VOL. 10, NO 20, NOVEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

ENHANCED CLASSIFICATION USING PSO WITH EMBEDDED ATTRIBUTE ELIMINATION TECHNIQUES

M. Balasaraswathi and B. Kalpana
Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India
E-Mail: baladars@yahoo.co.in

ABSTRACT

Massive information created in the current scenario has led to a major bottleneck in terms of processing. The vast data that is available is not completely usable, in the sense; it does not entirely contain data that guides to the final results. The data tends to contain missing or redundant information, or information that is irrelevant to the study. Removing these data will not only reduce the processing time, it also enhances the accuracy of the processing algorithm. This paper presents a modified PSO algorithm (HPSO) that has embedded attribute elimination techniques. Analysis proves that HPSO consumes less time and provide better accuracy when compared to PSO.

Keywords: attribute elimination, classification, filtering, PSO.

1. INTRODUCTION

The current information age has led to generation of huge amount of data, while the lowered cost of the hardware has led to easy and efficient storage of all the generated data [18]. The importance of a data is revealed only on extracting useful information from it. Due to the huge nature of the data, it becomes difficult for the algorithms to process the data to turn it into useful information. Further, not all the recorded data will be relevant for a particular task. The data is bound to contain irrelevant and redundant information, which needs to be identified and removed for improving the accuracy of the results.

Machine Learning algorithms are a class of structures that automatically improve their performance with training and experience. The major functionality of a machine learning algorithm is to perform prediction. The process of prediction is performed by learning the data associated with the problem. The process of learning is termed as the training phase, and as the algorithm encounters more and more data, it learns better and hence these class of algorithms can effectively predict even in dynamic environments. One major disadvantage of these algorithms is that they are time consuming. But, this downside can be overcome by including a heuristic in the design.

Heuristics, in algorithms are used to find a solution among all possible ones [19]. The identified solution is not guaranteed to be best solution; instead, it is guaranteed to be one of the optimal solutions that is closest to the best solution. The advantage of using such an approach is that they tend to provide near optimal solutions within a short span of time, which makes them ideal for most of the real time applications. Due to this nature, heuristics become the best candidates for incorporating into machine learning frameworks.

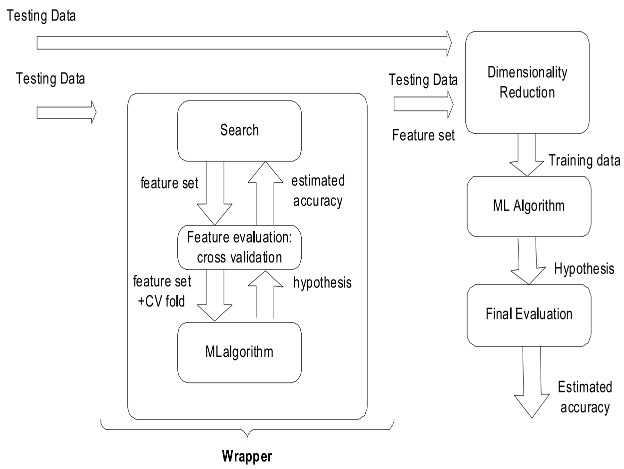

Figure-1. Wrapper.

Particle Swarm Optimization (PSO) [11, 20] is a metaheuristic optimization technique being used for the current study. PSO is a computational method that optimizes a problem by improving a candidate solution in an iterative manner with regard to a given quality measure. This paper proposes a method that uses a modified form of PSO by embedding it with feature selection techniques for accurate and faster results.

PSO not only lends itself for pure computational analysis, it also provides excellent results, when hybridized with other techniques.

Various approaches use the PSO algorithm for certain phases in their optimization process. Some of the mostly used approaches that integrate PSO to provide optimization in a hybrid manner includes; Artificial Neural Network [4] and Support Vector Machines [16]. These approaches tend to combine well with PSO and hence they are proven to provide effective results.

ARPN Journal of Engineering and Applied Sciences

A survey on the accuracy of Particle Swarm Algorithms in data mining is presented in [2, 5]. Three different PSO variants are used here and are tested against Genetic and J48 algorithms. The accuracy exhibited by PSO proved that PSO was a suitable algorithm for performing classification tasks. A PSO based Classifier for solving sequence classification problems is presented in [22]. A two stage SPM based method was proposed that addresses the current problems. This method mainly focuses on identifying and excluding redundant or unreliable patterns and to determine sequence similarities. A similar method was proposed in [6, 1] that performs multi class classification using PSO. It uses multiclass databases to perform the Classification process.

A study on how PSO can improve the process of Classification by improving the results of well knows machine learning approaches is discussed in [3]. A binary PSO was used in [3] and it provides a set of logical rules to perform mapping to the available classes. Well known problems were analyzed and comparisons were carried out to show that PSO is an effective candidate for the problem of classification.

A hybrid method using Support Vector Machines that utilizes both filters and wrappers to solve Classification problems was presented in [23]. The filter model provides feature relevance while the wrapper model uses the modified discrete PSO algorithm. A multi swarm optimization technique (MSPSO) was proposed in [14], that was designed to solve discrete problems. This method uses multiple sub swarms to implement PSO. It also uses a multi swarm scheduler to control the swarms. An enhancement to this technique, the Improved Feature Selection was proposed by integrating MSPSO, SVM with F-score method.

A fuzzy classification system with dynamic parameter adaptation is proposed in [13, 17, 2]. This method uses the concept of fuzzy logic in determining the classified set. This method proposes to improve the convergence and diversity of swarm in PSO. A distributed hybrid PSO SVM system is proposed in [10]. It provides mechanism to improve the classification accuracy of the system with a small and appropriate feature subset. It combines the discrete and the continuous valued PSO to perform simultaneous optimization of the input feature in a distributed architecture.

Feature selection is the process of analyzing data and eliminating certain attributes that do not contribute to the results. Feature selection methods fall into two broad categories, the wrappers, that evaluate the features using a learning algorithm and eliminate them on the basis of the resultant accuracies, and the filters that evaluate the importance of features using heuristics based on the general characteristics of data [12, 13]. Though wrappers provide better results than filters, they are more expensive and are intractable for large databases with many features. Wrappers are totally dependent on the learning algorithm being used, hence they need to be re-run when switching between learning algorithms.

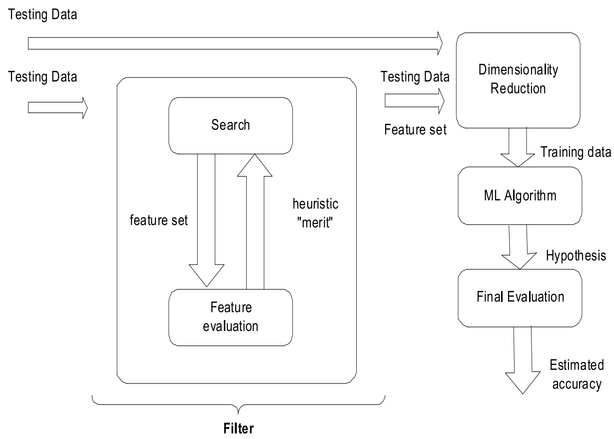

Figure-2. Filters.

Filters provide faster results and are learning algorithm independent, hence they are considered to be better then wrappers. Their downside is the lack of accuracy when compared to wrappers, but this outweighs their scaling nature with large databases. They function as effective subset selectors for wrapper methods in order to reduce the processing time of wrappers.

This paper presents a correlation based feature selection method that is embedded to PSO to create an optimized and faster classification algorithm.

The remainder of this paper is structured as follows; section II provides

2. ENHANCED CLASSIFICATION USING PSO WITH EMBEDDED ATTRIBUTE ELIMINATION TECHNIQUES

Incorporating attribute elimination techniques in the regular PSO enhances the conventional classification performed by PSO. The method of attribute elimination is embedded in PSO itself; hence it works side by side with the algorithm, which proves to be advantageous. The technique of embedded PSO is performed in two major phases. Figure presents an architecture overview of the PSO Classifier with embedded attribute filtering techniques.

The process begins with the evaluation of attributes using the CFS subset evaluator. The Greedy hill climbing method is used to filter attributes and the final pruned dataset is created. PSO is applied on the pruned dataset to produce efficient results.

VOL. 10, NO 20, NOVEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

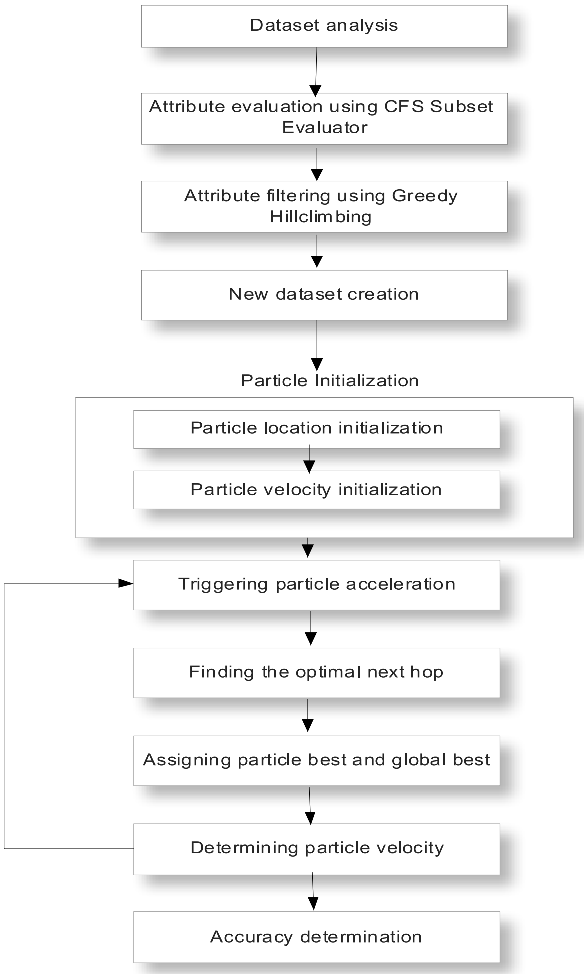

Figure-3. PSO with embedded feature selection: architecture.

A. Data preprocessing and pruning

1) CFS based feature selection

The CFS [15] based feature selection method evaluates the accuracy of the subset of attributes by considering the individual predictive ability of each of the feature and the degree of redundancy existing between them. Subsets containing attributes that have high correlation with the class attribute and low correlation between themselves are preferred. A good feature set is said to contain features containing most correlation with the class and least no correlation with each other.

A feature is said to be relevant iff there exists some vi and c for which p(Vi = vi) > 0 such that

$$p(C = c|V_i = v_i) \neq p(C = c) \tag{1}$$

CFS only measures the correlation between nominal features, so numeric features are first discretized and then the process is carried out. However, the generalized correlation-based feature selection does not depend on any particular data transformation; the correlation between any two variables is alone measured. Hence the technique can be applied to a variety of problems involving even numerical values. CFS is a completely automatic algorithm, which does not require any supervision in terms of threshold limits. It operates on the original feature space; hence it can be interpreted in terms of the original features. Hence the CFS filtering technique does not incur high computational cost, due to the repeated invoking of the learning algorithm.

If the correlation between the components are known, and the inter-correlation between is provided, then the correlation can be predicted by

$$r_{zc} = \frac{k\overline{r_{zi}}}{\sqrt{k+k(k-1)\overline{r_{ii}}}}, \tag{2}$$

Where r_zc is the correlation between the summed components and the outside variable, k is the number of components, (r_zi)¯ is the average of the correlations between the components and the outside variable, and (r_ii)¯ is the average inter-correlation between components [7, 8, 24].

The evaluator method used here is the Best first, which Searches the space of attribute subsets by greedy hillclimbing augmented with a backtracking facility.

Greedy HillClimbing algorithm

1. Let s ←start state.
2. Expand s by making each possible local change.
3. Evaluate each child t of s.
4. Let s 0 ←child t with highest evaluation e(t).
5. If e(s 0) ≥ e(s) then s ← s 0, goto 2.
6. Return s.

2) PSO based classification

Particle Swarm Optimization (PSO) is an evolutionary computation method used to optimize a problem by improving the candidate solutions in a continuous manner. Since classification is a discrete problem, the continuous solutions are discretized to find the final solutions. The process of optimization is carried out by components called particles. The movement of these particles in accordance to the fitness function determines the direction and velocity of movement of the particles. Termination is determined by defining the maximum time or the accuracy limit, depending on the applications requirement.

a) Particle initiation

The process of PSO begins by first initializing the number of particles. The optimal number of particles that can be set for best results is itself a non-trivial problem

ARPN Journal of Engineering and Applied Sciences

(refer Section 4 for details). These particles are distributed in a uniform manner in the search space. The dimensions of the particles are determined by the input data. For Example, if the input data comprises of 10 columns, the particles also contain 10 dimensions and movement is also triggered in all the 10 dimensions. Initial velocities are set to random values and the particle acceleration in triggered. Velocities are set to all the dimensions of the data; hence movement is triggered relative to all the dimensions.

b) Particle acceleration

After every displacement, the velocities of the particles are altered based on the global and the local best values determined by the fitness function. The velocity of the particles is calculated using the equation

$$V_{i,d} \leftarrow \omega V_{i,d} + \varphi_p r_p \left(P_{i,d} - X_{i,d}\right) + \varphi_p r_p \left(g_d - X_{i,d}\right) \quad (3)$$

Where rp and rg are the random numbers, Pi,d and gd are the parameter best and the global best values, xi,d is the value current particle position, and the parameters ω,φ_p,andφ_g are selected by the practitioner.

If the current known position of the particle is better than the particle best (pbest), then the pbest value for the particle is updated. Similarly, if the current pbest value is found to be greater than the global best (gbest), then the gbest value is updated to the current pbest.

This process is continued until the application reaches termination. The termination condition is determined by either the maximum time set or on reaching the maximum required accuracy limit.

Embedded PSO: Algorithm

1. *Attribute ranking and evaluation using CFS subset Evaluator.*
2. *Attribute filtering to generate pruned dataset.*
3. *Initialize particle location and velocity.*
4. *Triggering particle acceleration.*
5. *Particle location to the nearest available node.*
6. *Identify node fitness with respect to particle's initial location*
7. *Compare current fitness with pbest.*
8. *If pbest< current fitness*

a. *Assign current fitness to be the pbest*
b. *Compare pbest and gbest*
c. *If gbest < pbest*
i. *Assign pbest to be the latest gbest.*

9. *Generate new particle velocity using ()*
10. *If termination condition is not reached go to step4.*
11. *Perform classification to identify the accuracy.*

3. RESULTS AND DISCUSSIONS

The technique of embedded Particle Swarm Optimization (PSO) was implemented and tested using seven datasets from KEEL repository [9]. Details about the dataset are provided in Table-1.

Table-1. Dataset details.

Name	Attributes	Instances	Classes
Iris	4	150	4
Bupa	6	345	2
Heart	13	257	2
Sonar	60	208	2
Ionosphere	34	351	2
Libras-movement	90	360	15
Shuttle	9	58000	7

Figure-4 represents the attribute selection rates of the embedded PSO. It can be found from Figure-5 that the attribute reduction rate of the embedded hybrid PSO (HPSO) is high; hence the reduction rate of the data is also found to be high.

ARPN Journal of Engineering and Applied Sciences

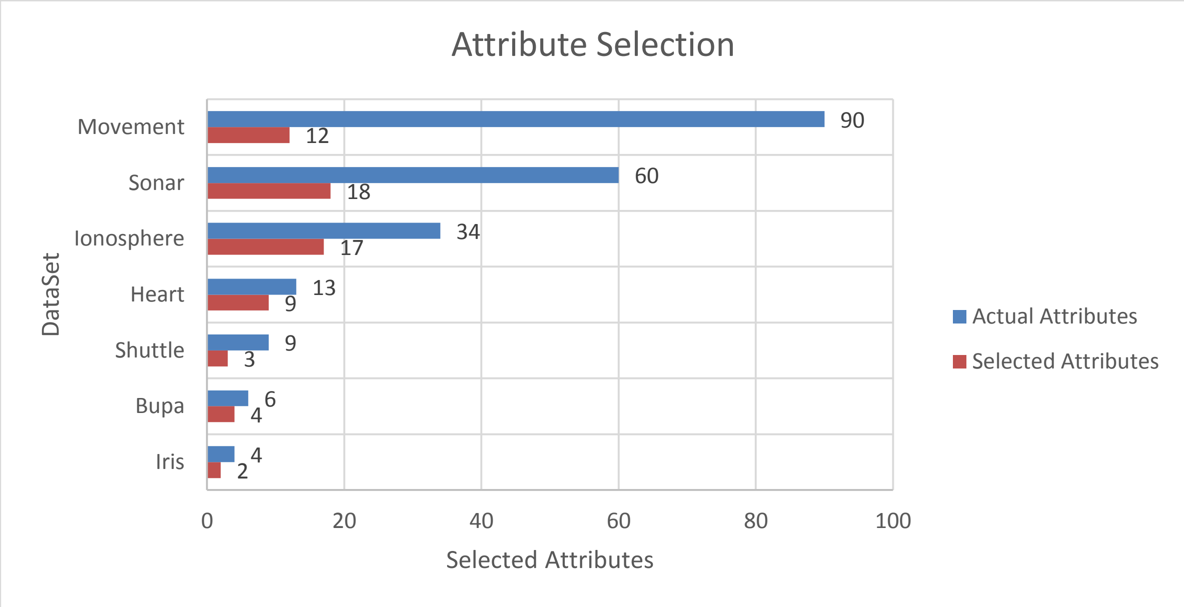

Figure-4. Attribute selection: Actual attributes vs. selected attributes.

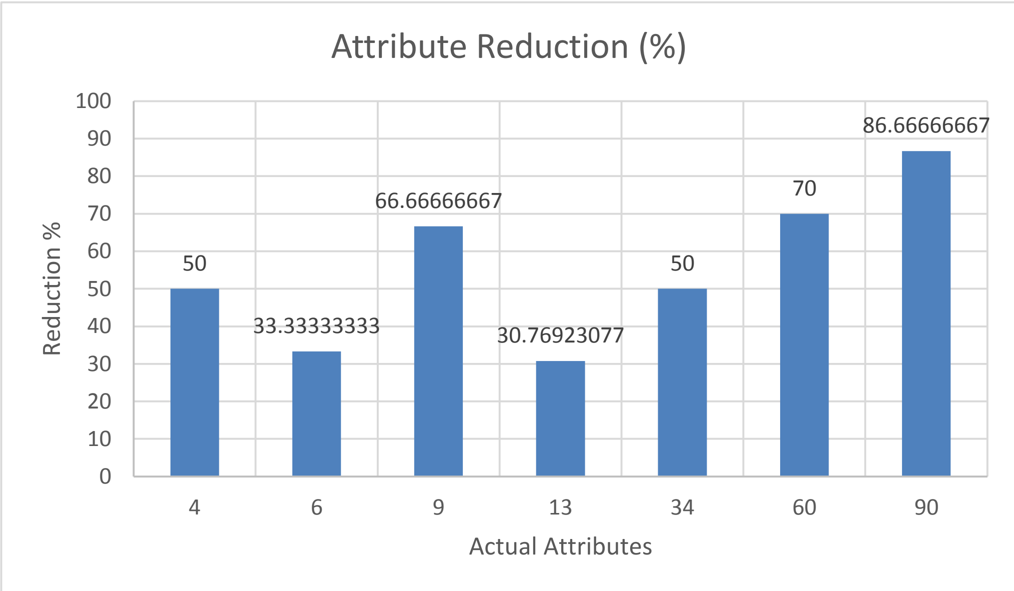

Figure-5. Attribute reduction %.

Figure-6 represents the accuracy rate of HPSO with respect to the number of particles used. The particle count was increased starting from 50% to the 100% of the dataset size. It can be observed that the accuracy of HPSO increases with respect to the particle count and reaches saturation when the particles are equal to the dataset size.

ARPN Journal of Engineering and Applied Sciences

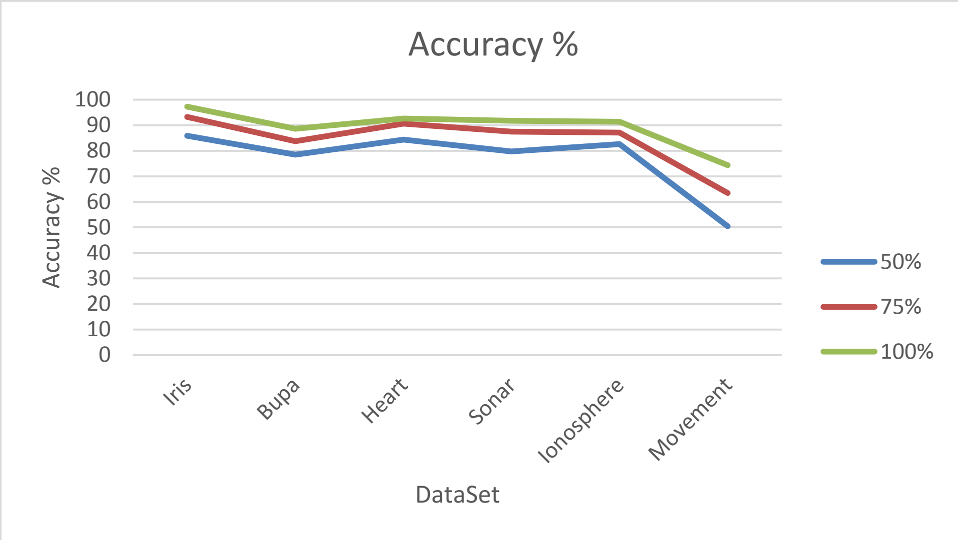

Figure-6. Accuracy obtained with varied particle sizes.

Figure-7 shows the time taken for HPSO to converge in the event of varied particle sizes. Though an increase in the time taken is observed, it is quite considerable when compared with the corresponding increase in the accuracy.

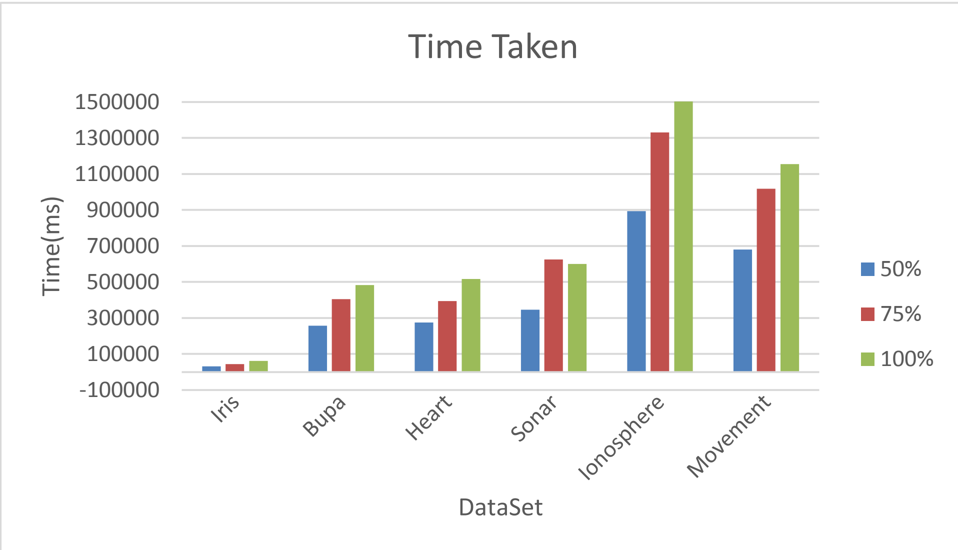

Figure-7. Time taken with varied particle sizes.

Figure-8 shows the accuracy and time comparison between PSO and HPSO on applying 10 particles in the search set. Though the increase in accuracy of HPSO is very low (~1% to 2%) when compared to PSO, time taken to compute the results show a huge difference, where it can be observed that HPSO takes very low time when compared to PSO. Similar observations can be made from Figure-10.

ARPN Journal of Engineering and Applied Sciences

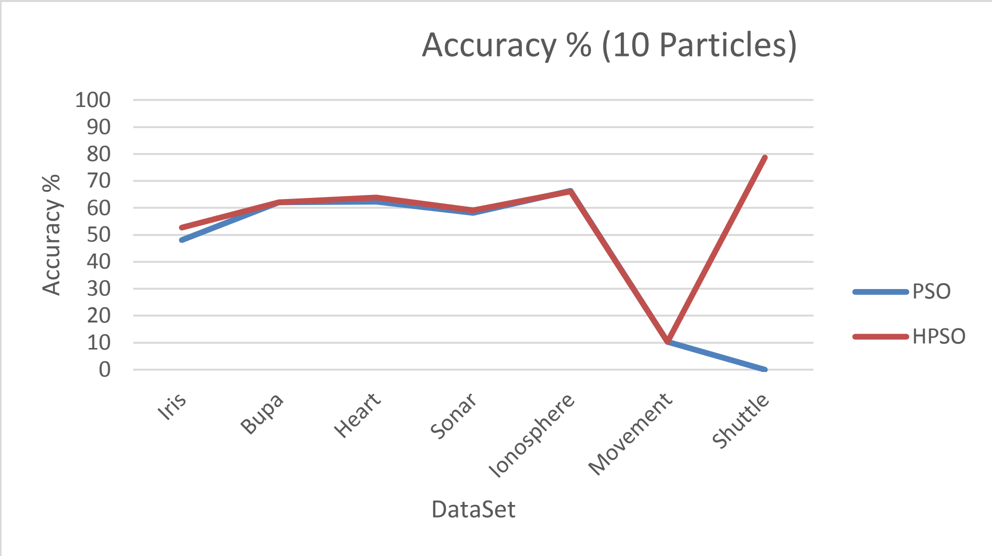

Figure-8. PSO Vs. HPSO: Accuracy comparison with 10 particles.

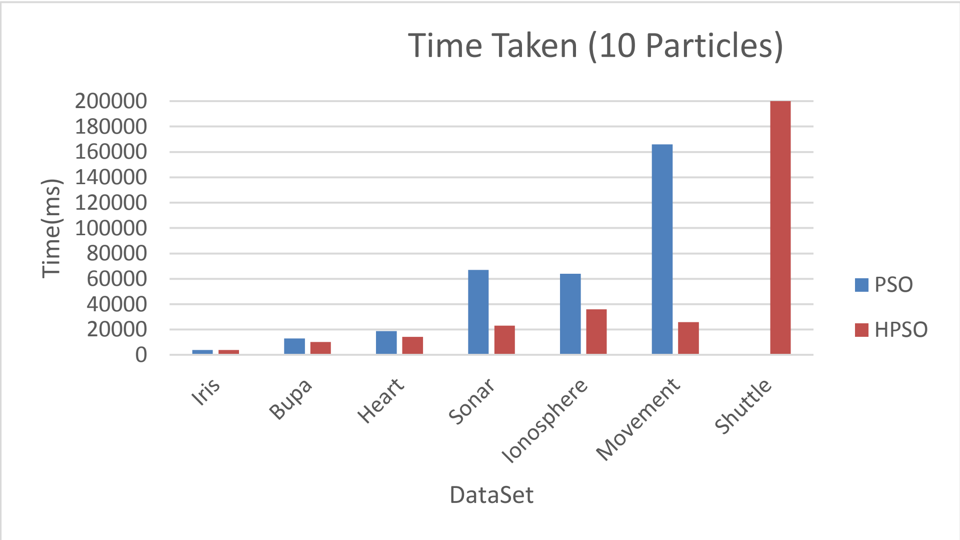

Figure-9. PSO Vs. HPSO: Time comparison with 10 particles.

ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

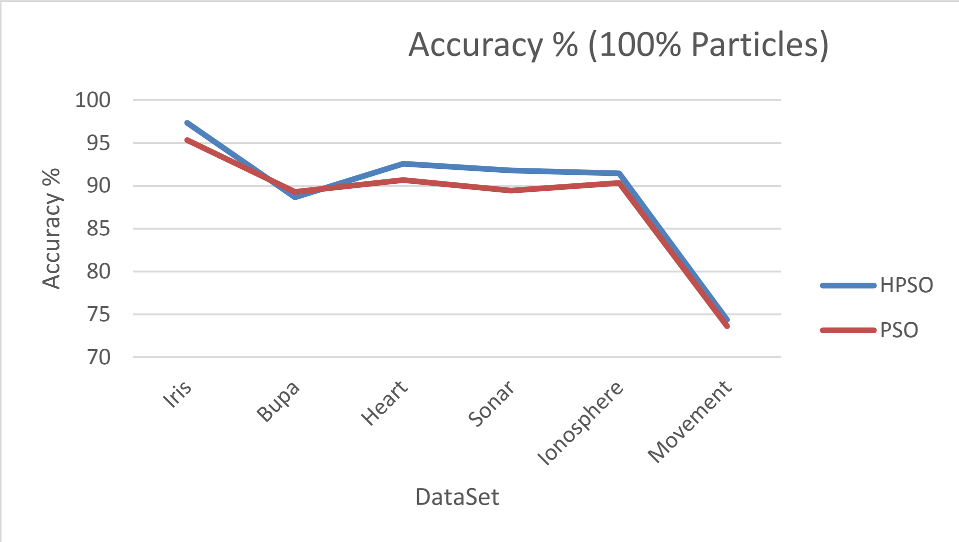

Figure-10. PSO Vs. HPSO: Accuracy comparison with 100% particles.

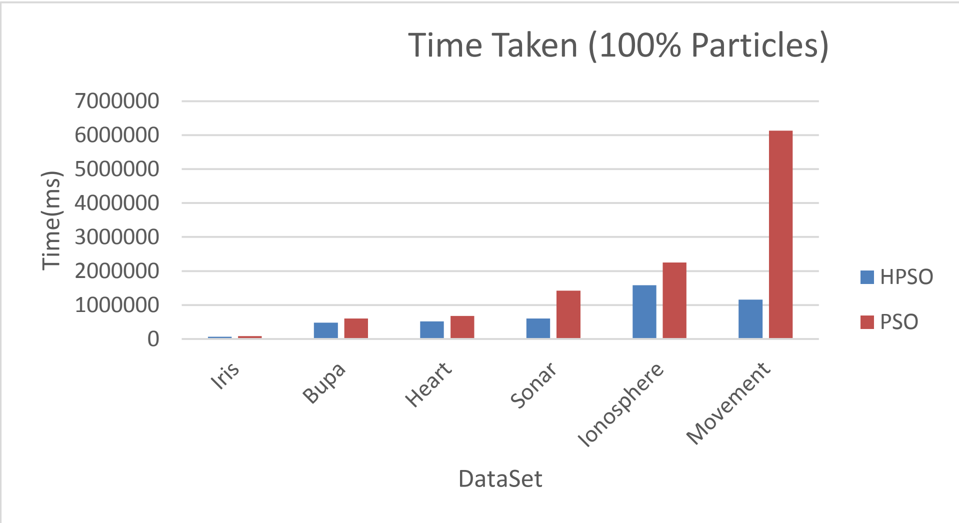

Figure-11. PSO Vs. HPSO: Time comparison with 100% particles.

4. CONCLUSIONS

Particle Swarm Optimization (PSO) is a metaheuristic optimization technique that provides enhanced results for a problem in a search space. This paper discusses the problems present in the current ‘big data’, and the characteristics or nature of such data. It discusses the presence of unrelated, redundant or unnecessary attributes present in such data and the increase in accuracy that can be observed by eliminating such attributes. A modified PSO algorithm (HPSO) has been proposed and it has been observed to provide a huge reduction in time and a considerable increase in the classification accuracy. In future, this approach can be extended or enhanced by incorporating machine learning techniques. Enhancements can also be observed by providing appropriate stopping criterion and effective discretization methods.

REFERENCES

[1] Bottou Léon, Bousquet Olivier. 2008. The tradeoffs of large scale learning. Advances in neural information processing systems.

[2] Bottou Léon. 2010. Large-scale machine learning with stochastic gradient descent: Proceedings of COMPSTAT'2010. Physica-Verlag HD. pp. 177-186.

ARPN Journal of Engineering and Applied Sciences

[3] Cervantes Alejandro, Inés M. Galván and Pedro Isasi. 2005. Binary particle swarm optimization in classification.

[4] Cortes Corinna and Vladimir Vapnik. 1995. Support-vector networks. Machine learning. 20.3, 273-297.

[5] Dalessandro Brian. 2013. Bring the noise: Embracing randomness is the key to scaling up machine learning algorithms. Big Data. 1.2: 110-112.

[6] De Falco, Ivanoe, Antonio Della Cioppa, and Ernesto Tarantino. 2007. Facing classification problems with particle swarm optimization. Applied Soft Computing. 7.3, 652-658.

[7] Ghiselli E.E. 1964. Theory of Psychological Measurement, McGraw Hill, New York.

[8] Hogarth R.M. 1977. Methods for aggregating opinions. In H. Jungermann and G. de Zeeuw, editors, Decision Making and Change in Human Affairs. D. Reidel Publishing, Dordrecht-Holland.

[9] http://sci2s.ugr.es/keel/category.php?cat=clas#sub2.

[10] Huang Cheng-Lung and Jian-Fan Dun. 2008. A distributed PSO–SVM hybrid system with feature selection and parameter optimization. Applied Soft Computing. 8.4, 1381-1391.

[11] Kennedy J., Eberhart R. 1995. Particle Swarm Optimization: Proceedings of IEEE International Conference on Neural Networks IV. pp. 1942-1948.doi:10.1109/ICNN.1995.488968.

[12] Kohavi R. 1995. Wrappers for Performance Enhancement and Oblivious Decision Graphs. PhD thesis, Stanford University.

[13] Kohavi R and John G. 1996. Wrappers for feature subset selection. Artificial Intelligence, special issue on relevance. 97(1-2): 273-324.

[14] Liu Yuanning, Gang Wang, Huiling Chen, Hao Dong, Xiaodong Zhu and Sujing Wang. 2011. An improved particle swarm optimization for feature selection. Journal of Bionic Engineering. 8.2. 191-200.

[15] Mark A. Hall. 1991. Correlation-based Feature Selection for Machine Learning. PhD thesis, University of Waikato.

[16] McCulloch, Warren S. and Walter Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics. 5.4, 115-133.

[17] Melin Patricia, Frumen Olivas, Oscar Castillo, Fevrier Valdez, Jose Soria and Mario Valdez. 2013. Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. Expert Systems with Applications. 40.8, 3196-3206.

[18] Mukhopadhyay A., Maulik U.; Bandyopadhyay S.; Coello Coello C.A. 2013. A Survey of Multiobjective Evolutionary Algorithms for Data Mining: Part I: Evolutionary Computation, IEEE Transactions on. 18(1).

[19] Mukhopadhyay Anirban, Ujjwal Maulik, Sanghamitra Bandyopadhyay and Carlos Artemio Coello Coello. 2014. A survey of multi-objective evolutionary algorithms for data mining: Part-II. 1(1).

[20] Shi Y., Eberhart R.C. 1998. A modified particle swarm optimizer. Proceedings of IEEE International Conference on Evolutionary Computation. pp. 69-73.

[21] Sousa Tiago, Arlindo Silva and Ana Neves. 2004. Particle swarm based data mining algorithms for classification tasks. Parallel Computing. 30.5, 767-783.

[22] Tsai Chieh-Yuan and Chih-Jung Chen. 2014. A PSO-AB Classifier for Solving Sequence Classification Problems. Applied Soft Computing.

[23] Unler Alper, Alper Murat and Ratna Babu Chinnam. 2011. Mr^2PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification. Information Sciences. 181.20, 4625-4641.

[24] Zajonic R.B. 1962. A note on group judgements and group size. Human Relations. 15, 177-180.