International Journal of Applied Metaheuristic Computing
Volume 13 • Issue 1

Hybrid Binary Butterfly Optimization Algorithm and Simulated Annealing for Feature Selection Problem

Mohd Faizan, Babasaheb Bhimrao Ambedkar University, India*
Fawaz Alsolami, King Abdulaziz University, Saudi Arabia
Raees Ahmad Khan, Babasaheb Bhimrao Ambedkar University, India

ABSTRACT

Feature selection is performed to eliminate irrelevant features to reduce computational overheads. Metaheuristic algorithms have become popular for the task of feature selection due to their effectiveness and flexibility. Hybridization of two or more such metaheuristics has become popular in solving optimization problems. In this paper, the authors propose a hybrid wrapper feature selection technique based on binary butterfly optimization algorithm (bBOA) and simulated annealing (SA). The SA is combined with the bBOA in a pipeline fashion such that the best solution obtained by the bBOA is passed on to the SA for further improvement. The SA solution improves the best solution obtained so far by searching in its neighborhood. Thus, the SA tries to enhance the exploitation property of the bBOA. The proposed method is tested on 20 datasets from the UCI repository, and the results are compared with five popular algorithms for feature selection. The results confirm the effectiveness of the hybrid approach in improving the classification accuracy and selecting the optimal feature subset.

KEYWORDS

Binary Butterfly Optimization Algorithm (bBOA), Classification, Feature Selection, Hybrid Optimization Algorithm, Simulated Annealing

1. INTRODUCTION

Feature Selection is a crucial research area in the development of efficient classification algorithms for high dimensional datasets (Khaire and Dhanalakshmi, 2019). A proper feature selection technique should enhance the performance of the classification model. High dimensional dataset suffers from the famous problem of "curse of dimensionality" (Xue et al., 2016). The original dataset contains irrelevant and redundant features that degrade the performance of the algorithm and also increase computational overheads. Feature selection methods aim to reduce the feature space by removing superfluous and insignificant elements. These methods can be broadly classified into *wrapper* and *filter* methods. In the *wrapper* approach, the classifier is used to evaluate the quality of the selected features. The filter methods use concepts from information theory to obtain a relevant set of features. Wrapper methods are usually slower than filter methods. However, they can generate better classification performance than filter methods (Grande et al., 2007).

The feature selection technique must be capable of searching every possible subset of the original feature set to obtain the optimum feature subset. However, it is impractical to achieve this task owing to the high computational cost that comes along with it which consequently results in the selection of

DOI: 10.4018/IJAMC.2022010104

*Corresponding Author

a sub-optimum feature subset. Nowadays, evolutionary algorithms and metaheuristic techniques are popular for solving the problem of feature selection. The metaheuristic approaches did not guarantee the best results; however, it will produce better solutions within time bounds (Talbi et al., 2009). Some of the most popular metaheuristics are genetic algorithm (GA) (Holand et al., 1992), differential evolution (DE) (Storn and Price, 1997), tabu search (TS) (Hedar et al., 2006), simulated annealing (SA) (Kirkpatrick et al., 1983), particle swarm optimization (PSO) (Kennedy and Eberhart, 1995), ant colony optimization (ACO) (Dorigo et al., 2006), artificial bee colony (ABC) (Yi and He, 2014) and Cuckoo Search (CS) (Yang and Deb, 2009), ant lion optimization (ALO) (Emary et al., 2016) and sine-cosine algorithm (SCA) (Mirjalili, 2016).

A good metaheuristic should be capable of maintaining a balance between exploration and exploitation strategies during the search process (Talbi et al., 2009). Population-based methods like PSO or ACO are better in exploring the search space while single solution-based methods employ the right exploitation strategies. To achieve a balance between the two, hybrid methods can be used that combine the merits of two or more metaheuristic to achieve better optimization results.

Butterfly Optimization Algorithm (BOA) is a recent nature-inspired swarm algorithm that takes inspiration from the food searching behavior of butterflies in a natural environment (Arora and Singh, 2018). BOA has produced superior results for continuous optimization problems when compared with other recent metaheuristics. The binary version of BOA called bBOA was proposed by Arora and Anand (2019) for feature selection problems in wrapper mode. The binary version bBOA has the same structure as that of native BOA except that it utilizes two transfer functions for generating binary solutions.

A metaheuristic based on hill-climbing called Simulated Annealing was proposed by Kirkpatrick et al (1983). The method tries to improve the current solution at each iteration by generating a trial solution in its vicinity. The improved solution with better fitness value is always accepted while the solution with no improvement is taken with a specific probability value to avoid getting trapped in the local optima. The probability of acceptance of a worse solution is dependent on the temperature parameter. The temperature parameter decreases with every iteration at a fixed rate called the cooling schedule.

Despite producing superior results, the weak exploitation ability of BOA prematurely converges it to local optima. Also, the random transition between the exploration and exploitation phase in BOA is determined by the value of the switching probability which sometimes distracts the BOA from attaining the global optima (Arora et al., 2018). On the other hand, the SA algorithm possesses excellent exploitation properties.

In this paper, we introduce a hybrid wrapper method of bBOA and SA algorithms that can enhance the exploitation capability of the native bBOA for feature selection. The hybrid method combines bBOA with SA to further improve the fitness value of the candidate solutions. The SA algorithm enhances the exploitation property of bBOA by iteratively adjusting the best solution obtained so far at the end of every iteration. In this way, the SA algorithm is embedded into the bBOA and is executed in each iteration of the bBOA. The efficiency of the proposed approach has been evaluated against 20 benchmark datasets from the UCI machine learning repository.

The rest of the paper is organized as follows: Section 2 describes the related work in the area of feature selection using metaheuristic algorithms. Section 3 describes the background and the proposed hybrid approach. Section 4 contains the result and discussion part and Section 5 draws the conclusion and future work.

2. RELATED WORK

The objective of feature selection is to select a reduced subset of features by removing irrelevant and redundant features. The task of feature selection requires the creation and evaluation of every possible subset of features which makes it an NP-hard problem (Chandrashekar and Sahin, 2014).

However, to find an optimal or near-optimal subset in a reasonable time, a tradeoff between quality and computational complexity should be found. Therefore the metaheuristics and evolutionary search methods can be applied to find the near-optimum solutions in a feasible time limit (Saeys et al., 2007).

Feature selection methods based on swarm and evolutionary intelligence algorithms have attracted several researchers in recent years. PSO, GA, ABC and ACO (Inbarani et al., 2014; De Stefano et al., 2014; Schiezaro and Pedrini, 2013; Aghdam et al., 2009) are some of the methods that have been successfully applied in the task of feature selection. Furthermore, the researchers have proposed hybrid methods based on the combination of two or more algorithms. The very first hybrid approach incorporated the local search techniques into the GA to modify the search scheme (Oh et al., 2004).

The combination of GA with TS was proposed for feature selection where TS performs the local search while the global search was controlled by GA. The hybrid method produced better results than the native GA on several publicly available datasets (Tang, 2007). GA was also hybridized with SA for efficient feature selection by replacing the selection process in GA by SA based method (Olabiyisi Stephen et al., 2012). The fusion of SA prevents the GA from getting trapped in the local optima.

The hybrid approaches based on ABC and DE were suggested both for continuous and binary optimization problems with good results. (Alizadegan et al., 2012; Zorarpacı and Özel, 2016). A new method called Genetical Swarm Optimization (GSO) for selecting features in the Digital mammogram dataset was proposed (Jona and Nagaveni, 2012). The GSO employs the GA operators in PSO to improve the convergence rate. A combination of ACO and CS was proposed for selecting features in the Digital mammogram dataset (Jona and Nagaveni, 2014). The local search of ACO was improved by CS to achieve better results.

In yet another hybrid approach, PSO was combined with a binary bat algorithm (BBA) for feature selection (Tawhid and Dsouza, 2018). The high exploration capability of BBA and the convergence characteristics of PSO facilitates the generation of functional feature space. The BBA was also used to solve the 0-1 knapsack problems. The novel BBA along with the local search scheme attains better exploration and exploitation capability that helps avoid the local optima. The proposed method shows promising results when tested on different 0-1 knapsack problems (Rizk-Allah and Hassanien, 2018).

A hybrid approach HPSO-LS was suggested (Moradi and Gholampour, 2016) that embeds the local search into the PSO to obtain a reduced feature set 13 UCI datasets. PSO was modified to solve the multi-objective resource allocation problem (MORAP). The update method for the particles was developed to speed up the convergence of the algorithm. Moreover, the selection of the local and global best employs the neighborhood strategy to find optimal solutions. The proposed method outperformed other methods in the literature (Fan et al., 2013)

The whale optimization algorithm was hybridized with the simulated annealing (WOASA) to enhance the exploitation capability of the whale optimization algorithm (WOA). The WOA was combined with the SA in two different fashions. In the first approach, the SA was incorporated into the WOA while in the second approach; the SA was invoked at the end of every iteration to further enhance the quality of the solution returned by the WOA (Mafarja and Mirjalili, 2017). A binary version of the WOA for feature selection was proposed wherein the V-shaped hyperbolic tangent function was used for transforming the continuous values to binary form. The experimental results suggest that the proposed binary WOA was able to reduce the number of features and improve the accuracy of several datasets (Houssein et al., 2017). In another work, an S-shape sigmoid function was employed in WOA to convert the continuous values to binary. The proposed method shows significant improvement in classification accuracy with an optimal feature set on eleven UCI datasets (Hussien et al., 2019).

The bio-inspired algorithms have also been used in chemical research. A study has employed the swarming behavior of the salps algorithm to predict activities in the chemical compound. With three different initialization techniques, the proposed salp algorithm was used to select relevant chemical descriptors with high classification accuracy as compared to other optimization algorithms (Hussien et al., 2017). The hybrid approach for feature selection was also applied in the biomedical field.

A novel approach based on the improved coral reefs optimization (CRO) and SA was proposed to enhance the search process (Yan et al., 2019). The proposed method outperforms the other state of the art methods on 13 medical datasets publicly available.

A novel study combines the Fuzzy Min-Max (FMM) neural network with the Brain Storm Optimization (BSO) for feature selection in classification (Pourpanah et al., 2019). The combined scheme was able to generate good results on benchmark and real-world datasets. Another study has proposed the amalgamation of the harmony search algorithm with the Pareto optimization approach (Dash, 2018) for high dimensional datasets. The hybrid algorithm produced good classification accuracy and smaller feature subsets.

The No-Free Lunch (NFL) theorem in the optimization field confirms that no single algorithm is capable of solving all optimization problems. The number of studies discussed above for feature selection is just one example to prove it. It can be said that there is always a place for new algorithms and improvements in the existing one. Therefore in this work, we present another hybrid approach for feature selection for classification.

3. METHOD

3.1 Background

3.1.1 Butterfly Optimization Algorithm

Butterfly optimization algorithm (BOA) is a novel population-based bio-inspired algorithm suggested by Arora and Singh (2018) that mimics the behavior of butterflies while searching for food. In nature, the butterflies possess chemoreceptors spread over their body that helps them in sensing and discovering food and other butterflies in the surrounding. In BOA, each butterfly is supposed to produce a scent or smell that can be perceived by the other butterflies. The intensity of the scent of a butterfly corresponds to its fitness.

A butterfly senses the best butterfly in the search space and moves towards it; this step is called the global search. If a butterfly could not detect another butterfly in its vicinity, it moves randomly in the space, and this step is called the local phase.

The perceived fragrance with which each butterfly sense other in the surrounding is given by Eq. (1):

$$pf_i = cI^a \tag{1}$$

where pf_i is the perceived magnitude of the fragrance of i^{th} butterfly, *c* is the sensory modality, *I* is the stimulus intensity, and *a* is the power exponent.

In BOA, butterflies move both in local and random phases, so a switch probability is used for the transition between exploration and exploitation.

The original BOA was developed for the continuous optimization problem. However, the values in the candidate solution for feature selection are limited to {0,1} only. The binary version bBOA for the task of feature selection was proposed and is described as follows. The bBOA make use of a transfer function for converting the continuous value solutions into the binary solutions for feature selection. This transfer function provides the probability value that determines the values (0 or 1) in the position vector. Two transfer functions were employed for performing this task, namely sigmoid and V-shaped transfer function. In this work, we shall be using the sigmoid transfer function as it produces better results than the V-shaped function as confirmed in their study. The sigmoid function is given by Eq. (2):

$$S\left(t\right)=\frac{1}{1+e^{-t}} \tag{2}$$

The output of the sigmoid function is still a continuous value that needs to be limited to generate a binary value. The common stochastic threshold given by Eq. (3) is used for obtaining binary values.

$$y=\begin{cases}0 \ \mathit{if\,rand} < S\left(t\right)\\ 1 \ \mathit{if\,rand} \geq S\left(t\right)\end{cases} \tag{3}$$

The steps of the algorithm are:

Step 1: A random population of *n* butterflies called search agents is generated. The original fragrance of each butterfly is obtained by Eq. (1).

Each butterfly is represented by x_i *(i=1,2….n)* as a *k*-dimensional vector where *k* represents the number of features in the problem to be solved.

Step 2: The fitness value of each of the randomly generated butterflies is calculated by the objective function. The butterfly with the highest fitness is regarded as the current best of the population.

Step 3: After initialization, all butterflies move around the region in search of food. While searching the position of the i^{th} butterfly is updated depending on the local or global phase. The position of the butterfly in the global and local phase is updated by the Eq. (4) and Eq. (5) respectively:

$$x_i^{t+1}=x_i^t+\left(r^2\times g^*-x_i^t\right)\times pf_i \tag{4}$$

$$x_i^{t+1}=x_i^t+\left(r^2\times x_j^t-x_k^t\right)\times pf_i \tag{5}$$

Where x_i^t and x_i^{t+1} are the position vectors of the i^{th} butterfly at *t* and *t+1* iteration. g^* is the current best solution at the t^{th} iteration, pf_i is the perceived fragrance of i^{th} butterfly, x_j^t and x_k^t are random butterflies in the solution space and r is a uniform random number in [0, 1]. A probability value *p* is used for switch between the global and local search phases.

Step 4: Use Eq.(2) and Eq. (3) to obtain a binary position vector corresponding to the i^{th} butterfly position.

Step 5: After exploring the search space and updating the positions, the fitness value of each butterfly is calculated. If the new value is better than that of the current value, then the butterfly retains this new position and discard the old position. At the end of the current iteration, the current best butterfly is also updated if it has a higher fitness value than the old one.

Step 6: Update the sensory modality *c*.

The steps 3 to 5 are repeated until the stopping criteria are not met. The best solution found at the termination of the algorithm is the optimum solution.

3.1.2 Simulated Annealing

SA is a local search based stochastic algorithm that iteratively improves the initial solution by exploring its neighborhood. The improved version of the initial solution is always accepted, but the worst solution is allowed with a certain probability. The SA derives its working from the movements of particles in the physical systems. The particles move freely at higher temperatures with minimal changes in structure. As the temperature decreases, the probability that controls the particle movement also decreases until a ground state of lowest energy is achieved by the system.

Kirkpatrick et al. (1983) relate these points to design a heuristic technique called SA for solving optimization problems. In SA, the state of a physical system is equivalent to the current solution, the ground state achieved by the system is the global optima, and the movement of particles that lead to the change of state corresponds to the neighborhood searching of the current solution. The temperature that governs the particle movement is the 'temperature' parameter in the SA. The steps of SA are:

Step 1: Set the initial temperature T_0 to T, set the current best solution g^* to S and the fitness of the current best solution $f\left(g^*\right)$ to $f\left(S\right)$.

Step 2: Generate a random trial solution Tr in the neighborhood of S and calculate its fitness $f\left(Tr\right)$.

Step 3: If $f\left(Tr\right) > f\left(S\right)$, accept the trial solution Tr, set S to Tr and its corresponding fitness value $f\left(S\right)$ to $f\left(Tr\right)$, also update the g^* and $f\left(g^*\right)$ to Tr and $f\left(Tr\right)$ respectively. Otherwise, go to Step 4.

Step 4: If $\left|Tr\right| < \left|S\right|$ and $f\left(Tr\right) = f\left(S\right)$, set S to Tr and its corresponding fitness value $f\left(S\right)$ to $f\left(Tr\right)$, also update the g^* and $f\left(g^*\right)$ to Tr and $f\left(Tr\right)$ respectively. Otherwise, go to Step 5.

Step 5: Calculate $\theta = f\left(Tr\right) - f\left(S\right)$, generate a random number *P* in [0,1]. If $P \le e^{-\theta/T}$ then accept the worse solution and set S to Tr and $f\left(S\right)$ to $f\left(Tr\right)$.

Step 6: Update the temperature parameter T.

Step 7: Repeat steps 2 to 6 until the termination criteria are not met.

Step 8: Output the best solution.

3.2 The Proposed Approach

Feature selection is a process of optimization where the solution is represented as a binary vector. The binary version of the BOA was proposed for the task of feature selection. In bBOA, a solution is described in the form of a single-dimensional vector of length equal to the original number of features in the dataset. Each element of the vector is either "0" or "1", where "1" denotes that the feature is selected and "0" indicates that it is not selected.

Feature Selection demands a balance between two conflicting criteria: number of features and the classification accuracy. A good solution is one with a fewer number of selected features and higher classification accuracy. The fitness function given by Eq. (6) is used to achieve this objective:

$$Fitness = \alpha Er + \beta R \tag{6}$$

where Er is the classification error rate, R is the ratio of the size of the selected feature subset to the total number of features in the dataset. The parameters α and β are the weights corresponding to the classification accuracy and the importance of the number of selected features such that $\alpha \epsilon [0,1]$ and $\beta=(1-\alpha)$. The values of the parameters α and β are taken from the existing work (Emary et al., 2016; Mafarja and Mirjalili, 2017; Arora and Anand, 2019) to fairly compare our results with the existing work.

The binary version of the algorithm has shown promising results for the challenge of feature selection for classification. A new method of feature selection for the task of classification is proposed here by combining the bBOA algorithm with the SA algorithm. The current best solution obtained by bBOA is passed to the SA algorithm. The SA algorithm repeatedly generates a trial solution by searching in the neighborhood of the current best. The fitness of the trial solution is obtained and compared with the current best solution. The current best solution is replaced by the trial solution having better fitness value. The SA algorithm is invoked in each iteration to improve the quality of the current best. Once the termination criteria are met, the best solution found so far is returned as the optimum solution. The flowchart of the proposed approach is presented in Figure 1.

The K-nearest neighbor (KNN) is a popular classification algorithm used in varied classification domains (Altman, 1992). It uses the labels of the K nearest neighbors of training data to assign a class of the test data. The Euclidean distance is used to calculate the distance between the test sample and its neighbors. The computational time required by the KNN for the classification is inversely related to the size of the dataset. In this work, the wrapper approach for feature selection is used with KNN as the learning algorithm to evaluate the effectiveness of the candidate solution.

3.2.1 Computational Complexity

The computational complexity of bBOA is of $O(i(v \times s + C \times s))$ where i denotes the number of iterations, v denotes the number of attributes, s denotes the number of solutions and C is the fitness function cost. The computational complexity of SA is of $O(n^2)$ because for each unit change in the temperature parameter is followed by the search in the neighborhood. The SA enhances the solution returned by the bOA therefore the computational complexity of the hybrid method will be the summation of the complexity of bBOA and SA. Finally, the computational complexity of bBOA-SA will be of order $O(n^2)$.

4. EXPERIMENTS

4.1 Datasets

The performance of the proposed hybrid approach is evaluated on the 20 benchmark datasets for classification from the UCI repository (Bache and Lichman, 2019). The description of the datasets is given in Table 1. A varied range of datasets has been selected for evaluations, including ample feature space with a small number of instances and vice-versa.

4.2 Parameter Settings and Implementation

The number of search agents in the bBOA is set to seven as employed in its original paper (Arora and Anand, 2019). The two parameters of bBOA i.e. the sensory modality and the power exponent were also adopted from the original bBOA paper as they provide the best possible performance. The SA algorithm is run for a fixed number of times (30) for improving the global best after each iteration of bBOA. The maximum number of iterations for the proposed hybrid method was set to 100 as there was no significant improvement in performance after 100 iterations.

k-fold cross-validation is applied at every run on the dataset for better validation of the results. In *k*-fold cross-validation, the dataset is divided into *k* folds such that the k-1 folds are used for training the classifier and the remaining one fold is used for the testing purpose. For an adequate comparison

Table 1. Description of datasets used in the experiment

Dataset	Features	Samples
Breastcancer	9	699
BreastEW	30	569
Clean1	166	476
Colon	2000	62
CongressEW	16	435
Exactly	13	1000
Exactly2	13	1000
HeartEW	13	270
IonosphereEW	34	351
Leukemia	7129	72
Lymphography	18	148
M-of-n	13	1000
PenglungEW	325	73
SonarEW	60	208
SpectEW	22	267
Tic-tac-toe	9	958
Vote	16	300
WaveformEW	40	5000
WineEW	13	178
Zoo	16	101

of the proposed method with bBOA, the proposed method is run for 20 number of times. The KNN classifier with K=5 as the best choice is selected as it gives the best results (Emary et al., 2016) and had been used in the previous work (Mafarja and Mirjalili, 2017; Arora and Anand, 2019). The α parameter in the fitness function is set to 0.99. The proposed hybrid method is implemented in Python v3.8 (Python Software Foundation, 2019) on an Intel 1.6 GHz Core i5 processor machine having 4 GB of RAM running under Windows 8.1 operating system.

4.3 Results and Discussion

This section reports the results obtained from the proposed hybrid method. The effect of embedding SA into bBOA is evaluated by comparing its results with the results obtained from the original bBOA. The following criteria were used for comparison:

Average Classification Accuracy: The average of the classification accuracy obtained from 20 runs of the algorithm.

Average of the number of selected features: The average size of the feature subset obtained across 20 runs of the algorithm.

Average fitness value: The mean of the fitness values across 20 runs is calculated and compared.

The mean fitness measure of the proposed method, SA and the original bBOA algorithm for all the datasets are shown in Table 2. It can be seen that the bBOA-SA produces better results than the bBOA alone. The search mechanism of the bBOA-SA has greater exploitation capability than the bBOA due to the ability of the SA to repeatedly exploit the best solution obtained so far in order to find a better alternative. The properly balanced exploration and exploitation in bBOA-SA produce much better results than the original bBOA. The use of SA after bBOA pulls out the best possible solution that may not be achieved if bBOA alone gets stuck in local optima.

Table 3 shows the results of classification accuracy and the average size of the feature subset. The bBOA-SA produces better accuracy on 17 datasets. The proposed approach achieved 100% accuracy for Exactly and M-of-n dataset, which indicates the excellent performance of the proposed technique. Moreover, in the Exactly dataset, absolute accuracy is attained just by selecting only around six features. The excellent capability of bBOA-SA in selecting small feature subset can be attributed to the use of SA methodology in selecting even weak solutions with a certain degree of probability that prevents from getting trapped in local optima. This proposed approach garners the exploration characteristics of a global optimization method combined with the local search algorithm having good exploitation properties. Moreover, the proposed method also produces a much smaller feature subset for almost all the datasets. Though the hybrid approach produced below-par classification accuracy on BreastEW dataset, it was able to significantly reduce the size of the feature subset of BreastEW dataset by more than 90 percent. However, the proposed approach underperformed on the WineEW dataset in terms of all the evaluation criteria.

Table 4 shows the average running time (in seconds) required by the SA, bBOA and bBOA-SA approach to obtain optimal results. The computational time of the three methods is comparable as all of them are tested on the common datasets and the same parameter values. As evident in Table 4 that the hybrid approach requires more significant time than the native bBOA. The increase in computational time is due to the addition of running time incurred by the SA algorithm in each iteration. However, the rise in time brings a significant improvement in the classification accuracy and dimensionality reduction for the majority of the datasets.

The analysis of the results so far shows the superiority of the hybrid approach over the native bBOA both in terms of accuracy and feature selection. Now, the bBOA-SA is compared with the other state of the art approaches for the problem of feature selection: ALO, GA, PSO, SCA, WOASA. The values of the specific parameters of these algorithms are given in Table 5. These parameter values are adopted from the original bBOA paper for a fair comparison of the results. Table 6 shows the average classification accuracy of bBOA-SA and other state of the art algorithms. The classification accuracy obtained by the KNN classifier with the full feature set without any feature selection technique is also shown in Table 6. The result shows that the bBOA-SA outperforms other state of the art methods on seventeen datasets whereas the SCA and WOASA generate better results on the other dataset. Table 7 compares the bBOA-SA with other state of the arts in terms of the average size of the feature subset. The proposed approach also select a smaller number of features on seventeen datasets. GA produces a slightly smaller subset on the two datasets than bBOA-SA. The enhanced exploitation property of the proposed method enables it to intensively search the optimal region of the search space.

From Tables 6 and 7, it is clear that the proposed method performs better than the other algorithms. In fact, a significant difference can be seen between the classification accuracy and the feature subset size between the proposed approach and most of the other algorithms. This is due to the structure of the proposed method wherein even the sub-optimal solution returned by bBOA was exploited by the SA algorithm that increases the chance of escaping from the local optima. The explored regions of the feature space by the bBOA are further exploited by SA for the subsequent reduction in the feature space.

Table 8 shows the p-values of the Wilcoxon test of the proposed bBOA-SA with the other state of the art algorithms. p-values <0.05 indicates that there is a significant difference between the two optimization algorithm for a specific dataset. As given in Table 7, the proposed bBOA-SA is significantly different from the other algorithms for most of the dataset.

Overall the proposed method generated better results than the original bBOA on small as well as relatively large datasets. The addition of SA to bBOA enhances the performance of bBOA particularly the exploitation property of bBOA.

Table 2. Comparison of the proposed approach with the original bBOA and SA in terms of mean fitness measure

Dataset	Mean Fitness Measure		
	SA	bBOA	bBOA-SA
Breastcancer	0.039	0.040	**0.032**
BreastEW	0.068	0.040	0.049
Clean1	0.224	0.112	**0.048**
Colon	0.174	0.135	**0.118**
CongressEW	0.058	0.045	**0.033**
Exactly	0.322	0.040	**0.005**
Exactly2	0.271	0.260	**0.240**
HeartEW	0.260	0.180	**0.159**
IonosphereEW	0.150	0.096	**0.087**
Leukemia	0.056	0.030	**0.019**
Lymphography	0.220	0.139	**0.136**
M-of-n	0.130	0.025	**0.007**
PenglungEW	0.160	0.118	**0.087**
SonarEW	0.371	0.086	0.147
SpectEW	0.212	0.160	**0.142**
Tic-tac-toe	0.255	0.205	**0.172**
Vote	0.07	0.044	**0.044**
WaveformEW	0.233	0.264	**0.166**
WineEW	0.130	0.023	0.048
Zoo	0.104	0.034	**0.032**

4.4 Limitations

The proposed hybrid method is limited by its inability to generate exact optimization results at different runs. Although the proposed method produces good results at every application, the selected features contain discrepancy which puts the user in a state of a dilemma on which subset to be selected. Moreover, the proposed method also uses the KNN for the task of classification and the computational time of the hybrid method may change when a different classifier like Naive Bayes or support vector machines are used. This thing should be kept in mind when dealing with real-world scenarios.

5. CONCLUSION

Feature selection in one of the foremost task in the field of data mining and requires great importance in developing efficient classifier models. To address this task, a hybrid feature selection technique based on the binary butterfly optimization algorithm is proposed. This technique combines the exploitation property of the simulated annealing with the bBOA. The SA algorithm exploits the best solution obtained by the bBOA in search of a better alternative. The effectiveness of the hybrid approach over the original bBOA and other optimization algorithms is evaluated by conducting an experiment on twenty commonly used datasets from the UCI repository for classification. It was found that the proposed method was very effective in selecting small feature subset that can improve the

Table 3. Comparison of the proposed approach with the original bBOA and SA in terms of average classification accuracy and the average size of the feature subset

Dataset	Classification Accuracy			Average Feature Set		
	SA	bBOA	bBOA-SA	SA	bBOA	bBOA-SA
Breastcancer	0.9676	0.9692	**0.9744**	5.8	5.6	**5.4**
BreastEW	0.9362	0.9716	0.9509	15	16.0	**2**
Clean1	0.7790	0.8832	**0.8847**	83.15	91.80	**74.2**
Colon	0.8294	0.8680	**0.8857**	992.1	959	**914.0**
CongressEW	0.9460	0.9600	**0.9722**	7.85	6.4	**6**
Exactly	0.6811	0.9736	**1.0000**	7.9	7.6	**6**
Exactly2	0.7305	0.7596	**0.7602**	5.85	4.8	**2.0**
HeartEW	0.7426	0.8237	**0.8444**	6.05	5.8	**5.6**
IonosphereEW	0.8534	0.9072	**0.9296**	14.35	16.2	**4.8**
Leukemia	0.9479	0.9748	**0.9857**	3539.8	3471.5	**3440.0**
Lymphography	0.7822	0.8682	**0.8730**	9.25	8.40	**8.2**
M-of-n	0.8757	0.9270	**1.0000**	7.8	6.8	**6.8**
PenglungEW	0.8437	0.8792	**0.9296**	162.35	170.5	**111.2**
SonarEW	0.6303	0.9375	0.8684	29.6	30.80	**17.6**
SpectEW	0.7901	0.8485	**0.8620**	10.45	10.80	**10.20**
Tic-tac-toe	0.7487	0.8010	**0.8354**	5.95	5.6	8
Vote	0.933	0.9653	**0.9677**	7.95	5.0	**3**
WaveformEW	0.7718	0.7465	**0.8386**	27.0	25.0	**24**
WineEW	0.8734	0.9844	0.9570	6.1	6.2	7
Zoo	0.9000	0.9775	**0.9781**	7.75	5.20	**5.2**

classification accuracy. The hybrid approach outperforms the original bBOA on seventeen datasets in terms of classification accuracy and has generated smaller feature subsets for eighteen datasets under study. The proposed approach also generated good results when compared with other optimizers for feature selection. For future work, we aim to extend our study for even higher dimensional datasets in the field of medical sciences for effective feature selection. Moreover, we also like to assess the performance of the proposed approach with other classifiers like artificial neural networks, random forests, etc.

Table 4. Comparison of the proposed approach with the original bBOA and SA in terms of average computational time

Dataset	Average Computational Time		
	SA	bBOA	bBOA-SA
Breastcancer	0.71	2.61	42.56
BreastEW	0.78	2.72	41.48
Clean1	1.33	4.06	54.63
Colon	1.37	40.0	40.05
CongressEW	0.68	3.01	37.25
Exactly	1.11	5.26	59.45
Exactly2	1.14	4.54	56.92
HeartEW	0.64	2.82	33.52
IonosphereEW	0.74	2.93	34.74
Leukemia	2.27	228.03	210.23
Lymphography	0.61	2.89	27.62
M-of-n	1.27	4.49	55.31
PenglungEW	0.73	2.78	31.33
SonarEW	0.70	2.71	34.44
SpectEW	0.66	2.59	32.15
Tic-tac-toe	0.89	4.68	54.79
Vote	0.68	2.74	32.65
WaveformEW	13.36	103.22	2041.35
WineEW	0.60	2.67	30.87
Zoo	0.62	2.53	29.37

Table 5. Parameter values of the other algorithms

Parameter	Value
Mutation rate r in ALO	Min = 0 and max = 0.9
Crossover ratio in GA	0.9
Mutation ratio in GA	0.1
Selection Mechanism in GA	Roulette wheel
Inertia w in PSO	[0.9,0.6]
Acceleration constants in PSO	[2,2]
r_4 parameter in SCA	0.5
A parameter in SCA	2
a parameter in WOA	Min = 0 and max = 0.9

Table 6. Comparison of the proposed approach with other state of the art algorithms in terms of average classification accuracy

Dataset	ALO	GA	PSO	SCA	WOASA	All Features	bBOA-SA
Breastcancer	0.9625	0.9482	0.9609	0.9637	0.9634	0.9671	0.9744
BreastEW	0.9335	0.9456	0.9378	0.9420	**0.9685**	0.9298	0.9509
Clean1	0.8475	0.8701	0.8573	0.8598	0.8549	0.7738	0.8847
Colon	0.8596	0.8260	0.8440	0.8610	0.8692	0.8048	0.8857
CongressEW	0.9327	0.9410	0.9265	0.9066	0.9673	0.9217	0.9722
Exactly	0.7240	0.7358	0.7493	0.7220	1.0000	0.7440	1.0000
Exactly2	0.6996	0.7050	0.7135	0.6980	0.7364	0.7050	0.7602
HeartEW	0.7768	0.7875	0.7880	0.7704	0.8353	0.6630	0.8444
IonosphereEW	0.8600	0.8972	0.8823	0.8938	0.9083	0.8234	0.9296
Leukemia	0.9137	0.8710	0.8660	0.9470	0.9678	0.9179	0.9857
Lymphography	0.7875	0.8180	0.8105	0.7920	0.8567	0.7571	0.8730
M-of-n	0.8150	0.7990	0.8425	0.8560	0.9932	0.8755	1.0000
PenglungEW	0.8071	0.6719	0.8136	0.8048	0.8924	0.8321	0.9296
SonarEW	0.8487	**0.8761**	0.8676	0.8514	**0.9367**	0.5952	0.8684
SpectEW	0.7883	0.8092	0.7856	0.7896	0.8422	0.7822	0.8620
Tic-tac-toe	0.7587	0.7616	0.7512	0.7568	0.7846	0.7320	0.8354
Vote	0.9262	0.9335	0.9256	0.9195	0.9433	0.9167	0.9677
WaveformEW	0.7063	0.6920	0.7187	0.7038	0.7328	0.8084	0.8386
WineEW	0.9549	0.9540	0.9519	**0.9670**	**0.9857**	0.6755	0.9570
Zoo	0.9220	0.9294	0.9451	0.9647	0.9670	0.8910	0.9781

Table 7. Comparison of the proposed approach with other state of the art algorithms in terms of average feature set size

Dataset	ALO	GA	PSO	SCA	WOASA	bBOA-SA
Breastcancer	6.6	**5.1**	5.68	6.1	5.6	5.4
BreastEW	18	14.2	16.6	20.32	10.8	2
Clean1	140	96	102	109	87	74.2
Colon	1020	954	977	940.0	932	914.0
CongressEW	7	6.74	8	7	6.4	6
Exactly	8	8.05	9.1	10.73	6.2	6
Exactly2	10.7	6.32	9.4	9.1	2.6	2.0
HeartEW	10.2	8.2	8.1	6.47	6.8	5.6
IonosphereEW	12	14.05	19.16	12.09	10.6	4.8
Leukemia	3511	3491	3474	3451	3484	3440.0
Lymphography	16	8.6	8.6	9.68	8.6	8.2
M-of-n	11.3	7.34	9.4	10.5	7.0	6.8
PenglungEW	166.42	172	180	178.21	124.2	111.2
SonarEW	47.2	30	32	33.73	25.8	17.6
SpectEW	12	**9**	11.94	10.60	**9.6**	10.20
Tic-tac-toe	8.6	7	6.6	**5.8**	**6**	8
Vote	8.8	6.3	8	8.21	5.6	3
WaveformEW	32	26	24	34.15	24.8	24
WineEW	11	6.7	8.2	8.97	7.2	7
Zoo	11.8	8	7.82	9.52	5.6	5.2

Table 8. p-values of the Wilcoxon test of bBOA-SA vs other feature selection algorithms (p-values>0.05 are in bold)

Dataset	ALO	GA	PSO	SCA	WOASA	bBOA
Breastcancer	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
BreastEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Clean1	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Colon	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
CongressEW	**0.1965**	<0.05	<0.05	**0.068**	<0.05	**0.3321**
Exactly	**0.1543**	<0.05	<0.05	**0.278**	**0.1254**	**0.1493**
Exactly2	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
HeartEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
IonosphereEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Leukemia	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Lymphography	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
M-of-n	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
PenglungEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
SonarEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
SpectEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Tic-tac-toe	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Vote	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
WaveformEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
WineEW	<0.05	<0.05	<0.05	<0.05	<0.05	<0.05
Zoo	<0.05	**0.1478**	**0.1480**	**0.1551**	<0.05	**0.167**

REFERENCES

Aghdam, M. H., Ghasem-Aghaee, N., & Basiri, M. E. (2009). Text feature selection using ant colony optimization. *Expert Systems with Applications*, *36*(3), 6843–6853. doi:10.1016/j.eswa.2008.08.022

Alizadegan, A., Meybodi, M. R., & Asady, B. (2012). A novel hybrid artificial bee colony algorithm and differential evolution for unconstrained optimization problems. *Advances in Computer Science and Engineering.*, *8*(1), 45–56.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*, 175–185.

Arora, S., & Anand, P. (2019). Binary butterfly optimization approaches for feature selection. *Expert Systems with Applications*, *116*, 147–160. doi:10.1016/j.eswa.2018.08.051

Arora, S., & Singh, S. (2018). Butterfly optimization algorithm: A novel approach for global optimization. *Soft Computing*, *23*(3), 715–734. doi:10.1007/s00500-018-3102-4

Arora, S., Singh, S., & Yetilmezsoy, K. (2018). A modified butterfly optimization algorithm for mechanical design optimization problems. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, *40*(1), 21. doi:10.1007/s40430-017-0927-1

Bache, K., & Lichman, M. (2019). *UCI Machine Learning Repository*. http://archive.ics.uci.edu/ml

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, *40*(1), 16–28. doi:10.1016/j.compeleceng.2013.11.024

Dash, R., (2018). An Adaptive Harmony Search Approach for Gene Selection and Classification of High Dimensional Medical Data. *Journal of King Saud University - Computer and Information Sciences*. 10.1016/j.jksuci.2018.02.013

De Stefano, C., Fontanella, F., Marrocco, C., & Freca, A. S. (2014). A GA-based feature selection approach with an application to handwritten character recognition. *Pattern Recognition Letters*, *35*, 130–141. doi:10.1016/j.patrec.2013.01.026

Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine*, *1*(4), 28–39. doi:10.1109/CI-M.2006.248054

Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary ant lion approaches for feature selection. *Neurocomputing*, *213*, 54–65. doi:10.1016/j.neucom.2016.03.101

Fan, K., You, W., & Li, Y. (2013). An effective modified binary particle swarm optimization (mBPSO) algorithm for multi-objective resource allocation problem (MORAP). *Applied Mathematics and Computation*, *221*, 257–267. doi:10.1016/j.amc.2013.06.039

Grande, J., Del Rosario Suárez, M., & Villar, J. R. (2007). A feature selection method using a fuzzy mutual information measure. In *Innovations in Hybrid Intelligent Systems* (pp. 56–63). Springer. doi:10.1007/978-3-540-74972-1_9

Hedar, A. R., Wang, J., & Fukushima, M. (2006). Tabu search for attribute reduction in rough set theory. *Soft Comput. Fusion Found. Methodol. Appl.*, *12*(9), 909–918.

Holand, J. H. (1992). Genetic algorithms. *Scientific American*, *267*(1), 66–72. doi:10.1038/scientificamerican0792-66

Houssein, E.H., Hussien, A.G., & Hassanien, A.E. (2017). *A binary whale optimization algorithm with hyperbolic tangent fitness function for feature selection*. .10.1109/INTELCIS.2017.8260031

Hussien, A. G., Hassanien, A. E., Houssein, E. H., Bhattacharyya, S., & Amin, M. (2019). S-shaped Binary Whale Optimization Algorithm for Feature Selection. In S. Bhattacharyya, A. Mukherjee, H. Bhaumik, S. Das, & K. Yoshida (Eds.), *Recent Trends in Signal and Image Processing. Advances in Intelligent Systems and Computing*. Springer. doi:10.1007/978-981-10-8863-6_9

Hussien, G., Hassanien, A. E., & Houssein, E. H. (2017). Swarming behaviour of salps algorithm for predicting chemical compound activities. *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, 315-320. doi:10.1109/INTELCIS.2017.8260072

Inbarani, H. H., Azar, A. T., & Jothi, G. (2014). Supervised hybrid feature selection based on PSO and rough sets for medical diagnosis. *Computer Methods and Programs in Biomedicine*, *113*(1), 175–185. doi:10.1016/j.cmpb.2013.10.007 PMID:24210167

Jona, J. B., & Nagaveni, N. (2012). A hybrid swarm optimization approach for feature set reduction in digital mammograms. *WSEAS Transactions on Information Science and Applications.*, *9*, 340–349.

Jona, J. B., & Nagaveni, N. (2014). Ant-cuckoo colony optimization approach for feature selection in digital mammogram. *Pakistan Journal of Biological Sciences*, *17*(2), 266–271. doi:10.3923/pjbs.2014.266.271 PMID:24783812

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95—international conference on neural networks*, 1942-1948. doi:10.1109/ICNN.1995.488968

Khaire, U.M., & Dhanalakshmi, R. (2019). Stability of feature selection algorithm: A review. *Journal of King Saud University - Computer and Information Sciences*. 10.1016/j.jksuci.2019.06.012

Kirkpatrick, S., Gelatt, C. D. Jr, & Vecchi, M. P. (1983). Optimization by simulated annealing. *Sci.*, *220*(4598), 671–680. doi:10.1126/science.220.4598.671 PMID:17813860

Mafarja, M. M., & Mirjalili, S. (2017). Hybrid whale optimization algorithm with simulated annealing for feature selection. *Neurocomputing*, *260*, 302–312. doi:10.1016/j.neucom.2017.04.053

Mirjalili, S. (2016). SCA: A sine cosine algorithm for solving optimization problems. Knowl.-. *Based Syst.*, *96*, 120–133. doi:10.1016/j.knosys.2015.12.022

Moradi, P., & Gholampour, M. (2016). A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. *Applied Soft Computing*, *43*, 117–130. doi:10.1016/j.asoc.2016.01.044

Oh, I. S., Lee, J. S., & Moon, B. R. (2004). Hybrid genetic algorithms for feature selection. Pattern Analysis and Machine Intelligence. *IEEE Transactions.*, *26*(11), 1424–1437.

Olabiyisi Stephen, O., Fagbola Temitayo, M., Omidiora Elijah, O., & Oyeleye Akin, C. (2012). Hybrid metaheuristic feature extraction technique for solving time tabling problem. *International Journal of Scientific and Engineering Research*, *3*(8).

Pourpanah, F., Lim, C. P., Wang, X., Tan, C. J., Seera, M., & Shi, Y. (2019). A hybrid model of fuzzy min–max and brain storm optimization for feature selection and data classification. *Neurocomputing*, *333*, 440–451. doi:10.1016/j.neucom.2019.01.011

Python Software Foundation. (2019). *Python Language Reference, version 3.6.* http://www.python.org

Rizk-Allah, R. M., & Hassanien, A. E. (2018). New binary bat algorithm for solving 0–1 knapsack problem. *Complex Intell. Syst.*, *4*(1), 31–53. doi:10.1007/s40747-017-0050-z

Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics (Oxford, England)*, *23*(19), 2507–2517. doi:10.1093/bioinformatics/btm344 PMID:17720704

Schiezaro, M., & Pedrini, H. (2013). Data feature selection based on artificial bee colony algorithm. *EURASIP Journal on Image and Video Processing*, *2013*(1), 47. Advance online publication. doi:10.1186/1687-5281-2013-47

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*(4), 341–359. doi:10.1023/A:1008202821328

Talbi, E. G. (2009). *Metaheuristics: From Design to Implementation*. Wiley. doi:10.1002/9780470496916

Tang, W. C. (2007). *Feature Selection For The Fuzzy Artmap Neural Network Using A Hybrid Genetic Algorithm And Tabu Search*. USM.

Tawhid, M. A., & Dsouza, K. B. (2018). *Hybrid Binary Bat Enhanced Particle Swarm Optimization Algorithm for solving feature selection problems*. Appl. Comput. Informatics., doi:10.1016/j.aci.2018.04.001

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2016). A Survey on Evolutionary Computation Approaches to Feature Selection. *IEEE Transactions on Evolutionary Computation*, *20*(4), 606–626. doi:10.1109/TEVC.2015.2504420

Yan, C., Ma, J., Luo, H., & Patel, A. (2019). Hybrid binary Coral Reefs Optimization algorithm with Simulated Annealing for Feature Selection in high-dimensional biomedical datasets. *Chemometrics and Intelligent Laboratory Systems*, *184*, 102–111. doi:10.1016/j.chemolab.2018.11.010

Yang, X., & Deb, S. (2009). *Cuckoo search via levy flights. In: 2009 world congress on nature biologically inspired computing*.

Yi, Y., & He, R. (2014). A novel artificial bee colony algorithm. *Proceedings of 2014 sixth international conference on intelligent human–machine systems and cybernetics*, 271–274. doi:10.1109/IHMSC.2014.73

Zorarpacı, E., & Özel, S. A. (2016). A hybrid approach of differential evolution and artificial bee colony for feature selection. *Expert Systems with Applications*, *62*, 91–103. doi:10.1016/j.eswa.2016.06.004

Mohd Faizan is currently working as Assistant Professor in the Department of Computer Application, Integral University, Lucknow, India. He did his Ph.D. from Department of Information Technology, Babasaheb Bhimrao Ambedkar University, Lucknow, India.. His research interests include evolutionary intelligence, optimization algorithms, intelligent systems, machine learning, Internet crime and dark web.

Fawaz Alsolami received the M.A.Sc in Electrical and Computer Engineering from University of Waterloo, Canada, in 2008, and his Ph.D. degree in Computer Science from KAUST University, Thuwal, Saudi Arabia, in 2016. Fawaz joined computer science at King Abdulaziz University as an assistant professor of Computer Science. His research interests are artificial Intelligence, machine learning and data mining, and combinatorial optimization. He also published many articles and one monograph. He has been the chairman of the Computer Science department at King Abdulaziz University since 2018.

Raees Ahmad Khan is currently a Professor, the Head of the Department with the Department of Information Technology, Babasaheb Bhimrao Ambedkar University, (A Central University), Lucknow, India. He has more than 20 years of teaching and research experience. He has published a number of national and international books, including Chinese Language, Technical Article, Research Papers, Reviews and Chapters on Software Security, Software Quality, and Software Testing. His research interests include software security, software quality, and software testing.