Binary Border Collie Optimization Algorithm for Feature Selection

Salah Mortada
Department of Petroleum Project Management, Basra University for Oil and Gas, Iraq
salah.mortada@buog.eud.iq

Abstract: *Effective performance enhancement and feature reduction can be achieved by feature selection, which is the procedure of evaluating and choosing the most informative features. Consequently, this paper proposes a Binary Border Collie Optimization (BBCO) to address the feature selection problem in classification tasks. The sigmoidal function is used in the proposed algorithm to compress the continually updated position in order to achieve BBCO. Therefore, the proposed algorithm is utilized to determine the ideal feature subset from the initial feature set. To assess the performance of the proposed algorithm, BBCO is compared with Binary Firefly Algorithm (BFA), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Binary Gray Wolf Optimization (BGWO). The experiments on eighteen datasets collected from University of California Irvine (UCI) machine learning data repository results show the superiority of BBCO in 15 datasets, which means 83.3% in terms of classification accuracy with a reduced features number being chosen. Furthermore, BBCO has a very low average selected feature ratio, it is more beneficial for applications in the actual world.*

Keywords: *Metaheuristic, classification, real-world optimization, reduction.*

Received April 14, 2025; accepted September 11, 2025
https://doi.org/10.34028/iajit/23/1/5

1. Introduction

Data representation has recently emerged as a significant element influencing the performance of classification models. In the process of gathering data, many high-dimensional datasets have been created, which is creating a challenge for data mining [3, 4, 5]. In addition, the dataset often includes irrelevant and duplicated features, which seriously impedes the performance of the classification model. Not only does an excessive number of features add to the computational complexity, but it also raises the prediction inaccuracy [23]. Consequently, the significance of Feature Selection (FS) has led to it being a crucial phase in the data mining process [1, 2]. The main objective of FS is to identify the optimal set of possible features that contributes to comprehension of the classification model. The selection of important features has a dual benefit: It increases the accuracy of predictions and lowers the dimension of data [22]. Nevertheless, the FS is regarded as an NP-hard problem [17].

There are two strategies for FS: wrapper and filter. The filter technique in FS utilizes distance, mutual information, dependence, and information theory [21]. Instead of using a filter, a wrapper strategy optimizes classification performance by choosing the pertinent features using a classifier as the learning algorithm. Typically, the filter strategy is more rapidly than the wrapper strategy since it requires less computational work. However, wrapper strategy can usually provide superior performance [24]. Wrapper strategy employs a metaheuristic algorithm, such as grey wolf optimization, genetic algorithm, binary gravitational search algorithm, ant colony optimization, and bat algorithm, to choose the ideal feature subset [11, 16, 20].

Kennedy and Eberhart [15] previously presented discrete Binary version of the Particle Swarm Algorithm (BPSO), which modified the original PSO technique to address binary optimization issues. The results showed that the BPSO implementation is capable of quickly tackling these varied challenges. By applying BPSO to the FS setting, Firpi and Goodman [13] showed that it outperforms genetic FS. Rashedi *et al*. [20] suggested BGSA, which is a binary form of the Gravity Search Algorithm (GSA) for choosing features. The results showed that the BGSA is effective in handling several nonlinear functions benchmark, and Ramos *et al*. [19] proposed a new rapid and accurate approach for FS by producing their own version of the Harmony Search (HS) and comparing it to the performance of the Optimal-Path Forest classifier. The proposed version outperformed previous pattern recognition and FS methods. Emary *et al*. [12] propose a system for FS based on Gray Wolf Optimization (GWO) intelligent search has been suggested. In contrast to PSO and GA across a collection of University of California Irvine (UCI) datasets, GWO demonstrates superior performance in addition to robustness and convergence speed. Additionally, a wide variety of optimization algorithms have been employed to address the same issue in numerous studies [6, 7, 9, 14, 18]. According to previous research, FS is crucial to achieving the best classification ideal.

Border Collie Optimization (BCO) is one of the newly suggested metaheuristic algorithms that has been proposed for continuous optimization problem [10], which is focused on imitating the methods used by border collie dogs to manage sheep, its unique as herding styles of BCO have not been studies earlier, the algorithm is good in balancing exploration and exploitation and have successfully avoid local optima [10]. Nevertheless, the standard BCO was designed to tackle continuous optimization problem, not binary variable problems. Thus, the sigmoidal function is used to transform the continuous form of BCO into the binary form. The BCO uses a sigmoidal function to squash the continually updated position into a binary representation exclusively for the dog's position vector. The primary objective of this work is to provide an innovative BCO-based FS approach for the identification of a limited set of features and produce classification accuracy that is equivalent to or even better than that obtained by employing all features and traditional features reduction methods. The efficiency of binary BCO is assessed using 18 UCI machine learning data repository datasets. The efficiency of the proposed method is assessed by contrasting binary BCO with Binary Firefly Algorithm (BFA), GA, PSO, and BGWO. The experimental findings show that binary BCO maintains a competitive performance in FS while having a very efficient computing complexity. Improving search efficiency with respect to selection functions. Furthermore, the classification accuracy is enhanced by attaining the optimal goodness-of-fit score.

The subsequent portions of the paper are organized as follows: Section 2 presents the standard border collie optimization. The new FS algorithm, binary BCO, is discussed in brief detail as well. Section 3 describes the experimental setup and empirical findings. The findings from section 3 are discussed as well. Lastly, section 4 concluded the findings of this research.

2. Feature Selection Using Border Collie Optimization

2.1. Border Collie Optimization

BCO is a swarm intelligence algorithm that was recently introduced by Dutta *et al*. [10]. BCO was inspired by the idea of imitating nature border collie dogs' sheep-herding styles. The candidate solutions in BCO are represented by dogs and sheep, and the best solution (best fitness) is referred to as lead dog. In real life scenario, a dog can handle the herd on its own. Nonetheless, due to the expansive search area required for various optimization issues, thus three dogs are taken into consideration. In addition, When the algorithm is initialized, three sheep and three dogs are displayed. As the sheep wander off to graze, the dogs fetch them back to the property.

The positions of dogs and sheep are generated using random variables. The lead dog, left dog, and right dog are the names given to the dogs based on their placements. The lead dog is in charge of the herd from the front. The lead dog in each iteration is the individual with the highest fitness (fit_f). Their main job is gathering. The second and third highest individuals are chosen to represent the right and left dogs, correspondingly. For the purpose of selecting the right and left dogs, the tournament selection procedure is used. These dogs are primarily in charge of eyeing and stalking the herd. The symbols (fit_{le}) and (fit_{ri}), sequentially represent their fitness values. Those that remain are sheep, whose lower fitness values than dogs, making up the remainder of the population. The sheep's fitness is denoted as (fit_s).

The dogs that guide the sheep to the plantation are the best option. They migrate from field to plantation. Direction and distance are determined by sheep and dog velocity, acceleration, and time, as illustrated in Figure 1.

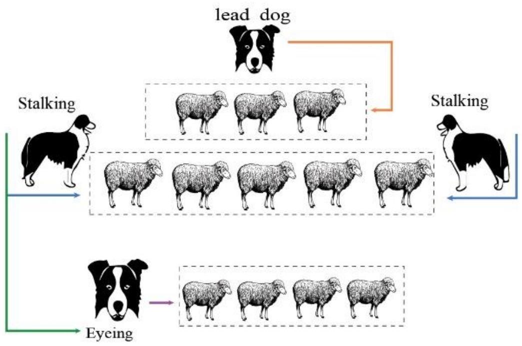

Figure 1. Border collie herding techniques.

- Dogs' velocity: the velocity of all three dogs over time is determined using the equation $(t+1)$.

$$V_f(t+1) = \sqrt{V_f(t)^2 + 2 \times Acc_f(t) \times Pop_f(t)} \quad (1)$$

$$V_{ri}(t+1) = \sqrt{V_{ri}(t)^2 + 2 \times Acc_{ri}(t) \times Pop_{ri}(t)} \quad (2)$$

$$V_{le}(t+1) = \sqrt{V_{le}(t)^2 + 2 \times Acc_{le}(t) \times Pop_{le}(t)} \quad (3)$$

$V_f(t+1)$, $V_{ri}(t+1)$ and $V_{le}(t+1)$ are the velocity at time $(t+1)$ for the lead, right, and left dogs, sequentially, in Equations (1), (2), and (3). Likewise, $V_f(t)$, $V_{ri}(t)$ and $V_{le}(t)$ are the velocity at time (t) for the lead, right, and left dogs. $Acc_f(t)$, $Acc_{ri}(t)$ and $Acc_{le}(t)$ are the acceleration at time (t) for the lead, right, and left dogs, sequentially.

$Pop_f(t)$, $Pop_{ri}(t)$ and $Pop_{le}(t)$ are the locations of the lead, right, and left dogs at time (t), sequentially.

- Sheep velocity: the three herding techniques are used to update the velocity of the sheep.
- Gathering: the sheep that are near the lead dog follow its lead. Consequently, these are only gathering sheep. Fitness values determine their selection.

$$D_g = (fit_f - fit_s) - \left(\left(\frac{fit_{le} + fit_{ri}}{2}\right) - fit_s\right) \quad (4)$$

If the value of (D_g) in Equation (4) is positive, the sheep is getting closer to the lead dog. In this case, the following equation is used to modify the sheep's velocity.

$$V_{sg}(t+1) = \sqrt{V_f(t+1)^2 + 2 \times Acc_f(t) \times Pop_{sg}(t)} \quad (5)$$

The lead dog's acceleration at time t and the velocity of the sheep V_{sg} are both directly influenced by the lead dog's velocity at time (t+1) in Equation (5). The sheep to be gathered are currently located at Pop_{sg}, as illustrated in Figure 2.

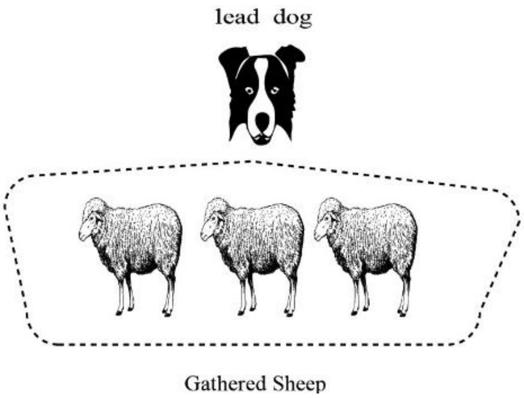

Figure 2. Sheep are gathered by the lead dog.

- Stalking: keep the left and right dogs on track by stalking the sheep from the sides. These are the sheep whose *Dg* values have been discovered to be negative. These sheep's velocity is more influenced by the left and right dogs' velocities. The equations for the stalked sheep's velocity updation are shown below.

$$V_{ri} = \sqrt{(V_{ri}(t+1)\tan(\theta_1)^2 + 2 \times Acc_{ri}(t) \times Pop_{ri}(t)} \quad (6)$$

$$V_{le} = \sqrt{(V_{le}(t+1)\tan(\theta_1)^2 + 2 \times Acc_{le}(t) \times Pop_{le}(t)} \quad (7)$$

$$V_{ss}(t+1) = \frac{v_{le} + v_{ri}}{2} \quad (8)$$

Equation (8) states that the left and right dogs' velocities dictate the stalked sheep's velocity, represented by V_{ss}. Since the dogs lead the sheep from the sides, θ_1 and θ_2 are the tangents of the random traversing angles. The value of the two variables, θ_1 and θ_2, range from (1-89) and (91-179). These values are randomly designated.

- Eyeing: the sheep that have gone totally astray are the ones that need to be eyeing. When an individual's fitness does not increase in successive iterations, eyeing is used. In this case, it is believed that the dog with the least amount of fitness would follow after the sheep and give them an eye. Consequently, it is expected that they will experience retardation, as shown by the equations below.

$$V_{se}(t+1) = \sqrt{(V_{le}(t+1)^2 - 2 \times Acc_{le}(t) \times Pop_{le}(t)} \quad (9)$$

$$V_{se}(t+1) = \sqrt{(V_{ri}(t+1)^2 - 2 \times Acc_{ri}(t) \times Pop_{ri}(t)} \quad (10)$$

As shown in Equation (9), $V_{le}(t+1)$ and $Acc_{le}(t)$ display the left dog's speed and acceleration when it is the least fit of the three. As shown in Equation (10), $V_{ri}(t+1)$ and $Acc_{ri}(t)$ display the right dog's speed and acceleration when it is the least fit of the three. The sheep that need to be gathered are at Pop_{se} right now. According to the theory, the dog that is the least fit is the most like a sheep, so that dog is taken into account, as seen in Figure 3.

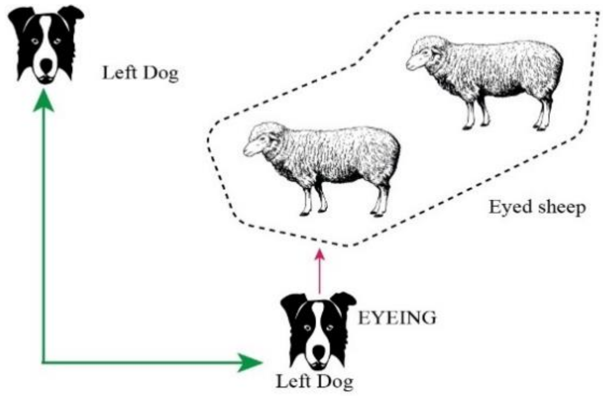

Figure 3. Eyeing of sheep by left dog.

- Acceleration Dogs and Sheep: the most common physics equation gives the acceleration updates below.

$$Acc_i(t+1) = \frac{(V_i(t+1) - V_i(t))}{Time_i(t)} \quad (11)$$

Where Acc_f (t+1), $Acc_{le}(t+1)$, Acc_{ri} (t+1), Acc_{sg} (t+1), Acc_{ss} (t+1) and Acc_{se} (t) as the acceleration of all dogs and sheep are updated utilizing in Equation (11). $i \in \{f, le, ri, sg, ss,$ till $se\}$.

- Sheep and dog time: the following equation is utilized to update the traversal *Time* (*T*) for each separate.

$$Time_i(t+1) = Avg \sum_{i=1}^{d} \frac{(V_i(t+1) - V_i(t))}{Acc_i(t+1)} \quad (12)$$

where each individual's average traversal time is of dimension (d).

- Dogs population updating: the fundamental physics displacement equation is used to update the locations of the dogs.

$$Pop_f(t+1) = V_f(t+1) \times Time_f(t+1) + \frac{1}{2} Acc_f(t+1) \times Time_f(t+1)^2 \quad (13)$$

$$Pop_{le}(t+1) = V_{le}(t+1) \times Time_{le}(t+1) + \frac{1}{2} Acc_{le}(t+1) \times Time_{le}(t+1)^2 \quad (14)$$

$$Pop_{ri}(t+1) = V_{ri}(t+1) \times Time_{ri}(t+1) + \frac{1}{2} Acc_{ri}(t+1) \times Time_{ri}(t+1)^2 \quad (15)$$

Equations (13), (14), and (15) are used to update the positions of the dogs that are in the lead, right, and left.

- Population updating of sheep: the following equations are used to update the positions of sheep that are part of gathering and stalking groups, as illustrated in Figure 4.

$$Pop_{sg}(t+1) = V_{sg}(t+1) \times Time_{sg}(t+1) + \frac{1}{2} Acc_{sg}(t+1) \times Time_{sg}(t+1)^2 \quad (16)$$

$$Pop_{ss}(t+1) = V_{ss}(t+1) \times Time_{ss}(t+1) - \frac{1}{2} Acc_{ss}(t+1) \times Time_{ss}(t+1)^2 \quad (17)$$

When it comes to eyed sheep, the equation shown below is applied.

$$Pop_{se}(t+1) = V_{se}(t+1) \times Time_{se}(t+1) - \frac{1}{2} Acc_{se}(t+1) \times Time_{se}(t+1)^2 \quad (18)$$

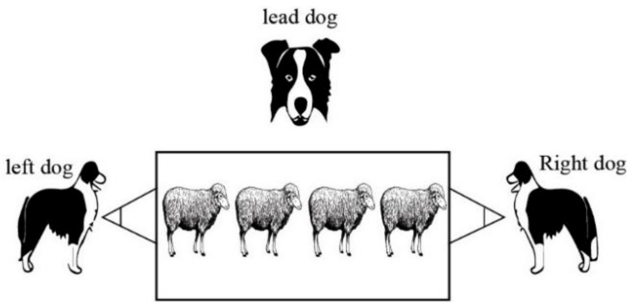

Figure 4. Sheep are being stalked by both the left and right dogs.

2.2. Algorithm

The BCO algorithm is designed primarily using four parameters. Time and velocity are independent parameters that perform significant importance in the updating of the states. Population and acceleration, the remaining two dependent parameters, are simply obtained from the aforementioned independent parameters. We derive from Equation (11), that if time and velocity are known, one can compute $Acc_i(t+1)$. The following equation is obtained by replacing the value of $Acc_i(t+1)$ in Equation (13) in a similar manner.

$$Pop_f(t+1) = V_f(t+1) \times Time_f(t+1) + \frac{1}{2} \frac{(V_f(t+1) - V_f(t))}{Time_i(t)} \times Time_f(t+1)^2 \quad (19)$$

or,

$$Pop_f(t+1) = V_f(t+1) \times Time_f(t+1) + \frac{1}{2}\left(V_f(t+1) - V_f(t)\right) \times Time_f(t+1) \quad (20)$$

In a similar manner, the populations of eyed sheep, stalked sheep, right dog, left dog, and gathered sheep can be derived by replacing the value of $Acc_i\ (t+1)$ in Equations (14), (15), (16), (17) and (18), respectively.

2.3. Avoid Being Trapped in Local Optima

The fitness in BCO of each sheep is evaluated at each iteration to see whether it is or is not trapped in local optima. The sheep are assumed to be trapped in local optima if their fitness does not enhance during the process of five steps. The dog then looks at the sheep to get it back on its path.

2.4. Exploitation and Exploration

As a critical step to obtain the optimum results, the search space must be explored and exploited. The algorithms that can balance the two have a better chance of avoiding being trapped in local optima. In the search space, exploration focuses on identifying possible solution regions. The exploration capability of the BCO algorithm is governed by the movements of the three dogs: lead, right, and left. The dogs move in separate directions and are completely independent of one another. Consequently, they possess the capability to identify the most promising regions inside the search space. Conversely, exploitation pertains to the enhancement of search results. The three dogs have a direct influence on the gathered and stalked sheep's movements. Consequently, they focus their efforts on finding more optimum solutions inside the portion of the search area occupied by dogs. Furthermore, to rescue the BCO algorithm from the region of local optima, the "eyed sheep" employs the notion of retardation.

2.5. The Binary Border Collie Optimization

This approach mandates that only the revised position vector of the border collie be binary; refer to Figure 5, using the primary updating equations presented Equations (21) and (22). The collection of the Binary Border Collie Optimization (BBCO) solutions will be in binary representation, where all solutions are on the Boolean lattice. The original algorithm BCO concept will be used to update the positions of a particular dog while maintaining the binary restriction depending on the position updating of the dog's equations, per the equation below.

$$X_d^{t+1} = \begin{cases} 1 \ if \ sigmoid\ (updating\ the\ position\ of\ dog's) \\ \qquad \geq\ rand \\ 0\ otherwise \end{cases} \quad (21)$$

And updating position of sheep Equations (16), (17) and (18) as showing in following equation.

$$X_d^{t+1} = \begin{cases} 1 \ if \ sigmoid(updating\ the\ position\ of\ sheep\ ) \\ \qquad \geq\ rand \\ 0\ otherwise \end{cases} \quad (22)$$

In this case, X_d^{t+1} is the current binary position in dimension d as of iteration t, $rand$ is a number chosen at random from a uniform distribution $\epsilon[1, 0]$, and $sigmoid\ (a)$ is defined as explained below.

$$sigmoid(a) = \frac{1}{1 + e^{-10(x-0.5)}} \quad (23)$$

Figure 5 illustrates the display of a solution for FS in BBCO. The solution's location may assume a value of "1". An attribute will be selected if the value is "1" and will be disregarded if the value is "0". The BBCO is employed in the selection of features for classification problems in this field. For a feature vector of size *N*, the number of different feature redactions would be 2^N. This would create a huge space of features that need to be thoroughly studied. Therefore, the BBCO is used to adaptively investigate the feature space for optimum feature pairings. The ideal feature combination enhances classification efficacy while reducing the quantity of selected attributes. The fitness function used in BBCO to assess the locations of individual agents is represented by the following equation.

$$F = \alpha\gamma R(D) + \beta\frac{|C - R|}{|C|} \tag{24}$$

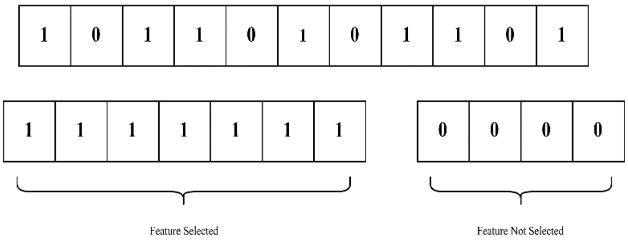

Figure 5. Illustrates the depiction of a solution for FS in BBCO.

Where $\gamma R(D)$ is the classification fineness of state attribute set R for decision D, R is the length of the selected attribute subset, C is the total number of attributes, and α and β correspond to the importance of 2 parameters subset length and classification quality. $a \in [0, 1]$ and β=1- a. As shown by $\gamma R(D)$ [11], the number of unselected features as a percentage of all the features; $\frac{|C-R|}{|C|}$, and the fitness function's ability to maximize classification quality.

Employing error rate instead of classification quality, and chosen feature ratio rather than unselected feature size, it is simple to turn the aforementioned equation into a minimization issue. Equation can be used to formulate the minimization issue.

$$F = \alpha E_R(D) + \beta\frac{|R|}{|C|} \tag{25}$$

Where R is the size of the selected feature subset, C is the total number of features, and $E_R(D)$ is the classification error rate of the classifier. $a \in [0, 1]$ and β=1- a are constants that regulate how important feature reduction and classification accuracy; β=0.01 in current experiments.

The main aspect of wrapper techniques is using the classifier as a guide for FS. The three following categories can be used to group wrapper-based FS:

1. Feature evaluation criteria.
2. Search method.
3. Classification method.

A simple and widely used classification method is the K-Nearest Neighbor (KNN) technique. The dominant KNN category is used to classify an unknown sample instance using KNN, a supervised learning technique. Classifiers are determined by minimizing the distance between the query instance and the training instances; the KNNs model is not employed. The minimum distance between the query instance and the training examples is used to ascertain classifiers; no KNNs model is applied. The KNN method is a widely used classifier since it is straightforward and simple to build. This approach use KNN as a classifier to ascertain the efficacy of the chosen attributes. To maximize the feature assessment criterion, BBCO is used as a search method as it has the capability to search the feature space in an adaptable manner. Due to the fact that each feature is only represented by a single dimension in the search space, the location of the search agent corresponds to either a solution or a single feature combination. The BBCO algorithm are shown in Algorithm (1).

Algorithm 1: Binary border collie optimization.

```
1.  Initialize
    Pop_t→A random population of n individuals having d dimensions each, 3 dogs and (n - 3) sheep;
      Acc_t → Random acceleration for each of the n individuals having d dimensions;
      Time_t→ Random time for each of the n individuals;
      V_t→ Zero velocity for n individuals having d dimensions;
      K=0;
2.  while t<max_Iterations do
3.     Eyeing=0
4.     fit_t=Calculate fitness of n individuals
5.  if fit_t<fit_t -1 then
6.     k=k + 1
7.  end if
8.  if k=5 then
9.     Eyeing=1
10.    k=0
11. end if
12.    LeadDog=Individual with best fitness (fit_d))
13.    R=Random Number [2, 3]
14. if R=2 then
15.    RightDog=Individual with 2nd best fitness (fit_ri)
16.    LeftDog=Individual with 3nd best fitness (fit_le)
17. else
18.    LeftDog=Individual with 2nd best fitness (fit_le)
19.    RightDog=Individual with 3nd best fitness (fit_ri)
20. end if
21.    Sheep=Rest of the individuals excluding top three (fit_s)
22.    Update velocity of dogs (using (1), (2), (3) and (21))
23. while i>3 and i<=n do
24. if Eyeing=1 then
25.    Update velocity if sheep (using (9) and (22))
26. else
27. if Dg>0 then
28.    Update velocity of sheep (using (5) and (22))
29.   else
30.    Update velocity of sheep (using (8) and (22))
31.     end if
32.  end if
33. end while
34.    Update Acceleration of n individuals (using (11) and (21))
35.    Update Time of n individuals (using (12) and (22))
36.    Update Population of Dogs (using (13), (14) and (15))
37. while i>3 and i<=n do
38.     if Eyeing=1 then
39.     Update Population of sheep (using (18) and (22))
40.     else
41.      Update Population of sheep (using (16), (17) and (22))
42.     end if
43.  end while
44. end while
```

3. Results

There were 18 datasets from the UCI machine learning library [8] used in the experiments and comparisons. Table 1 demonstrates how the datasets were selected to have a wide range of cases and attributes to illustrate the many types of problems the suggested technique will be evaluated on. It is common practice to use cross-validation to randomly split instances from each dataset into a training set, a validation set, and a test set. A wrapper technique is used to classify features in this study. In trials that rely on trial and error, KNN is a popular and easy-to-understand learning approach, with the superior value for K (K=5) being selected across all datasets. Each dog's location serves to symbolize a subset of characteristics throughout training. During the optimization phase, the training set is used to assess the KNN's accuracy on the validation set, which helps guide the FS process.

Table 1. Dataset's description.

Dataset	No. Instances	No. Attributes
Zoo	101	16
WineEW	178	13
WaveformEW	5000	40
Vote	300	16
Tic-tac-toe	958	9
SpectEW	267	22
SonarEW	208	60
PenglungEW	73	325
M-of-n	1000	13
Lymphography	148	18
KrvskpEW	3196	36
IonosphereEW	351	34
HeartEW	270	13
Exactly2	1000	13
Exactly	1000	13
Congress	435	16
BreastEW	569	30
Breastcancer	699	9

Specific datasets are randomly divided into three equal segments for testing, validation, and training purposes. The data is segmented 20 times to ensure statistical significance and outcome stability. From the validation information for each run, the following procedures are listed:

- Average classification accuracy is a measure that displays how excellent the subset of features classifier is according to the equation below:

$$AAP = \frac{\sum_{i=1}^{run} AC_i}{run} \tag{26}$$

- The statistical best fitness function produced for a certain optimizer at the various X processes of an optimization technique is shown in Equation (27) below.

$$best = min_{i=1Y_*^i}^{X} \tag{27}$$

Where X is the times number of the optimization approach was used to choose the features subset, and Y_*^i denotes the ideal solution achieved from run number i.

- The statistical worst is the poorest solution discovered after running an optimization method X times to identify the best solutions. The equation can be used to show that the worst solution is the pessimistic one.

$$Worst = min_{i=1Y_*^i}^{X} \tag{28}$$

The number of times the optimization method to be performed to pick the feature subset is denoted by X, and the optimum solution obtained from run number i, is represented by Y_*^i.

- The statistical mean is the average of outcomes obtained by conducting an optimization procedure over several iterations. The mean represents the average efficacy of a certain stochastic optimizer, which may be articulated in an equation.

$$Main = \frac{1}{X}\sum_{i=1}^{X} Y_*^i \tag{29}$$

The typical ratio of selected characteristics to all features is represented by the average selection size. It is possible to express this metric as an equation.

$$AVGSelection = \frac{1}{X}\sum_{i=1}^{X} \frac{size\ (Y_*^i)}{C} \tag{30}$$

C is the number of features in the real dataset, and Size (Y_*^i) is the number of on values for the vector Y_*^i .

3.1. Experimental Results

The suggested method was put into practice using the MATLAB R2019 a tool on an Intel Core I7 computer running at 5.00 GHz with 16 GB of RAM. MATLAB's ease of use and the availability of supporting toolboxes, such as the parallel toolbox, are two advantages. Mathematical formula 2022, 10, 999, 10 of 16, which facilitates the search. Moreover, Python employs the Pandas and Sklearn libraries for data preparation and preprocessing, which include methods and techniques for data preprocessing and transformation similar to those in the preprocessing library, MATLAB is used to process complicated data and solve sophisticated simulations and engineering challenges. The platform, programming language, and parameters used to implement the suggested and compared methodologies were all the same (seed distribution, population size, number of iterations, and fitness function). PSO, BGWO, GA, and BFFA are contrasted with the suggested approach BBCO to assess its efficacy in FS. The following is a description of parameter setting for FS methods: The iterations number and the population size, N, are set at 200 and 30, sequentially. These optimum parameters are obtained from the original BCO [6]. It is important to note that BGWO, BFFA and BBCO do not have any additional parameter settings. For PSO, the minimum and maximum velocities are set at 6 and 6, respectively, and the weight of inertia, denoted by w, is falling in a linear fashion from 0.9 to 0.4. In addition, the acceleration coefficients, C1 and C2, are both set to the value of 2. In GA, the Crossover Rate (CR) and Mutation

Rate (MR) are both set to 0.6. A roulette wheel is employed to apply the primary selection, and the MR is set at 0.01. The single point crossover is implemented.

The classification accuracy of the proposed BBCO is shown for individual datasets in Figure 6. As can be seen, the classification accuracy for 15 out of the 18 datasets was the highest with BBCO. BFFA produces the greatest results for datasets 5, 6, and 8. From this perspective, BBCO is better equipped to choose the pertinent features. In comparison to BGWO, GA, and PSO. The second-best FS method is BFFA, as seen in Figure 6. Across all datasets, BBCO obtains the highest mean classification accuracy of 83.3%, followed by BFFA. This is because leaders in BBCO are able to improve their performance over time. Therefore, BBCO has a better chance of avoiding becoming stuck in the local optimum.

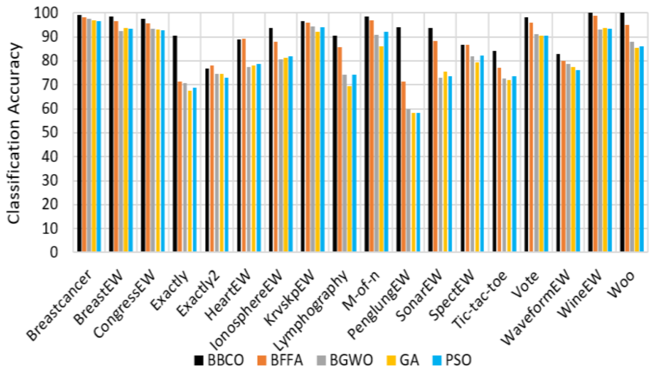

Figure 6. Performance average for the attributes chosen by the various optimizers with uniform initialization.

Table 2. The average fitness function obtained utilizing uniform initialization from the various optimizers.

Dataset	BBCO	BFFA	BGWO	GA	PSO
Zoo	0.000	0.051	0.127	0.118	0.133
WineEW	0.000	0.010	0.044	0.020	0.034
WaveformEW	0.171	0.200	0.214	0.206	0.221
Vote	0.016	0.039	0.056	0.040	0.042
Tic-tac-toe	0.156	0.230	0.233	0.233	0.223
SpectEW	0.131	0.132	0.178	0.160	0.166
SonarEW	0.060	0.116	0.174	0.154	0.154
PenglungEW	0.057	0.286	0.242	0.250	0.250
M-of-n	0.014	0.030	0.135	0.067	0.068
Lymphography	0.094	0.143	0.196	0.168	0.159
KrvskpEW	0.034	0.039	0.065	0.047	0.055
IonosphereEW	0.060	0.121	0.106	0.111	0.101
HeartEW	0.110	0.108	0.136	0.142	0.144
Exactly2	0.230	0.219	0.244	0.242	0.249
Exactly	0.095	0.286	0.315	0.291	0.277
CongressEW	0.022	0.043	0.057	0.044	0.048
BreastEW	0.013	0.034	0.037	0.027	0.033
Breastcancer	0.007	0.019	0.030	0.027	0.028

In the experiment of Table 2, the BBCO performed better than the other algorithms. The uniform initialization strategy, which made sure the mean fitness criterion in the early iterations and improved the results, that allowed the BBCO to perform better than the other algorithms. This indicates that the algorithm is more proficient at identifying features in the data pertinent to fitness across all used datasets. In uniform initialization, the search agents are spread out evenly across the search area using numbers that are chosen at random.

Table 3 consolidates the statistical outcomes from the diverse optimization runs across all data sets. The results demonstrate that, when measuring performance using the best fitness criterion, the suggested algorithm performs better than PSO, BFFA, BGWO, and GA.

Table 3. Best fitness function achieved from various optimizers with uniform initialization.

Dataset	BBCO	BFFA	BGWO	GA	PSO
Zoo	0.000	0.000	0.077	0.000	0.077
WineEW	0.000	0.000	0.000	0.000	0.000
WaveformEW	0.163	0.187	0.206	0.199	0.205
Vote	0.016	0.010	0.030	0.000	0.000
Tic-tac-toe	0.146	0.213	0.216	0.200	0.212
SpectEW	0.113	0.101	0.146	0.124	0.146
SonarEW	0.024	0.058	0.145	0.072	0.101
PenglungEW	0.000	0.042	0.167	0.167	0.125
M-of-n	0.000	0.000	0.087	0.021	0.000
Lymphography	0.034	0.082	0.143	0.122	0.143
KrvskpEW	0.026	0.028	0.056	0.035	0.031
IonosphereEW	0.057	0.077	0.085	0.094	0.085
HeartEW	0.092	0.078	0.122	0.111	0.122
Exactly2	0.230	0.195	0.213	0.234	0.234
Exactly	0.000	0.275	0.275	0.257	0.180
CongressEW	0.011	0.028	0.041	0.028	0.034
BreastEW	0.008	0.016	0.021	0.005	0.016

The ability of the dogs to balance feature space exploration and exploitation throughout the course of optimization iterations can be used to interpret this enhanced performance. In broad search spaces, where performance is more obvious, dogs outperform humans in handling enormous amounts of data. It is also clear that the suggested algorithm outperforms PSO, BFFA, BGWO and GA based on the best and worst results as shown in Table 4.

Table 4. The worst fitness function that each optimizer found when starting with a uniform initialization.

Dataset	BBCO	BFFA	BGWO	GA	PSO
Zoo	0.000	0.176	0.176	0.176	0.176
WineEW	0.000	0.034	0.119	0.051	0.051
WaveformEW	0.179	0.212	0.232	0.221	0.229
Vote	0.016	0.060	0.070	0.060	0.080
Tic-tac-toe	0.183	0.247	0.256	0.253	0.231
SpectEW	0.150	0.180	0.213	0.202	0.180
SonarEW	0.073	0.188	0.246	0.246	0.203
PenglungEW	0.071	0.542	0.417	0.458	0.417
M-of-n	0.060	0.078	0.201	0.141	0.123
Lymphography	0.103	0.204	0.265	0.204	0.184
KrvskpEW	0.039	0.050	0.078	0.064	0.072
IonosphereEW	0.071	0.162	0.120	0.128	0.120
HeartEW	0.129	0.144	0.156	0.167	0.189
Exactly2	0.240	0.240	0.260	0.251	0.260
Exactly	0.225	0.308	0.335	0.326	0.323
CongressEW	0.023	0.069	0.083	0.069	0.076
BreastEW	0.017	0.058	0.053	0.047	0.047
Breastcancer	0.007	0.034	0.043	0.039	0.034

A number of different dataset optimizations are shown in Table 5, which shows the ratio of the initial size to the size of the features that were actually picked. The table shows that, while maintaining comparable selected feature sizes, the proposed algorithm beats the other two approaches (BFFA, BGWO, PSO, and GA) in classification performance.

Table 5. Average selected attribute ratio for uniform initialization optimizers.

Dataset	BBCO	BFFA	BGWO	GA	PSO
Zoo	0.512	0.455	0.662	0.600	0.575
WineEW	0.553	0.516	0.677	0.554	0.554
WaveformEW	0.617	0.782	0.750	0.540	0.570
Vote	0.425	0.509	0.537	0.475	0.463
Tic-tac-toe	0.733	0.587	0.800	0.578	0.644
SpectEW	0.554	0.487	0.582	0.482	0.509
SonarEW	0.548	0.602	0.620	0.517	0.510
PenglungEW	0.504	0.495	0.494	0.489	0.513
M-of-n	0.576	0.484	0.815	0.600	0.523
Lymphography	0.480	0.532	0.533	0.456	0.544
KrvskpEW	0.626	0.631	0.739	0.528	0.472
IonosphereEW	0.536	0.605	0.576	0.482	0.506
HeartEW	0.411	0.604	0.708	0.662	0.677
Exactly2	0.565	0.780	0.646	0.400	0.462
Exactly	0.607	0.747	0.662	0.662	0.600
CongressEW	0.453	0.518	0.438	0.412	0.563
BreastEW	0.480	0.638	0.700	0.600	0.580
Breastcancer	0.500	0.571	0.644	0.556	0.556

Additionally, in order to confirm the stability and reproducibility of the stochastic algorithms' convergence, Table 6 displays the standard deviation of the fitness values obtained from the 20 iterations. It is clear that both BCO and PSO have low standard deviations, indicating their reliability, consistency, and capacity to provide the best outcome irrespective of the kind of randomization used or the search agents' initial positions.

Table 6. Standard deviation of the fitness function values for each approach employing uniform initialization across the 20 runs.

Dataset	BBCO	BFFA	BGWO	GA	PSO
Zoo	0.000	0.058	0.043	0.069	0.038
WineEW	0.000	0.013	0.046	0.022	0.021
WaveformEW	0.005	0.009	0.011	0.009	0.009
Vote	0.000	0.019	0.019	0.025	0.029
Tic-tac-toe	0.013	0.015	0.015	0.021	0.009
SpectEW	0.009	0.035	0.027	0.029	0.015
SonarEW	0.014	0.047	0.042	0.069	0.051
PenglungEW	0.028	0.159	0.104	0.121	0.114
M-of-n	0.016	0.028	0.041	0.048	0.053
Lymphography	0.018	0.044	0.055	0.033	0.017
KrvskpEW	0.003	0.007	0.010	0.012	0.016
IonosphereEW	0.006	0.027	0.016	0.012	0.014
HeartEW	0.012	0.025	0.014	0.021	0.026
Exactly2	0.005	0.014	0.021	0.008	0.011
Exactly	0.072	0.012	0.024	0.025	0.059
CongressEW	0.003	0.015	0.017	0.017	0.020
BreastEW	0.004	0.017	0.013	0.016	0.013
Breastcancer	0.000	0.011	0.012	0.009	0.007

3.2. Discussion

This section summarizes that BBCO was suggested to tackle the issue of FS across many application domains. BBCO was evaluated and compared with other common attribute selection technique such as BFFA, PSO, GA, and BGWO. The results of this investigation show how effective the BCO is at choosing the best collection of features. Competitive strategies are put in place for BFFA, BGWO, PSO, and GA to maintain the quality of solutions and promote cooperation among search agents. By using a t-test (p-value), it can be demonstrated that the BBCO performs significantly better than the BFFA, BGWO, GA, and PSO in terms of classification accuracy (p=0.0021, 0.0023, 0.0019, and 0.011, respectively). This may be seen as the capacity of BBCO to evade local minima. The p-values that were estimated for the average fitness that was attained by using a variety of optimizers across all of the data sets are shown in Table 7.

Table 7. The p-values for the average fitness achieved by the various optimizers.

	BBCO				
BFFA	0.0045	0.0532	0.0522	0.0021	0.0078
BGWO	0.0023	0.105	0.085	0.0256	0.023
GA	0.047	0.0088	0.0019	0.0275	0.0964
PSO	0.011	0.0424	0.0759	0.0127	0.0521

This indicates that BBCO performance is noticeably superior to that of BFFA, GA, BGWO, and PSO. The statistical finding shows BBCO advantage over other FS algorithms. Moreover, the proposed model was specified for a specific informative feature, and BCO achieved an accuracy of 83.3% using nearly 18 data sets. Nonetheless, this observed accelerated convergence relative to traditional BCO, which may adversely impact performance when dealing with class-imbalanced datasets (Exactly2, HeartEw, and SpectEW), particularly for FS. In this study, BBCO showed exciting results for a variety of applications intended goal by improving search efficiency with respect to selection functions. Furthermore, It shown that classification accuracy is enhanced by attaining the greatest goodness-of-fit score relative to other optimization algorithms. Finally, metaheuristic techniques can be used to improve multiple domains and applications, due to the difficulty to find algorithms that are applicable to all optimization problems.

4. Conclusions

BBCO is proposed in this study. The algorithm's design was initially inspired by the herding techniques used by Border Collie dogs. In addition, these clever dogs eagerly obey their master's commands, yet what makes them far more intriguing is their capacity for quick thought and action. In terms of FS, BFA, GA, PSO, and BGWO are compared with BBCO. The experimental findings demonstrated that BBCO outperformed other algorithms in FS. BBCO not only had the lowest average selected feature ratio, but it was also ranked the best fitness function for FS. Substantially, the proposed BBCO is effective and more suitable for usage in healthcare. As for future work, the parameters of BBCO can be fine-tuned with the use of a chaotic map. It is possible to increase the diversity in BBCO by increasing the velocity. Besides, it will be utilized on other applications areas, such as numerical problems, postal delivery, and school bus routing.

References

[1] Al-Behadili H., "Improved Firefly Algorithm with Variable Neighborhood Search for Data Clustering," *Baghdad Science Journal*, vol. 19, no. 2, pp. 409-421, 2022. DOI: 10.21123/BSJ.2022.19.2.0409

[2] Al-Behadili H., "Stochastic Local Search Algorithms for Feature Selection: A Review," *Iraqi Journal for Electrical and Electronic Engineering*, vol. 17, no. 1, pp. 1-10, 2021. DOI: 10.37917/ijeee.17.1.1

[3] AL-Behadili H., Mahamud K., and Sagban R., "Hybrid Ant Colony Optimization and Genetic Algorithm for Rule Induction," *Journal of Computer Science*, vol. 16, no. 7, pp. 1019-1028, 2020. DOI: 10.3844/JCSSP.2020.1019.1028

[4] Almazini H. and Mahamud K., "Adaptive Technique for Feature Selection in Modified Graph Clustering-Based Ant Colony Optimization," *International Journal of Intelligent Engineering and System*, vol. 14, no. 3, pp. 332-345, 2021. DOI: 10.22266/ijies2021.0630.28

[5] Almazini H. and Mahamud K., "Grey Wolf Optimization Parameter Control for Feature Selection in Anomaly Detection," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 2, pp. 474-483, 2021. DOI: 10.22266/ijies2021.0430.43

[6] Al-shalabi L., "Feature Selection Method Based on Consecutive Forward Selection and Backward Elimination Concepts Using a Weighted Vector," *The International Arab Journal of Information Technology*, vol. 22, no. 4, pp. 801-813, 2025. DOI: 10.34028/iajit/22/4/13

[7] Babatunde O., Armstrong L., Leng J., and Diepeveen D., "A Genetic Algorithm-Based Feature Selection," *International Journal of Electronics Communication and Computer Engineering*, vol. 5, no. 4, pp. 899-905, 2014. file:///C:/Users/acit2k/Downloads/Publication2.pdf

[8] Bache M. and Lichman K., UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/index.php, Last Visited, 2025.

[9] Banati H. and Bajaj M., "Fire Fly Based Feature Selection Approach," *International Journal of Computer Science Issues*, vol. 8, no. 4, pp. 473-479, 2011. https://www.researchgate.net/publication/235323871_Fire_Fly_Based_Feature_Selection_Approach

[10] Dutta T., Bhattacharyya S., Dey S., and Platos J., "Border Collie Optimization," *IEEE Access*, vol. 8, pp. 109177-109197, 2020. DOI: 10.1109/ACCESS.2020.2999540

[11] Emary E., Zawbaa H., and Hassanien A., "Binary Grey Wolf Optimization Approaches for Feature Selection," *Neurocomputing*, vol. 172, pp. 371-381, 2016. DOI: 10.1016/j.neucom.2015.06.083

[12] Emary E., Zawbaa H., Grosan C., and Hassenian A., *Afro-European Conference for Industrial Advancement, Advances in Intelligent Systems and Computing*, Springer, 2015. DOI: 10.1007/978-3-319-13572-4_1

[13] Firpi H. and Goodman E., "Swarmed Feature Selection," *in Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop*, Washington (DC), pp. 112-118, 2004. DOI: 10.1109/AIPR.2004.41

[14] Huang J., Cai Y., and Xu X., "A Hybrid Genetic Algorithm for Feature Selection Wrapper Based on Mutual Information," *Pattern Recognition Letters*, vol. 28, no. 13, pp. 1825-1844, 2007. DOI: 10.1016/j.patrec.2007.05.011

[15] Kennedy J. and Eberhart R., "Discrete Binary Version of the Particle Swarm Algorithm," *in Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Orlando, pp. 4104-4108, 1997. DOI: 10.1109/icsmc.1997.637339

[16] Mahamud K. and Almazini H., "Heuristic Initialization Using Grey Wolf Optimizer Algorithm for Feature Selection in Intrusion Detection," *International Journal of Intelligent Engineering and System*, vol. 16, no. 1, pp. 410-418, 2023. DOI: 10.22266/ijies2023.0228.36

[17] Mortada S. and Yusof Y., "An Improved Artificial Bee Colony with Perturbation Operators in Scout Bees' Phase for Solving Vehicle Routing Problem with Time Windows," *IAES International Journal of Artificial Intelligence*, vol. 12, no. 2, pp. 656-666, 2023. DOI: 10.11591/ijai.v12.i2.pp656-666

[18] Nakamura R., Pereira L., Costa K., Rodrigues D., and et al., "BBA: A Binary Bat Algorithm for Feature Selection," *in Proceedings of the 25th SIBGRAPI Conference on Graphics, Patterns and Images*, Ouro Preto, pp. 291-297, 2012. DOI: 10.1109/SIBGRAPI.2012.47

[19] Ramos C., Souza A., Chiachia G., Falcao A., and Papa J., "A Novel Algorithm for Feature Selection Using Harmony Search and its Application for Non-Technical Losses Detection," *Computers Electrecal Engineering*, vol. 37, no. 6, pp. 886-894, 2011. DOI: 10.1016/j.compeleceng.2011.09.013

[20] Rashedi E., Nezamabadi-Pour H., and Saryazdi S., "BGSA: Binary Gravitational Search Algorithm," *Natural Computing*, vol. 9, no. 3, pp. 727-745, 2010. DOI: 10.1007/s11047-009-9175-3

[21] Shunmugapriya P. and Kanmani S., "A Hybrid Algorithm Using Ant and Bee Colony Optimization for Feature Selection and Classification (AC-ABC Hybrid)," *Swarm and*

Evolutionary Computation, vol. 36, pp. 27-36, 2017. DOI: 10.1016/j.swevo.2017.04.002

[22] Too J., Abdullah A., and Saad N., "A New Quadratic Binary Harris Hawk Optimization for Feature Selection," *Electron*, vol. 8, no. 10, pp. 1-27, 2019. DOI: 10.3390/electronics8101130

[23] Yang L., Li K., Zhang W., Wang Y., and Ke Z., "Application of a New $_{m}$Ant-Miner$_{PR}$ Algorithm in Classification Rule Mining," *Journal of Digital Information Management*, vol. 13, no. 5, pp. 311-320, 2015. https://dline.info/fpaper/jdim/v13i5/v13i5_1.pdf

[24] ZorarpacI E. and Ozel S., "A Hybrid Approach of Differential Evolution and Artificial Bee Colony for Feature Selection," *Expert Systems with Applications*, vol. 62, no. C, pp. 91-103, 2016. DOI: 10.1016/j.eswa.2016.06.004

Salah Mortada is a Doctor at Basra University for Oil and Gas. He holds a Ph.D. in Information Technology with a specialization in Artificial Intelligence from Universiti Utara Malaysia (UUM). His research interests include Artificial Intelligence, Evolutionary Computation, Meta-Heuristic Algorithms, Swarm Intelligence, and their Applications in the Design and Optimization of Intelligent Transportation Management Systems such as Vehicle Routing Problems (VRPs).