Feature Selection using Modified Particle Swarm Optimization

Khushboo Jain
Computer Engineering Department
Shri G.S. Institute of
Technology and Science
Indore-452003 (M.P.) India

Anuradha Purohit
Computer Technology and
Application Department
Shri G.S. Institute of
Technology and Science
Indore-452003 (M.P.) India

ABSTRACT
Feature selection is the process by which relevant features are selected from large datasets in order to improve the performance of the classification systems. There are various approaches that are used for feature selection such as Soft Computing, Hill Climbing etc. Particle Swarm Optimization is now a days popularly used soft computing technique for feature selection due to its searching ability, simplicity and low computation cost. But the main problem with Particle Swarm Optimization is premature convergence which in turn affects the classification performance. In this paper, a modified Particle Swarm Optimization is proposed for feature selection. To handle the problem of premature convergence, a flipping operator is introduced before the updation of velocity and position of the particle. Fitness of each particle is computed using Support Vector Machine based fitness function. To establish the effectiveness of proposed approach, testing is done on various benchmark datasets like wine, zoo, sonar etc. Results obtained on these datasets are compared with the standard approach and satisfactory improvements are observed.

Keywords
Feature Selection, Particle Swarm Optimization, Classification, Support Vector Machine.

1. INTRODUCTION
Classification is an important task in data mining. It is helpful in predicting the class of data available on the basis of knowledge extracted from the existing data. Various classifiers are used for classification and their performance depends on the number of features present in the data-set and their values. Hence, features play important role in classification. The irrelevant and redundant features present in the data-set may reduce the performance of the classifier. Feature selection algorithms are then used to remove the irrelevant and redundant features from the original data-set in order to improve the classification accuracy. There are many approaches used for feature selection such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Graph Based Clustering, Relief Algorithm etc.

PSO is one of the most widely used approach among swarm intelligence in which the swarm searches the optimal solution. PSO was motivated from simulation of social behavior of bird flocking. Each particle in the swarm searches for the best solution by updating the position and velocity based on the best experience of its own and its neighbor particles [9]. Particle Swarm Optimization is now a days popular among all the Soft Computing techniques due to its characteristic of being simple to understand, easy to implement with the adjustment of few parameters, limited calculations, and searching capabilities.

Support Vector Machine (SVM) is supervised learning algorithm in which data are analyzed and patterns are recognized for classification [10]. In SVM, hyper plane or set of hyper-planes is constructed in a high dimensional space which can be used for classification, regression or other tasks. A good separation is achieved with the maximal margin hyper plane which gives lower generalization error of the classifier.

In this paper, a modified Particle Swarm Optimization is proposed for feature selection. The proposed method uses a flipping operator before the updation of velocity and position of each particle. The main reason for using this operator is to explore the search space efficiently which improves the accuracy of the classification. Fitness of each particle is computed using Support Vector Machine based fitness function.

This paper is organized as follows: Section 2 discusses the background study of the techniques used. Related work done by various researchers in the field of particle swarm optimization and Support Vector Machine have been presented in section 3. The proposed approach with detail steps are described in section 4. Experimentation done and results obtained are presented in section 5. Finally, section 6 outlines the main conclusion of the work.

2. BACKGROUND
2.1 Particle Swarm Optimization
Particle Swarm Optimization was introduced by Rousell Eberhart and James kennedy in 1995 [9]. PSO was mainly focused to solve non-linear continuous optimization problem but recently used for real life application problem. PSO was inspired by sociological behavior of birds flocking. Birds have a capability to fly in large group without collision for long distance and maintain their optimum distance between themselves and their neighbor.

Particle Swarm Optimization (PSO) is a population-based stochastic optimization technique. PSO consist of set of solution called particles and a group of particles called swarm. Particles moved in a search space with a specified velocity for finding the optimal result. Every particle contains its memory that helps it in keeping the information of its previous best position. The position of the particles is distinguished as personal best and global best. The velocity of the particle is adjusted according to the historical behavior of each particle and its neighbor while they fly through the search space. Each move of the particle is influenced by its current position, its memory of previous useful parameter, and the group knowledge of the swarm. The velocity of the particles are updated using equation (1).

$$v_{pd}^{new} = w \times v_{pd}^{old} + c_1 r_1 \left(pbest_{pd} - x_{pd}^{old}\right) + c_2 r_2 \left(gbest_d - x_{pd}^{old}\right) \quad (1)$$

Where, c_1 and c_2 are acceleration constant, r_1 and r_2 are the random values in the range between 0 and 1, w is a inertia weight, x_{pd}^{old} shows the position of each particles in d-dimensional space, v_{pd}^{new} and v_{pd}^{old} are the particle velocity, $pbest_{pd}$ is the best previous position of each particle called particle best position, $gbest_d$ is the best position of particles called the global best. The position of the particles is updated using the equation (2).

$$x_{pd}^{new} = x_{pd}^{old} + v_{pd}^{new} \quad (2)$$

where, x_{pd}^{new} is the new position of the particle, x_{pd}^{old} is the previous best position of the particle. The sequential flow of this process is given step-by-step using following algorithm:

Algorithm: **PSO**

Step1: Initialize particles, velocity and position.

Step2: Calculate fitness of each particle.

Step3: Update local best.

Step4: Update global best.

Step5: Update velocity.

Step6: Update position.

Step7: If acceptable solution is found or some other stopping criterion is met return the best solution else go to Step 2.

2.2 Support Vector Machine

Support Vector Machine is a data classification technique that was first developed by Vapnik in 1995 [10]. It is based on supervised learning. SVM works on high dimensional data and avoids curse of dimensionality problem. It represents the decision boundary using a subset of training examples, known as support vectors. SVM can be trained to classify both linearly separable and non-linearly separable data.

For linearly separable classes the training data set contain k cases and represented as $\{x_i, y_i\}$, i=1,...,k, where $x_i \in R^N$ is an N-dimensional space and $y \in \{-1,+1\}$ is the class label. These training patterns are linearly separable if there exists a vector w (determining the orientation of a discriminating plane) and a scalar b (determining the offset of the discriminating plane from the origin) as given in equation (3).

$$Y_i (w * x_i + b) - 1 \geq 0 \quad (3)$$

For linearly non-separable classes, the restriction that all the training cases of a given class lie on the same side of the optimal hyper-plane can be relaxed by the introduction of a "slack variable" $\xi_i \geq 0$.

For nonlinear decision surfaces, a feature vector $x_i \in R^N$ is mapped into a higher dimensional feature space F via a linear vector function $\Phi : R^N \rightarrow F$ The optimal margin problem in F can be written by replacing $x_i * x_j$ with $\Phi(x_i)*\Phi(x_j)$ which is computationally expensive. To address this problem, Vapnik [10] introduced the concept of using a kernel function K in the design of nonlinear SVM as shown in equation (4).

$$K(x_i, x_j) = \Phi(x_i)*\Phi(x_j) \quad (4)$$

The other kernel functions that can be used are:

- Linear: $K(x_i, x_j) = x_i^T x_j$
- Polynomial of power p: $K(x_i, x_j) = (1 + x_i^T x_j)^p$
- Sigmoidal: $K(x_i, x_j) = \tanh(\beta_0 x_i^T x_j + \beta_1)$

3. RELATED WORKDONE

Feature Selection is one of the technique of dimensionality reduction in which small subset of relevant features are selected that minimize redundancy and maximize relevance to the target such as class labels in classification [4]. Various techniques have been proposed by researchers to perform feature selection. A lot of work using soft computing technique has also been reported in the literature.

Jiliang Tang et al. in [5] have given a survey on feature selection for classification. They have discussed various types of features available in the datasets and a review on various methods available for performing feature selection like filter, wrapper and embedded algorithms.

Stanislaw Osowski et al. in [6] proposed application of Genetic Algorithm and Support Vector Machine to recognize the blood cell by evaluating the image of the bone marrow aspirate. By using GA, the features are selected which are further used by the SVM to recognize and to classify the cells. The main advantage that GA method provide lies in combining the ranking of all features and then finding the optimal number of them. But Genetic Algorithm (GA) suffers from a few shortcomings such as it doesnot has a memory concept. If a chromosome is not selected, the information contained by it is lost. To remove this limitation PSO can be used.

Bing Xue et al. in [7] proposed a PSO based feature selection approach for selecting a smaller number of features and achieving similar or even better classification performance than using all features. They developed three new initialization strategies motivated by forward selection and backward selection, and three new pbest and gbest updating mechanisms considering both the number of feature and the classification performance to overcome the limitation of the traditional updating mechanism.

Chung- Jui Tu et al. [8] proposed an approach for feature selection using PSO-SVM. PSO is used to perform feature selection. Further for the classification process support vector machine is used and one-versus-rest method serve as a fitness function of PSO. The main problem in the algorithm is the premature convergence of PSO. Due to this, searching abilities of PSO can be reduced and inturn affects the classification performance.

4. PROPOSED APPROACH

In this paper, a modified PSO based feature selection approach is proposed. Using this approach irrelevant and redundant feature can be removed from the data-set. For this PSO is combined with SVM. PSO is used for feature selection and SVM is used to calculate the fitness of the particle. A flipping operator is introduced to remove premature convergence problem of PSO so that better results can be achieved. This operator is applied on half of the elites having low fitness value before updating the position and velocity of each particle in swarm. The block diagram of proposed approach is as shown in Figure 1.

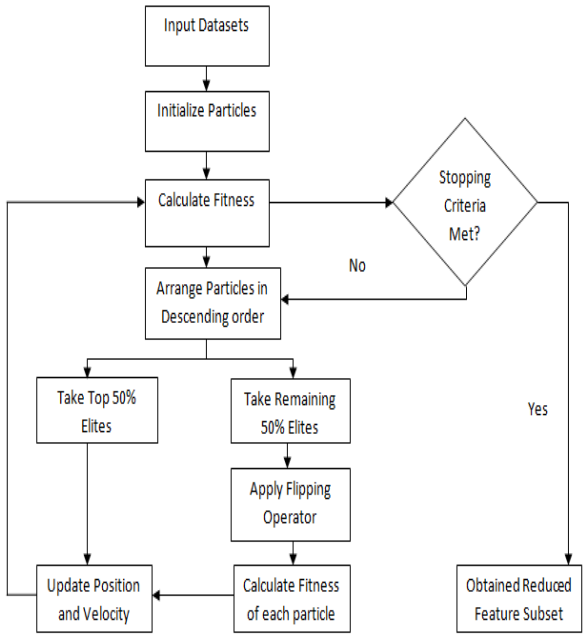


Fig 1: Flow Chart of Proposed Approach

The modified PSO is carried out using following steps:

(i) Initialization of swarm.

(ii) Fitness Evaluation Process.

(iii) Partitioning of the Elites.

(iv) Applying Flipping Operator.

(v) Update Position and Velocity of the Particle.

(vi) Termination Criterion.

The steps identified are explained in detail as follows:

(i) Initialization of swarm:
Initialization of population is done by generating particles in form of 0, 1 strings. For this, random generation operator is used. Here 0 represent the feature is not selected and 1 represent the feature is selected. The size of the particles depends on the attribute present in the datasets.

(ii) Fitness Evaluation Process:
Fitness function is used to compute the fitness of each particle. Fitness value of particle represents its chances of survival. To evaluate the significance of feature, classifier dependent fitness function is used. For calculating the fitness, the dataset is divided into two sets known as training set and testing set. The fitness is calculated using the fitness function which gives accuracy of the particles. The fitness is calculated by the following fitness function:

$$Fitness = Accuracy\ of\ Classification \quad (5)$$

$$Accuracy = \frac{\text{Number of samples correctly classified}}{Total\ number\ of\ samples} \quad (6)$$

$$Accuracy = \frac{tp+tn}{tp+tn+fp+fn} \quad (7)$$

Where, tp corresponds to the number of positive samples correctly predicted by the classified model, tn corresponds to the number of negative samples correctly predicted by the classified model, fp corresponds to the number of negative samples wrongly predicted as positive by the classified model, and fn corresponds to the number of positive samples wrongly predicted as negative by the classified model.

(iii) **Partitioning of the Elites:**
According to the fitness value, evaluated particles are arranged in descending order. Then partitioning of particles is done by selecting 50% of the initial particles from the sorted list. These elites are selected for directly updation of their position and velocity. The remaining are selected for flipping operation and then are sent for updation process.

(iv) Applying Flipping Operator:
In this step, the flipping operation is applied on the particles having low fitness value. In flipping, bits are flipped according to the probability decided for flipping. The main reason of flipping the bit is to explore the search space. The probability of the flipping operator is 0.1.

(v) Update Position and Velocity of the Particle:
Velocity represents the probability of a bit to take the value 0 or 1. For updating the position and velocity of particles, first the random velocity and positions are initialized to each particle. For determining the global best among swarm, other particles are updated. In each iteration local best velocity and position of each particle is updated by comparing it with global best and fitness is calculated again. If any other particle has large fitness value than previous global best then the global best is updated. Position of each particle is represented as attribute subset and velocity as positive integer which lies between 1 and V_{max} (maximum velocity) of the particle. This implies the changes to be made in the particle to come closer to the global best position. The number of different bits between the particles represent the difference between their positions.

For example, to update velocity and position of the particles the global best is computed from the swarm. Suppose the global best of the particle P_{gbest} is [1,0,1,0,0,1,1,0,1] and there are other particles local best position X_i is [0,1,1,0,0,0,1,1,1]. Then to update the velocity of particles, the difference between gbest and particles current position are Pgbest - X_i, this gives with the values [1, -1, 0, 0, 0, 1, 0, -1, 0] here 1 signifies that compared with the best position, this feature should be selected but it is not, -1 means that compared with the best position, this feature should not be selected but it is present. 0 signifies that the bit does not require any changes. After updating velocity, a particles position will be updated by a new velocity.

(vi) Termination Criterion:
To stop the process, following termination criteria are used:

(a) Predefined number of iterations are completed, or
(b) Fitness with 100 \% accuracy is obtained.

5. EXPERIMENTAL RESULTS

Proposed approach is implemented on R language using RStudio framework. To evaluate the results obtained using existing approach and proposed approach six benchmark datasets are taken from UCI Machine Learning Repository [11]. These datasets are Wine, Breast Cancer Wisconsin Diagnostics (WDBC), Sonar, Glass, Zoo and Ionosphere. A brief description of the datasets used is summarized in the Table 1.

Table 1. Datasets used for Experimentation

Datasets	Number of features	Number of classes	Number of Instances	Attribute Type
Wine	13	3	178	Real, Integer
Ionosphere	34	2	351	Integer, Real
Sonar	60	2	208	Real
WDBC	32	2	569	Real
Zoo	17	7	101	Categorical, Integer
Glass	10	6	214	Real

Various parameters are used to achieve best results on the datasets selected. 10 fold cross validation method is used for testing. Results are achieved on following parameter settings:

- Number of Generations: 10 - 100
- Particle Size: 10 - 100
- Training Samples (in %): 10 - 90
- Testing Samples (in %): 10 - 90
- Flipping Probability: 0.1

The proposed approach is tested on the basis of classification accuracy of the feature subsets obtained. Table 2 shows the average number of features obtained and their computed classification accuracy.

Table 2. Results Obtained for the Proposed Approach

Dataests	Number of Features Present	Average Number of features obtained	Classification Accuracy (in %)
Wine	14	6	98.645
Ionosphere	34	14	93.039
Sonar	60	18	78.846
WDBC	32	12	99.381
Zoo	17	6	99.634
Glass	10	5	94.244

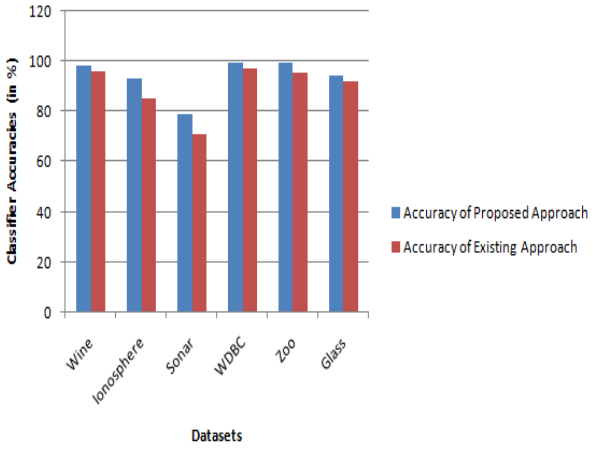
Results obtained using proposed approach is compared with the results obtained using existing approach as shown in the Table 3.

Table 3. Comparison of Existing and Proposed Approach

Datasets	Proposed Approach		Existing Approach	
	Average Number of features Obtained	Classification Accuracy (in %)	Average Number of features Obtained	Classification Accuracy (in %)
Wine	7	98.645	7	95.830
Ionosphere	14	93.039	16	85.533
Sonar	18	78.846	19	71.042
WDBC	12	99.381	12	97.200
Zoo	6	99.634	8	95.327
Glass	5	94.244	6	92.051

Comparative results shows that the proposed approach performs better than the existing approach in average number of features obtained and classification accuracy for all the datasets taken. The existing approach gives less classification accuracy, as it starts giving constant results in early generations (premature convergence) which keeps it away from exploring good results. Whereas due to the use of new flipping operator in proposed approach, the results obtained in each generation are improved. This is because search space is explored efficiently by flipping operator. The modification in the approach makes it more efficient in searching which gives better results than the existing approach in terms of classification accuracy.

Comparison between proposed approach and existing approach as shown in Table 3 is represented graphically in Figure 2. Horizontal axis presents the six bench mark datasets used in this project and vertical axis presents the classification accuracy (in percentage).



As compared to the existing approach, 2.815\% improvement in classification accuracy is obtained for Wine dataset; 7.506\% improvement in classification accuracy is obtained for Ionosphere dataset; 7.804\%, 2.181\%, 4.327\%, and 2.193\% improvement in classification accuracy is obtained for Sonar, WDBC, Zoo and Glass datasets respectively. The overall accuracy of the classification system is improved for all the datasets.

6. CONCLUSION

In this paper, a modified Particle Swarm Optimization approach is proposed to perform feature selection in order to improve classification accuracy. The standard Particle Swarm Optimization suffers from the problem of premature convergence and hence unable to explore good results at some point of time. For this a new flipping operator is introduced. This operator is applied to the particles having low fitness value. Particle Swarm Optimization is used as a search

technique. The flipping operator introduced, helps in exploring the search space efficiently and improves the performance of classification. For evaluating classification accuracy of the obtained feature subset, Support Vector Machine based fitness function is used. Results obtained for the datasets taken are compared with existing approach and satisfactory improvements are observed. An overall improvement of 2% to 8% has been observed for datasets taken.

7. REFERENCES

[1] M. Dash and H. Liu, "Feature selection for classification," Intelligent Data Analysis, vol. 1, no. 1–4, pp. 131–156, 1997.

[2] A. Unler and A. Murat, "A discrete particle swarm optimization method for feature selection in binary classification problems," Science Direct Trancastion on European Journals of Operational Research, vol. 206, no. 3, pp. 528–539, Nov. 2010.

[3] Xiaodong zhu, Yoanning Liu, Gang Yang , Hao Dong, Sujing Wang, Huiliag Chen, "An Improved particle swarm optimization for Feature Selection," Science Direct Transaction on Bionic Engg., vol. 8, issue 2, pp. 191-200, June 2011.

[4] Isabelle Gyuon, Andre Elisseeff, "An Introduction to variable and Feature Selection," Journal of Machine Learning Research, pp. 1157-1182, March 2003.

[5] Jiliang tang, Saleem Alelyani and Huan Liu, " Feature selection for classification: A Review," Thesis, pp. 1-29, 2014.

[6] Stanislaw Osowski, Robert Siroi, Tomasz Markiewicz, and Krzysztof Siwek, "Application of Support Vector Machine and Genetic Algorithm for Improved Blood Cell Recognition," IEEE transaction instrumentation and measurement, vol. 58, no. 7, pp. 2159-2166, July 2009.

[7] Bing Xue, Mengjie Zhang, Will N.Browne, "Particle swarm optimization for feature selection in classification: Novel initalisation and updating mechanism," Applied Soft Computing, pp. 261-276, New Zealand, May 2014.

[8] Chung-Jui TuSSS, "Feature Selection using PSO-SVM," IAENG International Journal of Computer Science, Feb. 2007.

[9] J. Kennedy and R. C. Eberhart, "Particle swarm optimization" In Proceedings of the 1995 IEEE International Conference on Neural Networks, volume 4, pages 1942–1948, IEEE Press, Piscataway, NJ, 1995.

[10] Vapnik V., " The nature of statistical learning theory," Statistics of engg. and Tnformation science, Springer-Verlag, New York NY, 1995.

[11] http://archive.ics.uci.edu/ml/