An Empirical Study on Classification Using Modified Teaching Learning Based Optimization

[1]Amaresh Sahu, [2]Sushanta Kumar Panigrahi, [3]Sabyasachi Pattnaik

[1] Computer Science Engineering Department, SOA University
Bhubaneswar, Odisha 751030, India

[2, 3] Information & Communication Technology Department, Fakir Mohan University
Balasore, Odisha 756019, India

Abstract

In this paper the modification to 'Teaching–Learning Based Optimization (TLBO) called Modified Teaching–Learning Based Optimization (MTLBO) based on particle swarm optimization principle has been proposed. Unlike TLBO, this population based method works on the effect of influence of a teacher on learners to find the optimum solution. The process of MTLBO is divided into two parts, the first part consists of the 'Teacher Phase' means learning from the teacher and the second part consists of the 'Learner Phase' means learner learns by interacting with other learner having better knowledge and from the best learner knowledge treated as team leader among all learners. The effectiveness of the method is tested on many benchmark problems with different characteristics and the results are compared with other population based methods and finally it is implemented on classification using neural network in data mining.

Keywords: *Artificial neural network, evolutionary algorithm, genetic algorithm, particle Swarm Optimization, teaching learning based optimization.*

1. Introduction

Data mining [1] is a field of many disciplines, there core is existing at the intersection of statistics, machine learning, and databases. In data mining several tasks are emphasized like classification, association rule mining, clustering, regression, summarization etc. Each of these tasks can be viewed as a type of problem to be solved by a data mining method. Therefore, the initial step in designing a data mining algorithm is to represent which task the algorithm will consider. In this paper the main focus is given on task of supervised classification. Classification is treated as one of the most frequently encountered decision making tasks of our activity and also it is a data mining technique used to predict group membership for data instances. A classification problem occurs when an object needs to be allotted into a predefined group or class based on a number of observed attributes related to that object.

Many problems in science, business, industry, and medicine can be treated as classification problems. Several major kinds of classification method including decision tree induction, Bayesian networks, k-nearest neighbor classifier, case-based reasoning, fuzzy logic and neural network classifier techniques etc, but they have their own pops and cons [7][8].

Artificial neural networks (ANNs) have been applied widely in many application problem domains including classification in recent years. Most of the applications use feed forward ANNs that use the standard back-propagation (BP) learning algorithm or some improved BP algorithms [4]. The BP algorithm is a gradient-based method, therefore some inherent problems are frequently encountered in the application of this algorithm, e.g., very slow convergence speed in training process, easily to get stuck in a local optima, etc. In addition to that the BP learning also needs to predetermine some important learning parameters such as learning rate, momentum and a predetermined structure.

Evolutionary algorithms (EAs) are promising non-gradient approaches for training ANNs for solving several application domains. EAs are heuristic and stochastic methods based on populations made up of individuals with a specified behavior similar to biological phenomenon, simultaneously they are also robust and efficient at exploring an entire solution space of optimization problems. Therefore, EAs are less likely to get stuck in local optima. These characteristics result in the development of evolutionary computation algorithms as an increasing important field in application. EAs have been successfully used by several researchers to evolve weights, structure, and learning parameters of ANNs in recent years. EAs are considered to be capable of reducing the ill effects of the BP algorithm, because they do not require gradient and differentiable information like BP algorithm. In general, several categories of evolutionary methodologies are genetic algorithm (GA), evolution strategies (ES), evolutionary programming (EP), particle swarm optimization (PSO), artificial Bee colony (ABC) and HS algorithm etc., where a change in the algorithm parameters changes the effectiveness of the algorithm. Among all the existing EAs, one of the most popular branch is GA. The main attractive feature of GA is that it supports for generic implementation of some major operators like crossover, mutation or selection operators. However, GA suffers from many drawbacks like competing conventions, premature convergence problem, lengthy local searches near a local optima and difficulty in determining the optimum controlling parameters such as population size, crossover rate and mutation rate [5].

Differential Evolution (DE) is similar to GA with specialized crossover and selection method. The particle swarm optimization (PSO), was proposed by Kennedy and Eberhart [6] as an alternative to GA in many aspects. The PSO was inspired by bird swarms and has been proven to be a good competitor and alternative to GA in many optimization problems. In comparison with GA, PSO has some attractive inherent characteristics. It retains the good knowledge studied by all particles in population; whereas GA destroys the previous knowledge of the problems once the population changes their state. PSO encourages constructive cooperation and information sharing between particles, which enhance the search ability of particles for getting a global optimal solution. PSO suffers from drawback of premature convergence. PSO also find difficulty in determining inertia weight, social and cognitive parameters.

Artificial Bee Colony (ABC) is another most recently defined algorithm by Dervis Karaboga in 2005, motivated by the intelligent behavior of honey bees. It is as simple as PSO algorithms, and it uses only common control parameters such as colony size and maximum cycle number. But, ABC requires optimum controlling parameters of number of bees limit. Therefore, the efforts have been given to develop an optimization technique which is free from the algorithm parameters, i.e. no algorithm parameters are required for the working of the algorithm. This parameter free aspect is considered in the TLBO, which is based on the effect of the influence of a teacher on the outcome of learners in a classroom. Here, output is results considered in terms grades. In our society, the teacher is generally considered as a highly learned or experienced person who shares his or her knowledge with the learners. The quality of a teacher determines the outcome of learners. It is obvious that a good teacher always try to train learners such that they can have better results in terms of their marks or grades. Also, learners also learn from interaction between themselves, which also helps in their outcomes [2] [3].

2. Particle Swarm Optimization

PSO is population based search technique for solving numerical optimization problems. Its mechanism is inspired from the complex social behavior shown by the natural species like flock of birds, school of fish and even crowd of human beings. Each member of swarm (called particle) has a velocity and a position. The particles fly through a multidimensional search space looking for a potential solution. Each particle adjusts its position in the search space from time to time according to the flying experience of its own and its neighbors [9]. For an n-dimensional search space, the movement of each particle is governed by the following two equations (1) and (2),

$$v_{ij}(t+1) = w.v_{ij}(t) + c_1.\text{rand1}.(pbest_{ij}(t) - x_{ij}(t)) + c_2.\text{rand2}.(gbest_j(t) - x_{ij}(t)) \quad (1)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (2)$$

where x_i is the position of the i^{th} particle; v_i represents the velocity of the i^{th} particle; $pbest_i$ is the previous best particle of the i^{th} particle; and $gbest$ is the global best particle found by all particles so far. $rand_1$ and $rand_2$ are vectors of random numbers, uniformly sampled within [0, 1], w is an inertia weight, c_1 and c_2 are two learning factors and t =1, 2,…, indicates the iterations.

3. Teaching Learning Based Optimization

Generally, the process of Teaching Learning Based Optimization (TLBO) is divided into two parts. The first part considered as the ‘Teacher Phase’ and the second part considered as the ‘Learner Phase’. The ‘Teacher Phase’ means learners learn from the teacher and the ‘Learner Phase’ means learners learn through the interaction between them.

3.1 Teacher Phase

In our society the best learner is treated as teacher, who has the better knowledge than other learners. Teacher tries to disseminate knowledge among students or learners to enhance their knowledge in the classroom, i.e. the mean of a class increases from MA to MB depending upon the ability of a good teacher. Good teacher ability is estimated by how much he can bring his or her learners up to his level in terms of knowledge. But, practically this is not possible and a teacher can only move the mean of a class up to some extent depending on the capability of the class. This follows a random procedure depending on many factors. Let Mi be the mean and Ti be the teacher at any iteration i. Ti always try to move mean Mi towards its own level, so now the new mean will be designated as Mnew. The solution is updated according to the difference value between the existing and the new mean and is given by the expression,

$$\text{Difference\_Mean}_i = r_i (M_{new} - T_F M_i) \tag{3}$$

where TF is a teaching factor that decides the value of mean to be changed, and ri is a random number in the range [0, 1]. The value of TF can be either 1 or 2, which is again a heuristic step and decided randomly with equal probability as,

$$T_F = \text{round}[1 + \text{rand}\,(0,1)\,\{2 - 1\}\,] \tag{4}$$

This difference modifies the existing solution to enhance the mean according to the following expression,

$$X_{new,\,i} = X_{old,\,i} + \text{Difference\_Mean}_i \tag{5}$$

3.2 Learner Phase

Learners enhance their knowledge by two different means, one through input knowledge from the teacher and the other through interaction of knowledge between themselves. A learner interacts randomly with other learners with the help of presentations, group discussions and formal communications, etc. A learner learns something new if the other learner has more knowledge than him or her. For population size Pn learner phase is expressed as,

```
For i = 1 : Pn
    Randomly select two learners Xi and Xj, where i <> j
    If f (Xi) < f (Xj)
    Xnew,i = Xold,i + ri(Xi − Xj)
    Else
    Xnew,i = Xold,i + ri(Xj − Xi)
    End If
End For
```

Accept Xnew if it gives a better function value.

4. Modified TLBO

In the modified TLBO (MTLBO) the teacher phase is similar to TLBO algorithm. In the learner phase the algorithm is modified. . A learner interacts randomly with other learners with the help of group discussions, presentations, formal communications, etc. A learner learns something new if the other learner has more knowledge than him or her and also he or she follows the best learner as team leader. This representation mimics the PSO activities where the particle update it's position by following it's previous best as well as global best position found by all particle. In TLBO, a learner learns something new if the other learner has more knowledge than him or her can be treated as learner's previous best position. In modified TLBO in addition to previous best the learner also learns from the best learner acting as team leader unlike PSO. Any learner in the Learner phase modification is expressed as,

```
For i = 1 : Pn
    Randomly select two learners Xi and Xj, where i <> j
    If f (Xi) < f (Xj)
    Xnew,i = Xold,i + ri (Xi − Xj) + ri (Xg − Xi)
    Else
    Xnew,i = Xold,i + ri (Xj − Xi) + ri (Xg − Xi)
    End If
End For
```

where Xg is the Knowledge of best learner acting as team leader. Accept Xnew if it gives a better function value.

5. Experimental Studies

5.1 Experiment 1

In the area of optimization different benchmark problems are considered having different characteristics such as separability, multimodality, and regularity. A function is multimodal if it has two or more local optima. A function is separable if it can be written as a sum of functions of variables separately. A function is regular if it is differentiable at each point of its domain. Non-separable functions are more difficult to optimize and difficulty increases if the function is multimodal. Complexity increases when the local optima are randomly distributed. Moreover, complexity increases with the increase in the dimensionality. To validate the proposed algorithm, its results are compared with the results of different algorithms for different benchmark problems available in the literatures. Details of benchmark functions are given in the Appendix.

In this experiment, eight different benchmark functions were taken as given by Akay and Karaboga [10] and Rao, Savsani, Vakharia [3].These functions are optimized and the results are compared with those given by PSO, DE, ABC, TLBO and MTLBO algorithms. Details of the properties of functions are given in Table 1.

Table 1: Details of benchmark function

Function	*Range*	*Min.*	*Multimodal ?*	*Separable ?*	*Regular?*
F1	[-100;100]	0	No	Yes	Yes
F2	[-100;100]	0	No	Yes	No
F3	[-500;500]	0	Yes	Yes	No
F4	[-100;100]	0	No	No	Yes
F5	[-5.12;5.12]	0	Yes	Yes	Yes
F6	[-600;600]	0	Yes	No	Yes
F7	[-32;32]	0	Yes	No	Yes
F8	[-50;50]	0	Yes	No	Yes

In this experiment the mean of results are taken as comparison criterion for different runs. This experiment is conducted for very high dimension of 500 for all the considered functions evaluation. The results obtained in this experiment are for 30 independent runs. Maximum number of function evaluations considered by Akay and Karaboga [10] was 100,000. For TLBO and MTLBO maximum number of function evaluations is set as 2000 (i.e. 1/50th of that given by Akay and Karaboga [10] with population size of 10) except for Rosenbrock, Schwefel and Penalized functions for which the maximum number of function evaluations is set as 100,000 (with population size of 10 for Schwefel and Penalized functions and 50 for Rosenbrock function). The results are given in Table 2. It can be observed from Table 2 that with only 2000 function evaluations, MTLBO has outperformed all the algorithms for Sphere, Rastrigin and Ackley functions. For Step and Griewank functions, TLBO and ABC have shown the same result but TLBO and MTLBO require only 1/50th of the function evaluations than those required by ABC. For the same number of function evaluations of 100,000, TLBO has shown better result for Rosenbrock function than all the algorithms. However, TLBO has shown inferior result for the Penalized function compared to that given by ABC, but still the result of TLBO is better than those given by PSO and DE algorithms. For Schwefel function, the results of TLBO and ABC are nearly the same. This experiment shows that TLBO is effective at very high dimensions for the functions having different characteristics like separability, multimodality, and regularity.

Table 2: Margin specifications

Function	*PSO*	*DE*	*ABC*	*TLBO*	*MTLBO*
F1	181.16	20.33	8.71E-07	2.02073E-52	0
F2	1621	1998.03	0	0	0
F3	-98168.1	-138152.03	-190906.66	-184297.381	-180206.28
F4	1.09E+06	8.72E+10	1007.87	497.91	497.61
F5	1033.04	594.69	87.96	0	0
F6	2.2	0.645	0	0	0
F7	3.69	13	0.058	0	0
F8	5.29	1.48E+10	3.46E-08	0.06292	1.2E-02

5.2 Experiment 2

The performance of the proposed model is evaluated using the four-benchmark databases taken from the UCI machine repository [11]. Out of these, the most frequently used in the area of neural networks and of neuro-fuzzy systems are IRIS, WINE, PIMA and BUPA Liver Disorders datasets. In addition, we have compared the results of FLANN with other competing classification methods using the aforesaid datasets

5.2.1 Description of the Dataset

Let us briefly discuss the datasets, which we have taken for our experimental setup.

IRIS: A classification data set based on characteristics of a plant species (length and thickness of its petal and sepal) divided into three distinct classes (Iris Setosa, Iris Versicolor and Iris Virginica).

WINE: Data set resulting from chemical analyses performed on three types of wine produced in Italy from grapevines cultivated by different owners in one specific region.

PIMA Indians diabetes: Data set related to the diagnosis of diabetes in an Indian population that lives near the city of Phoenix, Arizona.

BUPA liver disorders: Data set related to the diagnosis of liver disorders and created by BUPA Medical Research, Ltd.

Table 3 presents a summary of the main features of each database that has been used in this study.

Table 3: Details of database employed

Dataset	*Pattern*	*Attribute*	*Class*	*Patterns in*		
				Class1	*Class2*	*Class3*
IRIS	150	4	3	50	50	50
WINE	178	13	3	71	59	48
PIMA	768	8	2	500	268	-
BUPA	345	6	2	145	200	-

5.2.2 Classification Performance

We have adopted the 2-fold cross-validation strategy with the intension of getting good classification accuracy. The dataset is randomly partitioned into two sets of equal size that are in turn used for building and testing the proposed model. While one part is used for building the proposed model the other part is used for testing the model, in a way that each one is used for opposite purpose. The percentage of correct classification for each dataset using the TLBO and proposed MTLBO model is presented in Table 4 and Table 5 respectively.

Table 4: Classification accuracy using TLBO model

Datasets	*Hit Percentage in Training Sets*			*Hit Percentage in Test Sets*		
	Set 1	*Set 2*	*Average*	*Set 1*	*Set 2*	*Average*
IRIS	92.69	86.00	89.35	100.00	97.33	98.67
WINE	95.45	90.67	93.06	98.68	96.60	96.64
PIMA	76.04	76.56	76.30	80.72	80.46	80.59
BUPA	68.60	74.49	71.54	79.76	69.77	74.77

Table 5: Classification accuracy using MTLBO model

Datasets	*Hit Percentage in Training Sets*			*Hit Percentage in Test Sets*		
	Set 1	*Set 2*	*Average*	*Set 1*	*Set 2*	*Average*
IRIS	93.33	97.33	97.33	100.00	100.00	100.00
WINE	98.60	96.56	97.58	100.00	100.00	100.00
PIMA	75.52	75.00	74.35	83.60	82.81	84.11
BUPA	69.00	74.83	71.92	82.08	78.26	80.17

5.2.3 Result Comparison with Other Algorithms

The results obtained for the Iris Dataset, Wine Dataset, Pima Indians Dataset and Bupa Liver Disorders Dataset were compared with the results described in [13] [14] where the performance of several models is presented: NN (nearest neighbor), KNN (k- nearest neighbor, FSS (nearest neighbor with forward sequential selection of feature) and BSS (nearest neighbor with backward sequential selection of feature). MFS (multiple feature subsets), CART (CART decision tree), C4.5 (C4.5 decision tree), FID3.1 (FID3.1 decision tree), MLP (multilayer perceptron) and NEFCLASS. FLANN (functional link artificial neural), simulated annealing (SA) [12], Simple genetic algorithms (SGA) [14], Particle Swarm Optimization (PSONN) [15].TLBO and MTLBO.

Table 5: Comparison results on average Performance

Methods	*Dataset Average Performance (in %)*			
	Iris	*Wine*	*Pima*	*Bupa*
NN	---	95.2	65.1	60.4
KNN	---	96.7	69.7	61.3
FSS	---	92.8	73.6	56.8
BSS	---	94.8	67.7	60.0
MFS1	---	97.6	68.5	65.4
MFS2	---	97.9	72.5	64.4
C4.5	94.0	---	74.7	---
FID3.1	96.4	---	75.9	---
MLP	---	---	75.2	---
NEF Class	96.0	---	---	---
HNFB	98.67	98.31	77.08	74.49
HNFB1	98.67	99.44	78.26	73.33
Fixed HNFB	98.67	97.8	78.6	---
Adaptive HNFQ	98.67	98.88	77.08	75.07
FLANN	98.67	95.51	78.13	76.23
SGANN	92.93	94.15	74.84	---
PSONN	98.66	99.44	---	---

TLBONN	94.01	94.85	78.38	73.15
MTLBONN	98.67	98.79	79.56	76.05

The Table 6 depicts a comparative performance of series of classification techniques over a length of four diversified datasets i.e. Iris Dataset, Wine Dataset, Pima Indians Dataset and Bupa Liver Disorders Dataset. All classification techniques included here are highly prominent and widely accepted. The figure reveals the advancement of the proposed technique i.e. MTLBONN over the newly introduced technique TLBO. MTLBONN yields 98.67% level of accuracy for Iris Dataset, 98.79% for Wine Dataset, 79.56% for Pima Indian Dataset and 76.05% for Bupa Liver Disorder Dataset. MTLBO as empirically established proves to be a better classification method.

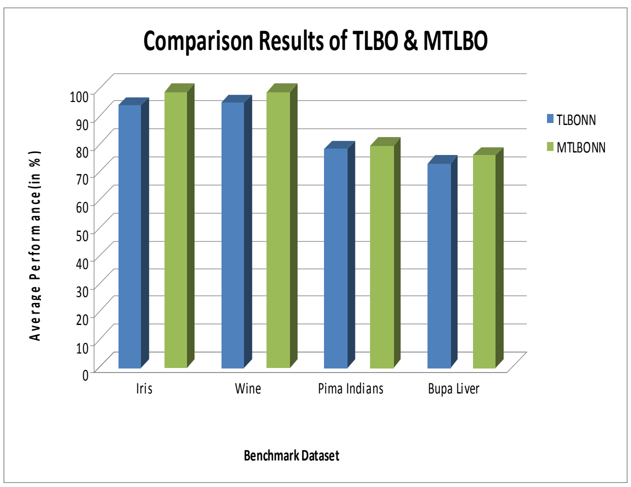

Fig. 1 Average performance chart among TLBO & MTLBO.

In all the methods used for classification the accuracy level varies due to the problems of improper setting of parameters, deficiency in the methods and improper selection of network structure. But in TLBO and the proposed method MTLBO there is no need of setting parameters and the results are found out by taking minimum network structure in hidden layers and in output layers. In case of Wine dataset, the result is slightly less than PSONN; it is because of improper setting of parameters, improper selection of network structure.

4. Conclusion

In this paper, the performance of the proposed MTLBO method has been given due emphasis with recent and well-known optimization algorithms such as PSO, DE, ABC, TLBO, etc. by experimenting with eight different benchmark functions with different characteristics like, multimodality, separability, regularity, and dimensionality. The effectiveness of MTLBO method is also evaluated using four benchmark databases for classification performance in terms of accuracy of hit percentage in training sets, hit percentage in test sets, average performance etc. The results show better performance of MTLBO method over almost all other natured-inspired optimization methods for the considered benchmark problems, except PSONN for Wine dataset, due to improper setting of parameters and selection of network structure. The MTLBO method shows better performance with less computational efforts for the large scale problems, i.e. problems with high dimensions. The results of TLBO and MTLBO can be enhanced by evolving the neural network properly, which will be the future scope to our research work.

Appendix

F1: Sphere

$$\sum_{i=1}^{n} x_i^2$$

F2: Step

$$\sum_{i=1}^{n} \left[x_i + 0.5 \right]^2$$

F3: Schwefel

$$-\sum_{i=1}^{n} x_i \sin\left(\sqrt{|x_i|}\right)$$

F4: Rosenbrock

$$\sum_{i=1}^{n-1} \left(100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right)$$

F5: Rastrigin

$$\sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi 0_i) + 10)$$

F6: Griewank

$$\frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

F7: Ackley

$$\sum_{i=1}^{n} - 20\exp\left(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{30}\sum_{i=1}^{n}\cos 2\pi o_i\right)$$

F8: Penalized

$$\frac{\pi}{n}\left[10\sin^2(\pi\pi_1) + \sum_{i=1}^{n-1}(y_i-1)^2\left\{1+10\sin^2(\pi\pi_{i+1})\right\} + (y_n-1)^2\right] + \sum_{i=1}^{n} u(x_i,10,100,4)$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i-a)^m, & x_i > a, \\ 0, & -a \le x_i \le a, \\ k(-x_i-a)^m, & x_i < -a, \end{cases}$$

$$y_i = 1 + \frac{1}{4}(x_i + 1)$$

References

[1] Ghosh A., Dehuri S., Ghosh S., "Multi-objective Evolutionary Algorithms for Knowledge Discovery from Databases", Springer-Verlag, Heidelberg, Germany, 2008.

[2] Rao R.V., Savsani V.J., Vakharia D.P., "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems", Computer-Aided Design, 43 (2011), 303–315.

[3] Rao R.V., Savsani V.J., Vakharia D.P., "Teaching–Learning-Based Optimization: An optimization method for continuous non-linear large scale problems", Information Sciences, 183 (2012) 1–15.

[4] Hush D.R., Horne N.G., "Progress in supervised neural networks", IEEE Signal Process Mag (1993), 10:8–39.

[5] Jianbo Yu, Lifeng Xi, Shijin Wang, "An Improved Particle Swarm Optimization for Evolving Feed forward Artificial Neural Networks", Neural Process Litt (2007), 26:217–231.

[6] Kennedy J, Eberhart R.C., "Particle swarm optimization", IEEE Int Conf. on Neural Networks, Piscataway (1995), pp 1942–1948.

[7] Thair Nu Phyu, "Survey of Classification Techniques in Data Mining", Proceedings of the International Multi Conference of Engineers and Computer Scientists, Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong.

[8] Guoqiang Peter Zhang, "Neural Networks for Classification: A Survey", IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews, VOL. 30, NO. 4, November 2000.

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

[10] Akay B., Karaboga D., "Artificial bee colony algorithm for large-scale problems and engineering design optimization", Journal of Intelligent Manufacturing (2010), doi:10.1007/s10845-010-0393-4.

[11] Blake C.L., Merz, C.J.,UCI Repository of Machine Learning Databases. www.ics.uci.edu/~mlearn/MLRepository.

[12] Shih Wei Lin, Zne Jung Lee, Shih Chieh Chen, Tsung Yuan Tseng, "Parameter determination of support vector machine and feature selection using simulated annealing approach", Applied Soft Computing 8 (2008) 1505–1512.(SA-SVM).

[13] Mishra B.B., Dehuri S., "Functional Link Artificial Neural Network for Classification Task in Data Mining", Journal of Computer Science 3 (12): 948-955, 2007.

[14] Erick Cant´u-Paz, "Pruning Neural Networks with Distribution Estimation Algorithms", GECCO 2003, LNCS 2723, pp. 790–800, 2003.c_Springer-Verlag Berlin Heidelberg 2003.

[15] Massod Zamani, Alireza Sadeghian, "A Variation of Particle Swarm Optimization for Training of Artificial Neural Networks", Computational Intelligence and Modern Heuristics.

Mr.Amaresh Sahu received a M. Tech. (CS) from Utkal University, Bhubaneswar 2005.Currently he is HOD in MCA at Ajay Binay Institute of Technology, Cuttack, India. He is performing research in Cluster Analysis, Neural Network and Soft Computing in Computer Science at SOA University, Bhubaneswar. He is published 16 research paper in national & international journals and conferences.

Mr.Sushanta Kumar Panigrahi received MCA from IGNOU in 2002.Currently he is Head in IT at Interscience Institute of Management & Technology, Bhubaneswar, India. He is performing research in Cluster Analysis, Neural Network, Optimization Technique and Soft Computing in ICT at Fakir Mohan University, Balasore. He is published 14 research paper in national & international journals and conferences.

Dr Sabyasachi Pattnaik has done his B.E in Computer Science, M Tech. from IIT Delhi. He has received his PhD degree in Computer Science in the year 2003, now working as Professor and Head in the Department of Information and Communication Technology, in Fakir Mohan University, Vyasavihar, Balasore, Odisha, India. He has got 15 years of teaching and research experience in the field of neural networks, soft computing techniques. He has got 52 publications in national & international journals and conferences. He has published three books in office automation, object oriented programming using C++ and artificial intelligence. At present he is involved in guiding 6 scholars in the field of neural networks in cluster analysis, bio-informatics, computer vision & stock market applications. He has received the best paper award & gold medal from Orissa Engineering Congress in 1992 and institution of Engineers in 2009.