

Annealing strategy for an enhance rule pruning technique in ACO-based rule classification

Hayder Naser Khraibet AL-Behadili¹, Ku Ruhana Ku-Mahamud², Rafid Sagban³

¹Computer Science Department, Shatt Alarab University College, Iraq

²School of Computing, Universiti Utara Malaysia (UUM), Malaysia

³Department of Software, University of Babylon, Iraq

Article Info

Article history:

Received Feb 21, 2019

Revised Jun 28, 2019

Accepted Jul 22, 2019

Keywords:

Data mining

Metaheuristic

Parameter control

Rule induction

Swarm-intelligent

ABSTRACT

Ant colony optimization (ACO) was successfully applied to data mining classification task through ant-mining algorithms. Exploration and exploitation are search strategies that guide the learning process of a classification model and generate a list of rules. Exploitation refers to the process of intensifying the search for neighbors in good regions, whereas exploration aims towards new promising regions during a search process. The existing balance between exploration and exploitation in the rule construction procedure is limited to the roulette wheel selection mechanism, which complicates rule generation. Thus, low-coverage complex rules with irrelevant terms will be generated. This work proposes an enhancement rule pruning procedure for the ACO algorithm that can be used in rule-based classification. This procedure, called the annealing strategy, is an improvement of ant-mining algorithms in the rule construction procedure. Presented as a pre-pruning technique, the annealing strategy deals first with irrelevant terms before creating a complete rule through an annealing schedule. The proposed improvement was tested through benchmarking experiments, and results were compared with those of four of the most related ant-mining algorithms, namely, Ant-Miner, CAnt-Miner, TACO-Miner, and Ant-Miner with hybrid pruner. Results display that our proposed technique achieves better performance in terms of classification accuracy, model size, and computational time. The proposed annealing schedule can be used in other ACO variants for different applications to improve classification accuracy.

Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Hayder Naser Khraibet AL-Behadili,

Computer Science Department,

Shatt Alarab University College, Iraq.

Email: haider872004@gmail.com

1. INTRODUCTION

Data mining involves methods used at the intersection of a multidisciplinary field, and it aims to discover knowledge from databases. Data mining tasks can be categorized into classification, clustering, regression, and association [1-4]. Classification involves introducing a classification model from a database by finding the relationship between data attributes and their classes. This model can then be utilized to classify the class values of other instances. Examples of classification tasks include intrusion detection, medical diagnosis, handwritten digit recognition, spam email detection, and bankruptcy determination [5-9]. In recent years, the field of swarm intelligence (SI) has been derived by observing the swarming behavior of some animals, such as ant colonies, flocking birds, and fish schools [10, 11]. The SI field, particularly ant colony optimization (ACO), has received widespread attention in data classification tasks. ACO is an SI metaheuristic algorithm that is used to solve combinatorial optimization problems [12]. ACO was first used

to solve the traveling salesman problem. Since then, it has widely been applied to various NP-hard problems such as vehicle routing, quadratic assignment, grid computing, and data mining [13-15]. The ant system (AS) was the first version of ACO algorithm which was proposed by Dorigo et al. in the 1990s. ACO uses the stigmergy method to find good paths when searching for a solution and then deposits pheromones to mark these paths. The amount of pheromones laid depends on the distance and the solution quality. As each ant moves randomly, other ants can detect previously visited paths and decide whether to visit them. The pheromones of other ants are then deposited along these paths. The process is described by a positive feedback iteration and stochastic behavior. Therefore, indirect communication among ants provides them with the ability to find the optimal solution for complex optimization problems. In addition, ACO was successfully applied to data mining classification tasks, particularly through the Ant-Miner algorithm. This algorithm, which simulates the foraging behavior of ants, is considered one of the most successful SI algorithms due to its competitive accuracy compared with other well-known classification algorithms [16]. Ant-Miner uses ACO techniques to discover classification rules, which are presented as *IF* (conditions)–*THEN* (class) statements. Rule-based classification introduces an understandable model to the user. Thus, the discovered rules have natural language representations that are easy to interpret.

Overfitting is a problem that is frequently encountered when dealing with rule classification for a large number of terms. Rules are perfectly fit a particular dataset from which they are created, and thus, hard to generalize those rules to a new dataset. Rule pruning is a routine utilized in the rule discovery process to avoid overfitting. Many pruning methods, such as pre, post, and hybrid methods, have been proposed in Ant-Miner literatures to omit irrelevant terms that do not give correct classification [17]. Most of these algorithms are based on post-pruning, which removes irrelevant terms after the rules are expanded to their maximum size. Subsequently, the quality of a rule is measured using a quality function. If the quality can be further improved by simplifying the rule, then simplification will be performed repeatedly until further simplification will degrade the quality of the rule or only one term is left in the rule. Post-pruning method is considered a time-consuming, complex, and costly process when the training dataset consists of a considerable amount of attributes and instances [18, 19]. In addition, post-pruning incorporates greedy search biases to remove one term at a time. In contrast, pre-pruning techniques eliminate irrelevant terms before developing a complete rule. The stopping criteria addresses unrelated terms during the construction process (e.g., terms are no longer selected when the information measured for some terms do not exceed the predetermined value) [20]. The limitation of pre-pruning techniques is the difficulty in selecting the appropriate threshold to omit unrelated terms. The construction rules will underfit the data when a high threshold value is applied. Conversely, a small value of the threshold will not be able to overcome the overfitting problem. The threshold value is data-dependent, and it is considered very critical. Hence, selecting an unsuitable value will fail to overcome the problem of overfitting. Another study has proposed a hybrid pruning procedure that combines post-pruning and pre-pruning based on a user-specified parameter [21]. The parameter determines the allowable number of terms in a rule. Pre-pruning is applied to each rule that exceeds the predefined parameter value. Then, the number of terms in the selected rule is reduced to be equal to that of the parameter value. Thereafter, the original Ant-Miner's rule pruning method will be applied. The parameter value of this mechanism is critical, and any mistake related to this value may lead to an extremely simple model that does not consider the implicit structure of the data. Thus, the classification model will be inappropriate and perform poorly. This is known as underfitting problem.

The current study presents an enhanced Ant-Miner algorithm called ACO-Annealing, that uses an annealing strategy to find a trade-off between exploration and exploitation in the rule construction progress and prevent the inclusion of unfit terms. This strategy is based on a new parameter, called Importance Rate (IR), which controls the strength or importance of each term added to the rule. Our proposed technique deals with irrelevant terms during learning instead of finding a complete rule and then pruning it. Each ant is assigned its own IR value during the model learning process. Over time, the IR value is modified linearly to ensure diversity. Other terms that can be added to the rules are explored, and the learning process is prevented from being stuck in local optima. Furthermore, this strategy reduces post-pruning computational complexity by eliminating the majority of unfit terms during the construction process. In addition, this strategy overcomes the pre-pruning limitation by using dynamically adjusted rejection criteria during the learning process rather than keeping such criteria constant.

2. ENHANCED ANT-MINER ALGORITHM

Ant-mining rule-based classifiers follow the ACO variants. The variants are AS, ant colony system (ACS), max–min AS (MMAS) algorithms [22]. The first developed classifier, known as Ant-Miner, is based on the well-known AS algorithm. Since then, numerous classifiers have been developed. Among these improvements, one of the most successful is the Antminer+ algorithm, which is based on the MMAS

strategy. Other classifiers, such as Ant-Miner2, Ant-Miner3, and Ant-Miner4, are based on ACS [23-25]. The three aforementioned ACO variants were developed to enhance balance between exploration and exploitation and to produce better classification rules. However, the MMAS classifier is expensive in terms of its rule construction procedure, and its learning time is longer than those of the two other variants. MMAS also uses a less accurate heuristic function. By contrast, the ACS-based classifier is improved by using an adaptive-state transition strategy to develop the exploration mechanism; however, it does not enhance the exploitation strategy. During the learning process, an imbalance between exploration and exploitation affects the learning model, which results in a non-optimal rule set. Therefore, the present study focuses on the Ant-Miner variant, which is a prominent ACO classification algorithm, for discovering rules [26, 27].

Exploitation and exploration are the primary search strategies that guide the rule construction process in the Ant-Miner algorithm. Exploitation refers to the process of intensifying the search for neighbors in good regions, whereas exploration transfers the search toward new promising regions. Therefore, the success of the Ant-Miner algorithm depends on the balance between exploitation and exploration to arrive at an optimal global model during the learning process that can be used during classification. Ant-Miner starts with high exploration due to the randomness of the selection terms that must be included in the rule. After producing a sufficient number of rules, ants narrow down the search to promising regions (exploitation), continue with the exploitation results in generating the same rule set, and then get trapped in local optima. Therefore, the algorithm restarts the search with random term selection.

An annealing strategy is a technique used to improve the performance of the Ant-Miner classifier through better balance between exploration and exploitation and preventing the inclusion of irrelevant terms. During rule construction, probabilities are used by ants to determine the paths in traveling through a constructed graph. Ants start from an initial node and then select terms on the basis of the probability function. A probability function is used by each ant to determine the importance and strength of the term being selected while constructing a rule. The probability function of each term is calculated in accordance with its heuristic information and pheromone amount. Thus, our proposed method uses the parameter Importance Rate (IR) (Figure 1, Line 6) and an annealing schedule to determine the degree of uphill movement allowed during a model's learning process. This annealing schedule is critical for the credibility of terms and aims to find the global optimum for a given classification problem. Therefore, IR is a pre-pruning criterion that is responsible for adding only the most important terms to each rule (Figure 1, Lines 8 to 13). The value of IR linearly increases over time, i.e., from its minimum value to its maximum value, to escape local optima. This strategy increases search diversity by assigning a unique value to each ant in the colony (Figure 1, Line 20). A high-level description of our proposed algorithm is provided in Figure 1.

```

1  TrainingSet= {all Training instances};
2  InitialConstructionRuleList =[];
3  WHILE(TrainingSet>MaxUncoveredInstances)
4  AntIndex=1; ConvergenceTest=1;
5  InitializePheromone;
6  Initialize IR= Mini-Value; /* determine the importance or the strength of each term */
7  REPEAT
8      WHILE (TermsNotSelected==True)
9          IF (TheProbabilityOfChosenTermij is >= IR,)
10             THEN AddTerm;
11             ELSE RejectTermFromInclusion;
12             END IF;
13         END WHILE
14  RulePrune;
15  UpdatePheromone;
16  IF(CurrentRule=PreviousRule)
17  THEN ConvergenceTest = ConvergenceTest + 1;
18  ELSE ConvergenceTest = 1;
19  END IF AntIndex = AntIndex + 1;
20  GraduallyIncreaseParameter IR;
21  UNTIL(AntIndex>=AntNo)OR(ConvergenceTest>=RuleConvergenceNo)
22  SelectBestRule;
23  AddBestRule to DiscoveredRuleList;
24  TrainingSet = TrainingSet - {InstancesSetCoveredByRule};
25  END-WHILE

```

Figure 1. High-level pseudocode of the proposed algorithm

The annealing strategy suggests initializing the algorithm with a minimum IR value to explore the search space. During the learning process, each ant is assigned its own value and terms are randomly added to the rule. Over time, exploitation becomes essential for the learning process to find the most important terms and include them in the construction rule. The value of IR gradually increases when our developed mechanism is used, as presented in Equation (1), to increase the parameter for the next ant.

$$IR_t = (\text{Max})f(t) \quad (1)$$

$$\text{where } f(t) = e^{-t \frac{\ln(a)}{a}}$$

where IR represents the importance of each term that will be selected; Max is the maximum IR value; F(t) is a function that increases the IR value for the next ant; and a and t are the number of ants and the ant index, respectively.

The IR parameter has minimum and maximum values as boundaries. Annealing strategy process is illustrated in Figure 2, starting with the minimum value. Our study shows that the minimum value of IR is nearly a nonentity (0.0). Then, an ant constructs the rule, prunes it, and updates the pheromone in accordance with the minimum value. Therefore, the annealing strategy gradually increases the IR value for the next ant to avoid falling into local optima by allowing the selection of non-improving moves as other terms to explore new rules. Over time, the significance of the IR parameter will increase until it reaches the maximum value. This process iterates until convergence occurs or all ants discover their own rules.

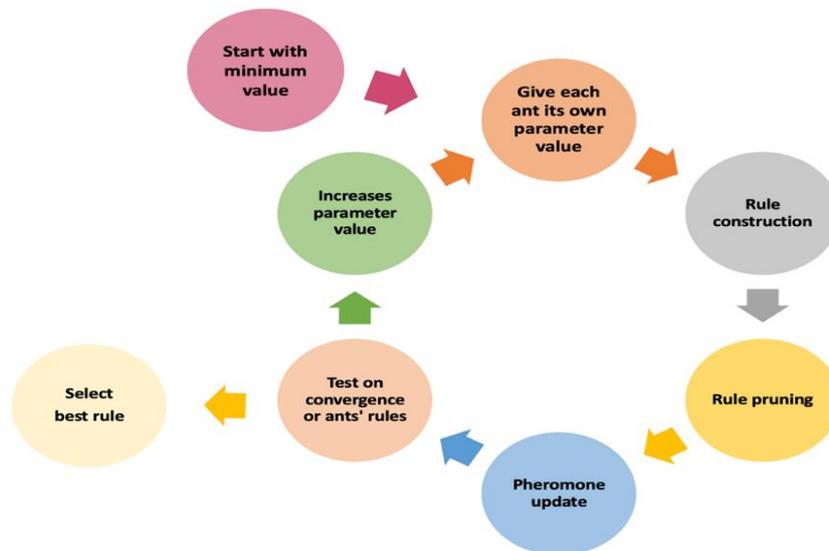


Figure 2. Process of the annealing strategy

3. RESEARCH METHOD

A research method is a systematic process that aims to solve a particular research problem. A researcher should not only know existing research methods/techniques but also evaluation criteria that must be used. The current study uses a 10-fold cross-validation method, which is considered the best approach in evaluating the performances of algorithms when dealing with datasets of various sizes [28]. This procedure randomly divides a dataset into 10 approximately equally sized subsets or partitions. Two major stages are used to build the classification model during the learning process. The algorithm in the first stage utilizes nine partitions in the learning process to construct the classification model based on ant capabilities. Meanwhile, in the second stage, the classification model is tested with the remaining partition that was not used by the algorithm. This procedure is repeated ten (10) times. Figure 3 depicts the validation of the classification model which has been adopted and adapted from [29].

The performance of each repetition (10 times) is statistically computed for three (3) metrics: classification accuracy, model size, and elapsed computational time (in second). The classification accuracy is also a measure for error while model size can be used to analyse the model complexity. Then, the overall performance is averaged, and standard deviation is calculated as shown in Figure 4.

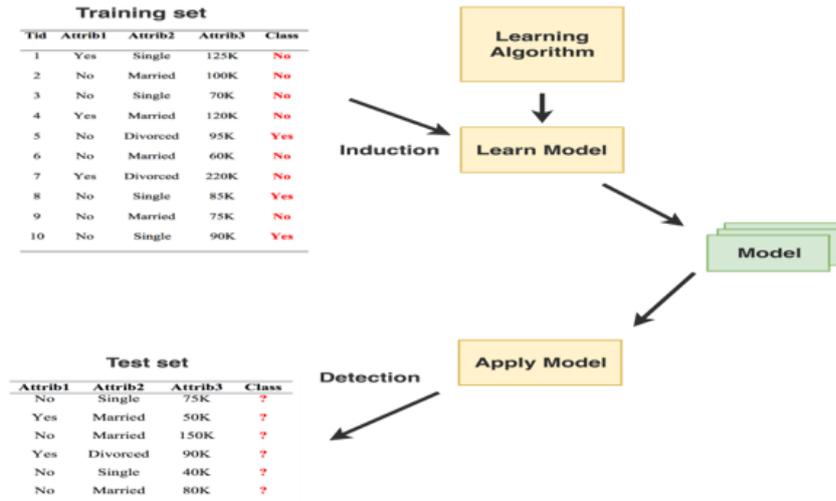


Figure 3. Validation process for classification model

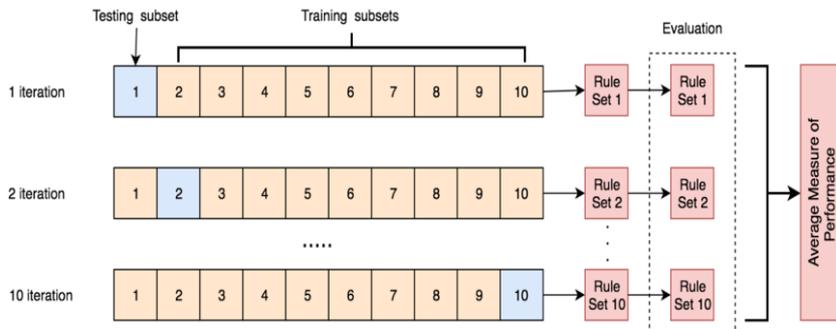


Figure 4. 10-fold cross-validation method

4. RESULTS AND DISCUSSION

We conducted several experiments using 15 publicly available benchmark databases from the repository of the University of California, Irvine (UCI) [30] to test and study the performance of our proposed algorithm. These datasets are widely used in the literature of ant-mining algorithms, and they vary in their attribute numbers (i.e., between 4 and 60). In addition, attribute numbers consist of continuous and categorical types. The datasets also vary in terms of instance number (within the range of 148–8124) and target class numbers. The major characteristics of the datasets are summarized in Table 1.

Table 1. Major Features of the Datasets Used in Our Experiments

Datasets Name	Description		
	Instances number	Attributes Number	Classes Number
BalanceScale	625	4	3
Ljubljana (Breast Cancer)	286	9	2
Wisconsin (Breast Cancer)	699	9	2
Credit (a)	690	15	2
Credit (g)	1000	20	2
Diabetes dataset	768	8	2
Cleveland (Heart)	303	13	5
Statlog (Heart)	270	13	2
Hepatitis dataset	155	19	2
Ionosphere data	351	34	2
Iris (flower)	150	4	3
Lymphography	148	18	4
Mushroom	8124	22	2
Segment	2310	19	7
Sonar	208	60	2

The Ant-Miner algorithm does not work with continuous features during the training process. Thus, a discretization method is required to address such attribute types. In the Ant-Miner algorithm literature, discretization is performed either before the rule construction process [31] or during the rule generation process [32]. In our experiment, we use the well-known C4.5-disc method as a preprocessing step. This method has also been used in other studies on the the Ant-Miner algorithm.

The proposed algorithm is evaluated by comparing its performance with those of the Ant-Miner and other related ant-mining algorithms, namely, CAnt-Miner, TACO-Miner, and Ant-Miner with hybrid pruner. The CAnt-Miner classification algorithm is an extended version of the Ant-Miner classifier that integrates an entropy discretization method to split the continuous attributes during the model training [33]. In contrast, TACO-Miner provides a threshold value based on the information gain of each term. If the information associated with the selected term is below the threshold value, then the information will be rejected. The threshold value is considered a pre-pruning criterion that determines the acceptance or rejection of terms [34]. The Ant-Miner with hybrid pruner presents a new hybrid rule pruner that combines the original Ant-Miner’s rule pruner with a rule pruner based on information gain and the r parameter. This parameter determines the number of terms allowed in the rule. The new method will be applied to each rule that exceeds the number of terms in a rule. This number of terms is a user specified parameter. Then, the number of terms in the selected rule is reduced until it is equal to the specified value. This method is implemented on the basis of information gain and roulette wheel for selection. Thereafter, the procedure of the Ant-Miner’s original rule pruner is applied [21]. The experiments have been conducted using a MacBook Pro (2016) with a 2.9 GHz Intel Core i5 processor and 8 GB 2133 MHz memory. We have not attempted to optimize the parameters of ACO-Annealing in the experiments. The parameter values for the Ant-Miner classifiers are adopted from Raul et al. (2016) to ensure that each classifier will work with the same parameters’ values and to provide a fair evaluation of the results [35]. The parameters we used for the Ant-Miner classifiers are as listed in Table 2.

Table 2. Experiment Parameters

Parameter	Description	Value
NA	Number of ants in the colony.	10
MCR	Minimum number of cases that each rule must cover.	5
MUC	Maximum of uncovered cases by the discovered rules.	10
RC	Number of rules for convergence.	10
NI	Number of iterations.	10

Figure 5 displays the comparison of the classification accuracy of ACO-Annealing algorithm with those of the Ant-Miner, CAnt-Miner, TACO-Miner, and Ant-Miner with hybrid pruner algorithms. Each column in the figure displays classification accuracy obtained through 10-fold cross-validation. Among the evaluated algorithms, our proposed technique achieves the highest classification accuracy on 11 datasets (73% of the results), thereby indicating that the performance of proposed technique is above average.

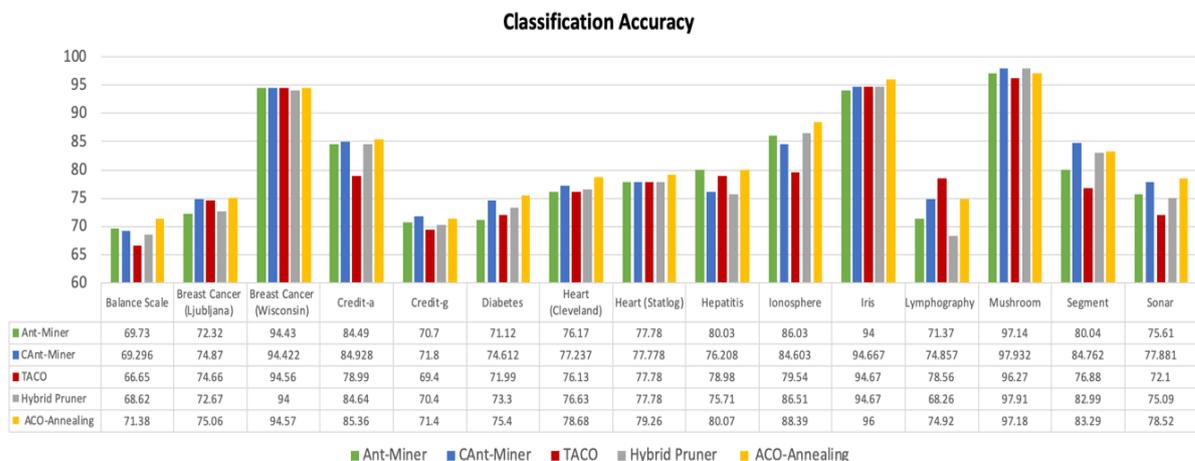


Figure 5. Comparison of the classification accuracies of different ant-mining classification algorithms

Figure 6 shows the performance evaluation of the algorithms based on model size (average number of terms in a rule). Again, the 10-fold cross validation has been used in the experiment. The overall results show that the model size discovered by TACO-Miner is simpler on nine datasets, whereas our proposed technique ranks second place, with a simpler model size on seven datasets. However, the no-free-lunch theorem states that no algorithm performs best on all datasets [36, 37].

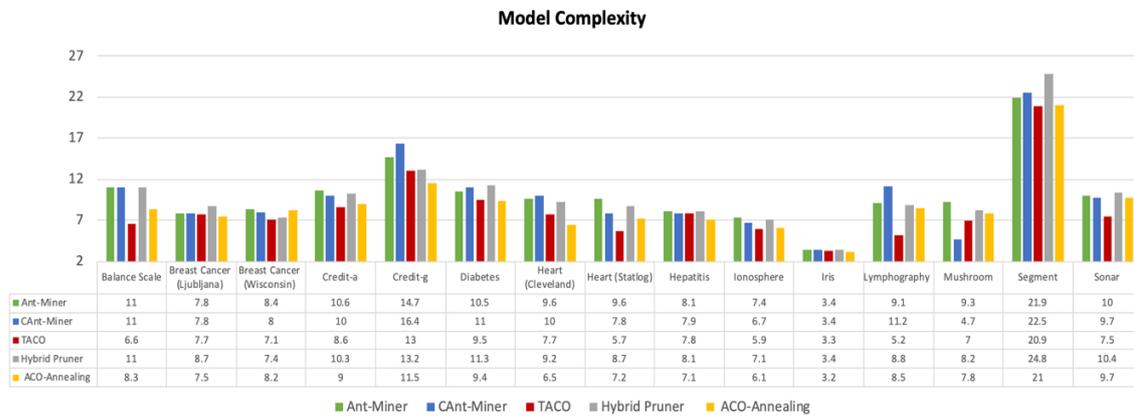


Figure 6. Comparison of model sizes of different ant-mining classification algorithms

Table 3 provides the overall average statistical test results for classification accuracy, model size, and computational time (in second) on all the datasets used in our experiments. In general, our proposed algorithm performs the best for classification accuracy and computational time. In addition, our algorithm achieves the second-best average for model size which shows that the complexity of the algorithm is lower than Ant-Miner, CAnt-Miner and Hybrid Pruner algorithms.

Figure 7 shows the plot of average classification accuracy rank against the average model size rank. A lower average rank indicates better performance of the algorithms. The result shows that our proposed algorithm outperforms the other four classification algorithms based on ideal rank between the model size and classification accuracy. Trade-off between classification accuracy and model size has been considered in developing our proposed algorithm. In contrast, the most accurate classification algorithm requires more terms per rule to produce better classification accuracy because our proposed strategy aims to find the appropriate pruning criteria for each dataset by considering classification accuracy.

Table 3. Test Results Based on Average Performance on All Datasets

Datasets	Ant-Miner	CAnt-Miner	TACO	Hybrid Pruner	ACO-Annealing
Average Predictive Accuracy	80.06	81.06	79.14	79.95	81.97
Average Model Size	10.09	9.87	8.23	10.04	8.73
Average Computational Time	24.93	7.27	21.33	20.80	5.80



Figure 7. Comparison between the averaged classification accuracy rank vs. the averaged complexity size rank of the ant-mining classification algorithms

5. CONCLUSION

ACO is an SI metaheuristic algorithm used to solve complex problems. The Ant-Miner algorithm is a typical example of ACO for classification. The strength of this algorithm lies in its ability to provide a complete comprehensible result, which is a key requirement in different application domains, such as medical diagnosis and bankruptcy. In this paper, we elucidate the importance of the pruning component of the Ant-Miner algorithm and discuss the primary strategies (exploitation and exploration) that guide the rule construction process. We then modify these strategies in accordance with an online control parameter, which varies during the learning process by using an annealing strategy. This method, which is presented as a pre-pruning technique, deals with irrelevant terms before creating a complete rule. Experiments were conducted using publicly available benchmarking datasets from the UCI repository. Results indicate that our proposed strategy improves the overall performance of the Ant-Miner classifier algorithm and yields better results than CAnt-Miner, TACO-Miner, and Ant-Miner with hybrid pruner algorithms. Further research is required to improve the adaptability of the proposed parameter value using different search spaces and considering feedback from the rule construction process. This mechanism involves assigning rewards, and the action of the Ant-Miner algorithm may determine whether a new value will prevail in the search space.

ACKNOWLEDGEMENT

Funding for this study is from the Ministry of Higher Education Malaysia under the Transdisciplinary Research Grant Scheme (S/O code 14163).

REFERENCES

- [1] D. Martens, et al., "Editorial survey: Swarm intelligence for data mining," *Mach. Learn.*, vol. 82, pp. 1-42, 2011.
- [2] A. Gavrilovski, et al., "Challenges and Opportunities in Flight Data Mining: A Review of the State of the Art," *AIAA Infotech @ Aerospace*, pp. 1-18, 2016.
- [3] A. M. Jabbar and K. Ku-Mahamud, "Ant-based sorting and ACO-based clustering approaches: A review," *IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, pp. 217-223, 2018.
- [4] M. Z. Hossain, et al., "A dynamic K-means clustering for data mining," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, pp. 521-526, 2019.
- [5] A. Martin, et al., "An Analysis on Qualitative Bankruptcy Prediction Rules using Ant-Miner," *Int. J. Intell. Syst. Appl.*, vol. 6, pp. 36-44, 2014.
- [6] H. N. K. Al-behadili, "Classification Algorithms for Determining Handwritten Digit," *Iraq J. Electr. Electron. Eng.*, vol. 12, pp. 96-102, 2016.
- [7] M. Durgadevi and R. Kalpana, "Medical distress prediction based on Classification Rule Discovery using ant-miner algorithm," *Proc. 2017 11th Int. Conf. Intell. Syst. Control. ISCO 2017*, pp. 88-92, 2017.
- [8] D. Albashish, et al., "A hierarchical classifier for multiclass prostate histopathology image gleason grading," *J. Inf. Commun. Technol.*, vol. 2, pp. 323-346, 2018.
- [9] Y. Ouaïd, et al., "Handwritten tiffinagh character recognition using simple geometric shapes and graphs," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, pp. 598-605, 2019.
- [10] S. Bashath and A. R. Ismail, "Comparison of swarm intelligence algorithms for high dimensional optimization problems," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 11, pp. 300-307, 2018.
- [11] Z. Al Nuaimi and R. Abdullah, "Neural Network Training Using Hybrid Particle-Move Artificial Bee Colony Algorithm for Pattern Classification," *J. Inf. Commun. Technol.*, vol. 2, pp. 314-334, 2017.
- [12] R. Sagban, et al., "ACO ustic: A nature-inspired exploration indicator for ant colony optimization," *Cient. World J.*, 2015.
- [13] H. J. A. Nasir, et al., "Load balancing using enhanced ant algorithm in grid computing," *Proceedings - 2nd International Conference on Computational Intelligence, Modelling and Simulation, CIMSIm 2010*, pp. 160-165, 2010.
- [14] M. Dorigo and T. Stützle, "Ant Colony Optimization," Cambridge, MA, USA, MIT Press, 2004.
- [15] A. M. Jabbar, "Controlling the Balance of Exploration and Exploitation in ACO Algorithm," *J. Univ. Babylon*, vol. 26, pp. 10-17, 2018.
- [16] R. Parpinelli, et al., "Data Mining With an Ant Colony Optimization Algorithm," *IEEE Trans. Evol. Comput.*, vol. 47, pp. 321-332, 2002.
- [17] H. N. K. Al-behadili, et al., "Rule pruning techniques in the ant-miner classification algorithm and its variants: a review," *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, pp. 78-84, 2018.
- [18] R. Saian and K. R. Ku-Mahamud, "Ant colony optimization for rule induction with simulated annealing for terms selection," *Proc. - 2012 14th Int. Conf. Model. Simulation, UKSim 2012*, pp. 33-38, 2012.
- [19] F. E. B. Otero, et al., "A new sequential covering strategy for inducing classification rules with ant colony algorithms," *IEEE Trans. Evol. Comput.*, vol. 17, pp. 64-76, 2013.
- [20] S. Tripathy, et al., "MTACO-Miner : Modified Threshold Ant Colony Optimization Miner for Classification Rule Mining," *Emerging Reserch in Computing, Information, Communication and Application*, pp. 1-6, 2013.
- [21] A. Chan and A. Freitas, "A new classification-rule pruning procedure for an Ant Colony Algorithm," *International*

- Conference on Artificial Evolution (Evolution Artificielle)*, vol. 3871, pp. 25-36, 2006.
- [22] H. N. K. Al-behadili, et al., "Ant colony optimization algorithm for rule-based classification: Issues and potential solutions," *J. Theor. Appl. Inf. Technol.*, vol. 96, pp. 7139-7150, 2018.
- [23] B. Liu, et al., "Classification rule discovery with ant colony optimization algorithm," *IEEE Computational Intelligence Bulletin*, vol. 7435, pp. 678-687, 2004.
- [24] Liu, et al., "Density-Based Heuristic for Rule Discovery With Ant-Miner," *6th Australas. Jt. Work. Intell. Evol. Syst.*, pp. 1-6, 2002.
- [25] B. Fakhar, "A Novel Method for Extracting Classification Rules based on Ant-Miner," *J. Math. Comput. Sci.*, vol. 8, pp. 377-386, 2014.
- [26] J. Wahid and H. F. A. Al-Mazini, "Classification of Cervical Cancer Using Ant-Miner for Medical Expertise Knowledge Management," *Knowledge Management International Conference (KMICE)*, 2018.
- [27] H. N. K. Al-behadili, "Intelligent Hypothermia Care System using Ant Colony Optimization for Rules Prediction," *J. Univ. Babylon*, vol. 26, pp. 47-56, 2018.
- [28] I. M. Anwar, et al., "ADR-Miner: An ant-based data reduction algorithm for classification," *2015 IEEE Congr. Evol. Comput. CEC 2015 - Proc.*, pp. 515-521, 2015.
- [29] R. H. A. Alsagheer, et al., "Popular Decision Tree Algorithms of Data Mining Techniques: A Review," *Int. J. Comput. Sci. Mob. Comput.*, vol. 6, pp. 133-142, 2017.
- [30] D. Dua and T. Karra, "UCI Machine Learning Repository," *Irvine, CA: University of California, School of Information and Computer Science*, 2017. Available: <http://archive.ics.uci.edu/ml>.
- [31] R. Saian and K. R. Ku-Mahamud, "Hybrid Ant Colony Optimization and Simulated Annealing for Rule Induction," *2011 UKSim 5th Eur. Symp. Comput. Model. Simul.*, pp. 70-75, 2011.
- [32] F. Otero, et al., "Handling continuous attributes in ant colony classification algorithms," *2009 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2009 - Proceedings*, pp. 225-231, 2009.
- [33] F. Otero, et al., "cAnt-miner: An ant colony classification algorithm to cope with continuous attributes," *International Conference on Ant Colony Optimization and Swarm Intelligence*, pp. 48-59, 2008.
- [34] K. Thangavel and P. Jaganathan, "Rule Mining Algorithm with a New Ant Colony Optimization Algorithm," *Int. Conf. Comput. Intell. Multimed. Appl. ICCIMA 2007*, vol. 1, pp. 561-563, 2007.
- [35] R. Robu, et al., "A study on Ant Miner parameters," *6th International Conference on Information, Intelligence, Systems and Applications*, 2016.
- [36] D. Martens, et al., "Classification with ant colony optimization," *IEEE Trans. Evol. Comput.*, vol. 11, pp. 651-665, 2007.
- [37] R. P. Datta and S. Saha, "Applying rule-based classification techniques to medical databases : an empirical study," *Int. J. Bus. Intell. Syst. Eng.*, vol. 1, pp. 32-48, 2016.

BIOGRAPHIES OF AUTHORS



Hayder Naser Khraibet: received the Bachelor in Computer Sciences from Shatt Al-Arab University College, Iraq in 2008. His Master's degree in Information Technology from University Utara Malaysia in 2011. Since 2016, he has been working toward the Ph.D. degree in Information Technology. His research is mainly focused on development of rule-based classification techniques based on Ant Colony Optimization.



Ku Ruhana Ku-Mahamud: She holds a Bachelor in Mathematical Sciences and a Masters degree in Computing, both from Bradford University, United Kingdom in 1983 and 1986 respectively. Her PhD in Computer Science was obtained from Universiti Pertanian Malaysia in 1994. As an academic, her research interests include ant colony optimization, pattern classification and vehicle routing problem.



Rafid Sagban holds a Bachelor in Computer Sciences from University of Babylon, Iraq in 1999 and Higher Diploma in Data Security from Iraqi Commission for Computers and Informatics in 2001. His Master's degree and Ph.D in Information Technology are both from Universiti Utara Malaysia in 2015. Rafid's research and development experience includes over 13 years in the Academia and Industry. He works in a multi-disciplinary environment involving computational intelligence, swarm intelligence algorithms, business intelligence and web design and development.