An innovative approach for feature selection based on chicken swarm optimization

Ahmed Ibrahem Hafez[1,*], Hossam M. Zawbaa[2,4,*], E. Emary[3,*], Hamdi A. Mahmoud[2,*], Aboul Ella Hassanien[2,3,*]
[1]Faculty of Computer and Information, Minia University, Egypt
[2]Faculty of Computers and Information, Beni-Suef University, Egypt
[3]Faculty of Computers and Information, Cairo University, Egypt
[4]Faculty of Mathematics and Computer Science, Babes-Bolyai University, Romania
[*]Scientific Research Group in Egypt (SRGE), http://www.egyptscience.net

***Abstract*—In this paper, a system for feature selection based on chicken swarm optimization (CSO) algorithm is proposed. Datasets ordinarily includes a huge number of attributes, with irrelevant and redundant attribute. Commonly wrapper-based approaches are used for feature selection but it always requires an intelligent search technique as part of the evaluation function. Chicken swarm optimization (CSO)is a new bio-inspired algorithm mimicking the hierarchal order of the chicken swarm and the behaviors of chicken swarm, including roosters, hens and chicks, CSO can efficiently extract the chickens' swarm intelligence to optimize problems. Therefore, CSO was employed to feature selection in wrapper mode to search the feature space for optimal feature combination maximizing classification performance, while minimizing the number of selected features. The proposed system was benchmarked on 18 datasets drawn from the UCI repository and using different evaluation criteria and proves advance over particle swarm optimization (PSO) and genetic algorithms (GA) that commonly used in optimization problems.**

***Keywords*-Chicken swarm optimization; feature selection; bio-inspired optimization; evolutionary computation; wrapper-based feature selection.**

I. Introduction

A feature is an measurable property of the problem under observation, over the past years the domain of features in machine learning and pattern recognition applications have expanded from tens to hundreds of variables or features used in such applications. Hence the use of reduction or selection techniques is essential to reduce the large number of feature in the problem. Feature selection is a process of selecting a subset of features from a larger set of features, which leads to the reduction of the dimensionality of features space for a successful classification task. Feature selection provides a way for identifying the important features and removing irrelevant or redundant features from a dataset [1]. Feature Selection helps in understanding data, reducing computation requirement, reducing the effect of curse of dimensionality and improving the predictor performance [2].

Formerly, an exhaustive search for the optimal or near to optimal solution in a enormous search space may be impracticable, many researches seek to model the feature selection as a optimization problem [3]. One of the most used methods to solve the feature selection problems are evolutionary and swarm intelligence methods. Swarm intelligence is a computational intelligence-based approach which is made up of a population of artificial agents and inspired by the social behavior of animals (fish, birds, fireflies, etc.) from the real world. Example of such methods are ant colony optimization [4], [5], bat algorithm [6], and particle swarm optimization (PSO) [7], [8]. Genetic algorithms (GA) was the first evolutionary based algorithm introduced in the literature and developed based on the natural process of evolution through reproduction [9]. A hybrid methods can also be applied in which two evolutionary algorithms are used to solve the problem for example [10] proposed new feature selection approach that is based on the integration of GA and PSO.

The organization of this paper as the following: section II presents background of chicken swarm optimization algorithm and behavior of chickens' movement. The proposed system for feature selection using chicken swarm optimization algorithm describes in section III. The experimental results are discussed in section IV. Finally, conclusions are stated in section V.

II. Preliminaries

Chicken swarm optimization (CSO) is bio-inspire meta-heuristic optimization algorithm proposed by Meng, X.B. et al. [11]. The algorithm mimics the hierarchal order of a chicken swarm and the behaviors of its individuals chickens. The hierarchal order of a chicken swarm is divided into several groups, each group consists of one rooster and many hens and chicks. Each type of chickens follow different laws of motions. A hierarchal order plays a significant role in the social lives of chickens. The superior chickens in a flock will dominate the weak ones. There exist the more dominant hens that remain near to the head roosters as well as the more submissive hens and roosters who stand at the periphery of the group.

The mathematical model of CSO proposed in [11] was based on the following rules that summarize the chickens' behaviors:

1) The chicken swarm is divided into several groups. In each groups there is a dominant rooster, following it some hens and chicks.
2) The fitness value of the chickens outlines the hierarchy of the swarm, the individuals with the best fitness will be the roosters each one will be a group leader, the individuals with the worst fitness values will be considered as chicks. The others would be the hens.
3) The swarm hierarchy, dominance relationship and mother-child relationship in a group will remain unchanged. These statuses only update every several (G) time steps.
4) The swarm consists of N virtual chickens divided as follow: RN, HN, CN, and MN which are the number of roosters, the hens, the chicks, and the mother hens, respectively. Each individual is represented by their positions in a D-dimensional space by $x_{i,j} (i \in [1, ..., N], j \in [1, ..., D])$.

A. Chickens Movements

Rooster Movement: Roosters with better fitness values can search for food in a wider range of place than those with worse fitness values, such movement is depicted as in equations (1) and (2).

$$x_{i,j}^{t+1} = x_{i,j}^{t} * (1 + Randn(0, \sigma^2)), \tag{1}$$

$$\sigma^2 = \begin{cases} 1, & \text{if } f_i \le f_k, \\ \exp(\frac{f_k - f_i}{|f_i| + \epsilon}), & \text{otherwise}, \end{cases} \quad k \in [1, N], k \ne i, \tag{2}$$

where $x_{i,j}$ is the selected rooster with index i, $Randn(0, \sigma^2)$ is a Gaussian distribution with mean 0 and standard deviation σ_2, ϵ s the smallest constant in the computer used to avoid zero-division-error, k a randomly chosen roosters index selected from the roosters group, f_i is the fitness value of the corresponding rooster x_i.

Hen movement: Hens follow their group-mate roosters to search for food. Moreover, they would also randomly steal the good food found by other chickens, though they would be repressed by the other chickens. The more dominant hens would have advantage in competing for food than the more submissive ones. These phenomena can be formulated mathematically as in equations (3) and (4).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + S1 * \text{Rand} * (x_{r_1,j}^{t} - x_{i,j}^{t}) + S2 * \text{Rand} * (x_{r_2,j}^{t} - x_{i,j}^{t}) \tag{3}$$

$$S1 = \exp((f_i - f_{r_1}) / (\text{abs}(f_i) + \epsilon)), \tag{4}$$

$$S1 = \exp((f_{r_2} - f_i)), \tag{5}$$

where $Rand$ is a uniform random number over [0, 1]. $r_1 \in [1, ..., N]$ is an index of the rooster, which is the ith hen's group-mate, while $r_2 \in [1, ..., N$, is randomly chosen index of a chicken (rooster or hen) from the swarm.

Chick movement: The chicks move around their mother to search for food. This is formulated as in equation (6).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + FL * (x_{m,j}^{t} - x_{i,j}^{t}), \tag{6}$$

where $x_{m,j}^{t}$ is the position of the ith chick's mother such that $m \in [1, N]$, FL is parameter that represent how much speed a chick would follow its mother, to consider the differences between each chick FL is chosen randomly in the range [0, 2].

Based on the previous description CSO pseudo-code can be formulated as given in algorithm 1. Basically CSO has many advantage over similar optimization algorithm such as PSO. First CSO can behave similar to the standard PSO by setting $RN = CN = 0$ and letting $S1$ and $S2$ be the parameters $c1$ and $c2$ from PSO algorithm. Hence CSO can inherit many advantages of PSO. However CSO is designed to be innate multi-swarm method which enhance the whole swarm performance. Second, the chickens' swarm intelligence can be efficiently utilized in CSO. Given the diverse laws of the chickens' motions and cooperation between the multi-groups, the search space can be efficiently explored. Moreover, the whole chicken swarm may behave like a team to forage for food, which can be associated with the objective problems to be optimized.

III. Chicken Swarm Optimization (CSO) Algorithm for Feature Selection

The proposed system makes use of the chicken swarm optimization (CSO) algorithm to find combinations of features that maximizes the classification accuracy with minor number of selected features. The feature space with each feature represented in an individual dimension and the span of each dimension ranges from 0 to 1 is very huge and hence requires an intelligent searching method to find optimal point in the search space that maximizes the given fitness function. The fitness function for the CSO is to maximize classification performance over the validation set given the training data, as shown in equation (7) while keeping minimum number of features selected.

$$f_\theta = \omega * E + (1 - \omega) \frac{\sum_i \theta_i}{N}, \tag{7}$$

where f_θ is the fitness function given a vector θ sized N with 0/1 elements representing unselected / selected features, N is the total number of features in the dataset, E is the classifier error rate and ω is a constant controlling the importance of classification performance to the number of features selected.

The used variables is the same as the number of features in the given dataset. All variable are limited in the range [0, 1], where the variable value approaches to 1; its corresponding feature is candidate to be selected in classification. In individual fitness calculation, the variable is threshold to decide the exact features to be evaluated as in the equation (8).

$$f_{ij} = \begin{cases} 1 \text{ if } X_{ij} > 0.5 \\ 0 \text{ otherwise}, \end{cases} \quad (8)$$

where X_{ij} is the dimension value for search agent i at dimension j. While updating the firefly position; solution, at some dimensions the updated value can violate the limiting constrains; [0, 1], and hence we used simple truncation rule to ensure variable limits.

The classifier used in fitness function; equation 7, is the well-known K-nearest neighbor (KNN) [12]. KNN is a supervised learning algorithm that classifies an unknown sample instance based on the majority of the K-nearest neighbor category. Classifiers do not use any model for K-nearest neighbors and are determined solely based on the minimum distance from the query instance to the training samples. KNN is utilized in the experiments based on trial and error basis where the best choice of K is selected $(K = 5)$ as the best performing on all the datasets. Through the training process, each chicken's position represents one attribute subset. Training set is used to evaluate the KNN on the validation set throughout the optimization to guide the feature selection process. The test data are kept hidden from the optimization and is let for final evaluation. In this proposed system, the KNN is used as a classification to ensure the goodness of the selected features. The classifier is evaluated on a validation set inside the fitness function. In addition, the used fitness function incorporates both classification accuracy and reduction size [13].

```
Initialize RN, HN, CN, MN, G;
Randomly initialize each chicken in the swarm
X_i(i = 1, 2, ..., N).;
Initialize the max numbers of iteration T_max;
while T < T_max do for each iteration
    if T % G equals 0 then
        Rank the chickens fitness values and establish a
        hierarchal order in the swarm;
        Divide the swarm into different groups, and
        determine the relationship be- tween the chicks
        and mother hens in a group;
    end
    foreach chicken X_i in the swarm do
        if X_i is a roster then
            Update X_i's location using equation 1;
        end
        if X_i is a hen then
            Update X_i's location using equation 3;
        end
        if X_i is a chick then
            Update X_i's location using equation 6;
        end
        Evaluate the new solution using equation 7;
        If the new solution is better than its previous
        one, update it;
    end
end
```

Algorithm 1: Chicken swarm optimization (CSO) algorithm

IV. Experimental results

The chicken swarm optimization (CSO) algorithm was used to select optimal features combination to maximize classification performance and minimize the number of selected features in this case study. 18 datasets from the UCI machine learning repository [14] and [15] are used in the experiments and comparisons results. The 18 datasets were selected to have various numbers of attributes and instances as representatives of various kinds of issues that the proposed technique will be tested on, as shown in table (I).

For each dataset, the instances are randomly divided into three sets namely *training*, *validation*, and *testing* sets in a cross validation manner. The training set is used to train the used classifier while the validation set is used to evaluate the classifier performance and is applied inside the optimization fitness. The test data are kept hidden for both the classifier and the optimizer for the final evaluation of the whole feature selection and classification system.

Individual solutions in the CSO are points in the feature space; $d-$dimensional space, where d is the number of features in the original dataset in the range $[0, 1]$. The well-known K-nearest neighbor (KNN) classifier was used in the fitness function. CSO is randomly initialized with solutions in the feature space and is applied to minimize the fitness function in equation (7) but a solution with all the features selected is forced to be one of the initial solutions. The global parameter set for all the optimizers are decided by experiment experience as shown in table (II). The used specific optimizer's parameter set for GA and PSO are previous work [16] [17], and parameter set for CSO are decided by experiment experience as shown in table (III).

The genetic algorithm (GA) [16] optimizer and particle swarm optimizer (PSO) [17] are used in the same manner to be compared with the CSO to evaluate its classification performance with parameters indicated in table (III).

Tables (IV) and (V) displays the best, worst, mean, and std fitness values obtained by running each optimization algorithm for $NRuns$; see table (II). We can see that CSO obtains much enhanced fitness values over both PSO and GA on the average fitness values obtained from the different $NRuns$. The advance in the obtained fitness value can be interpreted by the clever capability of CSO to search the feature space adaptively and distributed searching capability

Table I
DATASETS DESCRIPTION

Dataset	No. of attributes	No. of instances
Lymphography	18	148
WineEW	13	178
BreastEW	30	569
Breastcancer	9	699
CongressEW	16	435
Exactly	13	1000
Exactly2	13	1000
HeartEW	13	270
IonosphereEW	34	351
KrvskpEW	36	3196
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
Zoo	16	101

Table II
GLOBAL PARAMETER SETTING

Parameter	Value	Meaning
ω	0.9999	Fitness function constant
$NIter$	70	The Number of iterations for optimization
$NAgents$	10	Number of used search agents in the optimization
$NRuns$	20	The number of times repeating the stochastic optimization

of CSO that always avoid algorithm stagnation. The same remark can be assured by remarking the best, worst, mean, and std solutions obtained with the different optimizers used which are depicted in tables (IV) and (V). In addition, we can remark that the output of CSO fitness even is better than using the whole feature set while it keeps less number of features.

Wilcoxon rank sum test proposed by Frank Wilcoxon [18] as a nonparametric test. The test assigns ranks to all the scores considered as one group, and then sums the ranks of each group. The rank-sum test is often described as the nonparametric version of t-test for two independent groups. It tests the null hypothesis that data in x and y vectors are samples from continuous distributions with equal medians against the alternative that they are not. We measured the Wilcoxon test among the used optimizers and the results are shown in table (VI). We can see that the advance in CSO is significant comparing to PSO (0.01) and GA (0.091) which motivates using CSO for feature selection and optimization problems.

Table III
INDIVIDUAL OPTIMIZER PARAMETER SETTING

Parameter	Value	Meaning
CSO		
r	0.15	The population size of roosters
h	0.7	The population size of hens
m	0.5	The population size of mother hens
PSO		
w	0.1	value of the inertia factor
c	0.1	individual-best acceleration factor
GA		
$Crossover\_Fraction$	0.8	Crossover Fraction
$Migration\_Fraction$	0.2	Migration Fraction

Table VI
WILCOXON TEST ON THE OBTAINED AVERAGE FITNESS AMONG DIFFERENT OPTIMIZERS

Optimizers	Wilcoxon Test value
CSO-PSO	0.01
CSO-GA	0.091
PSO-GA	0.029

Table VII
MEAN CLASSIFICATION ERROR ON TEST DATA FOR DIFFERENT OPTIMIZERS IN COMPARISON WITH THE DATA WITH ALL FEATURES

Dataset	All Features	GA	PSO	CSO
Breastcancer	0.0343	0.0369	**0.0206**	**0.0206**
BreastEW	0.0621	0.0579	0.0379	**0.0284**
CongressEW	0.0883	0.0759	0.0483	**0.0317**
Exactly	0.3303	0.2967	0.2895	**0.2378**
Exactly2	0.2745	0.2517	0.2432	**0.2348**
HeartEW	0.2156	0.1911	0.1489	**0.1467**
IonosphereEW	0.1624	0.1897	0.1299	**0.1094**
KrvskpEW	0.0905	0.0496	0.0475	**0.0391**
Lymphography	0.2760	NaN	0.1760	**0.1280**
M-of-n	0.1471	0.0817	0.0799	**0.0697**
PenglungEW	0.3600	0.3120	0.2880	**0.1840**
SonarEW	0.3229	0.2886	0.1743	**0.1057**
SpectEW	0.1978	0.2067	0.1528	**0.1213**
Tic-tac-toe	0.2589	0.2577	0.2169	**0.2119**
Vote	0.1100	0.0820	0.0560	**0.0400**
WaveformEW	0.2345	0.2253	**0.2028**	0.2029
WineEW	0.0700	0.0833	0.0267	**0.0167**
Zoo	0.2735	0.2127	0.1826	**0.1455**
Average	0.1949	0.1706	0.1401	**0.1152**

The performance of the different optimizers over different dataset is outlined in table (VII). We can remark that the performance of CSO is much better than PSO and GA, which proves that the selected feature combinations are much better. In addition, we can remark that using CSO outperforms the full features used in classification.

Values in table (VIII) are the average ratios of features selected to the total number of features for different optimizers over different datasets. We can remark that CSO selects minimum number of features in comparison with PSO and GA, while it keeps better classification performance as outlined in table (VII).

V. CONCLUSION AND FUTURE WORK

The objective of this paper was to propose a chicken swarm optimization (CSO) algorithm for feature selection to choose minimal number of features and to obtain compa-

Table IV
BEST AND WORST OBTAINED FITNESS VALUES FOR DIFFERENT OPTIMIZERS

	All Features		GA		PSO		CSO	
Dataset	**Best**	**Worst**	**Best**	**Worst**	**Best**	**Worst**	**Best**	**Worst**
Breastcancer	0.0343	0.0515	0.0215	0.0301	0.0130	0.0258	**0.0129**	**0.0258**
BreastEW	0.0421	0.0737	**0.0158**	0.0422	0.0211	0.0474	0.0158	**0.0316**
CongressEW	0.0414	0.1034	0.0207	**0.0345**	0.0277	0.0552	**0.0069**	0.0483
Exactly	0.3054	0.3563	**0.0629**	0.3204	0.2763	0.3034	0.0661	**0.3004**
Exactly2	0.2515	0.2725	0.2126	**0.2366**	**0.2043**	0.2733	**0.2043**	0.2552
HeartEW	0.1222	0.2111	**0.0778**	**0.1556**	0.1001	0.1889	0.1222	0.1778
IonosphereEW	0.1368	0.1709	0.0940	0.1453	0.1026	0.1624	**0.0855**	**0.1368**
KrvskpEW	0.0722	0.0975	0.0347	**0.0450**	0.0348	0.0611	**0.0263**	0.0620
Lymphography	0.2041	0.3469	0.0817	**0.2245**	0.1201	0.2801	**0.0800**	0.2600
M-of-n	0.1497	0.1766	0.0360	0.1108	**0.0151**	0.1112	0.0451	**0.0992**
PenglungEW	0.2917	0.5000	0.1667	0.3750	0.1200	0.4400	**0.0400**	**0.3200**
SonarEW	0.2464	0.4058	0.1015	0.1884	0.1286	0.2143	**0.0857**	**0.1286**
SpectEW	0.1124	0.2584	0.1012	0.2023	0.0675	0.2247	**0.0675**	**0.1910**
Tic-tac-toe	0.2188	0.2844	0.1938	0.2532	**0.1756**	**0.2320**	**0.1756**	**0.2320**
Vote	0.0800	0.1100	**0.0200**	0.0600	0.0201	0.0800	0.0300	**0.0500**
WaveformEW	0.2222	0.2437	0.1929	0.2126	**0.1874**	0.2145	0.1947	**0.2102**
WineEW	0.0339	0.1186	**0.0000**	**0.0170**	0.0001	0.0501	0.0000	0.0334
Zoo	0.2353	0.5294	0.0770	0.5294	0.0606	0.3636	**0.0000**	**0.3030**
Average	0.1556	0.2395	0.0839	0.1768	0.0931	0.1849	**0.0699**	**0.1592**

Table V
MEAN AND STD OBTAINED FITNESS VALUES FOR DIFFERENT OPTIMIZERS

	All Features		GA		PSO		CSO	
Dataset	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
Breastcancer	0.0412	0.0078	0.0275	**0.0038**	0.0207	0.0070	**0.0206**	0.0070
BreastEW	0.0611	0.0121	**0.0274**	0.0114	0.0380	0.0101	0.0285	**0.0071**
CongressEW	0.0786	0.0261	**0.0276**	**0.0069**	0.0483	0.0119	0.0317	0.0173
Exactly	0.3251	0.0192	0.2461	0.1056	0.2895	**0.0107**	**0.2379**	0.0968
Exactly2	0.2623	**0.0081**	**0.2216**	0.0102	0.2433	0.0328	0.2349	0.0241
HeartEW	0.1644	0.0328	**0.1112**	0.0324	0.1489	0.0374	0.1467	**0.0288**
IonosphereEW	0.1538	**0.0160**	0.1197	0.0191	0.1300	0.0221	**0.1094**	0.0229
KrvskpEW	0.0872	0.0101	0.0411	**0.0041**	0.0476	0.0123	**0.0391**	0.0137
Lymphography	0.2776	0.0551	0.1429	**0.0520**	0.1761	0.0623	**0.1280**	0.0756
M-of-n	0.1623	**0.0128**	0.0779	0.0293	0.0799	0.0376	**0.0697**	0.0243
PenglungEW	0.3750	**0.0780**	0.3000	0.0801	0.2880	0.1145	**0.1840**	0.1220
SonarEW	0.3449	0.0659	0.1624	0.0389	0.1743	0.0310	**0.1057**	**0.0192**
SpectEW	0.2112	0.0581	0.1483	**0.0367**	0.1529	0.0608	**0.1214**	0.0473
Tic-tac-toe	0.2481	0.0311	0.2144	0.0235	0.2170	0.0234	**0.2119**	**0.0215**
Vote	0.0940	0.0152	**0.0380**	0.0148	0.0561	0.0230	0.0400	**0.0071**
WaveformEW	0.2356	0.0088	**0.2013**	0.0091	0.2028	0.0101	0.2029	**0.0061**
WineEW	0.0678	0.0360	**0.0068**	**0.0093**	0.0267	0.0190	0.0167	0.0167
Zoo	0.3504	**0.1194**	0.1978	0.1894	0.1826	0.1417	**0.1455**	0.1345
Average	0.1967	**0.0340**	0.1285	0.0376	0.1401	0.0371	**0.1153**	0.0384

rable or even better classification accuracy from utilizing all attributes. This study shows that CSO is an effective search algorithm for feature selection problems. The used fitness function targets both the classification accuracy and reduction size, which means we can obtain a set of minimum selected features with maximum accuracy. CSO is hired in the feature selection domain for evaluation and results are compared against two of the well-known feature selection methods particle swarm optimization (PSO) and genetic algorithm (GA), which proves an advance in classification performance using a set of machine learning datasets. The CSO proves an advance in both reduction size and classification accuracy over PSO and GA. In our future work, we will work on the updating mechanisms in CSO to resolve feature selection to further minimize the number of attributes, maximize the classification accuracy. Also, we will examine the employ of chicken swarm optimization (CSO) algorithm for feature selection on datasets with a large number of attributes.

ACKNOWLEDGMENT

This work was partially supported by the IPROCOM Marie Curie initial training network, funded through the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement No. 316555. This fund only apply for one co-authors (Hossam M. Zawbaa).

REFERENCES

[1] B. Chizi, L. Rokach, O. Maimon, and J. Wang, "A survey of feature selection techniques." 2009.

Table VIII
AVERAGE FEATURE REDUCTION FOR DIFFERENT OPTIMIZERS OVER DIFFERENT DATASETS

Dataset	GA	PSO	CSO
Breastcancer	0.5111	0.6444	**0.4444**
BreastEW	0.5667	0.6733	**0.4800**
CongressEW	0.4875	0.6375	**0.1500**
Exactly	**0.6154**	0.7077	0.6308
Exactly2	0.7385	0.8769	**0.5231**
HeartEW	0.6615	0.6769	**0.4923**
IonosphereEW	0.4824	0.6882	**0.4353**
KrvskpEW	0.6444	0.6833	**0.6111**
Lymphography	0.5111	0.6778	**0.3444**
M-of-n	0.6769	0.7077	**0.6000**
PenglungEW	0.4892	0.6018	**0.1102**
SonarEW	0.5500	0.6700	**0.4533**
SpectEW	**0.3727**	0.6182	0.4182
Tic-tac-toe	0.8000	0.8222	**0.5778**
Vote	0.4875	0.6125	**0.3375**
WaveformEW	**0.7350**	0.7700	0.7400
WineEW	0.4308	0.6154	**0.4000**
Zoo	0.4625	0.4500	**0.2750**
Average	0.5680	0.6741	**0.4457**

[2] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[4] M. M. Kabir, M. Shahjahan, and K. Murase, "A new hybrid ant colony optimization algorithm for feature selection," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3747–3763, 2012.

[5] R. Forsati, A. Moayedikia, R. Jensen, M. Shamsfard, and M. R. Meybodi, "Enriched ant colony optimization and its application in feature selection," *Neurocomputing*, vol. 142, pp. 354–371, 2014.

[6] D. Rodrigues, L. A. Pereira, R. Y. Nakamura, K. A. Costa, X.-S. Yang, A. N. Souza, and J. P. Papa, "A wrapper approach for feature selection based on bat algorithm and optimum-path forest," *Expert Systems with Applications*, vol. 41, no. 5, pp. 2250–2258, 2014.

[7] A. Unler, A. Murat, and R. B. Chinnam, "mr 2 pso: a maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification," *Information Sciences*, vol. 181, no. 20, pp. 4625–4641, 2011.

[8] H. H. Inbarani, A. T. Azar, and G. Jothi, "Supervised hybrid feature selection based on pso and rough sets for medical diagnosis," *Computer methods and programs in biomedicine*, vol. 113, no. 1, pp. 175–185, 2014.

[9] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[10] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *Geoscience and Remote Sensing Letters, IEEE*, vol. 12, no. 2, pp. 309–313, 2015.

[11] X. Meng, Y. Liu, X. Gao, and H. Zhang, "A new bio-inspired algorithm: chicken swarm optimization," in *Advances in swarm intelligence*. Springer, 2014, pp. 86–94.

[12] C. T. C. Y. L.Y. Chuang, H.W. Chang, "Improved binary pso for feature selection using gene expression data," *Computational Biology and Chemistry*, vol. 32, pp. 29–38, 2008.

[13] W. N. B. Bing Xue, Mengjie Zhang, "Particle swarm optimisation for feature selection in classification," *Novel initialisation and updating mechanisms, Applied Soft Computing*, vol. 18, pp. 261–276, 2014.

[14] A. A. A. Frank, "Uci machine learning repository," *Expert Systems with Applications*, 2010.

[15] T. I. B. Raman, "Instance-based filter for feature selection," *Machine Learning Research*, vol. 1, pp. 1–23, 2002.

[16] J.-Y. C. H.-H. L. P.-C. K. Cheng-Hong Yang, Chung-Jui Tu, "Dimensionality reduction using ga-pso," in *Proceedings of the 2006 Joint Conference on Information Sciences (JCIS)*. Atlantis Press, Kaohsiung, Taiwan, 2006.

[17] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceeding of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan*, 1995, pp. 39–43.

[18] F. Wilcoxon, "Individual comparisons by ranking methods," *Bio-metrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.