International Journal of Engineering Research and Development
e-ISSN: 2278-067X, **p**-ISSN: 2278-800X, www.ijerd.com
Volume 6, Issue 4 (March 2013), PP. 114-121

A Hybrid Classifier based Rule Extraction System

Tulips Angel Thankachan[1], Dr. Kumudha Raimond[2]
[1]M.Tech Student, Department of Computer Science & Engineering, Karunya University, Coimbatore.
[2]Professor, Department of Computer Science & Engineering, Karunya University, Coimbatore.

Abstract:- Classification is one of the common tasks performed in knowledge discovery and machine learning. Different learning algorithms can be applied to induce various forms of classification knowledge. This type of learning process results in creating classification system called classifier. Measure used to evaluate such systems is classification accuracy. Classification accuracy of individual classifiers can be improved through combining classifiers. This paper proposes a hybrid classifier model called DTABC (Decision Tree and Artificial Bee Colony Algorithm) to improve the classification accuracy irrespective to size, dimensionality, class distribution and domain of the dataset. This proposed system comprises of two phases: in the first phase, rules are extracted from the training dataset using C4.5 decision tree algorithm. In the second phase, ABC algorithm is applied over C4.5 produced rules to produce more optimized rules. The proposed system has been compared with C4.5, DTGA and Naïve Bayes. The results show that the proposed hybrid classifier provides better classification accuracy**.**

Keywords:- artificial bee colony algorithm, classification, c4.5, hybrid classifier, fitness function, accuracy.

I. INTRODUCTION

Nowadays, the volume of digital data is increasing tremendously, so the amount of raw data extracted is also increasing [1]. Data mining (DM) is one of the methods to customize and manipulate the data according to our need. The knowledge extracted by DM should be readable, accurate and ease to understand. Its process can also be called as knowledge discovery. It can be applicable in areas such as parallel computing, artificial intelligence and database. Various types of algorithms such as Evolutionary Algorithm (EA) [2], Classification algorithm, Clustering algorithm can be applied in DM. Genetic algorithm (GA) [3], Artificial bee colony (ABC) algorithm [4], Ant colony optimization algorithm (ACO) [5], Particle swam optimizations (PSO) [6] are some of the EA algorithms which are being used in DM.

Classification includes assigning a decision class label to a set of unclassified objects described by a fixed set of attributes. Different learning algorithms can be applied to bring on various forms of classification knowledge from provided learning examples. This knowledge can be successively used to classify new objects. In this sense learning process results in creating classification system called classifier. Classifier is to categorize datasets into various classes based on the features of the dataset. A typical measure used to evaluate such systems is classification accuracy. Classification algorithms have been implemented over various learning tasks in different domains such as financial classification, manufacturing control chemical process control, and robot control.

Many methods are applied in DM for classification and rule extraction processes[16]. One approach in DM for classification is GA. From a set of possible solutions GA generates a best new solution. It replaces the worst solution with best solutions. The main application done with GA are oil collecting routing problem [7]. Another method used in DM is PSO [1]. It simulates the coordinated motion in flocks of birds. The main benefit of using PSO is that it reduces the complexity and speeds up the DM process [8]. The main application done with PSO is mining of breast cancer pattern [6]. Another method used in DM is ACO. It explains the social behavior of ant colonies. An example of ACO application is prediction of protein activity [5]. The major approach involves combining GA and ACO in prediction postsynaptic activity of proteins. The hybrid algorithm takes the advantages of both ACO and GA methods by complementing each other for feature selection in protein function prediction. Another hybrid PSO/ACO algorithm also has been proposed for discovering classification rules. Another hybrid model in DM for classification and rule extraction is Decision Tree and Genetic algorithm (DTGA) [12]. In this hybrid model the primary rules are generated by C4.5 and those rules are given as the input to GA. GA replaces the worst rules with the best one and produce more optimized set of rules. It provides better classification accuracy than individual classifiers. GA is a primeval algorithm and more efficient algorithms are available to overcome the limitations of GA. As ABC is a recent optimization algorithm,

it is used in this work along with C4.5 algorithm to improve the classification accuracy of the dataset irrespective of domain, size and dimensionality.

The rest of this paper is organized as follows: Section 2 covers the proposed system and the working of the optimization algorithm. Section 3 explains about the other classifiers used for comparing the performance of the proposed work. Experimental design and the results of this work are shown in section 4. Conclusions are given in Section 5.

II. DTABC

From the survey on different machine learning systems, some of the classification algorithm will solve only some specific problems. In the hybrid model of DTGA [12], GA is a primeval optimization algorithm. So that it is not providing much better classification accuracy and optimized rules. In this study a new hybrid model is proposed with a recent optimization algorithm for providing more optimized rules and better classification accuracy, irrespective of size, domain and dimensionality of the input dataset. DTABC mainly consists of three phases as shown in figure 1. In the first phase, the UCI (University of California at Irvine) repository datasets are given as input to C4.5 for classifying the dataset based on the classes. In the second phase, the IF-THEN form rules are extracted. In the third phase, IF-THEN form rules are eliminated and ABC algorithm is applied over the rules to produce more optimized set of rules which in turn increased the classification accuracy of the proposed system.

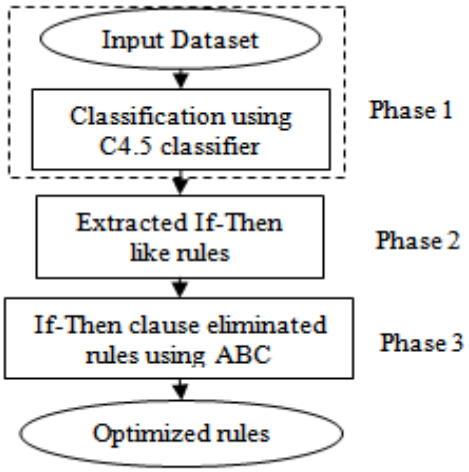

Figure 1 Discovering optimized rules by DTABC

A. C4.5: A Decision Tree Classifier

C4.5 is a statistical classifier used to generate decision tree [12]. It is developed by Ross Quinlan. It overcomes the limitations of Quinlan's ID3 algorithm. The C4.5 classifier is mainly used for classification. It is a directed tree which consists of nodes. It is having a node called a "root" along with various leaf nodes. The root node is not having any incoming node but it is having one or more outgoing nodes. All the leaf nodes have exactly one incoming node but it may or may not have outgoing nodes. Information gain indicates how much informative each node is. Based on the information gain of the nodes, the positions of nodes are decided. Therefore the instances are classified by navigating them from the most informative node down to a leaf. Pruning strategies are applied over the tree to stop the growth of the tree.

B. ABC Algorithm

ABC is one of the most recent optimization algorithm introduced by Dervis Karaboga in 2005 [4]. It is swam intelligent based algorithm. It explains the intelligent behaviour of honey bees. It provides a population-based search procedure. The main aim of bees is to discover the places of food sources which is having high nectar amount. The employed and onlooker bees fly around the multidimensional search space for finding the best food sources depending on their experiences and their neighbour bees. And the scout bees fly around and find the food positions randomly without any experience. If the new source is having the higher nectar amount, it replaces the previous source with the new one in the memory. Thus the algorithm combines the local and global search methods for balancing the exploitation and exploration processes. ABC algorithm is a simple concept and easy to implement. It is applied over various applications such as digital IIR filters [10], protein tertiary structures [9], artificial neural networks [11] etc.

The foraging selection of bees leads to the appearance of collective intelligence of honey bee swarms. It consists of three essential components: unemployed foragers, employed foragers and food sources.

1) *Unemployed foragers:* scouts and onlookers are the two types of unemployed foragers. The main aim of them are exploring and exploiting food source. At the initial stage of ABC algorithm the unemployed foragers are having two choices: either it becomes a scout or an onlooker.
2) *Employed foragers:* the employed foragers used to store the food source information and using it according to some calculated probability. When the food source has been exhausted the employed foragers will act as scout bees.
3) *Food sources:* The food sources represented the position of the solutions. It has been fixed by calculating the fitness function.

The colony of artificial bees in this algorithm divided into three phases: employed bees, onlooker bees and scout bees. The number of onlooker and employed bees are same and the sum of these two is the colony size. The overall flowchart of the ABC algorithm is shown in figure 2[16]. Initially the algorithm is generating an initial population by randomly selecting the food source positions. Next the initial nectar amount is calculated using equation 2 of newly generated population. Then the population is subjected to repeat cycles of the search courses of the three phases. Using the search equation as shown in equation 1 the employed bee is producing modifications on the population and calculating the nectar amount. If the new nectar amount is better than the previous nectar amount, the employed bees will memorize the new positions; otherwise it keeps the previous nectar amount. After completing the search process of all the employed bees, they share the information such as nectar amount, new food positions of food source with the onlooker bees.

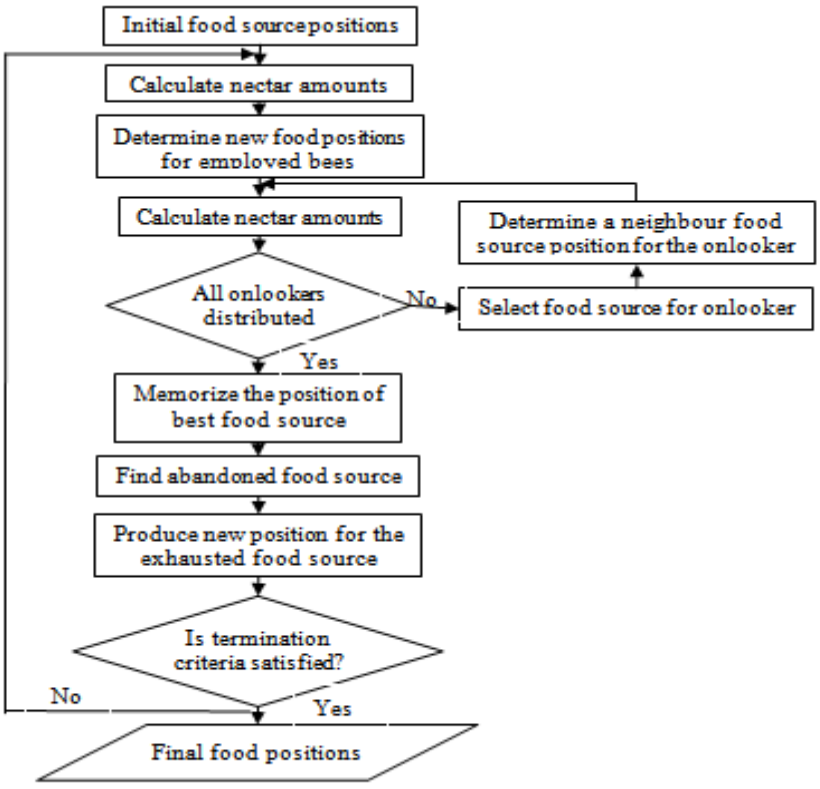

Figure 2 Flowchart of ABC algorithm

The onlooker bees evaluate the nectar amount information of the employed bees and they start the search process. The onlooker bees finding the food positions by depends on the probability equation 3. If the new nectar amount is better than the previous one they will memorize the new one and ignoring the previous one. The search process will complete when it satisfies the termination criteria.

C. Search Equation

In ABC algorithm, the bees are doing local search for finding the new possible food positions.

$$v_{ij} = x_{ij} + \phi_{ij}\left(x_{ij} - x_{kj}\right) \quad (1)$$

Equation 1 shows the local search strategy of the ABC algorithm. *Vij* is the new food position, *k* and *j* are the randomly chosen parameters and both are different, where $k \in [1,2,\ldots,SN]$ and $j \in [1,2,\ldots,D]$. *Xij* and *Xkj* are the old food source positions. The difference between these two positions is the distance from one food source to the other one. *D* is the number of optimization parameters and *SN* is the number of employed bees. Φij is a random number between [-1, 1].

D. Nectar Amount

Nectar amount is also called as fitness function. It is an objective function. The fitness formula used in ABC algorithm is shown in equation 2:

$$f(r_i) = \frac{n-m}{n+m} + \frac{n}{m+k} \quad (2)$$

Where, r_i is the i^{th} rule, '*n*' is the number of training examples satisfying all the conditions in the antecedent (A) as well as the consequent (C) of the rule (r_i). '*m*' is the number of training examples which satisfy all the conditions in the antecedent (A) part but not the consequent (C) of the rule (r_i). '*k*' is the predefined positive constant value. Here k=4.

E. Probability Equation

It is one of important functions that supports the algorithm[16]. It is shown in equation 3:

$$P_i = \frac{fit_i}{\sum_{n-1}^{SN} fit_n} \quad (3)$$

Where, P_i is the probability value associated with i^{th} food source. fit_i represents i^{th} food source's nectar amounts. *SN* is the number of food source which is equal to the number of employed bees.

F. Classification Accuracy

The overall classification accuracy explains the overall performance of the system. It is calculated using equation 4:

$$F = \frac{Number\ \ of\ training\ \ examples\ \ correctly\ \ classified}{Total\ \ number\ \ of\ training\ \ examples} * 100 \quad (4)$$

III. OTHER CLASSIFIERS

To compare the classification accuracy of the proposed system DTABC, three other classifiers such as C4.5, DTGA (Decision Tree and Genetic algorithm) [12] and Naïve Bayes are used.

A. DTGA

DTGA is a hybrid classifier used to improve the classification accuracy. It mainly consists of three phases [12]. In the first phase the UCI dataset is given as the input for C4.5 to generate the rules of 'IF-Then' form. In its next phase discretized rules are produced by eliminating the IF-Then clause. In the last phase, the GA is applied over the discretized rules to generate more optimized set of rules. The overall classification accuracy of the system is calculated using equation 4.

B. Naive Bayes

Naive Bayesian classifier is a statistical classifier. This classifier is based on the Bayes' Theorem and the maximum posteriori hypothesis. It is having strong independence (naïve) assumptions.

Bayes's rule : $$P(H|E) = \frac{P(E|H) \times P(H)}{P(E)} \quad (5)$$

The necessary scheme of Bayes's rule is that the product of a hypothesis or an event (H) can be predicted based on some evidences (E) that can be observed. From Bayes's rule, we have

(1) A priori probability of H or P(H): This is the probability of an event before the evidence is observed.
(2) A posterior probability of H or P(H | E): This is the probability of an event after the evidence is observed.

IV. EXPERIMENTAL DESIGN AND RESULTS

A. Experimental Study

This subsection describes the details of implementation of the proposed work. Totally four algorithms such as C4.5, DTGA, Naive Bayes and DTABC are used for the comparison purpose. C4.5 and Naive Bayes are implemented using WEKA (Waikato Environment for Knowledge Analysis, Version 3.4.2) DM tool. DTGA and DTABC are hybrid model classifiers. DTGA is composed with C4.5 and GA and DTABC is composed with C4.5 and ABC. GA and ABC are implemented in java-1.4.1 on Intel core i3 running on windows 7. All experiments are performed on the same machine. The algorithms are tested over 5 benchmark datasets from UCI.

Table 1 gives the features of the datasets used. All the algorithms are evaluated using these datasets irrespective of domain, size, dimensionality and class distribution.

Table1: Summary of UCI datasets

Dataset	Number of Non-Target Attributes	Number of Classes	Number of Examples
Contact- Lenses	4	3	24
Diabetes	8	2	768
Glass	9	7	214
Heart-Statlog	13	2	270
Weather	4	2	14

B. Experimental Analysis

1) Description of DataSets

As an example, weather dataset has been taken as input to DTABC. It is having 14 instances. It contains 4 non target attributes and 1 target attribute. Weather dataset is shown in table 4. There are two types of attributes in the dataset such as target and non target attributes. The attributes and its values are shown in table 2 and table 3.

Table 2 Non-Target attributes with value

Attribute	Attribute values		
Outlook	Sunny	Overcast	Rain
Humidity	Continuous		
Temperature	Continuous		
Windy	True	False	

Table 3 Target attribute with value

Attribute	Attribute values	
Playing- Decision	Yes	No

Table 4 Original weather dataset

Day	Outlook	Humidity	Temperature	Windy	Playing-Decision
1	Sunny	85	46	False	No
2	Sunny	88	45	True	No
3	Overcast	82	42	False	Yes
4	Rain	94	25	False	Yes
5	Rain	70	20	False	Yes
6	Rain	65	15	True	No
7	Overcast	66	14	True	Yes
8	Sunny	85	28	False	No
9	Sunny	70	15	False	Yes
10	Rain	72	36	False	Yes
11	Sunny	65	25	True	Yes
12	Overcast	90	22	True	Yes
13	Overcast	75	41	False	Yes

14	Rain	80	21	True	No

2) Phase 1

The attributes are discretized to reduce the range. The basic conditions used for discretizing the dataset are given in table 5. The discretized dataset corresponding to the original dataset is shown in table 6.

Table 5 Conditions used for discretizing the dataset

Attributes	**Basic conditions to discretize**		
Playing-decision	Play(1)	Yes	Don't(0): no
Outlook	Sunny(1)	Overcast(2)	Rain(3)
Humidity	High(1): >=75	Normal(2)	<75
Temperature	Hot(1): >36	Mild(2): (20,36)	Cool(3): <=20
Windy	True(1)	False(2)	

Table 6 Discretized form of weather dataset

Day	**Outlook**	**Humidity**	**Temperature**	**Windy**	**Playing-Decision**
1	1	1	1	2	0
2	1	1	1	1	0
3	2	1	1	2	1
4	3	1	2	2	1
5	3	2	3	2	1
6	3	2	3	1	0
7	2	2	3	1	1
8	1	1	2	2	0
9	1	2	3	2	1
10	3	2	2	2	1
11	1	2	2	1	1
12	2	1	2	1	1
13	2	2	1	2	1
14	3	1	2	1	0

3) Phase 2

From the discretized dataset, five primary rules are generated by C4.5. It is in IF- THEN structure. The following are the rules generated by C4.5.

Rule 1: IF (Outlook=sunny) AND (Humidity <=75) THEN (Playing-decision=Yes).
Rule 2: IF (Outlook=sunny) AND (Humidity >75) THEN (Playing-decision=No).
Rule 3: IF (Outlook=overcast) THEN (Playing-decision=Yes).
Rule 4: IF (Outlook=rainy) AND (Windy=True) THEN (Playing-decision=No).
Rule 5: IF (Outlook=rainy) AND (Windy=False) THEN (Playing-decision=Yes).

As C4.5 is a decision tree. The decision tree generated after the classification is shown in figure 3.

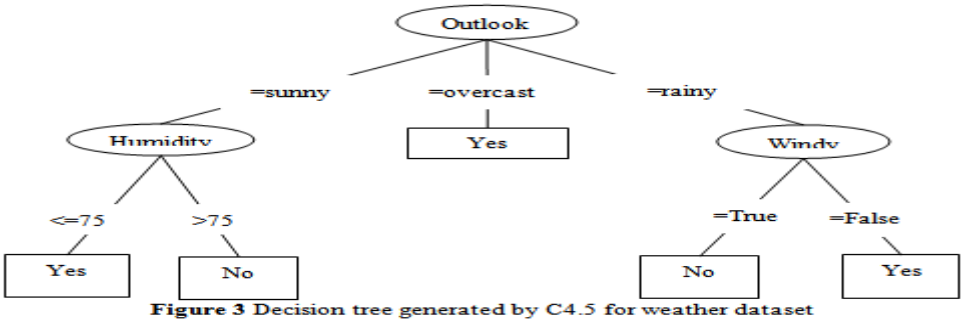

Figure 3 Decision tree generated by C4.5 for weather dataset

Next step is to eliminate the IF-THEN clause of C4.5 generated rules. i.e., discretizing the C4.5 generated rules (table 7) to apply ABC algorithm for getting more optimized set of rules.

Table 7 Discretized form of C4.5 generated rules

	Outlook	Humidity	Temperature	Windy	Playing-decision
Rule 1	1	2	*	*	1
Rule 2	1	1	*	*	0
Rule 3	2	*	*	*	1
Rule 4	3	*	*	1	0
Rule 5	3	*	*	2	1

('*' denotes the don't care condition. i.e., those attributes has no importance on that rule)

4) Phase 3

Next step is to apply ABC algorithm over the discretized ruleset. As ABC is an optimization algorithm, it generates more optimized set of rules which are shown in Table 8.

Table 8 Rule set generated by ABC algorithm

	Outlook	Humidity	Temperature	Windy	Playing-decision
Rule1	3	*	*	*	1
Rule 2	*	2	*	*	1
Rule 3	3	*	*	2	1
Rule 4	1	1	*	*	0
Rule 5	3	*	*	1	0

C. Comparison

For analyzing the performance of the proposed hybrid system DTABC, the accuracy of the system is compared with other three systems such as C4.5, Naïve Bayes and DTGA. From the four classifiers, DTABC and DTGA are hybrid classifiers and C4.5 and Naïve Bayes are single classifiers. The experiments are done over 5 datasets from different domain and size. The accuracies of each dataset from each classifier are shown in table 9.

Table 9: Comparative performances of C4.5, Naïve Bayes, DTGA and DTABC on the UCI datasets

	Classification accuracy in %			
Dataset	C4.5	Naïve Bayes	DTGA	DTABC
Contact-lenses	83.3333	70.8333	86.6667	93.33
Diabetes	73.8281	76.3021	76.7813	83.57
Glass	66.8224	48.5981	69.4953	74.84
Heart-statlog	76.6667	64.2587	79.7333	85.87
Weather	64.2857	59.4286	66.8571	72

The graph which represents the comparative performance of the classifiers is shown in figure 4.

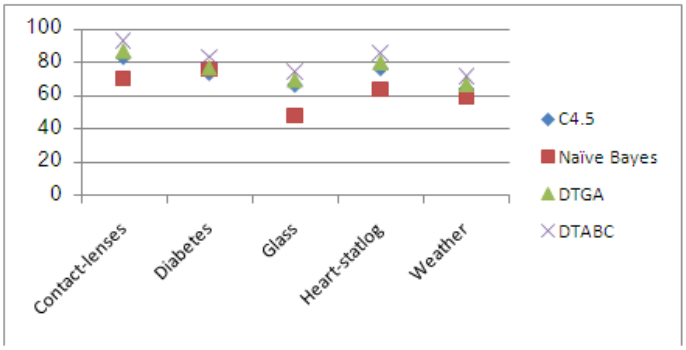

Figure 4 Comparative performance of classification accuracies (%).

From table 9, it is clear that proposed DTABC system is providing better classification accuracy than C4.5, Naïve Bayes and DTGA.

V. CONCLUSIONS

From the survey of the hybrid classifiers [12, 13, 14] and the proposed approach, it is clear that the works are done for improving the classification accuracy irrespective of domain, size and dimensionality. The proposed approach has been experimented over 5 datasets of different size and domain. From the analysis of the work, it is clear that the DTABC is providing optimized set of rules which is depicted through the better classification accuracy than other classifiers taken for comparison. As DTGA is also a hybrid classifier, it is having lesser performance than DTABC. So ABC algorithm is better optimization algorithm than GA to produce more optimized set of rules irrespective of domain, size and dimensionality.

REFERENCES

[1]. Sousa, T., Silva, A., & Neves, A. (2004). Particle Swarm based Data Mining Algorithms for classification tasks.*Parallel Computing, 30*(5-6), 767-783.

[2]. Tan, K. C., Teoh, E. J., Yu, Q., & Goh, K. C. (2009). A hybrid evolutionary algorithm for attribute selection indata mining. *Expert Systems with Applications, 36*(4), 8616-8630.

[3]. Sumida, B. H., Houston, A. I., McNamara, J. M., & Hamilton, W. D. (1990). Genetic algorithms and evolution.*Journal of Theoretical Biology, 147*(1), 59-84.

[4]. Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization:artificial bee colony (ABC) algorithm. *Journal of Global Optimization, 39*(3), 459-471.

[5]. Nemati, S., Basiri, M. E., Ghasem-Aghaee, N., & Aghdam, M. H. (2009). A novel ACO-GA hybrid algorithm for feature selection in protein function prediction. *Expert Systems with Applications, 36*(10), 12086-12094.

[6]. Kennedy, J. (2006). Swarm Intelligence (pp. 187-219).

[7]. Santos, H. G., Ochi, L. S., Marinho, E. H., & Drummond, L. M. A. (2006). Combining an evolutionary algorithm with data mining to solve a single-vehicle routing problem. *Neurocomputing, 70*(1-3), 70-77.

[8]. Yeh, W.-C., Chang, W.-W., & Chung, Y. Y. (2009). A new hybrid approach for mining breast cancer patternusing discrete particle swarm optimization and statistical method. *Expert Systems with Applications, 36*(4),8204-8211.

[9]. Bahamish, H. A. A., Abdullah, R., & Salam, R. A. (2009). *Protein Tertiary Structure Prediction Using Artificial Bee Colony Algorithm.* Paper presented at the Modelling & Simulation, 2009. AMS '09. Third Asia International Conference on Modelling & Simulation.

[10]. Karaboga, N. (2009). A new design method based on artificial bee colony algorithm for digital IIR filters.*Journal of the Franklin Institute, 346*(4), 328-348.

[11]. Karaboga, D., & Akay, B. B. (2005). *An artificial bee colony (abc) algorithm on training artificial neural networks* (Technical Report TR06): Erciyes University, Engineering Faculty, Computer Engineering Department.

[12]. Sarkar B.K., Sana S.S., Chaudhari K., *A genetic algorithm based rule extraction system.* Artificial soft computing,vol. 12., pp. 238–254., 2012.

[13]. B.K. Sarkar, S.S. Sana, K.S. Choudhury, Accuracy Based, Learning classification system, International Journal of Information and Decision Sciences 2 (1) (2010) 68–85.

[14]. B.K. Sarkar, S.S. Sana, A *hybrid approach to design efficient learning classifiers*, Journal, Computers and Mathematics with Applications 58 (2009) 65–73.

[15]. H. Langseth, T.D. Nielsen, *Classification using Hierarchical Naïve Bayes models*., Mach Learn., Volume: 63, Issue: 2, Pages: 135-159., 2006.

[16]. M. A. M. Shukran, Y. Y. Chung, W.C. Yeh, N. Wahid, A.M.A. Zaidi, Artificial Bee Colony based Data Mining Algorithms forClassification Tasks, Modern Applied Science, Vol. 5, No. 4; August 2011.