International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

Survey of Ant Colony Optimization with Classification Rule Discovery for Data Mining

Chandra Prakash[1], Mohammad Rahmatullah[2], Dr Priyanka Tripathi[3]

Department of Computer Engineering & Application, National Institute of Technical teacher's Training & Research Bhopal, India

cpnitttr@gmail.com[1]
mohdrahmatullahcse@gmail.com[2]
ptripathi@nitttrbpl.ac.in[3]

***Abstract*— Knowledge discovery needs effective classification rules in the area of data mining and it is a vibrant area of research in current years. In this paper we have discussed classification rules based on dissimilar sort of rule extraction algorithm. We have also discussed functioning of different mining algorithms. It is analyzed that how Ant colony optimization (ACO) algorithms are applied over combinatorial optimization problems. There are many algorithms were proposed for Ant Colony Optimization problem like remote logical problems, combinatorial problems, forecasting problems and the quadratic assignment problem. There is no single algorithm available which is efficient enough and able to deal with interrelated problems raised from different areas. Therefore in this paper we gave a brief survey in this field in order to fulfil initial need of research.**

Index Terms*— Ant Colony Optimization (ACO), Classification of rules, data mining, Ant Miner

I. INTRODUCTION

Classification is one of the most frequently occurring tasks of human decision making. A classification problem encompasses the assignment of an object to a predefined class according to its characteristics [1, 2]. Many decision problems in a diversity of domains, for example engineering, medical sciences, human sciences, and management science can be considered as classification problems. Popular examples are speech appreciation, character appreciation, medical analysis, bankruptcy prediction, and credit scoring. Throughout the years, a myriad of techniques for classification has been proposed [3,4] such as linear and logistic regression, decision trees and rules, k-nearest neighbour classifiers, neural networks, and support vector machines (SVMs).Since the models concern key decisions of a financial institution, they need to be validated by a financial regulator. Transparency and comprehensibility are, therefore, of crucial importance. Similarly, classification models provided to physicians for medical diagnosis need to be validated [5], demanding the same clarity as for any domain that requires regulatory validation.

II. ANT COLONY OPTIMIZATION (ACO)

An ACO employs artificial ants that cooperate to find good solutions for discrete optimization problems [6]. These software agents mimic the foraging behavior of their biological counter parts in finding the shortest-path to the food source. The first algorithm following the principles of the ACO met heuristic is the Ant System [7, 8] where ants iteratively construct solutions and add pheromone to the paths corresponding to these solutions. Path selection is a stochastic procedure based on two parameters, the pheromone and heuristic values. The pheromone value gives an indication of the number of ants that chose the trail recently, while the heuristic value is a problem dependent quality measure. When an ant reaches a decision point, it is more likely to choose the trail with the higher pheromone and heuristic values. Once the ant arrives at its destination, the solution corresponding to the ant's followed path is evaluated and the pheromone value of the path is increased accordingly. To summarize, the design of an ACO algorithm implies the specification of the following aspects.

- An associate degree setting that represents the matter domain in such how that it lends itself to incrementally building a solution to the matter.
- A retardant dependent heuristic analysis operate, which provides a top quality measure for the various solution components.
- A secretion change rule that takes under consideration the evaporation and reinforcement of the paths.

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

- A probabilistic transition rule supported the worth of the heuristic operate and on the strength of the pheromone trail that determines the trail taken by the ants.
- A clear specification of when the algorithm converges to a solution.

III. SCHEME OF ANT COLONY ALGORITHM BY ACO

Ant colony optimization (ACO) [9] is a branch of a newly developed form of artificial intelligence called swarm intelligence. Swarm intelligence (SI) is "the property of a system whereby the collective behaviours of (unsophisticated) agents interacting locally with their environment cause coherent function global patterns to emerge" [10]. In compilation of virus, which live in resolutions, such as ants and insects an individual cannot do a work on its own; colony's cooperative work is the main reason determining the intelligent behavior .Most real ants are blind, however real ant while walking randomly, leaves a chemical substance known as pheromone [9]. Pheromone attracts other ants to stay close to other ants. The pheromone evaporates over time to allow search evaporation. In number of experiments presented by Dorigo and Maniezzeo explains the complex behavior of colonies [11], where ants always prefer shortest path. Further Parpinelli, Lopes and his colleagues were the first to propose Ant Colony Optimization (ACO) for discovering classification rules, with the system Ant-Miner. They find out that an ant-based search is more flexible, robust and optimized than traditional approaches. Their method uses a heuristic value based on entropy measure. The basic aim of Ant- Miner is to extract classification rules from data [12]. Ant Miner follows a sequential covering approach to discover a list of classification rules from the given data set. These rules are added to the list of discovered rules and the training cases that are covered correctly by these rules, are removed from the training set. It covers all or almost all the training cases. Each classification rule has the form:

IF <term1 AND term2 AND…> Then <CLASS>

A. Pheromone Initialization

The initial amount of pheromone deposited at each path is inversely proportional to the number of values of all points, and is described by:

$$\tau_{ij\,(t=0)} = 1/\sum_{i=1}^{a} b_i \qquad (1)$$

Where 'a' is the total number of attributes, 'b_i' is the numbers of values in the domain of attribute i.

B. Rule Construction

Each rule in Ant miner contains a condition part as the antecedent and a predicted class. The condition part is a conjunction of attribute-operator-value tuples. Let us assume rule condition such as $term_{ij}$ $A_i = V_{ij}$, where A_i is the ith attribute and V_{ij} is the jth value in the area of A_i. The likelihood that this state is additional to the current partial rule that the ant is creating, is known by

$$P_{ij}\,(t) = \frac{\tau_{ij}\,(t).\eta_{ij}}{\sum_{i}^{a}\sum_{j}^{bi}\tau_{ij}\,(t).\eta_{ij}\,,\ \forall\, i \in I} \qquad (2)$$

Where T_{ij} is a problem dependent heuristic value for $term_{ij}$. T_{ij} is the amount of pheromone currently available (at time t) on the connection between attribute i and value I is the set of attributes that are yet used by the ant.

C. Heuristic Value

In Ant Miner, the heuristic value is taken to be an information theoretic measure for the quality of the term to be added to the rule which is measured in terms of the entropy for preferring this term to added, and is specified by the given equations:

$$\eta_{ij} = \frac{\log_2 k - info\ T_{ij}}{\sum_{i}^{a}\sum_{j}^{bi}\log_2 k - info\ T_{ij}} \qquad (3)$$

$$info\ T_{ij} = \sum_{w=1}^{k}\left[\frac{Freq\ T_{ij}^{w}}{|T_{ij}|}\right] + \log_2\left[\frac{Freq\ T_{ij}^{w}}{|T_{ij}|}\right] \qquad (4)$$

Where k is the number of classes, T_{ij} is the total number of cases in partition T_{ij} (partition containing cases where attribute A_i has value V_{ij}); Freq T_{ijw} is the number of cases in partition T_{ij} with class w. The higher value of T_{ij}, it is possible that the ant will select $term_{ij}$.

D. Rule Pruning

The rule pruning procedure iteratively removes the term whose removal will cause the maximum increase in the eminence of the rule. The quality of rule is deliberated by using the following equation:

$$Q = \left(\frac{\text{Truepos}}{\text{Truepos} + Falseneg}\right)\left(\frac{\text{Truepos}}{\text{Falsepos} + True\ neg}\right) \qquad (5)$$

Where, Truepos is the quantity of cases enclosed by the rule and having constant category that expected by the rule, Falsepos is that the range of cases lined by the rule and having a unique category that expected the rule, Falseneg is that the range of cases that don't seem to be lined by the rule, whereas having the category is predicted by the rule, Trueneg is that

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

the range of cases that don't seem to be lined by the rule that have a unique category from the category expected by the rule.

E. Pheromone Update Rule

After each ant completes the construction of its rule, pheromone updating is carried out as follows:

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \tau_{ij}(t).Q, \forall \text{ term ij } \epsilon \text{ the rule } \ldots \ldots \quad (6)$$

To simulate the phenomenon of pheromone evaporation in the real ant colony system, the amount of pheromone associated with each $term_{ij}$ which does not occur in the constructed rule must be reduced. The diminution of pheromone of an unexploited term is performed by dividing the value of each T_{ij} by the summation of all T_{ij}.

IV. cANT MINER ALGORITHM

Ant miner requires the discretization method as a preprocessing method and it is suitable only for the nominal attributes. Mostly real-world classification problems are described by nominal or discrete values and continuous attributes. There is a limitation with Ant-Miner that it is able to cope only with nominal attributes in its rule construction process [12]. So that discretization of continuous attributes is done in a preprocessing step. The disadvantage of this approach is that less information will be available to the classifier. Fernando, Frites, and Johnson proposed an extension to Ant-Miner, named cAnt Miner, which was able to cope with the continuous values as well [13].

A. Purpose

Ant miner that can handle continuous attributes to extract classification rules from data. Entropy based discretization used during rule construction to create Discrete intervals "on-fly".

B. Method:

Dynamic Discretization of continuous value is done by Entropy based discretization method. If nominal attribute, where every $term_{ij}$ has the form ($a_i = v_{ij}$), the entropy for the attribute-value pair is computed as in equation (7) – used in the original Ant-Miner,

$$entropy(a_i = v_{ij}) \equiv \sum_{c=1}^{k} -p(c \mid a_i = v_{ij}).log_2\,(p(c \mid a_i = v_{ij})) \quad (7)$$

Where, p(c|ai = vij) is the experiential probability of examining class c conditional on having observed $a_i = v_{ij}$, k is the number of classes. For computing the entropy of nodes representing continuous attributes ($term_i$) as these nodes do not represent an attribute-value pair, we need to choose a threshold value v to dynamically partition the continuous attribute ai into two intervals: a < v and a_i >=v. The best threshold value is the value v that minimizes the entropy of the partition, given by equation (8):

$$ep_v(a_i) \equiv \frac{|S_{a_i<v}|}{|S|} \cdot entropy(a_i < v) + \frac{|S_{a_i \geq v}|}{|S|} \cdot entropy(a_i \geq v) \quad (8)$$

Where S_{ai}<v is the total number of cases in the partition a_i < v (partition of training cases where the attribute ai has a value less than v), S_{ai}>=v is the total number of cases in the partition a_i>= v (partition of training cases where the attribute ai has a value greater or equal to v) S is the total number of training cases. After selecting threshold v best, the entropy of the $term_i$ matching to the minimum entropy value of the two partitions and it is defined as: equation (9)

$$entropy(term_i) \equiv \min\,(entropy(a_i < v_{best}), entropy(a_i \geq v_{best})) \quad (9)$$

Guidelines for further research: First, examine the performance of special discretization methods in the rule construction process. Second, evaluate other kinds of pheromone updating methods as dropping pheromone on the edges tends to a customized pruning process.

V. CANT $MINER_{PB}$

In this [14] a new approach is proposed, which directs the search performed by the algorithm using the quality of a candidate list of rules, instead of a sole rule. The main motivation is to avoid the problem of rule interaction derived from the order in which the rules are discovered i.e., the outcome of a rule affects the rules that can be discovered subsequently since the search space is modified due to the removal of examples covered by previous rules. In the new sequential covering strategy proposed, the pheromone matrix used by the ACO algorithm is extended to include a tour identification that indirectly encodes the sequence in which the rules should be created, allowing a more effective search for the best list of rules.

VI. C4.5 ALGORITHM

C4.5 is a popular decision tree based algorithm to solve data mining task. Professor Ross Quinlan from University of Sydney has developed C4.5 in 1993 [15]. Basically it is the advance version of ID3 algorithm, which is also proposed by Ross Quinlan in 1986 [16]. C4.5 has additional options like handling missing values, categorization of continuous attributes, pruning of call trees, rule derivation et al. C4.5 constructs a awfully giant sequoia by considering all attribute values and finalizes the choice rule by pruning. It uses a

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

heuristic approach for pruning supported the applied mathematics significance of splits. Basic construction of C4.5 call tree is [5].

- The foundation nodes square measure the highest node of the tree. It considers all samples and selects the attributes that square measure most vital.
- The sample data is passed to succeeding nodes, referred to as 'branch nodes' that eventually terminate in leaf nodes that provide selections.
- Rules square measure generated by illustrating the trail from the foundation node to leaf node.

A. Features of C4.5 Algorithm

There are several features of C4.5, which are discussed in below.

a. Continuous Attributes Categorization

Earlier versions of decision tree algorithms were unable to deal with continuous attributes. 'An attribute must be categorical value' was one of the preconditions for decision trees [17]. Another condition is 'decision nodes of the tree must be categorical' as well. Decision tree of C4.5 algorithm illuminates this problem by partitioning the continuous attribute value into discrete set of intervals which is widely known as 'discretization'. For instance, if a continuous attribute *C* needs to be processed by C4.5 algorithm, then this algorithm creates a new Boolean attributes C_b so that it is true if $C<b$ and false otherwise [18]. Then it picks values by choosing a best suitable threshold.

b. Handling Missing Values

Dealing with missing values of attribute is another feature of C4.5 algorithm. There are several ways to handle missing attributes. Some of these are Case Substitution, Mean Substitution, Hot Deck Imputation, and Cold Deck Imputation, nearest Neighbor Imputation [18]. However C4.5 uses chance worth s for missing value rather distribution existing commonest values of that attribute. This chance values are calculated from the determined frequencies in this instance. For instance, let A could be a mathematician attribute. If this attribute has six values with A=1 and 4 with A=0, then in accordance with applied math, the chance of A=1 is zero.6 and therefore the chance of A=0 is zero.4. At this time, the instance is split into two fractions: the 0.6 fraction of the instances is distributed down the branch for A=1 and the remaining 0.4 fraction is distributed down the other branch of tree. As C4.5 split dataset to training and testing, the above method is applied in both of the datasets. In a sentence we can say that, C4.5 uses most probable classification that is computed by summing the weights of the attributes frequency.

B. Limitations of C4.5 Algorithm

Although C4.5 one in all the favored algorithms, there are some shortcomings of this algorithmic program. Some imitations of C4.5 are mentioned below.

a. Empty branches

Constructing tree with meaning worth is one in all the crucial steps for rule generation by C4.5 algorithmic program. In our experiment, we've got found several nodes with zero values or near zero values. These values neither contribute to come up with rules nor facilitate to construct any category for classification task. Rather it makes the tree larger and additional advanced.

b. branches

Numbers of selected discrete attributes create equal number of potential branches to build a decision tree. But all of them are not significant for classification task. These insignificant branches not only reduce the usability of decision trees but also bring on the problem of over fitting.

c. Over fitting

Over fitting happens once algorithmic program model picks up information with uncommon characteristics. This cause several fragmentations is that the method distribution. Statistically insignificant nodes with only a few samples are called fragmentations [19]. Typically C4.5 algorithmic program constructs trees and grows it branches 'just deep enough to absolutely classify the coaching examples'. This strategy performs well with noise free information. However most of the time this approaches over fits the coaching examples with noisy information. Presently there are two approaches are wide exploitation to bypass this over-fitting in call tree learning [20].Those is:

- If tree grows terribly giant, stop it before it reaches highest purpose of excellent classification of the coaching information.
- Permit the tree to over-fit the coaching information then post prune tree.

VII. CART ALGORITHM

The CART algorithm generates a complicated tree at first, and then pruning the tree structure according to cross validation and test set validation [21]. When the CART algorithm built the original classification tree, it used the Gini ratio to measure the different degree of different classes in

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

every node. The formula of defined Gini ratio on the data set S is as follows:

$$Gini(S) = 1 - \sum_{i=0}^{c-1} p_i^2 \quad (1)$$

where the parameter c is the number of predefined classes, C_i respects a class, S_i is the number of samples belonging to the class C_i, $p_i = s_i/S$ is the relative frequency of class C_i in the data set S, where i = 1.., c-1. Gini ratio indicates the partitions purity of data set S [7]. For instance, for two classes of branch prediction, *Gini*(S) ∈ [0, 0. 5], all data in *S* belong to the same class, then *Gini*(S) =0 which represents that the data set S is pure; if *Gini*(S) =0.5, It indicates that the values of all observations in set S are evenly distributed on the two classes. The value of Gini ratio, which partitions the data set S into k subsets, is Gini ratio weighted sum of the resulting subset, the formula is:

$$Gini_{split}(S) = \sum_{i=0}^{k-1} \frac{n_i}{n} \times Gini(S_i). \quad (2)$$

Where the parameter n_i represents the number of samples in a subset of the partition and n represents the total number of samples in the dataset. When the sample set is split, the splitting rule uses the form of binary tree to represent. The CART algorithm starts from the root partition, calculates the value of $Gini_{split}$ for all properties, and selects the attribute which has the minimum value of Gini split as the split point, then processes the above operations for each child node recursively. The CART algorithm is described as follows:

Step 1: select the optimal split points for each attribute in each classification node. The process of select the optimal split point in an attribute is as follows: for the continuous variable X_i, $\{X_i \mid X_i>C\}$? i=1…m}, *C* is a constant in the range of variable X_i; for the discrete variable X_i, $\{X_i \mid X_i>V\}$? i=1... m}, V is a subset of U, which is the collection of all possible values of variable X_i in all sample space. According to compare the values of Gini split m all splitting rule, CART selects the splitting rule with the smallest value of $Gini_{split}$, and divides the node into left and right child nodes;

Step 2: select the optimal split points for this node from the optimal split points of each attribute. The rule of choice is that select the attribute which could achieve the smallest value of the formula:

$$Gini_{split}(S) = \frac{N_1}{N} Gini(S_1) + \frac{N_2}{N} Gini(S_2) \quad (3)$$

Step 3: execute the step 1 and step 2 for each divided child node recursively until the number of samples in the node is very small or the samples in the node belonging to the same category.

VIII. R2-CART ALGORITHM

Although the CART algorithm could process analysis of the sample data and build the classification tree model, the tree often has redundant tree leaf nodes. So that the description for the classification is not clear and simple enough, sometimes the classification results are also difficult to understand. In order to solve the problem of existing redundant tree leaf nodes in an classification tree of CART algorithm, this paper proposes an new improved classification and regression tree algorithm named R2- CART. At first, the algorithm uses Attribute Reduction Algorithm 1 based on rough set to reduce the attributes in the sample set; secondly, run the CART algorithm on the reduced attribute set of sample data and get the classification tree; then change paths of the classification tree root to the end of each leaf node into the classification rules; finally, use the Rule Reduction Algorithm 2 to reduce the rule of classification tree and express the reduction classification rules in the form of classification tree. Fig. 1 shows the basic process of the new improved algorithm R2 CART. The improved algorithm R2-CART is specifically described as follows:

Step 1: run Pre(D) to preprocess the sample data set which would be classified, remove the dirty data and fill in missing values, obtain a binary information system [U, Ω, V, f];

Step 2: run the Algorithm 1 to reduce the attributes of the sample data set D, get the minimal set of attributes A_{min}, a. Reduced sample data set D_{re} and a new binary information system [U, Ω_1, V, f_1], $\Omega_1 \in \Omega$;

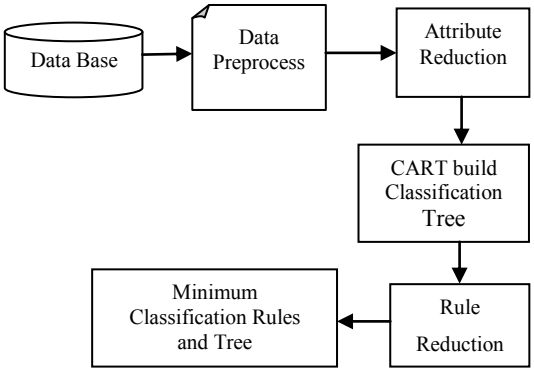

Figure 1. The basic process of the improved algorithm R2-CART

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

Step 3: run the algorithm CART to classify the sample data set D_{re} obtain the classification tree *Tc*;

Step 4: use the form of if - then to express the paths of the classification tree root to the all leaf nodes, then obtain the classification rule set R consisted of the classification rules corresponding to all paths;

Step 5: use the Algorithm 2 to reduce the rules in classification rule set R and obtain the minimum classification rule set R_{min};

Step 6: express the minimum classification rule set R_{min} as the brief classification tree T_{min}.

IX. MERITS OF DIFFERENT ALGORITHMS

The benefits of these algorithms are as follows:

A. cANT-MINER algorithm

cAnt-miner algorithm incorporates entropy based discretization method in order to cope with continuous attribute during the rule construction process. It does not require a discretization method in a preprocessing step, as ant-miner requires. Empirical results in paper [21] show that creating discrete intervals during the rule construction process facilitates the discovery of more accurate and significant simpler classification rules.

B. cANT MINERPB algorithm

In this a new approach is proposed, which directs the search performed by the algorithm using the quality of a candidate list of rules, instead of a sole rule. The main motivation is to avoid the problem of rule interaction derived from the order in which the rules are discovered. The pheromone matrix used by the ACO algorithm is extended to include a tour identification that indirectly encodes the sequence in which the rules should be created, allowing a more effective search for the best list of rules.

C. C4.5 Algorithm

C4.5 is one of the most well-liked algorithms for rule base classification. There are unit several empirical options during this rule like continuous range categorization, missing worth handling, etc. but in several cases it takes additional interval and provides less accuracy rate for properly classified instances. On the opposite hand, an outsized dataset may contain many attributes. We'd like to decide on most connected attributes among them to perform higher accuracy victimization C4.5. It's conjointly a tough task to decide on a correct rule to perform economical and ideal classification.

TABLE 1: FUNCTIONING OF DIFFERENT MINING ALGORITHMS

Algorithm	Functions
cANTMINER	Extension of Ant Miner deals with Continuous attributes. Entropy based discretization used during rule construction.
cANTMINER$_{PB}$	The main aim is to avoid the problem of rule interaction. The Best List of Rules is discovered by each ant instead of List of Best Rules.
C4.5	The main function of this algorithm is continuous number categorization and missing value handling, A large dataset might contain hundreds of attributes.
CART	One of the main functions of CART is that it automatically adjusts the tree model to minimize the effects of the measured impurities and determines the best node for classification. CART can be used to analyse either continuous or discrete data
R2-CART	R2 CART could maintain the accuracy of the Classification tree, simplify the complexity of the tree. Reduce the average length of classification rules effectively. To solve the problem of existing redundant tree leaf nodes in a classification tree of CART algorithm.

D. CART Algorithm

The CART algorithm has certain advantages. The CART uses the Gini ratio to estimate diversity of physical condition when the subjects are divided by a medical test indicator. One of the advantages of CART is that it automatically adjusts the tree model to minimize the effects of the measured impurities and determines the best node for classification. CART can be used to analyze either continuous or discrete data [21].

E. R2-CART Algorithm

During this at first, the algorithm uses Attribute Reduction Algorithm 1 based on rough set to reduce the attributes in the sample set; secondly, run the CART algorithm on the reduced attribute set of sample data and get the classification tree; then change paths of the classification tree

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

root to the end of each leaf node into the classification rules; finally, use the Rule Reduction Algorithm 2 to reduce the rule of classification tree and express the reduction classification rules in the form of classification tree.

X. Conclusions

Several strategies have been studied for controlling the influence of pheromone and all these methodologies have been demonstrated in this paper. However, to develop the understanding of parameters and effects of each parameter of every system needs a very detailed experimentation. The sole purpose of this paper is to help the researchers to select the one according to their need. Future research will focus on using these algorithms together, such that the strengths and efficiency of these techniques can be increased.

References

[1] R.O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification. New York: Wiley, 2000.

[2] D. J. Hand and S. D. Jacka, Discrimination and Classification. New York: Wiley, 1981.

[3] R.S.Parnelli, H.S.Lopes and A.A.Freitas, "Data Mining with an Ant Colony Optimization Algorithm", IEEE Trans. On Evolutionary Computation, special issue on Ant colony Algorithm, 6(4), 321-332, 2002.

[4] B. Baesens, "Developing intelligent systems for credit scoring using machine learning techniques," Ph.D. dissertation, K.U. Leuven, Leuven, Belgium, 2003.

[5] M. Pazzani, S. Mani, and W. Shankle, "Acceptance by medical experts of rules generated by machine learning," Methods of Inf. Med., vol. 40,no. 5, pp. 380–385, 2001.

[6] M. Dorigo and T. Stützle, Ant Colony Optimization. Cambridge, MA: MIT Press, 2004.

[7] M. Dorigo, V. Maniezzo, and A. Colorni, Positive feedback as a search strategy Elettronica e Informatica, Politecnico di Milano, Italy, Tech. Rep. 91016, 1991.

[8] "Ant system: Optimization by a colony of cooperating agents," IEEE Trans. Syst., Man, Cybern. Part B, vol. 26, no. 1, pp. 29–41, Feb.1996.

[9] Dorigo.M and Caro.G.D. "Ant Algorithm for Optimization", Artificial Life, 1999.

[10] Bonabeau. E, Dorigo.M & Theraulaz. G."Swarm Intelligence: From Natural to Artificial System", New York: Oxford University Press, 1999.

[11] Dorigo.M & Maniezzo.V "The ant System: Optimization by a colony of cooperating Agents", IEEE Transactions on Systems, Man, and Cybernetics, 26(1), 1-13, 1996.

[12] D. J. Hand, "Pattern detection and discovery," in Pattern Detection and Discovery, ser. Lecture Notes in Computer Science, David J. Hand , N. Adams, and R. Bolton, Eds. Berlin, Germany: Springer-Verlag, 2002, vol. 2447, pp. 1–12.

[13] Fernando E.B. Otero, Alex A. Freitas, and Colin G. Johnson, "cAnt-Miner: An Ant Colony Classification Algorithm to Cope with Continuous Attributes", Springer-Verlag Berlin, Heidelberg 2008.

[14] F. Otero, A. Freitas, and C. Johnson. "A new sequential covering strategy for inducing classification rules with ant colony algorithms", to appear in IEEE Transactions on Evolutionary Computation, 2012.

[15] J.R. Quinlan, C 4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[16] J.R Quinlan, Induction of Decision Trees, Machine Learning, 1986, pp81-106.

[17] A. B. M. S. Ali and S. A. Wasimi, Data Mining: Methods and Techniques, Thomson Publishers, Victoria, Australia, 2007.

[18] M. Singh, How to Handle Missing Values, Article base, viewed on Oct 2009, at

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 2, Issue 2, No 2, 2014. ISSN: 2348-6600
Reference ID: IJCS-057 PAGE NO: 328-335

http:www.articlebase.com/information-technology articles/how-to-handle-missing-values-538449.html.

[19] J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publish, 2001.

[20] Shawkat Ali and Kate a. Smith, On learning algorithm selection for classification, Applied Soft Computing, Dec 2004.

[21] Weiguang Wang, Cong Wang, Wanlin Gao and Jinbin Li "An Improved Algorithm for CART based on the Rough Set Theory" Fourth Global Congress on Intelligent Systems 2013.