employing 20 datasets is presented in Figure 4. Our put forward method gave dramatically better performance when compared to the standard method. Higher improvement was seen based on the results, particularly for data number 11 signifying Yeast dataset. The next Table 2 displays the proposed method's performance versus other six current instances selection methods. The put forward method provided the highest accuracy for 17 datasets employed in the experiment.

To obtain considerable improvement when employing the Cooperative PSO approach, a comparison of performance for our method was done with that of the current method that employed bio-inspired approaches. Better performance was seen with the proposed instances selection method when compared with the Generational Genetic Algorithm (GGA) and Adaptive Search Algorithm (CHC), particularly when handling the dataset that has large number of class. Table 2 lists out the result that shows best performance as well as accuracy as highlighted in bold.

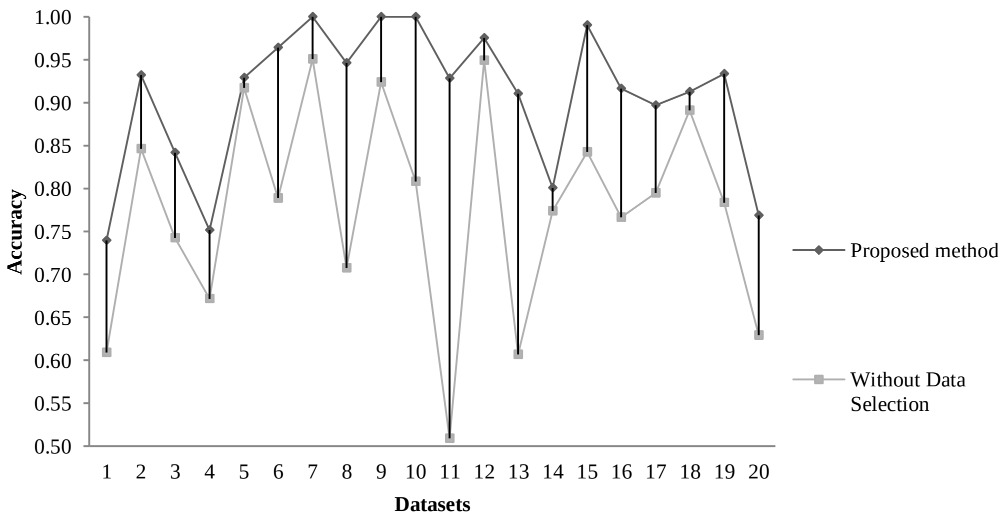
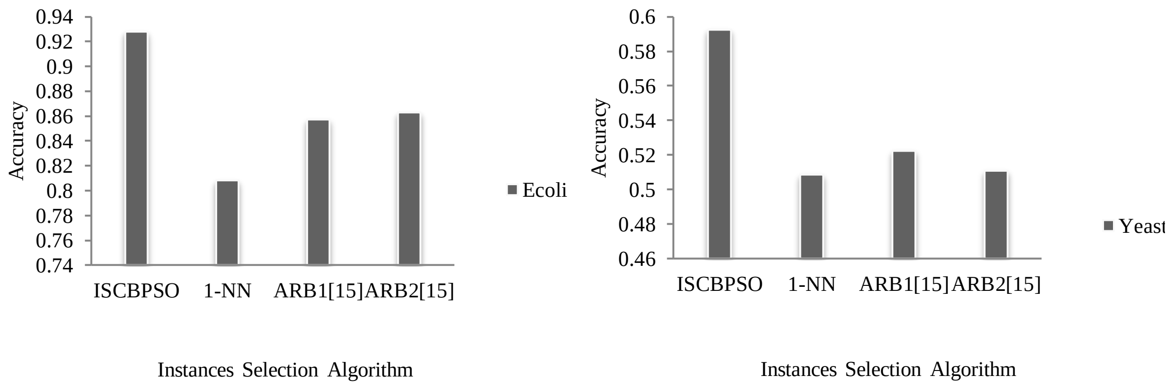


Figure 4. The performance of proposed method and standard k-NN classifier by using 20 datasets

Figure 5 shows the result when the Yeast and Ecoli dataset was employed. We intentionally selected this data due to the nature pertaining to these dataset classification problems as the features are not enough to allow distinguishing between classes. The yeast dataset was found to be a difficult classification issue. It included 1,484 instances signifying ten classes. Identical problem was seen even with the Ecoli dataset. However, a small number of instances were seen. For both datasets, the best accuracy result was achieved with our method versus the other two methods. Figure 6 displays the distribution pertaining to the selected data. The put forward method allows easy detection of the best instances pertaining to the training set that allows achieving result with better accuracy when employing the unseen data.

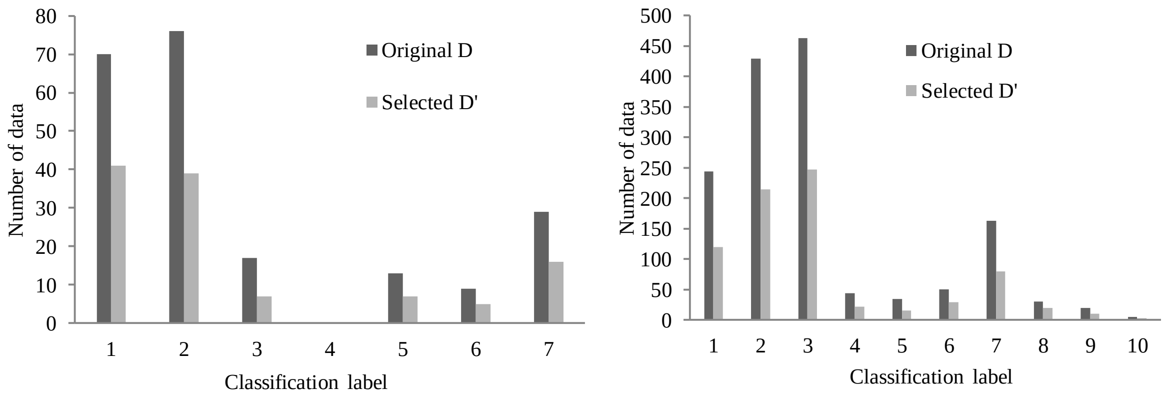
Table 2. Classification Rate for ISCBPSO vs. IS Algorithm (Instances Selection only)

Dataset	Proposed Method	HMN-C	HMN-E	HMN-EI	ICF	ENN	DROP3
Bupa	0.7400	0.6860	0.7270	0.7420	0.6990	0.7270	0.7350
Ionos	0.9322	0.9490	0.9230	0.8890	0.9070	0.9310	0.9190
Mamm	0.8416	0.9040	0.9200	0.9270	0.8820	0.9180	0.9160
Pima	0.7519	0.6540	0.7180	0.7130	0.6860	0.6970	0.7190
Wis	0.9293	0.9089	0.9036	0.9268	0.8821	0.9179	0.9161
Image	0.9644	0.7050	0.6570	0.7050	0.6140	0.5950	0.6190
Iris	1.0000	0.9490	0.9230	0.8890	0.9070	0.9310	0.9190
Glass	0.9462	0.8714	0.8714	0.8762	0.8524	0.8190	0.8286
Zoo	1.0000	0.9600	0.9700	0.9100	0.9500	0.9600	0.9200
Ecoli	1.0000	0.7910	0.8180	0.8240	0.8150	0.8520	0.8550
Yeast	0.9281	0.3490	0.3800	0.3750	0.3750	0.4030	0.4130
Segm	0.9757	0.9490	0.9230	0.8890	0.9070	0.9310	0.9190
Heart	0.9104	0.7670	0.8120	0.8060	0.7580	0.7840	0.7960
Magic	0.8011	0.6650	0.6550	0.6920	0.6580	0.6730	0.6710
Wine	0.9903	0.7240	0.6760	0.7000	0.6470	0.6470	0.6470
Libras	0.9166	0.6650	0.6550	0.6920	0.6580	0.6730	0.6710
Spam	0.8971	0.7500	0.7490	0.7420	0.7150	0.7410	0.7180
Sati	0.9126	0.9490	0.9230	0.8890	0.9070	0.9310	0.9190
Sonar	0.9340	0.4950	0.4700	0.4400	0.5000	0.5250	0.4850
Vehicle	0.7685	0.6380	0.5960	0.5830	0.6180	0.6360	0.5890



(a) (b)

Figure 5. Performance of several instances selection method; (a) Ecoli dataset and (b) Yeast



(a) (b)

Figure 6. Distribution of the selected data vs. the original data for Glass dataset (a) and Yeast dataset

Table 3 provides a comparison that was made between the two data selection techniques. By combining feature and instances selection, a better accuracy was achieved than by employing instances selection only. For each of the dataset employed in the experiment, equally interesting results were found with the reduction ratio of feature and instances. Table 4 presents the results. Overall, around 0.5 was the ratio for feature and instances, which signify that around 50% of the original data has been employed as the training dataset.

In the second experiment, four high dimensional datasets were employed as presented in Table 1. Table 5 presents the accuracy results achieved from these datasets. Here, our proposed method is compared with Interactive Genetic Algorithm (IGA) and Adaptive Search Algorithm (CHC) for feature and instances selection. In all cases, better performance was seen with our method versus other two methods. The data selected distribution with regards to their classification label is shown in Figures 7. In most cases, in our method, just 50% of the original data have been selected. Based on the results, it can be observed clearly that when our proposed method is employed for data selection, better balance was achieved in terms of the classification label. This is quite a significant concern which we handle, particularly when tackling imbalance datasets.

Table 3. Classification Rate for FISCBPSO vs. ISCBPSO

Dataset	Proposed Method	
	IS & FS	IS
Bupa	0.7418	0.7400
Mammo	0.8759	0.8416
Pima	0.7615	0.7519
Wisconsin	0.9650	0.9644
Ionos	0.9572	0.9322
Image	0.9533	0.9462

Table 4. Reduction Ratio Achieved using FISCBPSO

Dataset	Reduction Percentage	
	Feature	Instances
Bupa	0.43	0.51
Mammo	0.50	0.26
Pima	0.43	0.54
Wisconsin	0.50	0.50
image	0.58	0.49

Table 5. FISCBPSO vs. FIS Algorithm (Classification Rate) using Large Dataset

Dataset	IFS-BCPSO	CHC [13]	IGA [13]	1-NN
spam	0.9166	0.9071	0.9112	0.8945
libras	0.8971	0.6583	0.7234	0.8194
sati	0.9126	0.8611	0.8383	0.9058
Sonar	0.9340	0.7561	0.7878	0.8555

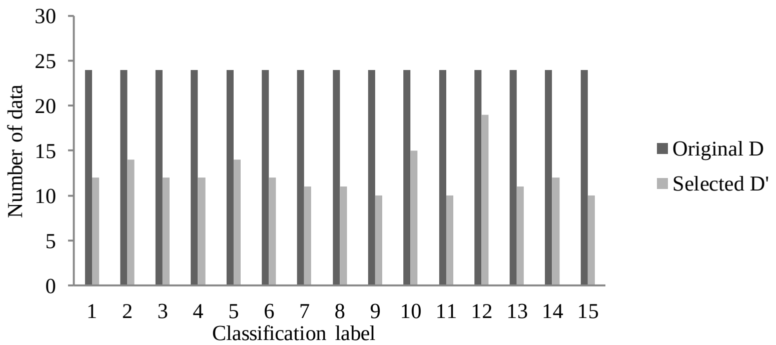


Figure 7. Distribution of the selected data vs. the original data for Libras dataset

4. CONCLUSION

In this study, we recommend an alternative tactic for the data selection approach by deploying the cooperative approach when seeking for the best data subset. The practice of choosing the subset of feature and data on the basis of the Cooperative Binary PSO approach overcomes the drawbacks of the existing approaches when tackling huge and imbalance datasets. The particles for Cooperative Binary PSO are split into multiple constituents known as sub-swarms. The first sub-swarm is for choosing the feature and the rest are for choosing the instances. Deploying the cooperative approach enables simultaneous assimilation of data selection and feature selection. Furthermore, this component by component optimisation facilitates fine tuning of every component per particle.

The outcomes accomplished by our proposed approach in the experimental study carried out have indicated that it provides improved data selection compared to another existing approaches. Furthermore, the approach which we recommended is efficient and coherent.

The recommended approach presents several prospects for future research. As far as data scalability is concerned, the selection algorithm can handle the issue by improving our proposed method of Cooperative Binary PSO. Here, we will experiment with extremely huge data sets and make a comparison of the results with another methodology which has a similar goal. Furthermore, we will seek to enhance our recommended method by deploying different learning algorithms and cracking different applications.

ACKNOWLEDGEMENTS

The authors would like to thank the Universiti Teknikal Malaysia Melaka for funding the study through PJP/2018/FTK(16A)/S01642. Besides, thank you to the Faculty of Information Technology and Communication for providing excellent research facilities

REFERENCES

[1] K. N. Stevens, T. M. Cover, and P. E. Hart, "Nearest Neighbor," vol. I, 1967.

[2] M. Muja and D. G. Lowe, "Scalable Nearest Neighbor Algorithms for High Dimensional Data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2227-2240, 2014.

[3] L. Zhang, Q. Liu, W. Yang, N. Wei, and D. Dong, "An Improved K-nearest Neighbor Model for Short-term Traffic Flow Prediction," *Procedia - Soc. Behav. Sci.*, vol. 96, no. Cictp, pp. 653-662, 2013.

[4] Z. Martinasek, V. Zeman, L. Malina, and J. Martinasek, "k-Nearest Neighbors Algorithm in Profiling Power Analysis Attacks," *Radioengineering*, vol. 25, no. 2, pp. 365-382, 2016.

[5] K. Millard and M. Richardson, "On the Importance of Training Data Sample Selection in Random Forest Image Classification: A Case Study in Peatland Ecosystem Mapping," *Remote Sens.*, vol. 7, no. 7, pp. 8489-8515, 2015.

[6] M. Huang, R. Lin, S. Huang, and T. Xing, "A Novel Approach for Precipitation Forecast Via Improved K-Nearest Neighbor Algorithm," *Adv. Eng. Informatics*, vol. 33, pp. 89-95, 2017.

[7] Y. Song, J. Liang, J. Lu, and X. Zhao, "An Efficient Instance Selection Algorithm for K Nearest Neighbor Regression," *Neurocomputing*, vol. 251, pp. 26-34, 2017.

[8] S. Nami and M. Shajari, "Cost-Sensitive Payment Card Fraud Detection Based on Dynamic Random Forest and K-Nearest Neighbors," *Expert Syst. Appl.*, vol. 110, pp. 381-392, 2018.

[9] S. A. Sharifah Sakinah, "Fuzzy Modeling Through Granular Computing," University of Alberta, 2012.

[10] H. Liu and H. Motoda, "On Issues of Instance Selection," *Data Min. Knowl. Discov.*, vol. 6, pp. 115-130, 2002.

[11] Y. B. Wah, H. A. A. Rahman, H. He, and A. Bulgiba, "Handling Imbalanced Dataset using SVM and K-NN Approach," *AIP Conf. Proc.*, vol. 1750, no. June 2016, 2016.

[12] W. Liu and S. Chawla, "Class Confidence Weighted k NN Algorithms for," *Adv. Knowl. Discov. Data Mining, Lect. Notes Comput. Sci.*, pp. 345-356, 2011.

[13] J. Metz, A. a Freitas, M. C. Monard, and E. A. Cherman, "A Study on the Selection of Local Training Sets for Hierarchical Classification Tasks," *Brazilian Natl. Meet. Artif. Intell.*, pp. 572-583, 2011.

[14] H. Brighton and C. Mellish, "Advances in Instance Selection for Instance-based Learning Algorithms," *Data Min. Knowl. Discov.*, vol. 6, pp. 153-172, 2002.

[15] S. S. S. Ahmad and W. Pedrycz, "Feature and Instance Selection Via Cooperative PSO," *2011 IEEE Int. Conf. Syst. Man, Cybern.*, pp. 2127-2132, 2011.

[16] I. Guyon and a Elisseeff, "An Introduction to Variable and Feature Selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157-1182, 2003.

[17] J. R. Cano, F. Herrera, and M. Lozano, "Using Evolutionary Algorithms as Instance Selection for Data Reduction in KDD: an Experimental Study," *IEEE Trans. Evol. Comput.*, vol. 7, no. 6, pp. 561-575, Dec. 2003.

[18] S. Dasgupta, "Consistency of Nearest Neighbor Classification under Selective Sampling," *JMLR Work. Conf. Proc.*, p. 16, 2012.

[19] N. García-Pedrajas, A. de Haro-García, and J. Pérez-Rodríguez, "A Scalable Approach to Simultaneous Evolutionary Instance and Feature Selection," *Inf. Sci. (Ny).*, vol. 228, pp. 150-174, Apr. 2013.

[20] J. R. Cano, F. Herrera, and M. Lozano, "On the Combination of Evolutionary Algorithms and Stratified Strategies for Training Set Selection in Data Mining," *Appl. Soft Comput.*, vol. 6, pp. 323-332, 2006.

[21] I. Czarnowski, "Prototype Selection Algorithms for Distributed Learning," *Pattern Recognit.*, vol. 43, pp. 2292-2300, 2010.

[22] N. O. Andrews and E. a Fox, "Clustering for Data Reduction : A Divide and Conquer Approach," pp. 1-10, 2007.

[23] A. de Haro-García, J. Pérez-Rodríguez, and N. García-Pedrajas, "Combining Three Strategies for Evolutionary Instance Selection for Instance-based Learning," *Swarm Evol. Comput.*, vol. 42, no. July 2017, pp. 160-172, 2018.

[24] C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke, "Evolutionary Instance Selection for Text Classification," *J. Syst. Softw.*, vol. 90, pp. 104-113, Apr. 2014.
[25] F. vandenBergh and A. P. Engelbrecht, "A Cooperative Approach to Particle Swarm Optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 225-239, Jun. 2004.
[26] R. C. Eberhart and Yuhui Shi, "Particle Swarm Optimization: Developments, Applications and Resources," *Proc. 2001 Congr. Evol. Comput. (IEEE Cat. No.01TH8546)*, vol. 1, pp. 81-86, 2001.

BIOGRAPHIES OF AUTHORS

Sharifah Sakinah Syed Ahmad is currently an associate professor in the Department of Intelligent Computing and Analytics, Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM). She received her Bachelors and Masters degrees of Applied Mathematics in School of Mathematics from University Science Malaysia. Following this, she received her Ph. D from the University of Alberta, Canada in 2012 in Intelligent System. Her research in graduate school focused on the granular computing and fuzzy modeling. Her current research interests including evolutionary optimization, fuzzy system, and granular computing. Her current research work is on the granular fuzzy rule-based system, Evolutionary Method, Modeling and Data Science.

Ezzatul Farhain Azmi started to be mathematics lecturer in 2013 at the age of 23. She is currently a lecturer in Department of Mechanical Engineering Technology, Faculty of Mechanical and Manufacturing Engineering Technology, Universiti Teknikal Malaysia Melaka (UTeM). She holds a Bachelor of Science in Mathematics and a Master of Science in Mathematics from Universiti Teknologi Malaysia (UTM). Her current research interest is on Mathematical Modelling and Artificial Intelligence.

Fauziah Kasmin received B. Sc (Mathematics) and M. Sc (Applied Statistics) at Universiti Putra Malaysia. She received the PhD degree from Universiti Kebangsaan Malaysia in 2018. She is currently a senior lecturer in Department of Intelligent and Computing Analytics, Faculty of Information and Communications Technology, Universiti Teknikal Malaysia Melaka. Her current research focuses on the algorithmic aspects of image processing.

Zuraini Othman is currently a senior lecturer in the Department of Intelligent Computing and Analytics, Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM). She received her Bachelors and Masters degrees of Applied Mathematics in School of Mathematics from University Science Malaysia. Her current research focuses on the mathematical modelling and algorithmic aspects of image processing.