A Hybrid Algorithm for Feature Selection and Classification

*B. R. Sathish, Radha Senthilkumar**

Department of Information Technology, Anna University, India
brsathishap@gmail.com, radhasenthilkumar@gmail.com

Abstract

With a recent spread of intelligent information systems, massive data collections with a lot of repeated and unintentional, unwanted interference oriented data are gathered and a huge feature set are being operated. Higher dimensional inputs, on the other hand, contain more correlated variables, which might have a negative impact on model performance. In our model a Hybrid method of selecting feature was developed by combining Binary Gravitational Search Particle Swarm Optimization (HBGSPSO) method with an Enhanced Convolution Neural Network Bidirectional Long Short Term Memory (ECNN-BiLSTM). In our proposed system, the Bidirectional Long Short Term Memory (BiLSTM) is introduced which extracts the hidden dynamic data and utilizes the memory cells to think of long-term historical data after the convolution process. In this paper, thirteen well-defined datasets are used from the machine learning database of UC Irvine to evaluate the efficiency of the proposed system. The experiments are conducted using K Nearest Neighbor (KNN) and Decision Tree (DT) which are used as classifiers to evaluate the outcome of selected features. The outcomes are contrasted and compared with the bio-enlivened calculations like Genetic Algorithm (GA), Grey Wolf Optimizer (GWO), and Optimization protocol using Particle Swarm Optimization (PSO).

Keywords: Feature Selection, Genetic Algorithm, Convolution Neural Network, Particle Swarm Optimization, Grey Wolf Optimizer

1 Introduction

Feature Selection (FS) has a significant region in data mining, which selects the subset of necessary features in the model designing. Complex features of data bring out the difficulty in data by achieving the common FS method for complex data [1]. The feasible FS methodology for big data in terms of specific data has been used to manage and analyze the application. FS contains static missing, big, heterogeneous, imbalanced, dynamic, and unreliable data. So, an essential source that has a direct influence on the effectiveness of knowledge discovery and other relevant activities would be how to pick a new function subset effectively [2]. The standard FS process is classified into four main phases, such as the production of subsets, assessment, outcome measures, and confirmation [3]. A subset is considered to be the search problem aimed at selecting the best subset of all-accessible. The validation is for assessing the quality result driven with the support of requirements for knowledge update; we have analyzed and represented the Bio-inspired algorithms for feature selection with the variation of numerous data [4]. The motive is to choose a subset of the actual high-dimensional features based on some performance aspects. Thus, they can save the actual features and generate a dimensionally decreased outcome that is more interpretable for domain experts [5]. Then the low-dimensional features have been selected and during the data acquisition, one should need to gather or evaluate the selected features. This will be useful when the measurement of the whole feature is at a higher rate or practically not possible [6].

Algorithms that are affected by the biological evolution of actions are used by algorithms that are bio-inspired. They prove to be better than the classical machine learning algorithms, rather they can define the optimal solution of complex issues [7]. FS (selection of variables or attributes) is an approach to data mining aimed at choosing an optimal subset that reduces the best output about the well-organized scenario [8]. Here, a feature is a data attribute that denotes unique requirements for the data function. Since FS performs well for deriving the model as well as reducing the variance, researchers can infer and know the data pattern model more easily with the help of FS [9]. A good FS method must be accomplished in selecting a various high range of correlation as well as the optimal classification outcomes [10].

Compared with conventional data certain significant data has to be pointed, in order to gather relevant data from complex data concerning three characteristics, the traditional FS methods manage the following threefold challenge in the event of big data [11]. Initially, conventional FS methods require a significant amount of learning time, hence it is difficult for information processing to keep up with the big data transition. Big data not only has an enormous amount of irrelevant characteristics but also contains irrelevant factors with variant degrees and shapes, which greatly leads to the high rationality of choosing characteristics [12]. Due to various means of retrieval, or even failure, some data is forged, which further increases the difficulty of FS [13].

Current FS approaches face demanding challenges in various stages due to the characteristics of big data, such as the processing of speed of data, outlining knowledge implications, and managing with inadequate or noisy data [14]. Thus, it is of great urgency to research specific FS

*Corresponding Author: Radha Senthilkumar; E-mail: radhasenthilkumar@gmail.com
DOI: 10.53106/160792642023052403004

techniques for big data. The methods available, however, are highly specific, and how to derive useful data based on resolving and analyzing these data poses severe problems [15]. Convolutional neural network (CNN) develops a simple auto-encoder to execute FS based on the reconstruction error [16]. It needs to refer to a simple network structure to confirm the error can be back propagated and the best feature can be studied [17]. The significant thing is that the simple network is also utilized for the actual data reconstruction. Such a network cannot design the complex manifold of data and proceed to high reconstruction error and non-optimal FS [18].

If a feature has mostly missing values, the feature itself can also be eliminated [19]. Next, the feature selection technique which selects the important data using Hybrid Binary Gravitational Search Particle Swarm Optimization (HBGSPSO) has been proposed. Generally, the search space of FS issues may be updated for every single dataset [20]. Hence, easy but effective concepts are highly emerging to alleviate the limitations of the gravitational search (GS) [21]. CNN is a deep learning algorithm which are comprised of input, hidden and an output layer that connects each node to another and has a weight and threshold associated with it [22]. The pre-processing needed in a convolution network is much lower as compared to the other classification algorithms [23]. Here, we use five layers such as two Convolution one dimensional layer and three BiLSTM layers. Generally, CNN is applied to the data classification that has kernels of several convolutions for extracting local features of different sizes and Long Short Term Memory is well suited for classifying, processing and making predictions on time series data [24]. Many crossover operators are problem-dependent and have different search abilities. Thus, it is a challenge to select the most efficient one to solve different feature selection problems, especially when the nature of feature selection problems is unknown in advance [25]. PSO-based feature selection approach, which can continuously improve the quality of the population at each iteration. Specifically, a correlation guided updating strategy based on the characteristic of data is developed, which can effectively use the information of the current population to generate more promising solutions [26-27]. Usually, FS is modelled as a bi-objective optimization problem whose objectives are classification accuracy and number of features. One of the main issues in real-world applications is missing data. Databases with missing data are likely to be unreliable [28-29]. In our proposed system, the BiLSTM is introduced which extracts the hidden dynamic data and utilize the memory cells to think long-term historical data after the convolution process.

2 Problem Formulation

The proposed study includes two new bio-inspired algorithm strategies for feature selection as well as classification. In Figure 1, the input is given as a dataset, and dataset is pre-processed. During pre-processing, it is very normal to have missing values in the given input dataset and also the irrelevant data which needs to be handled followed by feature selection using HBGSPSO and classification using CNN-BiLSTM Neural Network. BiLSTM is a technique for making Neural Network where the flow of information will happen in both forward and backward direction, through which we can preserve both the past and future information. Gravitational search algorithm (GSA) is an optimization algorithm which considers features as a set of masses which communicates with each other according to Newtonian gravity and laws of motion. PSO is a stochastic optimization technique, each particle in the swarm looks for its positional coordinates in the solution space, which are associated with the best solution that has been achieved so far by that particle. It is known as pbest or personal best. Another best value known as gbest or global best is tracked by the PSO. It is one of the simple and sometimes effective criteria. First, there is a lack of a defined but effective plan for transitioning from broad exploration to narrow exploitation. Next, an unstable equilibrium between discovery and exploitation is obtained, and then, searching at a progressive integration rate is established.

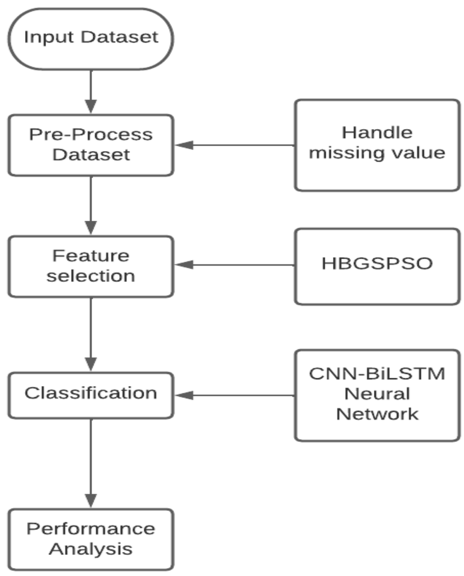

Figure 1. Overall flow of the proposed HBGSPSO with ECNN-BiLSTM

The disadvantages originate from the method's ineffective proclivity in the last iteration, which is dedicated to the execution phase, while the initial two iterations can experience stalling and insensitive major contractions in the middle of the iterations, lowering the quality of the final approaches significantly. For the first time in the segment, an improved hybrid binary version of GS is suggested. The key constraint is to enhance the efficiency of GS in resolving FS issues significantly while maintaining GS advantages [30-31]. As a result, we proposed that the GS, as well as the convergence rate, need to be improved. The logarithmic decreasing function is used in the procedure to achieve a steady change in the gravitational constant. The notion of social thinking in PSO has been applied here, and it is combined with the GS algorithm. To help the global best solution further improve the consistency of results and prevent stagnation to local optima, a mutation operator is implemented. Finally, we introduced the classification operation using an Enhanced Convolution Neural Network and Bidirectional Long Short Term Memory (ECNN-BiLSTM) to categorize the selected features following the feature selection procedure.

There are four input layers, convolution one-dimensional layer (COV1D), BiLSTM, and output layer in the Figure 2. CNN is a deep learning algorithm that can yield input data, assign significance to various aspects in the data, and be capable of differentiating one from another. The pre-processing needed in a convolution network is much lower as compared to the other classification algorithms. Here, we use five layers such as two COV1D and three BiLSTM layers. Generally, CNN is applied to the data classification with multiple convolution layers the extraction of local features of different sizes and Long short term memory (LSTM) rectifies the temporal relationship between data which increases the accuracy of dynamic data. In our system, the BiLSTM is introduced which extracts the hidden dynamic data and utilizes cells to think long term historical data after the convolution process. The performance measures are evaluated using the formulation and accuracy is compared with the classifier algorithms including Decision Tree and K Nearest Neighbor.

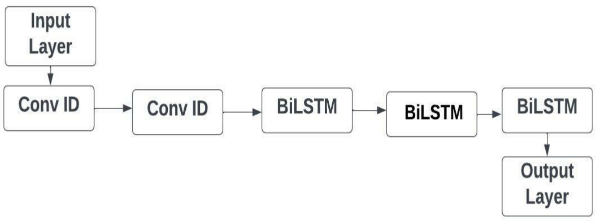

Figure 2. Neural network architecture

In the GSA process, the population's initialization is generated at random by n individuals, with the ith individual formulated as $x_i = (x_i^1, x_i^2, ..., x_i^d)$ is its position with respect to the d^{th} dimension. Then in the *d-th* dimension, individuals x_i and x_j interact with each other through the gravitational force $G_{ij}^d(t)$ in iteration t, denoted as,

$$G_{ij}^d(t) = P(t)\frac{H_i(t) \times H_j(t)}{K_{ij}(t) + €}(x_j^d(t) - x_i^d(t)). \tag{1}$$

Where P(t) denotes the gravitational constant for iteration t, $H_i(t)$ and $H_j(t)$ are the two individual masses. $K_{ij}(t)$ denotes the individual's Euclidean distance. The factor € is maintained a constant. The gravitational constant P (t) is described as

$$P(t) = P_0 \times e^{-\infty \frac{t}{T}}. \tag{2}$$

Where P_0 represents initial value and ∞ denotes the constant. T and t denote the maximum iteration and the current iteration. The mass $H_i(t)$ of variable x_i is expressed as follows

$$h_i(t) = \frac{g_i(t) - w(t)}{b(t) - w(t)}. \tag{3}$$

$$H_i(t) = \frac{h_i(t)}{\sum_{i=1}^{n} h_i(t)}. \tag{4}$$

Where $g_i(t)$ denotes the fitness value of individual, the worst and best fitness values of the current population in iteration t are represented by $w(t)$ and $b(t)$. For an individual x_i, the total gravitational force $G_{ij}^d(t)$ from the other individual in the dth dimension is calculated as follows.

$$G_{ij}^d(t) = \sum_{j \in K_{b, j \neq i}} rand_j G_{ij}^d(t). \tag{5}$$

Where K_b denotes the current population, there are K best individuals and from n to 2, the K value decreases in a linear fashion. $rand_j$ is an arbitrary value of the form [0,1] for the individual x_j. The acceleration

$$a_t^d(t) = \frac{G_{ij}^d(t)}{H_i(t)}. \tag{6}$$

Finally, the velocity $v_i^d(t+1)$ of the individual x_i is updated to rearrange its position in the next iteration t+1, given as follows.

$$v_i^d(t+1) = rand_i v_i^d(t) + a_i^d(t). \tag{7}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1). \tag{8}$$

Where $rand_i$ indicates an arbitrary value of the form [0, 1] for the individual x_i.

The PSO is a widely used approach derived from the exchange of knowledge among birds. The PSO first created a random population of particles, which then transferred at a specified velocity depending on an interchange with the population's particles.

Every particle is denoted by a D-dimensional vector, which is randomly initialised with binary values.

$$x_i = (x_{i1}, x_{i2}, ..., x_i D) \in S_s. \tag{9}$$

where S_s denotes the search space available.

The velocity is denoted by a D dimensional vector and is instantiated to zero

$$v_i = (v_{i1}.v_{i2}, ..., v_i D). \tag{10}$$

Each particles best recorded personal position is saved as

$$xp_i = (p_{i1}, p_{i2}, ..., p_i D) \in S_s. \quad \textbf{(11)}$$

At each repetition, each particle updated its position based on its personal best and the global best as follows:

$$\sum_i^{S_l} v_i^{t+1} = wv_i^t + c_1 rand_{i1} (Pbest_i(t) - x_i^{(t)} + c_2 rand_{i2} \cdot gbest - x_i^{(t)}. \quad \textbf{(12)}$$

Where c_1 and c_2 are acceleration constant. $rand_1$ and $rand_2$ are random values between [0, 1]. w represents the weight. It regulates how the particles prior velocity influences the velocity in next iteration. Then w is calculated by the following expression.

$$w = w_{\max} - iter \cdot \left(\frac{w_{\max} - w_{\min}}{Max_{iter}} \right). \quad \textbf{(13)}$$

Where $w_{\max}$ and $w_{\min}$ are constants. Max_{iter} is the maximum number of iterations.

The S-shaped output signal, which outlines the continuous velocity value to the particle in the swarm, determines the velocity of each particle. It's a sigmoid function that boosts the PSO.

$$s = \frac{1}{1+e^{-v_{i,j}}}, i = 1, ..., S_l. \; j = 1, ..., D. \quad \textbf{(14)}$$

$$X(i,j) = \begin{cases} 1 & if\,(random < s \\ 0 & otherwise \end{cases}. \quad \textbf{(15)}$$

In Binary Valued of the GSA (BGSA) the feature selection maximization problem is handled in binary space, the direction update in the binary value of 0 or 1. In Hybrid GSA (HGSA) which combines real valued and binary valued GSA, which uses real valued parameters to optimize the issue in continuous space and the direction are represented as difference in places of continuous proportions.

The flowchart in Figure 3 and Algorithm 1 stated that initially divide the given data set into two set trains and test sets. Then set the initial swarm size and dimension. The computation of the acceleration constant has been done and initializes the population randomly for each solution. The fitness evaluation will be computed and the value for all the solutions and also the velocity of the particles has been updated. At last, a mutation operator is designed to aid the global best solution to improvise the consistency of findings and preventing recession to local optima.

Algorithm 1. HBGSPSO

```
Divide the given data set into two set trains and test set {80:20 ratios}
Set the initial swarm size S_l(N) and set dimension D
Calculate Acceleration constants c_1, c_2, w_max, w_min, Max_iter, P_c, P_m
Initialize the population randomly as x according to the equation 9 for every result and initialize velocity vector 'v' as Dimension-zero vectors according to equation 10.
Assign t = 0
while t < Max_iter do
Compute fitness for all solutions using eq (5) and (6)
Update the value of P_best and g_best
    w = w_max - t((w_max - w_min)/Max_iter).
Update velocity of the particles using eq (12)
for i in S_l do
for j in D do
if (v(i, j) > v_max) then
        v(i, j) > v_max
end if
if (v(i, j) < v_max) then
        v(i, j) = - v_max
position updation using eq (14) and (15)
end for
end for
increment t by 1
end while
return g_best
```

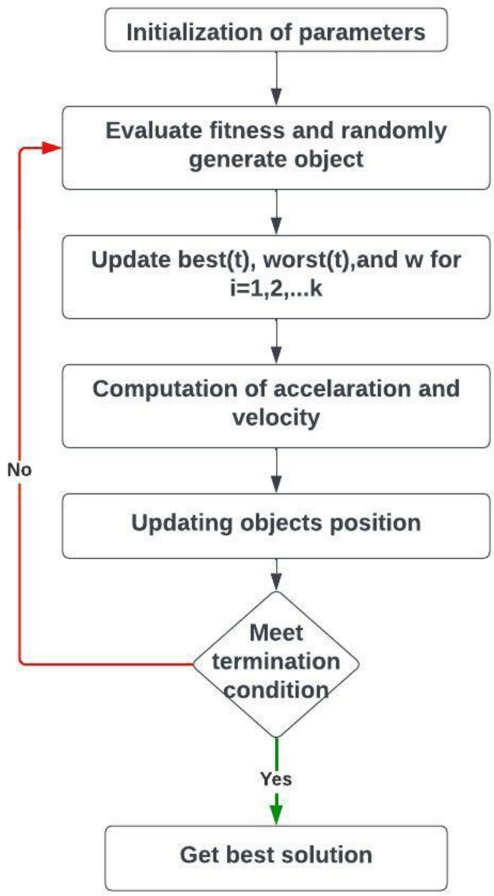

Figure 3. Flowchart for HBGSPSO

3 Experiment

The performance of HBGSPSO was tested on thirteen datasets taken from the UCI machine learning repository, ten cross fold validation is applied on the data. Where nine folds are used for training and validation purpose and one fold is used for testing purpose. The effect of the HBGSPSO with ECNN-BiLSTM is defined in this section. In this proposed job, the comparison is executed with two classifiers. The HBGSPSO findings were contrasted with the HGSA outcome. Since two classifiers are used to test the subsets of features and the effects within each classifier have been independently measured. Finally, HBGSPSO's findings on the KNN classifier were matched with that of HBGSPSO's findings on the DT classifier.

A system with an Intel Core i7 processor and 8GB RAM was utilized in all simulations. Table 1 shows all of the parameters used in this algorithm. Since the studies were based on algorithms, we recorded the average result for each data set. All of the algorithms in this architecture have the same iterations and population size for the entire system. To do the performance analysis, 13 distinct datasets from the UC Irvine database were used, with datasets with varying features being chosen.

Table 1. Parameter for proposed work

List of parameters
Number of max iteration = 100
move rate 0.5,
c1, c2 = 1, 1,
vmax search range = 4
Number of population- 30
Elite check = 1
Value between mass = 1
Value of kbest = 25

For comparative study, three performance measures are used, like classification accuracy, selection of features, and fitness value.

Average classification accuracy: This variable assesses the accuracy of the classifier while using the selected features to classify the correct class. The average accuracy is calculated as follows:

$$AvgAccuracy = \frac{1}{M}\sum_{j=1}^{M}\frac{1}{N}\sum_{i=1}^{N}(C_i = L_i). \quad \textbf{(16)}$$

Where, M seems to be the number of iterations for an algorithm to identify the trailing subset of feature.

Total number of dataset instances is given by N.

Class predicted C_i, and the initial class L_i.

Average Fitness: This parameter is a combination of the feature reduction rate and the error rate for KNN and DT. In the optimal solution, it's also employed as an objective function to find the appropriate subset. The following formula is used to obtain the average fitness value:

$$AvgFitness = \frac{1}{M}\sum_{j=1}^{M} fit_*^i. \quad \textbf{(17)}$$

Where N indicates the number of iterations fit_*^i is the fitness solution for i.

Average selection size: This shows how well an algorithm performed as per the selection size during the feature selection issue rectification. The following formula is used to determine this:

$$AvgAccuracy = \frac{1}{m}\sum_{i=1}^{M}\frac{d_i^*}{D}. \quad \textbf{(18)}$$

The performance measures are explained and the formulation was given in the above equation.

The significant features are identified from the number of selected characteristics. The correlation of HGSA and HBGSPSO over the minimum set of relevant features on all datasets is shown in Table 2. It should be noted that HBGSPSO substantially outshines the number of selected features when evaluating the recorded outcome. As can be seen in Table 2, in 89 percent of the dataset, the suggested work produced the best results. HBGSPSO also demonstrated superior performance over the HGSA method while using the KNN classifier. In 68 percent of the dataset, HBGSPSO was able to achieve a modest reduction, while in the remaining datasets, HGSA performed best.

Table 2. Selected features for KNN classifier

Dataset	KNN classifier					
	GWO	PSO	GA	BGSA	HGSA	HBGSPSO
BreastCancer	5.9	6.2	3.8	4.3	4.1	4
BreastEW	19.9	17	16.1	16.2	13.8	14
CongressEW	8.4	6.4	4.8	5.4	3.9	5
HeartEW	11.7	7.9	6.9	6.4	6.7	7
IonosphereEW	18.2	14.5	15.2	10.9	8.2	9
KrvskpEW	26.8	19.3	19.5	17	15.7	15
Lymphography	8.9	9.2	7.6	7.4	7	7
SonarEW	31.9	29.9	29.5	26	25.4	22
SpectEW	11.6	10.6	11.2	11.6	8	10
TicTacToe	8.4	7.2	5.8	8.8	8.6	9
WaveformEW	34.9	22.3	21.4	20	19.5	18
WineEW	7.8	7.6	6.9	6.2	5	5
Zoo	9.7	9	8.3	7.8	5.6	6

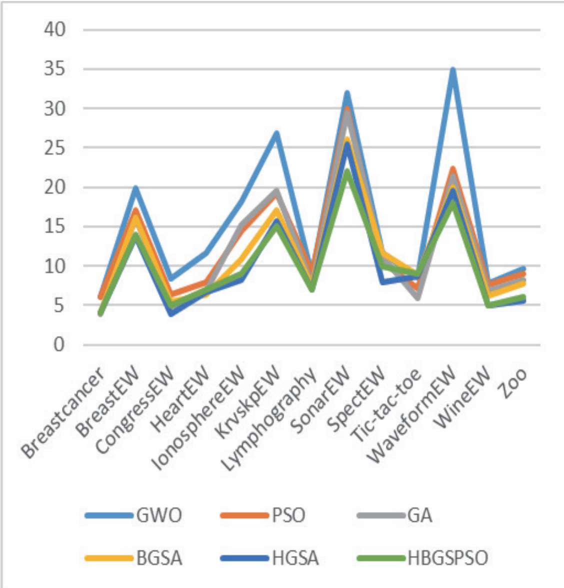

Figure 4. Selected features for KNN classifier

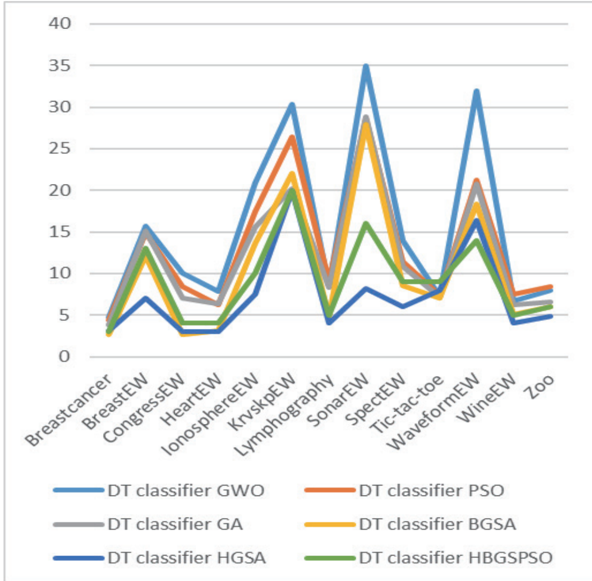

Figure 5. Selected features for DT classifier

The above Figure 4 represents the graphical representation of Table 3 which shows the line graph and the selected features have the minimal reduction KNN classifier by comparing both HGSA and HBGSPSO, where y axis denotes the number of features.

The amount of chosen attributes is a critical part of FS methodologies. Table 3 shows the relationship of HGSA and HBGSPSO over the entire number of chosen highlights across all datasets for Decision Tree classifier. While surveying the recorded yield, found to be seen that HBGSPSO model beat the quantity of chose highlights. HBGSPSO also demonstrated superior efficiency over the HGSA method when using DT classifier. In the 69 percent dataset, HBGSPSO was able to achieve the highest reduction, while HGSA performed best in the remaining datasets. These findings reflect HBGSPSO's ability to classify the most important characteristics through search space better than HGSA.

Table 3. Selected features for DT classifier

Dataset	DT classifier					
	GWO	PSO	GA	BGSA	HGSA	HBGSPSO
Breastcancer	4.7	4.4	3.8	2.6	3.1	3
BreastEW	15.7	14.9	15.1	12.1	7.1	13
CongressEW	10	8.4	7.1	2.7	3	4
HeartEW	7.9	6.2	6.4	3.1	3	4
IonosphereEW	20.9	17.4	15.6	13.6	7.5	10
KrvskpEW	30.3	26.4	20.2	22	20	20
Lymphography	8.4	9.1	8.3	4.8	4.1	5
SonarEW	35	28.6	28.9	27.9	8.2	16
SpectEW	14	11.4	10.9	8.5	6	9
Tic-tac-toe	7.3	7.4	7.2	7	8	9
WaveformEW	32	21.2	20.8	18.3	16.4	14
WineEW	6.7	7.5	6.2	5.1	4	5
Zoo	8	8.4	6.6	6	4.9	6

The above Figure 5 represents the graphical representation of Table 3 which shows the line graph and the selected features have the maximum reduction DT classifier by comparing both HGSA and HBGSPSO, where y axis represents the number of features.

4 Results

Table 4 and Figure 6 shows the accuracy classification of the proposed work for the KNN classifier. The values in the Figure 6 are scaled to 1. The highlighted value clearly displays that HBGSPSO has the better accuracy for the given 13 datasets. Moreover, it outperforms BGSA and HGSA.

Table 4. Accuracy comparisons for KNN classifier in percentage

Dataset	KNN classifier					
	GWO	PSO	GA	BGSA	HGSA	HBGSPSO
Breastcancer	**98**	97	97	97	97	98
BreastEW	95	94	95	**97**	97	98
CongressEW	92	95	94	**96**	96	97
HeartEW	85	79	83	**86**	85	88
IonosphereEW	89	88	87	90	**93**	95
KrvskpEW	95	96	93	97	**97**	98
Lymphography	83	75	81	88	**89**	92
SonarEW	79	76	85	89	**95**	97
SpectEW	83	81	86	89	**91**	93
Tic-tac-toe	79	78	75	**79**	78	86
WaveformEW	78	78	76	**81**	81	83
WineEW	98	93	96	**99**	98	99
Zoo	92	81	93	**99**	93	100

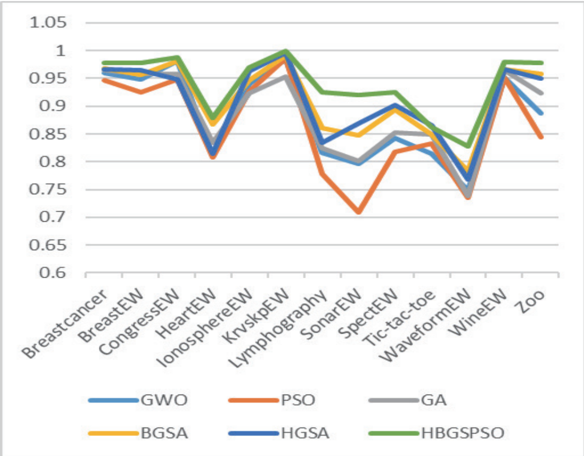

Figure 6. Comparison of accuracy for KNN classifier

Table 5. Accuracy comparisons for DT classifier in percentage

Dataset	DT classifier					
	GWO	PSO	GA	BGSA	HGSA	HBGSPSO
Breastcancer	96	94	96	**96**	96	97
BreastEW	94	92	95	95	**96**	97
CongressEW	**98**	94	95	98	94	98
HeartEW	80	80	83	**86**	81	87
IonosphereEW	94	92	92	94	**96**	97
KrvskpEW	98	98	95	99	**99**	99
Lymphography	81	77	82	**86**	83	92
SonarEW	79	70	80	84	**86**	92
SpectEW	84	81	85	89	**90**	92
Tic-tac-toe	81	83	85	84	**86**	86
WaveformEW	74	73	73	**78**	76	82
WineEW	95	95	96	**96**	**96**	98
Zoo	88	84	92	**95**	95	97

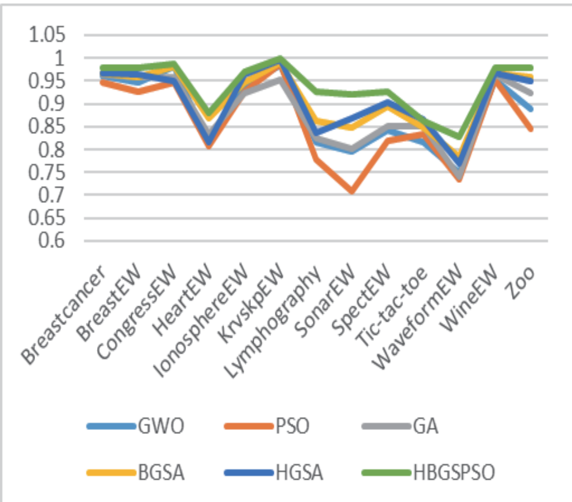

Figure 7. Comparisons of accuracy for DT classifier

The above Table 5 and Figure 7 depicts the accuracy classification of the proposed work (HBGSPSO) for the DT classifier. The values in the Figure 7 are scaled to 1. The highlighted value clearly displays that HBGSPSO has the better accuracy in the given dataset.

Table 6. Accuracy and execution time based on the population size

(a) Based on the population size (100)

Population size/iteration	100			
	Selected Features (SF)	SF count	Accuracy	Estimated time (sec)
5	110001111011	8	0.925926	3.443085
7	10011010011	6	0.944444	4.467649
10	1111011100011	9	0.925926	6.078099
20	110010111011	8	0.925926	9.250468
30	1110011110011	9	0.944444	11.7489

(b) Based on the population size (150)

Population size/iteration	150			
	Selected Features (SF)	SF count	Accuracy	Estimated time (sec)
5	1000111100011	7	0.925926	3.508795
7	10011010011	6	0.944444	4.155865
10	10011010011	6	0.944444	7.171657
20	110011110010	7	0.944444	10.01144
30	10011010011	6	0.944444	7.171657

Table 7. Accuracy and execution time based on the population size

(a) Based on the population size (25)

Population size/iteration	25			
	Selected Features (SF)	SF count	Accuracy	Estimated time (sec)
5	1010010111111	9	0.92593	1.26796
7	110101111011	9	0.92593	1.68409
10	10111111011	9	0.92593	2.298796
20	110011110010	7	0.94444	4.49113
30	1011011110011	9	0.925926	6.12943

(b) Based on the population size (50)

Population size/iteration	150			
	Selected Features (SF)	SF count	Accuracy	Estimated time (sec)
5	10011010011	6	0.944444	2.251476
7	111001111111	10	0.925926	3.147556
10	10011010011	6	0.944444	3.688807
20	11001111011	8	0.925926	6.864299
30	1010000110011	6	0.925926	7.955826

Table 6(a), Table 6(b), Table 7(a) and Table 7(b) show the Selection feature count and also the execution time for a particular population size with the number of iteration. For each iteration, the SF count and the execution time varies with the help of the feature selection approaches.

Table 8. Comparison of HBGSPSO with KNN to other techniques in terms of average classification accuracy

Dataset	GA	PSO	GWO	HBGSPSO
Breastcancer	95	94	97	98
BreastEW	92	93	92	98
CongressEW	89	93	93	97
HeartEW	73	74	77	88
IonosphereEW	86	87	80	95
KrvskpEW	94	94	94	98
Lymphography	75	75	74	92
SonarEW	83	80	73	97
SpectEW	75	73	82	93
Tic-tac-toe	76	75	72	86
WaveformEW	71	73	78	83
WineEW	94	93	93	99
Zoo	94	96	87	100

Table 9. Comparison of HBGSPSO with DT to other approaches with respect to average classification accuracy

Dataset	GWO	GA	PSO	HGSA	HBGSPSO
Breastcancer	97	96	96	97	98
BreastEW	93	93	93	97	98
CongressEW	93	93	92	96	97
HeartEW	77	78	78	85	88
IonosphereEW	83	81	81	93	95
KrvskpEW	95	92	94	97	98
Lymphography	70	69	74	89	92
SonarEW	72	75	73	95	97
SpectEW	82	79	82	91	93
Tic-tac-toe	72	71	73	78	86
WaveformEW	78	77	76	81	83
WineEW	92	93	93	98	99
Zoo	87	85	86	93	100

After reviewing the outputs of HBGSPSO and comparing it with existing procedures, the findings are correlated to other clearly-defined wrapping techniques. In terms of the consequences of classification accuracy, Table 8 and Table 9 compares the suggested approach to GWO, PSO, GSA, and HGSA and the results are listed, accordingly.

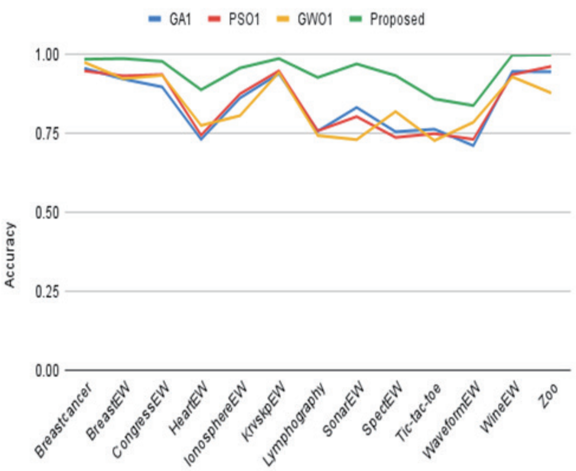

Figure 8. Average accuracy classification

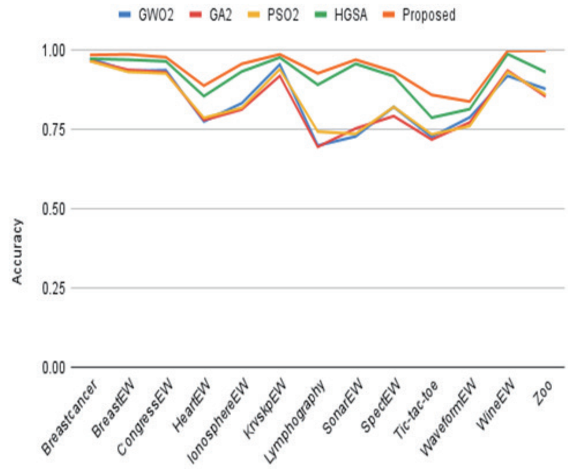

Figure 9. Average accuracy classification

In Figure 8 and Figure 9, the proposed work's accuracy rate is compared to that of various methodologies by scaling the table value of 8 and 9 to 1 respectively. Analyzing the outcome in the Table 8, Table 9, Figure 8 and Figure 9 above, it can be said that in interacting of almost all datasets, the HBGSPSO system can outperform other existing strategies. On all the data sets, the suggested wrapper achieved better results than the GA process. It is clear, though, that GWO, PSO and HGSA will be unable to demonstrate the superior outcome of challenging any of these datasets.

We compare our method with four latest existing methods like Laplacian Score (LS), Mutual Information (MI), Based on ReliefF (ReliefF), Fuzzy joint Mutual information (FJMI), Binary PSO and GSA (BPSOGSA), and found that HBGSPSO outperforms all the four methods in terms of accuracy as given in the following Table 10.

Table 10. Comparison of HBGSPSO four existing approaches with respect to average accuracy classification

Method name	Average accuracy on selected dataset
LS	0.9238
MI	0.8748
ReliefF	0.8958
FJMI	0.7962
BPSOGSA	0.8914
BPSOGSA	0.9345
HBGSPSO	0.9444

5 Conclusion

In this paper, an HBGSPSO with ECCN-BiLSTM was proposed to handle feature selection as well as classification. We substantially simulated the bio-inspired algorithm in a deep learning strategy. Various problems are listed out and solved using novel hybrid search optimization and enhanced neural network techniques and compared with the KNN and DT classifier. To assess the consistency of the effectiveness, we evaluated the proposed HBGSPSO with both KNN and DT classifiers as well as other standard approaches such as

GS, PSO, GWO, BGSA and HGSA methods. Based on 13 benchmark datasets HBGSPSO achieved the best result in accuracy, HBGSPSO outperforms the other five methods by the range from 1.5% - 9.8% using KNN, 0.9% – 9.3% using DT respectively. HBGSPSO achieves highly qualified results than the other five methods with respect to the average classification accuracy by range from 0.01% - 5.7%. The result revealed that the proposed approaches compared with other significant methods outperform the accuracy and execution time with the selected features and the dynamic data are used for the process with thirteen different datasets. Here, we use a hybrid and enhanced classifier to classify the accuracy. In the future, this system will be evaluated using image and multimedia approaches.

References

[1] W. Shu, W. Qian, Y. Xie, Incremental approaches for feature selection from dynamic data with the variation of multiple objects, *Knowledge-Based Systems*, Vol. 163, pp. 320-331, January, 2019.

[2] M. Rong, D. Gong, X. Gao, Feature Selection and Its Use in Big Data: Challenges, Methods, and Trends, *IEEE Access*, Vol. 7, pp. 19709-19725, January, 2019.

[3] A. Yadav, D. K. Vishwakarma, A comparative study on bio-inspired algorithms for sentiment analysis, *Cluster Computing*, Vol. 23, No. 4, pp. 2969-2989, December, 2020.

[4] H. Liu, B. Lang, M. Liu, H. Yan, CNN and RNN based payload classification methods for attack detection, *Knowledge-Based Systems*, Vol. 163, pp. 332-341, January, 2019.

[5] H. M. Zawbaa, E. Emary, C. Grosan, V. Snasel, Large-dimensionality small-instance set feature selection: A hybrid bio-inspired heuristic approach, *Swarm and Evolutionary Computation*, Vol. 42, pp. 29-42, October, 2018.

[6] J. Yang, V. Honavar, Feature Subset Selection Using a Genetic Algorithm, *IEEE Intelligent Systems and their Applications*, Vol. 13, No. 2, pp. 44-49, March-April, 1998.

[7] E. Emary, H. M. Zawbaa, A. E. Hassanien, Binary grey wolf optimization approaches for feature selection, *Neurocomputing*, Vol. 172, pp. 371-381, January, 2016.

[8] Z. Chen, X. Yuan, H. Tian, B. Ji, Improved gravitational search algorithm for parameter identification of water turbine regulation system, *Energy Conversion and Management*, Vol. 78, pp. 306-315, February, 2014.

[9] V. R. Balasaraswathi, M. Sugumaran, Y. Hamid, Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms, *Journal of Communications and Information Networks*, Vol. 2, No. 4, pp. 107-119, December, 2017.

[10] S. K. Baliarsingh, S. Vipsita, K. Muhammad, B. Dash, S. Bakshi, Analysis of high-dimensional genomic data employing a novel bio-inspired algorithm, *Applied Soft Computing*, Vol. 77, pp. 520-532, April, 2019.

[11] A. E. Hegazy, M. A. Makhlouf, G. S. El-Tawel, Improved salp swarm algorithm for feature selection, *Journal of King Saud University-Computer and Information Sciences*, Vol. 32, No. 3, pp. 335-344, March, 2020.

[12] H. M. Zawbaa, E. Emary, C. Grosan, Feature Selection via Chaotic Antlion Optimization, *PLOS ONE*, Vol. 11, No. 3, Article No. e0150652, March, 2016.

[13] E. Emary, H. M. Zawbaa, A. E. Hassanien, Binary ant lion approaches for feature selection, *Neurocomputing*, Vol. 213, pp. 54-65, November, 2016.

[14] S. L. Marie-Sainte, N. Alalyani, Firefly algorithm based Feature Selection for Arabic Text Classification, *Journal of King Saud University-Computer and Information Sciences*, Vol. 32, No. 3, pp. 320-328, March, 2020.

[15] H. M. Zawbaa, E. Emary, B. Parv, M. Sharawi, Feature selection approach based on moth-flame optimization algorithm, *2016 IEEE Congress on Evolutionary Computation (CEC)*, Vancouver, BC, Canada, 2016, pp. 4612-4617.

[16] V. R. E. Christo, H. K. Nehemiah, B. Minu, A. Kannan, Correlation-Based Ensemble Feature Selection Using Bioinspired Algorithms and Classification Using Backpropagation Neural Network, *Computational and mathematical methods in medicine*, Vol. 2019, Article No. 7398307, September, 2019.

[17] A. Chinnaswamy, R. Srinivasan, Hybrid Feature Selection Using Correlation Coefficient and Particle Swarm Optimization on Microarray Gene Expression Data, in: V. Snášel, A. Abraham, P. Krömer, M. Pant, A. Muda (Eds.), *Innovations in Bio-Inspired Computing and Applications*, Vol. 424, Springer, Cham, 2015, pp. 229-239.

[18] S. Fong, R. Wong, A. V. Vasilakos, Accelerated PSO Swarm Search Feature Selection for Data Stream Mining Big Data, *IEEE Transactions on Services Computing*, Vol. 9, No. 1, pp. 33-45, January-February, 2016.

[19] S. S. Naqvi, W. N. Browne, C. Hollitt, Feature Quality-Based Dynamic Feature Selection for Improving Salient Object Detection, *IEEE Transactions on Image Processing*, Vol. 25, No. 9, pp. 4298-4313, September, 2016.

[20] X. Wu, K. Yu, W. Ding, H. Wang, X. Zhu, Online Feature Selection with Streaming Features, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 5, pp. 1178-1192, May, 2013.

[21] K. Yu, X. Wu, W. Ding, J. Pei, Towards Scalable and Accurate Online Feature Selection for Big Data, *2014 IEEE International Conference on Data Mining*, Shenzhen, China, 2014, pp. 660-669.

[22] S. Jin, F. Ye, Z. Zhang, K. Chakrabarty, X. Gu, Efficient Board-Level Functional Fault Diagnosis With Missing Syndromes, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 35, No.6, pp. 985-998, June, 2016.

[23] Y. Lin, Q. Hu, J. Zhang, X. Wu, Multi-label feature selection with streaming labels, *Information Sciences*, Vol. 372, pp. 256-275, December, 2016.

[24] F. Barani, M. Mirhosseini, H. Nezamabadi-Pour, Application of binary quantum-inspired gravitational search algorithm in feature subset selection, *Applied*

Intelligence, Vol. 47, No. 2, pp. 304-318, September, 2017.

[25] M. A. Tawhid, K. B. Dsouza, Solving feature selection problem by hybrid binary genetic enhanced particle swarm optimization algorithm, *International Journal of Hybrid Intelligent Systems*, Vol. 15, No. 4, pp. 207-219, November, 2019.

[26] Y. Xue, H. K. Zhu, J. Y. Liang, A. Slowik, Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification, *Knowledge-Based Systems*, Vol. 227, pp. 1-9, September, 2021.

[27] K. Chen, B. Xue, M. Zhang, F. Zhou, Correlation-Guided Updating Strategy for Feature Selection in Classification with Surrogate-Assisted Particle Swarm Optimisation, *IEEE Transactions on Evolutionary Computation*, Vol. 26, No. 5, pp. 1015-1029, October, 2022.

[28] Y. Xue, Y. H. Tang, X. Xu, J. Liang, F. Neri, Multi-objective feature selection with missing data in classification, *IEEE Transactions on Emerging Topics in Computational Intelligence*, Vol. 6, No. 2, pp. 355-364, April, 2022.

[29] E. O. R. Flores, J. A. N. Acosta, A. H. Martínez, Optimal feature selection in industrial foam injection processes using hybrid binary Particle Swarm Optimization and Gravitational Search Algorithm in the Mahalanobis–Taguchi System, *Soft Computing*, Vol. 24, No. 1, pp. 341-349, January, 2020.

[30] S. Mirjalili, G. -G. Wang, L. d. S. Coelho, Binary optimization using hybrid particle swarm optimization and gravitational search algorithm, *Neural Computing and Applications*, Vol. 25, No. 6, pp. 1423-1435, November, 2014.

[31] J. -S. Pan, X. -F. Ji, A. Liang, K. -C. Huang, S. -C. Chu, Parallel Binary Cat Swarm Optimization with Communication Strategies for Traveling Salesman Problem, *Journal of Internet Technology*, Vol. 22, No. 7, pp. 1621-1633, December, 2021.

Biographies

B. R. Sathish is currently an Assistant Professor at the Department of Computer Science and Engineering in Loyola ICAM College of Engineering and Technology and pursuing his Ph.D at the Department of Information Technology in Anna University. His research interest includes Data Analytics, Machine Learning, Deep Learning and Soft Computing.

Radha Senthilkumar is currently an Associate Professor at the Department of Information Technology in Anna University, Madras Institute of Technology Campus, Chennai, India. Her research interest includes Data Mining and Knowledge Discovery focusing on developing novel methods that produce interpretable models.