High Dimensional Feature Selection Method of Dual Gbest Based on PSO

Hongbin Dong
Harbin Engineering University
School of computer science and technology
Harbin, China
donghongbin@hrbeu.edu.cn

Yuyao Pan
Harbin Engineering University
School of computer science and technology
Harbin, China
panyuyao96@163.com

Jing Sun
Harbin Engineering University
School of computer science and technology
Harbin, China
sunjing@hrbeu.edu.cn

***Abstract*—Particle swarm optimization (PSO) has a bright future in feature selection (FS). However, with the increase of data set dimension, the search space becomes larger, PSO is easy to fall into local optima and brings a lot of time and space overhead. It is still a big challenge to apply PSO to thousands of feature data sets. In this paper, the features are reordered to ensure that the shorter particles can get better classification performance. We propose a subset length constraint mechanism to reduce the number of selected features of particles gradually, so as to introduce particles into smaller and more effective space and jump out of local optima. Although the short running time of filter FS is more suitable for high-dimensional feature selection, the classification accuracy of filter FS is generally lower than that of wrapper FS. In order to achieve better classification performance in a short time, we propose a dual global optimal (Gbest) updating model. PSO has been modified. Compared with six algorithms on six high-dimensional data sets. The results show that the new dual Gbest update mechanism based on length restriction mechanism has higher efficiency and accuracy.**

Keywords—classification, data mining, feature selection, high-dimensional data, particle swarm optimization

I. INTRODUCTION

With the continuous generation and accumulation of data in various fields, high-dimensional data sets with thousands to tens of thousands of features become more and more common, which brings new challenges to machine learning tasks. Most of these data sets contain a lot of redundant and irrelevant features, which will significantly reduce the performance of the algorithm when performing machine learning tasks. Therefore, feature selection (FS) has become an important data preprocessing method. Effective feature selection algorithm can improve the accuracy and interpretability of learning model, and reduce the time and space cost of learning algorithm [1].

According to the different evaluation strategies of feature subsets, feature selection methods can be divided into wrapper method and filter method [2]. Filter methods only judge the quality of feature subset based on the intrinsic relationship of data, while wrapper methods use classifier to evaluate the subset. Therefore, filter methods need less training time than wrapper methods, but filters' performance are poor [3].

Although people have done a lot of researches in the field of FS, because the size of search space increases exponentially with the number of features, it is still a huge challenge to apply it to high-dimensional data. Feature sorting and feature weighting are common methods to implement highdimensional data FS. Some researchers use criteria such as feature dispersion [4] , classification accuracy [5] or granular computing[6] to measure the performance of each feature, and take the first several features to form a feature subset. Therefore, some domain experience is needed to determine the size of feature subset. Moreover, since the features are evaluated separately, and the redundancy and interaction between features are ignored, even the top features may become redundant due to the same function as another feature, and vice versa. Feature subsets can evaluate the performance of the whole subset at a time, and can better measure the relationship between features. Sequential forward feature selection (SFS) [7], sequential backward feature selection (SBS) [8] and greedy search [9] are classic feature subset selection methods. But SFS, SBS, greedy search and other search methods need huge time overhead, and are easy to fall into local optima.

Particle swarm optimization algorithm [10] (PSO) is an optimization algorithm to simulate the social behaviors of birds flocking. PSO has shown great potential in the field of FS [11]. However, due to the large search space, the high-dimensional feature selection methods based on PSO still face the problem of easily falling into local optimum. In order to solve this problem and improve the performance of the algorithm, people put forward different strategies. For example, the variation of PSO used in reference [12] is a competitive group optimizer (CSO) for large scale optimization to solve the problem of high-dimensional feature selection. In this method, all particles are divided into two groups, and the better of the two groups will help the other group to update. Literature [13] [14] divides filter and wrapper into two stages to reduce the number of features and time cost. The mixed methods of using wrapper and filter to form a single stage has also been proposed. For example, the PSO algorithm proposed in [15] uses classification accuracy to decide whether to update the Pbest, which has higher classification accuracy than the method using only filters. In reference [16], a new heuristic local search algorithm is proposed, which uses a combination of filters and wrappers in the fitness function, thus it is more likely to find a better solution. However, the different combination of the two shows that the more filters used in the algorithm, the worse the classification performance and the larger the subset length. Therefore, how to combine the advantages of filters and wrappers and avoid their dis-advantages is still a challenge.

In this paper, a new hybrid feature selection method is proposed. Firstly, attention mechanism is used to measure the contribution of features to the classification problem, and then all features are reordered according to the contribution. Each particle is initialized with a strategy that features in front are more likely to be selected.Then, a feature subset length limiting mechanism is proposed, which sets all feature bits

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

exceeding the limit to 0. A two-layer global optimal solution (Gbest) update mechanism is proposed. The outer layer is a particle swarm algorithm using information entropy as fitness function, and the inner layer further updates Gbest with KNN classification accuracy and subset length according to the results obtained from the outer layer. The method is evaluated on 6 high-dimensional public data sets and its good performance is proved. Comparing this method with other 6 feature selection methods, the results show that the method performance is better than the comparison algorithm in terms of accuracy, time and feature subset size.

The rest of this paper is arranged as follows. Section 2 introduces the background of the algorithm. Section 3 provides a detailed introduction of the algorithm. In Section 4, the experimental results of the algorithm and six comparison algorithms on six high-dimensional data sets are studied. Section 5 summarizes the whole paper.

II. BACKGROUND

A. Ranking of Characteristics

This algorithm needs to rearrange the order of features according to the importance of different features. Any measure can be used to evaluate features, such as unsupervised learning, random forest, etc. In this paper, we use an attention-based mechanism for feature selection (AFS)[17] to train feature ranking. Because it requires no prior knowledge, it can obtain the internal correlation between features in a relatively fast time, and obtain better experimental results than unsupervised and random forest. AFS consists of two modules: attention module and learning module. Attention module is responsible for calculating the weight of all features and is the core of the framework. By solving optimization problems, the learning module looks for the optimal correlation between weighted features and monitoring targets. The learning module connects the supervision target and features by the back propagation mechanism, and continuously corrects the feature weights during the training process. The two modules work together to build the correlation that best describes the degree of relevance of the target and the feature. Using this network, we can directly output the sorting of all features .

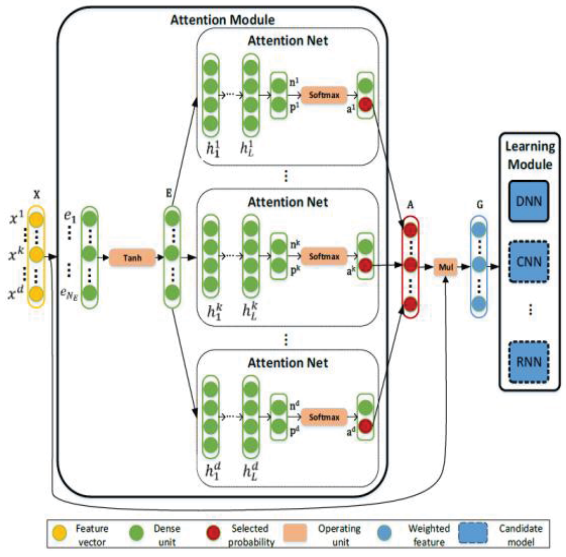

Fig. 1. AFS architecture.

B. Particle Swarm Optimization

1) Classic PSO:

PSO is a kind of optimization algorithm realized by individual cooperation. Each particle in the population has two properties: velocity and position. The velocity of the particle represents the direction and distance of the particle in the next iteration. The position of the particle indicates the position of the particle in the search space. The velocity and position of the particle are n-dimensional vectors composed of real Numbers, where n is the dimension of the problem to be optimized. In each iteration, fitness functions are used to evaluate the positions of all particles in the current population. The best location for each particle is Pbest, and all particles in the population share information to find the best location for the entire population, Gbest. Then, the velocity and position of the next generation of particles are determined according to the current position, individual optimal position and global optimal position of particles. The updated formula for the velocity and position of each particle in each generation is as follows:

$$\begin{cases} v_{id}^{k+1} = w * v_{id}^{k} + c_1 * r_1 * (p_{id}^{k} - x_{id}^{k}) + c_2 * r_2 * (p_{gd}^{k} - x_{id}^{k}) \\ x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \end{cases} \quad (1)$$

Where x^k_{id} and v^k_{id} are the position and velocity of the id-th particle at the k-th iteration respectively. p^k_{id} and p^k_{gd} are the positions of individual optimal values and global optimal values of particles. W is the inertia coefficient and is generally a constant that gradually decreases with the increase of iteration times. c_1 and c_2 are acceleration factors, usually taking the value of 2. r_1 and r_2 are random values evenly distributed among [0,1].

When the PSO is used to optimize the FS problem, each particle position corresponds to a feature subset, where 0 indicates that the feature subset does not contain the feature of the corresponding position and 1 indicates that it contains this feature.

2) Binary Backbone Particle Swarm Optimization Algorithm

Sun et al. [18] proposed a particle swarm optimization algorithm (QPSO) with quantum behavior based on PSO. QPSO cancels the attribute of speed, compared with PSO algorithm, it has simple evolution equation, few control parameters, fast convergence speed, simple operations, etc. In some practical applications, QPSO algorithm has also achieved better results than PSO algorithm [19]. Some inherent characteristics of feature selection problem decide that FS is suitable to be solved by binary algorithm. Sun et al. [20] proposed binary quantum particle swarm algorithm (BQPSO) based on QPSO.

III. PROPOSED METHOD

This section firstly introduces the initialization and length restriction mechanism of feature ranking. Then, we introduce the dual-layer Gbest update mechanism. At last, a particle swarm update mechanism is proposed.

A. Ranking of Characteristic

When the classical feature selection algorithm based on particle swarm optimization is initialized, each dimension of the particle needs to be compared with the threshold value (most values are 0.5). A value greater than this threshold value means that the particle contains this feature, otherwise it does not. However, this initialization method does not distinguish

the importance of different features. The number of features contained by all particles is a Gaussian distribution with n/2 as the standard deviation. For high-dimensional data, n/2 is still very large and can not get ideal feature selection results. This algorithm uses the method of section 2 A to reorder all features. In the first initialization of the population, all features are partitioned according to the ranking of features, and the threshold value of each zone becomes smaller as the ranking of feature zones becomes larger. That is, the better features in the front are selected with a higher probability, and the worse features in the back are selected with a lower probability.

In the evolution process, in order to help PSO jump out of the local optimum and reduce the number of features selected by FS, we propose a feature subset length restriction mechanism, which guides the particles in the population to more effective areas in the search space, so that PSO can get better solutions in a shorter time, especially in high-dimensional search space. As shown in the following formula, feature subset length (SubLen) is calculated based on feature subset scale (ps) and problem dimension (dim); iter is the current number of iterations and M is the maximum number of iterations. The change of ps is divided into two stages. In the first stage, the value of ps decreases according to the increase of iterations; in the second stage, the value of ps is determined according to the change of wrapper Gbest. When Gbest does not change for a predefined number of iterations, it can be considered that the algorithm falls into the local optimal solution, then reduce ps to jump out of this area. When the length of the feature subset of the current particle exceeds SubLen, the positions behind this particle are all taken as 0; if the length of the subset does not exceed SubLen, this particle does not change.

$$\mathrm{SubLen} = \mathrm{ps} * \dim \tag{2}$$

$$\begin{cases} iter < \frac{1}{4}M, ps = 0.4 \\ \frac{1}{4}M \le iter < \frac{1}{2}M, ps = 0.2 \\ \frac{1}{2}M \le iter \ and \ gnum > 3, ps = ps - 0.05 \end{cases} \tag{3}$$

The constants in (3) are set according to experience. This algorithm mainly considers the feature selection problem with large scale. DGPSO directly divides the length restriction mechanism in the operation process of the algorithm into two stages: the first stage is to forcibly limit the number of features. In the first quarter of the execution time of the algorithm, the length of the feature subset cannot exceed 40% of the length of the original set; in the next quarter of the time, the length of the feature subset cannot exceed 20% of the length of the original set. 20% may look small, but for a dataset with a dimension of 1000, 200 features are still selected. If the feature has been forced down, there is no way to measure how much reduction is appropriate.So DGPSO has a second stage: in the last $\frac{1}{2}$ of the running time, if the Gbest value does not change three times, we think the algorithm falls into the local optimum to a certain extent, and try to jump out of the local optimum by reducing 5% of the length each time. The number of features will not be infinitely reduced because it will be mentioned in (7) that the fitness function is affected by the number of features and the accuracy. If the number of features is infinitely reduced, the accuracy will be very poor.

Note that the features in DGPSO have been reordered by attention mechanism at this time, which is not the natural ordering of features.The dimension of all particles whose subset length exceeds the limit is set to 0 instead of directly shortening the particles, and the total length of the particles remains unchanged, so that all particles can learn from each other. In addition, when evaluating the particle performance with information entropy or KNN classifier and updating the particle position, the dimension with the value of 0 does not participate in the calculation, thus saving a lot of time. In Tran[21], a variable-length particle swarm optimization algorithm is proposed. By limiting the length, the particles search centrally. This method has achieved good results. However, features over than the limit lose the chance of being selected. Our algorithm only limits the length of feature subset instead of particle length. Even the last feature has the chance of being selected, but the probability of the latter features being selected is much smaller than the previous features.

For example, the particle is 10110011, ps is 0.5, dim is the length of this particle is 8, so SubLen is equal to 4, and the particle position after the selected fourth feature needs to be set to 0 to obtain the new particle 10110010. That is, the original feature subset is {F1,F3,F4,F7,F8}, and the transformed feature subset is {F1,F3,F4,F7}.

B. Mix of Wrapper and Filter Feature Selection

In order to obtain better results in a short period of time, the dual Gbest particle swarm optimal feature selection method (DGPSO) combines the advantages of filter and wrapper. Filter and wrapper evaluation criteria have strong consistency, and the selected direction of feature subsets is consistent [22]. Experiments show that the feature subsets selected by the filter mode and the feature subsets by the wrapper mode are usually inclusive rather than unrelated. Therefore, the consistency of the filter mode and the wrapper mode can be used, and the advantages of them can complement each other, combining the two modes, while taking into account the improvement of algorithm efficiency and the guarantee of result accuracy.

This algorithm uses a particle swarm nesting structure with a two-layer Gbest update mechanism. Information entropy (H) and mutual information (IG) come from information theory. They are the measure of the degree of interdependence between things. Symmetric uncertainty (SU) [23] is a form of normalized IG that can overcome the influence of variable units on the results. Standard mutual information is more inclined to choose the inherent disadvantages of multi-valued features. The outer PSO is a filter. The fitness function of the algorithm is shown in (6). where F is the features contained in the particles, C is the class attribute, and SubNum is the number of features in the feature subset.

$$\mathrm{IG(F|C)} = \mathrm{H(F)} - \mathrm{H(F|C)} \tag{4}$$

$$\mathrm{SU(F,C)} = \frac{IG(F|C)}{H(F) + H(C)} \tag{5}$$

$$\mathrm{F}_f\mathrm{(x)} = \frac{SU(F,C)}{SubNum} \tag{6}$$

Most SU-based fitness function the numerator is SU between the class attribute and the each feature in the subset, and the denominator is the SU between any two features in the subset. A larger value indicates that the particles are better, that is, large correlation with class, little redundancy with other features. This algorithm only calculates SU between classes and features, not calculates SU between features. When the dimension n of the problem is relatively large, the calculation amount of SU between features will increase in a square trend, which will bring huge time overhead to the

algorithm. Although the algorithm ignores the redundancy between the features, the outer PSO is to provide knowledge to the inner layer, and the final result is determined by the inner layer, so it will not reduce the performance of the algorithm. In addition, the length limitation mechanism reduces the number of features in the subset and reduces the redundancy between the subsets.

The inner layer of the algorithm is the wrapper Gbest update mechanism. The final output of feature selection is Gbest of the last iteration. In order to improve the accuracy of the results without significantly increasing the calculation time, the results of the improved algorithm can be regarded as improving Gbest of each generation. After the outer PSO has obtained the results, the inner layer algorithm is used to update the Gbest of the population. The population of the inner algorithm consists of two parts, one is a number of Pbests in front of each generation; the other is composed of several particles near Gbest, which draw lessons from the idea of that scout bees search for solutions around lead bees in bee swarm optimization algorithm.DGPSO uses KNN classification accuracy and subset size SubNum as fitness functions, and γ is a constant value between (0,1). Although classification accuracy can measure the classification performance of feature subsets, the size of subsets in feature selection problems is also an important measure.

$$F_w(x) = \gamma * \text{accuracy} + (1-\gamma)\frac{SubNum}{dim} \tag{7}$$

The outer layer of the algorithm is a PSO using a simplified SU as a fitness function, and the inner layer calculates the KNN classification accuracy of the particles that are most likely to obtain the optimal solution in each layer.

C. PSO without Speed

This algorithm discards the velocity of particles and combines with BQPSO, binary particle swarm optimization (BPSO), and feature ranking in the way of particle position update. Each dimension of the particle is one of three selected from Gbest, Pbest, and xn. The value of xn is 0 or 1, which is similar to the situation when the particle is initialized according to the feature ranking. The original positions of the particles x and x′ XOR . If they are the same, they are mutated with a small fixed probability, and if they are different, they are mutated with the pr probability.

$$\begin{cases} a \le \frac{1-w}{2}, x'(i,j) = Gbest(1,j) \\ \frac{1-w}{2} < a \le 1-w, x'(i,j) = Pbest(i,j) \\ 1-w < a \le 1, x'(i,j) = xn(i,j) \end{cases} \tag{8}$$

$$pr = b * \ln(\frac{1}{a}) \tag{9}$$

Where a is a random number between 0 and 1; w is the same as the coefficient of inertia in PSO. W decreases as the number of iterations increases, making the population more dependent on the local search capabilities of the Pbest and Gbest enhancement algorithms in the later stages. b =0.5 + 0.5 * (Maxiter-iter) / Maxiter, MaxIter represents the maximum number of iterations of the algorithm, and iter represents the current number of iterations. Variable b mutates with a high probability at the beginning of the algorithm, so the global search ability is strong. With the increase of the number of iterations, the global search ability of particles will decrease to a certain extent so as to enhance the local search ability.

Formula 8 determines the probability of choosing Gbest, Pbest and xn based on the value of w, the first two of which represent the experience gained by the particles in the search process. The third is the reinitialized value, not the particle's position. Because x′ calculated from (8) will continue the XOR operation with the position x of the particle. The generation strategy of xn is the same as that of particle initialization, still related to feature ranking. Xn is the ability of a particle to learn from its surroundings, to jump out of its place. In the early stage of the algorithm, particles gain less experience, so they learn more from xn. At this time, the global search ability is stronger. In the later stage of the algorithm, the particles are closer to the optimal solution, and the local search ability is stronger. Because the speed is abandoned, the position of the next generation is calculated by x XOR x′. (9) from BQPSO, $\ln(\frac{1}{a})$ is a random disturbance coefficient, b decreases with the number of iterations, thus reducing the probability of variation pr. DGPSO omits the velocity attribute in PSO, reduces the time cost, and the number of constants and variables.

The overall algorithm flowchart is shown in Fig. 2:

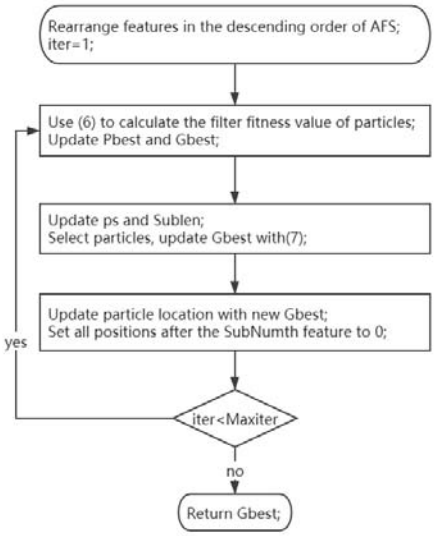

Fig. 2. DGPSO flowchart.

IV. Experimental Results And Discussion

TABLE I. Data Sets

Data set	Features	Ins.	Class	Smallest class	Largest class
Isolet	617	1560	26	4	4
SRBCT	2308	83	4	13	35
Tox_171	5748	171	4	23	26
Leukemia	7070	72	2	35	65
MLL	12533	72	3	28	39
Prostate	12533	102	2	49	51

A. Data Set

To test the performance of this algorithm, we used 6 public gene expression data sets. These data sets are publicly available atscikit-feacture feacture selection repository[24] and Biolab [25]. Table 1 shows the number of features, the number of instances, the number of classes, and the minimum and maximum number of instances in each data set for these data sets. As can be seen from Table 1, the number of instances of the data set other than the first data set is much smaller than

the number of features. As can be seen from the last two columns, some data sets also have data imbalance problems. These characteristics make FS and classification of these data sets very challenging.

B. Experimental Configuration and ParameterSettings

In order to test the performance of the DGPSO method, we compared the classification accuracy of the selected feature subset and the full feature set. Filter FS (referred to as FPSO) is a feature selection algorithm based on BPSO using (6) as the fitness function, and wrapper FS (referred to as WPSO) using KNN classification accuracy as the fitness function. We compare DGPSO with FPSO and WPSO. In addition, we compare this method with the recently proposed high-dimensional FS method using a competitive optimizer [10], and also compare with the recently proposed multi-subspace cooperative unsupervised feature selection algorithm (SRCFS) [26]. SRCFS uses random subspaces to improve the ability of unsupervised feature selection in high-dimensional space. Because SRCFS needs to manually set the feature subset size, in order to facilitate comparison with this algorithm, the subset sizes are all set to the integer up of the average subset size obtained by this algorithm.

We also compare this method with the two traditional FS, namely, giving a correlated FS (CFS) [27] and a fast correlation based FS (FCBF) [28]. We choose these two methods because they can automatically determine the number of selected features and they have a certain popularity, just like the method proposed in this paper. The CFS algorithm is a heuristic filter FS method. During the operation of the algorithm, it is preferred to select a subset that has high correlation with classes and small redundancy between features. FCBF is a two-stage algorithm that first sorts according to the relevance metric and then uses a heuristic search to remove redundant features in the subset.

For fair comparison, all methods choose the same parameters. The population size is set to 25 and the maximum number of iterations is set to 50. Because PSO is a random optimization algorithm, it runs independently thirty times on each data set. Due to the insufficient number of instances in the data set, we use ten-fold cross-validation (10F-CV). The test set is transformed according to the selected features, and the performance of the method is evaluated using KNN as a classifier, where K is set to 7.

C. Experimental Results

Table 2 shows the results of several feature selection algorithms. "Full" represents the KNN classification result using the full data set. The third and fourth columns of the table are the algorithms running time (in seconds) and the size of the feature subset selected by the algorithms. The fifth and sixth columns are the best and average classification accuracy of the algorithms. The seventh column represents the number of times DGPSO outperforms (w) / equal to (t) / inferior to (l) the comparison algorithms.

D. Results Analysis

1) DGPSO and Full: Since Full is a feature set without feature selection, there is no running time. As can be seen from table 2, the number of feature subsets obtained by DGPSO on all data sets except the first data set is 1-3 orders of magnitude smaller than the original size. The best performing data set is Leukemia, with a ratio of 1/175. Among all the algorithms, DGPSO obtained the smallest subset on almost all data sets, and significantly improved the performance of 4 of the 6 data sets. Leukemia's accuracy has increased the most, with an average accuracy improvement of approximately 13% and an optimal accuracy improvement of 21%. In the MLL data set, the algorithm selects 80 from 12,533 features, with an average accuracy improvement of 11% and a 15% improvement in the best accuracy. In Prostate, DGPSO selected about 78 features from 12533, with an average accuracy improvement of 5% over the full set and an optimal accuracy improvement of 10%.

TABLE II. AVERAGE TEST RESULTS

Data set	Method	Time(s)	Size	Best	Mean	W/t/l
Isolet	Full		617		**95.83**	
	FPSO	30.13	312.5	95.51	94.33	
	WPSO	869.33	323.6	95.51	95.22	
	CSO	2542.27	136.5	92.34	90.99	
	SRCFS	**4.46**	124	75.64	75.00	
	DGPSO	111.66	**123.8**	**97.12**	95.00	3/0/2
SRBCT	Full		2308		86.67	
	FPSO	26.64	1165.8	93.33	85.33	
	WPSO	30.08	1243.6	93.33	88.00	
	CSO	75.55	84.6	88.56	83.13	
	SRCFS	0.5	21	66.67	62.00	
	DGPSO	16.57	**20.4**	**100.00**	**89.33**	5/0/0
Tox_171	Full		5748		**96.97**	
	FPSO	61.04	2920.1	100.00	96.66	
	WPSO	162.8	2982.5	100.00	97.27	
	CSO	622.34	**80.7**	95.68	89.40	
	SRCFS	0.97	542	72.73	66.69	
	DGPSO	39.78	541.1	100.00	95.15	2/0/3
Leuke mia	Full		7070		78.57	
	FPSO	54.29	3630.5	85.71	81.42	
	WPSO	83.06	3859	95.71	81.71	
	CSO	309.02	171.6	91.22	89.45	
	SRCFS	0.725	41	71.43	65.07	
	DGPSO	28.55	**40.40**	**100.00**	**92.14**	5/0/0
MLL	Full		12533		84.62	
	FPSO	116.92	6381.1	85.21	83.73	
	WPSO	168.2	6603.3	85.35	84.61	
	CSO	481.48	365.1	88.92	86.75	
	SRCFS	0.84	81	84.62	80.33	
	DGPSO	58.07	**80.8**	**100.00**	**95.39**	5/0/0
Prostate	Full		12533		90.00	
	FPSO	101.60	6496.7	94.97	90.51	
	WPSO	184.58	6578.5	91.23	90.87	
	CSO	514.76	360.4	90.23	84.02	
	SRCFS	1.14	78	52.92	50.00	
	DGPSO	46.78	**77.8**	**100.00**	**95.00**	5/0/0

2) DGPSO and FPSO: According to Table 2, the feature subset generated by DGPSO provides higher accuracy than FPSO on 5 data sets of 6 data sets, and the average numbers of features selected on most data sets are at least one order of magnitude than FPSO. The largest decrease in the subset is the last data set. The feature selected by DGPSO and the FPSO ratio are 1/83, and the performance is still improved by 4%. Although the classification accuracy of FPSO is slightly higher than that of DGPSO on the third data set, the best classification accuracy of the two is the same, and the number of feature subsets has decreased by 2379. Therefore, the mixed mode of filter and wrapper proposed in this paper is effective. While maintaining or improving the performance of filter classification, the number of feature subsets is greatly reduced to achieve the purpose of removing redundant features.

3) DGPSO and WPSO: Although the number of features selected by WPSO is only half of the original, DGPSO selects at least one order of magnitude fewer features than WPSO on most data sets, and it is significantly better than the classification accuracy of WPSO in 4 data sets, one data set approximates the classification accuracy of WPSO. In Leukemia and MLL data sets, the average classification accuracy

improves by up to 10%. On the data set Prostate, the dimensionality decline is a maximum of 6,500, and the average performance is still improved by 4%. Only in Tox_171, the classification accuracy of DGPSO is 2.1% lower than that of WPSO, and the number of selected features is 1/5 of WPSO, and the best accuracy of both is 100. It can be seen that the classification results obtained by the hybrid mode in this paper are not worse than the wrapped type in most cases, and the number of features is greatly reduced.

4) DGPSO and CSO: Compared with CSO, DGPSO has obtained a smaller feature subset on 5 data sets. On the fifth data set, CSO selected 365 features, while DGPSO selected only 80.8 features. The performance of DGPSO on 6 data sets is significantly better than CSO.

5) DGPSO and SRCFS: Since SCRFS is a subset size set artificially by DGPSO, the subset sizes of the six data sets are the same. Because SRCFS is an unsupervised FS algorithm, it has fast speed but sacrifices classification accuracy to a certain extent. Therefore, the classification performance of DGPSO on six data sets is better than SRCFS.

In summary, DGPSO has won 25 times in classification performance in 30 comparisons and achieved the smallest feature subset in all 29 comparisons. The obtained results prove that DGPSO performs a better search than the comparison methods. DGPSO effectiveness is contributed by two mechanisms, the length limitation mechanism and the dual Gbest mechanism. The length limitation mechanism makes particles more concentrated in the effective space, reduces redundant features in the subset, and enables PSO to find a smaller subset of features. And when there is a local optimal situation, the length limitation mechanism can also make the population change the search space without discarding the learned experience. The dual Gbest mechanism can further improve the accuracy of the algorithm and lead the population to search in a more needed direction.

E. Calculation Time

The results show that the proposed algorithm is an effective combination of wrapper and filter. The running time is shorter than the filter type, the classification accuracy is better than the wrapped type in most cases, and it has a smaller feature subset.

F. Comparison with Traditional Methods

To compare the proposed algorithm with traditional methods, we compare DGPSO with CFS and FCBF. Table 3 shows the time (seconds), the selected feature subset size, the best classification accuracy and average accuracy. Bold indicates the best performing data.

It can be seen from Table 3 that DGPSO has the best classification accuracy in all data sets, the feature subset size is the smallest twice, and the algorithm running time is slower than FCBF and faster than CFS. Although the Tox_171 data set has many more features than CFS and FCBF, it is still only 1/10 the number of the full set, and the classification accuracy is improved by 21% and 15%, respectively. This shows that DGPSO can better explore the solution space to obtain a better solution than the traditional method in a reasonable running time.

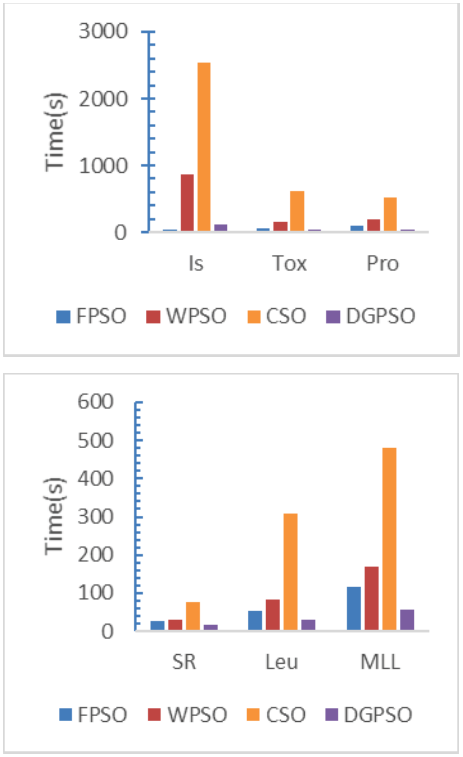

Fig. 3. Running time.

TABLE III. COMPARISON OF DGPSO AND TRADITIONAL METHODS

Data set	Method	Time(s)	Size	Best	Mean
Isolet	CFS	21.48	132.5	87.63	
	FCBF	5.81	**30.0**	79.17	
	DGPSO	111.66	123.8	**97.12**	95.00
SRBCT	CFS	37.02	72.7	99.17	
	FCBF	2.87	56.3	93.33	
	DGPSO	16.57	**20.4**	**100.00**	89.33
Tox_171	CFS	896.46	97.1	78.79	
	FCBF	2.67	**72.8**	84.85	
	DGPSO	39.78	541.1	**100.00**	95.15
Leukemia	CFS	211.42	58.4	97.30	
	FCBF	1.97	48.7	98.14	
	DGPSO	28.55	**40.4**	**100.00**	92.14
MLL	CFS	1516.52	104	92.31	
	FCBF	3.95	**68.6**	92.31	
	DGPSO	58.07	80.8	**100.00**	95.39
Prostate	CFS	1150.95	65.5	95.12	
	FCBF	4.14	**48.0**	93.96	
	DGPSO	46.78	77.8	**100.00**	95.00

G. Comparison with Traditional Methods

In order to investigate the effect of length restriction mechanism, we analyze the results of DGPSO with (W) and without (WO) length change mechanism to analyze the population change. Fig. 4 shows the change in the number of average feature subsets of W and wo in 50 iterations. The difference of running time between the two methods is contributed by the size of feature subset, which affects the adaptive evaluation time and particle update time. As can be seen from Fig. 4, the number of features of W decreases significantly in the previous iterations, and then decreases slowly; the number of features of Wo increases slightly in the previous iterations, and then tends to be stable. These figures clearly show that the number of feature subsets of W is much smaller than that of Wo, so the time of accuracy evaluation and speed update of W is significantly shortened. It also means that the final feature subset of W method is much smaller than that of Wo method.

V. Conclusions And Future Works

This paper aims to propose a new particle swarm optimization (PSO) -based FS method (DGPSO). This method mixes filter and wrapper to form a dual-Gbest update mode with fitness functions of filter and wrapper, respectively. In addition, a length limitation mechanism is proposed. As the number of iterations increases or falls into a local optimum, the length of the restriction is reduced to eliminate redundant features, jump out of the local optimal solution, and obtain a smaller feature subset and less operation time.

Experimental results on 6 high-dimensional data sets show that compared with 6 comparison algorithms, the proposed dual-Gbest method based on length limitation mechanism can obtain almost the smallest feature subset and good classification accuracy in a short time.

The results and analysis show that the proposed algorithm shows promise in FS. The trade-off between classification accuracy and subset length of the algorithm is controlled by simple constants. In future work, you can further analyze the relationship between the two, and use the current state of the algorithm or the population shape to control the relationship between the two. It is found through experiments that the unsupervised feature selection is fast, but the results are not ideal. We will consider this direction in the future.

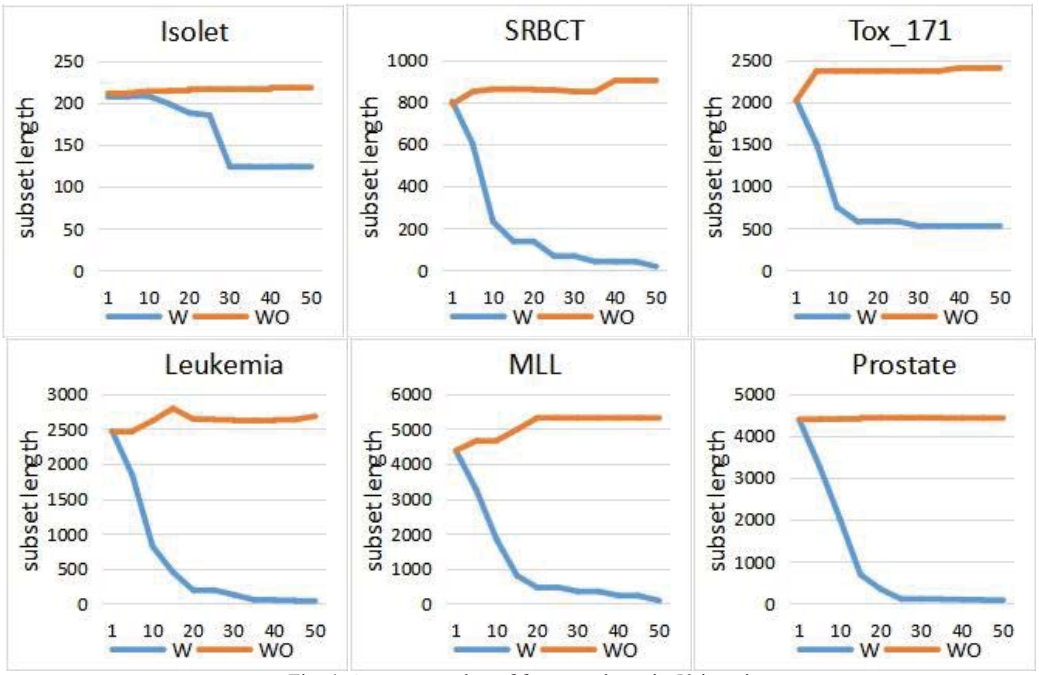

Fig. 4. Average number of feature subsets in 50 iterations.

Acknowledgment

We would like to acknowledge the support from the National Science Foundation of China (Nos. 61472095). This paper is funded by the International Exchange Program of Harbin Engineering University for Innovation-oriented Talents Cultivation.

References

[1] *M. Dash, "Feature selection via set cover," in Proc. IEEE Knowl. Data Eng. Exchange Workshop, Newport Beach, CA, USA, Nov. 1997, pp. 165–171.*

[2] *I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," J. Mach. Learn. Res., vol. 3, pp. 1157-1182, Mar. 2003.*

[3] *Y. Wang and L. Feng , " A new hybrid feature selection based on multi-filter weights and multi-feature weights," Applied Intelligence,vol.6 , 2019.*

[4] *A. J. Ferreira and M. A. Figueiredo, "Efficient feature selection filters for high-dimensional data," Pattern Recognition Letters, vol. 33, no. 13,pp. 1794–1804, 2012.*

[5] *B. Xue, M. Zhang, and W. N. Browne, "Single feature ranking and binary particle swarm optimisation based feature subset ranking for feature selection," in Proceedings of the Thirty-fifth Australasian Computer Science Conference-Volume 122. Australian Computer Society, Inc.,2012, pp. 27–36.*

[6] *D. Hongbin, L. Tao, D. Rui and S. Jing, "A novel hybrid genetic algorithm with granular information for feature selection and optimization, " Appl. Soft Comput. ,vol 65,pp. 33-46, 2018.*

[7] *A. W. Whitney, "A direct method of non parametric measurement selection," IEEE Trans. Comput., vol. C-20, no. 9, pp. 1100–1103,Sep. 1971.*

[8] *T. Marill and D. M. Green, "On the effectiveness of receptors in recognition systems," IEEE Trans. Inf. Theory, vol. IT-9, no. 1, pp. 11–17,Jan. 1963.*

[9] *H. Vafaie and I. Imam, "Feature selection methods: genetic algorithms vs. greedy-like search," Proceedings of the International Conference on Fuzzy and Intelligent Control Systems. 1994, pp.51- 28.*

[10] *J. Kennedy and R. Eberhart, "Particle swarm optimization," in IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948.*

[11] *W. Junliang , Z. Jie and W. Xiaoxi, "A Data Driven Cycle Time Prediction With Feature Selection in a Semiconductor Wafer Fabrication System," IEEE Transactions on Semiconductor Manufacturing, vol .31(1),pp. 173-182 , 2018.*

[12] *S. Gu, R. Cheng and Y. Jin, "Feature selection for high-dimensional classification using a competitive swarm optimizer,". Soft Computing, vol.22(3),pp.811-822, 2018.*

[13] *X. Bai, X. Gao and B. Xue , "Particle swarm optimization based two-stage feature selection in text mining,"2018 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2018,pp.1-8.*

[14] *M. Liu , L. Xu and J. Yi , et al, "A feature gene selection method based on ReliefF and PSO,"2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA). IEEE, 2018,pp. 298-301.*

[15] *T. Butler-Yeoman, B. Xue, and M. Zhang, "Particle swarm optimisation for feature selection: A hybrid filter-wrapper approach,"in IEEE Congress on Evolutionary Computation, 2015, pp. 2428–2435.*

[16] *B. Tran, M. Zhang andB. Xue , "A PSO based hybrid feature selection algorithm for high-dimensional classification,"2016 IEEE congress on evolutionary computation (CEC). IEEE, 2016,pp.3801-3808.*

[17] *N. Gui , D. Ge and Z. Hu, "AFS: An Attention-based mechanism for Supervised Feature Selection," AAAI, 2019.*

[18] *S. Jun , F. Bin and X. Wenbo , "Particle swarm optimization with particles having quantum behavior," Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753). IEEE, vol. 1,pp. 325-331, 2004.*

[19] *U. Rehman, S. Yang and S. Khan , et al, "A Quantum Particle Swarm Optimizer With Enhanced Strategy for Global Optimization of Electromagnetic Devices," IEEE Transactions on Magnetics, PP(99):1-4, 2019.*

[20] *X. Maolong ,S. Jun and W. yong, "A binary coded quantum particle swarm optimization algorithm," Control and decision making, pp(1):102-107, 2010.*

[21] *B. Tran,B. Xue and M. Zhang , "Variable-Length Particle Swarm Optimization for Feature Selection on High-Dimensional Classification," IEEE Transactions on Evolutionary Computation, vol. 23(3),pp. 473-487, 2018.*

[22] *Z. Yamin ,"Improvement of feature selection method based on genetic algorithm, "Chongqing University, 2008.*

[23] *W. H. Press, S. Teukolsky, W. Vetterling, and B. Flannery, Numerical Recipes in C, vol. 1. Cambridge, U.K.: Cambridge Univ. Press, 1988,p. 3.*

[24] *http://featureselection.asu.edu/datasets.php*

[25] *http://www.biolab.si/supp/bi-cancer/projections-/info/SRBCT.html.*

[26] *D. Huang,X. Cai and C D. Wang , "Unsupervised feature selection with multi-subspace randomization and collaboration," Knowledge-Based Systems, 2019, pp. 182: 104856.*

[27] *M. Gutlein, E. Frank, M. Hall, and A. Karwath, "Large-scale attribute selection using wrappers," in Proc. IEEE Symp. Comput. Intell. Data Min., 2009, pp. 332–339.*

[28] *L. Yu and H. Liu, ″Feature selection for high-dimensional data: A fast correlation-based filter solution,″ in Proc. 20th Int. Conf. Mach. Learn.(ICML), Washington, DC, USA, 2003, pp. 856–863.*