Zaria Journal of Electrical Engineering Technology, Department of Electrical Engineering, Ahmadu Bello University, Zaria – Nigeria. Vol. 10 No. 2, September 2021. ISSN: 0261 – 1570.

REVIEW ON METAHEURISTIC-BASED DECISION TREE INDUCTION

Ismaila Mahmud[1], Abubakar Abdulkarim[2], Sulaiman Haruna Sulaiman[3], Umar Musa[4]
Department of Electrical Engineering, Ahmadu Bello University,Zaria – Nigeria.
[*1]imahmud@abu.edu.ng, [2]abdulkarima@abu.edu.ng, [3]shsulaiman@abu.edu.ng,[4]umusa@abu.edu.ng

Keywords: –
Decision tree, swarm intelligence, metaheuristics.

Article History: –
Received: March, 2021.
Reviewed: June, 2021
Accepted: August, 2021
Published: September, 2021.

ABSTRACT

Decision tree is one of the well-known machine learning algorithms which remain popular due its simple implementation and ease of understanding. Existing techniques used in inducing decision tree have shown that they suffer data overfitting which leads to producing small size of decision tree model. To address the drawback and improve the performance of the algorithm, hybridization of metaheuristic algorithm and decision tree are done by many researchers. In this article, we undertake a review on the hybrid done with swarm intelligence on decision tree based on ant-miner and other different modifications. Different application domains executed using metaheuristic-based decision tree are described. Finally, we address some challenges and future research directions.

1. INTRODUCTION

In decision tree, patterns are discovered in the dataset through dividing instances into classes determined by set of tests. The greedy search heuristic algorithms are normally used in the induction of the decision tree. The basic concept of greedy heuristic is usual the divide and conquer where the algorithm start from the top to bottom fashion. In every iteration, the training dataset is partition according to the test on a discrete function of the input variable. The partition continues until stopping criteria are reached or the splitting measure is exhausted. Like all classification paradigms, a poor performance is obtained when the decision tree model is complex. This leads to many efforts in producing small size of decision tree model [1].

Several decision tree techniques were developed such as iterative dichotomiser 3 (ID3) or C4.5 [2], classification and regression tree (CART) [3], scalable parallelizable induction of decision tree (SPRINT) [4], supervised learning in quest (SLIQ) [5]. The techniques used in inducing some decision tree were being modified to improve on the drawbacks of the earlier techniques. Rainforest proposed by Gehrke et al [6] tend to have scalable and fast algorithm in constructing decision tree and making use of the available main memory [1] which falls under the ensemble decision tree. Metaheuristic techniques such as evolutionary algorithms and swarm intelligence are used in form of hybrid with the greedy search heuristic to improve optimal solution of a decision tree induction.

ID3 algorithm was proposed by Quinlan [7]. It uses information gain to partition the tree and has been considered to be a straightforward decision tree algorithm. The tree growth stops when the instances of the target feature are exhausted or when the threshold of the information gain is reached. Pruning techniques are not applied to ID3 and it can neither handle numeric attributes nor missing values [8].

C4.5 was also proposed by Quinlan as an improvement to ID3 in [2]. It uses different splitting criteria known as gain ratio. The tree growth is stop when a threshold is reached for splitting the number of instances and pruning is done using error-based pruning. As an improvement to ID3, numeric attributes are used in C4.5. It also has the ability of inducing a training set that has missing values through the use of the right gain ratio criteria.

Breiman et al [3] developed CART which stands for classification and regression trees. It has the characteristic of being a binary tree due to its structure, where each internal node has exactly two outgoing edges. Twoing criteria are used for the splitting and cost-complexity pruning is used for the pruning of developed tree.

CART has the ability to generate regression tree. In regression tree, the leaves predict a real number and not a class label. In the regression case, the split occurs when there is minimum squared error prediction (the least-squared deviation). Each leaf is predicted based on the weighted mean for node.

Several new method came up in the early seventies for generating decision tree, among them are: automatic interaction detection (AID) [9], theta automatic interaction detection (THAID) [10] and Chi-square automatic interaction detection (CHAID) [11]. The Chi-square automatic interaction detection (CHAID) algorithm was initially meant to work on nominal attributes only. It works using statistical test to determine the least significant difference value of each input attribute with respect to the target attribute. Depending on the type of target attribute used, the statistical test to be use is determined. For a continuous attribute, the F-test is used. Nominal attribute uses a Pearson chi-squared test and an ordinal attribute uses a likelihood-ratio test. CHAID treats missing values as a single valid category and does not perform pruning.

1. Oblivious decision trees: Oblivious trees have nodes at the same level testing the same attribute. Even with this drawback, it appears to be effective in feature selection as demonstrated in [12, 13]. While Langley and Sage [14] use the same means for backward selection. In [15], the authors showed that oblivious decision trees can be converted to a decision table. An algorithm that is based on information theory, called information fuzzy network was developed for constructing oblivious decision tree in [16].

2. METAHEURISTIC ALGORITHMS

Real-world problems come with various kinds of constraints that are non-linear, complex in nature and also having variables that are almost dependent of each other. The need to have a technique that could be used in solving real time optimization problems is paramount. This is where metaheuristic algorithms come in handy. These optimization methods provide good solutions within an acceptable time [17]. Another problem that requires applying optimization methods is when dealing with large number of dimension features (decision tree inclusive). The greedy search approach being used by most decision tree techniques suffered data over-fitting. This is so due to its recursive nature in partitioning the datasets that may result to very small datasets for attribute selection in the decision tree induction [18]. Metaheuristic algorithms could be used to handle such problems. They adapt their procedure through mimicking social or biological behavior [19]. Metaheuristics are algorithms in which natural learning concept are employed to obtain an almost optimal solutions through the exploitation and exploration of the search space. The search method is either a local or global search depending on the usage of the exploitation and exploration of the search space. In the local search, they make use of exploitation where optimal solution is found within the neighboring solutions and ends when there is no change in the optimal solution. While in the global search, both exploitation and exploration are utilized to obtain the optimal solution and its end when the user defined criteria is reached [20].

Metaheuristic algorithm could be divided into four categories based on the type of inspiration they adopt: swarm intelligence (SI) based, bio-inspired based, physics/chemistry based and others [21]. In SI-based, insects such as ants, bees, termites or flocks of fish and bird are the inspired agents used in the algorithms. Bio-inspired algorithms that mimic biological behavior such as genetic evolution in genetic algorithm, the pollination characteristic of flowing plants in flower pollination algorithm and the concept of shoots and roots growth of tree used in tree physiology algorithm. The physics and chemistry-based mimic the fundamental physics and chemistry laws such as electric charges, gravity etc. While, algorithms developed from inspiration away from nature fall into the other category. These algorithms are developed using some characteristic from its source such social, emotional etc.

Metaheuristic have been used in the induction of decision tree in the last two decades, not much reviews were carried out in its regard. Though, a survey on the evolutionary algorithms for decision tree induction was done by Barros et al [18]. This review will be for the swarm intelligence-based metaheuristic induction of decision tree.

2.1 METAHEURISTIC BASED DECISION TREE INDUCTION

In this section, the swarm intelligence algorithms type of metaheuristic algorithms that have been used in the induction of decision tree will be discussed here, as the evolutionary algorithms-based decision tree is thoroughly reviewed in [18].

2.1.1 Swarm Intelligence Algorithms

Several research on solving data-mining problem through the application of swarm intelligence have been conducted. Swarm Intelligence is a phenomenon where a set of self-organized agents conduct an activity in a coordinated manner. The swarms have the ability conduct a task collectively (global) which is often better than the one done by individual agent. Swarm intelligence have been thoroughly study in [22, 23]. There are several real-world problems that have been executed using swarm intelligence. In 1989, Beni and Wang coined the term “Swarm Intelligence” [24]. Summarily, swarm intelligence [22, 23] is the ability of a system to have unsophisticated agents behavior, to interact locally with their environment that will result to a coherent functional global pattern. In a swarm, its behaviors are distributed among its agents with little environmental model or perception but with strong bond to its level of adaptation and reaction [25].

One of the prominent SI-based decision tree algorithms is the Ant-Miner algorithm. It was proposed by Parpinelli et al [26] in 2002 as an ant colony optimization algorithm used in discovering rules. The algorithm shows improved performance from the results obtained as compared with other discovering rules algorithms such as CN2 [27] and C4.5 [2], and the authors continue to work on the algorithm as published in these articles [26, 27]. The Ant-Miner works on the principle of computing heuristic value based on entropy measure. For a classification rule to be established for a given dataset, sequential approach that determines the discovering rule for almost all the training sets, with each classification rule in the form: If <term1ANDterm2AND…> Then <Class>. The following stages are followed in executing Ant-Miner.

a. Initialization of Pheromone

The amount of pheromone released along the path at initial stage is inversely proportional to the number of all the attributes as given by

$$\tau_{ij}(t=0) = \frac{1}{\sum_{i=1}^{a} b_i} \tag{1}$$

Where b_i is the number of values of attribute *i*domain and *a* is the total number of attributes.

b. Rule Establishment

For every Ant-Miner rule, it has an antecedent and consequence (predictive class) condition. The condition deals with attribute-operator-value tuples. When term$_{ij}$ is considered to be the rule condition for $A_i = V_{ij}$, where A_i is the *i*th attribute and V_{ij} is the *j*th value in the domain of A_i. Then the probability that term$_{ij}$ is included to the current partial rule is given by

$$P_{ij} = \frac{\eta_{ij}.\tau_{ij}(t)}{\sum_{i=1}^{a} x_i \cdot \sum_{j=1}^{b_i}(\eta_{ij}.\tau_{ij}(t))} \tag{2}$$

Where η_{ij} is a problem dependent heuristic value for term$_{ij}$,$\tau_{ij}$is the amount of pheromone available at the time t, on the link between attribute.

c. Heuristic Value

The heuristic value in Ant-Miner is the information theoretic measure that estimate the quality of the term$_{ij}$ and determine if its able to the improve the predictive accuracy of the rule. It is given by

$$\eta_{ij} = \frac{\log_2(k) - Info\,T_{ij}}{\sum_{i=1}^{a}\sum_{j=1}^{b_i}(\log_2(k) - Info\,T_{ij})} \tag{3}$$

$$Info\,T_{ij} = -\sum_{w=1}^{k}\left[\frac{FreqT_{ij}^{w}}{|T_{ij}|}\right] * \log_2\left[\frac{FreqT_{ij}^{w}}{|T_{ij}|}\right] \tag{4}$$

Where k stands for the number of classes, $|T_{ij}|$ stands for the entire number of cases in the partition (where attribute A_i has value V_{ij}), $FreqT^{w}_{ij}$ stands for the number of cases in the partition with class w. The term is less likely to be chosen when the value of Info T_{ij} is high.

d. Rule pruning

In the pruning procedure, the term that its removal will provide maximum increase in the quality of the rule is removed. The quality of the rule is given by

$$Q = \frac{TP}{TP+FN} \cdot \frac{TN}{FP+TN} \tag{5}$$

Where TP is the cases treated by the rule and has the expected class, FP is the cases treated by the rule and does not have the expected class, FN is the cases not treated by the rule and has the expected class, TN is the cases not treated by the rule and does have the expected class.

e. Pheromone Update rule

At the end of rule establishment, the pheromone is update as follows:

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \tau_{ij}(t).Q \qquad (6)$$

A lot of modification and extension have been made to improve Ant-Miner performance. Among the modified or extended Ant-Miner algorithms are Ant-Miner I [28], ACO-Miner [29], Ant-Miner 3 [30] and Ant-Miner + [31, 32]. TACO-Miner is another improved Ant-Miner proposed by Thangaveli and Jaganathan[33]. The authors introduced changes that ensure improved classification accuracy and with simple construction rules. The trend continues with Salama et al. in 2011 developing Multi-pheromone Ant-Miner [34, 35], where they proposed a class-based pheromone. The rules established for particular classes during the algorithm run are based on different pheromone trail matrix and the decision class is being assigned to the ant agent [36]. In [37], Boryczka and Kozakproposed another ACO-based decision tree called ACDT which build a binary decision tree using ant colony algorithm. The decision tree is constructed using Twoing criterion (similar to that in CART algorithm [3]) as the heuristic function and pheromone values which is the quality of connection between a parent and child nodes. The authors modified the algorithm to ensure its handles continuous attributes as presented in [38]. Otero et al [39] proposed another version of ant colony induced decision tree which they termed Ant-Tree-Miner. The algorithm is constructed using pheromone value and heuristic function which determine the attributes to be selected as decision nodes through a top-down approach. The algorithm showed good performance in decreasing the size of the tree's nodes and an improved predictive accuracy. Boryczka and Kozak continued their work on ACDT in [25] where they further investigated the significance of the turning parameters of ACDT algorithm in quality evaluation for the decision tree induction. Sousa et al [40] proposes the use of the Particle Swarm Optimiser as a new tool for the induction of decision tree. The results obtained were compared with evolutionary algorithms induced decision tree and J48 (a java implementation of C4.5). They also induced the decision tree with an improve version of PSO and compared it with their earlier results. The disadvantage of this algorithm is, it works for discrete attributes only [40]. A hybrid of PSO and ACO based decision tree inductions were proposed in [41, 42] by Holden and Freitas. The algorithms have the ability to handle tasks with all types of data (continuous and nominal) without need for transformation. In [43], they propose an artificial bee colony data miner where the algorithm was able to work on both continuous and discrete attributes. The work was compared to a PSO induced decision tree and C4.5 and had performed better than the two in terms of accuracy and stability. Chen et al [44] proposed a novel method in gene selection using a PSO algorithm combined with a C4.5 classifier. Their PSOC4.5 method was compared with other well-known classifiers using eleven cancer datasets and performed better than most. As with many PSO, there is need to further investigate the parameters tuning in order to avoid local optimal trapping. In [45], a bat algorithm is used in inducting decision tree and the authors referred the algorithm as Bat Tree Constructor. The algorithm was compared with Ant Miner, CART and random forest where it performed better mostly.

3. APPLICATION OF METAHEURISTICS BASED DECISION TREE

As most of the work done on metaheuristic-based decision tree induction are validated on a public benchmark dataset for comparison reason, there are some that are specifically applied to a particular application domain. We highlight some in this section.

In [46], the authors proposed a modified ACDT which is applied to email folder problem. The pheromone trail is strengthened for the classifiers, which assigned new email messages to folders based on the previous decision made by the recipients. This implied that the messages are addressed to multiple persons and place into the targeted categories [19]. The authors extend their work in [47, 48] by hybridization of the algorithm with social network analysis. The modified algorithm showed significant improvement after creating the social network, which allows communication between the users to be analyzed. The last version of the algorithm provides better labels assignment to the analyzed email messages.

Finance is another area where ACDT found to be attractive in predicting changes in forex market. In [49], ACDT was used as association problem rather than the classical classification problem to tackle the discrepancies found between the technical analysis indicators used in forex market. The main task of the algorithm is to identify subset of the indicators that allow further effective analysis of the market by professional expert.

In [44], a PSOC4.5 hybrid was used as a novel method for gene selection. The algorithm was tested on eleven cancer datasets and compared with other well-known classifiers. The algorithms achieved better accuracy compared with all the other methods. Table 1 gives the summary of the applications where metaheuristics-based decision trees are being used.

Table 1: Summary of the applications of the Methaheuristics-bade methods

Algorithms	Application area	References
PSO-based decision tree	Medicine	[44]
ACO-based decision tree	Finance	[49]
ACO-based decision tree	Software Engineering	[46], [47], [48],[50]
ACO-based decision tree	Medicine	[51], [52], [53], [54]

4. CHALLENGES AND RESEARCH DIRECTIONS

Despite the successes made by metaheuristic algorithms in inducting decision tree, there is still remain some gaps that need to be bridge so as to attain a level of maturity as compared to similar established algorithms in the same field of research. This section also provides some of the related but potential areas of research that may build future literature. The some of the challenges often encountered by metaheuristic-based decision tree algorithms are as follows:

1. Ant-Miner has a drawback in the way its update the pheromone trail. Whenever the pheromone is laid, it has its values normalized. Causing decrease of the heuristic function value and not marking the time which the pheromone values were laid. As such the full potential of ant colony algorithms is not utilized.

2. Another disadvantage of Ant-Miner is it does not work well with a continuous attributes dataset being typical ant algorithm in its classical version, where its population is a single ant agent [19].

3. Particle swarm optimization has shown promising improvement of accuracy of decision tree as exhibited in [44]. They suffer, the inherent drawback associated to PSO of local optima trapping problem and dealing with many tuning parameters which increase its complexity.

Future research should investigate the use of new or improved metaheuristic algorithms in inducing decision tree in order to optimize its performance. Other application areas where metaheuristic-based decision tree have not been used such image processes, predictive maintenance, astronomy could explore its potential advantages, as it has proven to be capable in handling other applications.

5. CONCLUSION

In this article, we have presented a comprehensive review on metaheuristic-based decision tree induction. In the algorithms, swarm intelligence techniques were hybridized with decision tree to improve its performance. There are drawbacks that were discovered in some of the algorithms such as local optima trapping, complex operators, not working on all type of datasets and many parameters tuning problem. Therefore, using new or improved metaheuristic algorithms could tackle the drawbacks and is a research field to explore.

REFERENCE

[1] S. B. Kotsiantis, “Decision trees: a recent overview,” *Artificial Intelligence Review*, vol. 39, no. 4, pp. 261-283, 2013.

[2] J. Quinlan, “C4. 5: Programs for machine learning Morgan Kaufmann San Francisco,” *CA, USA*, 1993.

[3] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*: CRC press, 1984.

[4] J. Shafer, R. Agrawal, and M. Mehta, "SPRINT: A scalable parallel classifier for data mining." pp. 544-555.

[5] M. Mehta, R. Agrawal, and J. Rissanen, "SLIQ: A fast scalable classifier for data mining." pp. 18-32.

[6] J. Gehrke, R. Ramakrishnan, and V. Ganti, "RainForest—a framework for fast decision tree construction of large datasets," *Data Mining and Knowledge Discovery*, vol. 4, no. 2-3, pp. 127-162, 2000.

[7] J. Quinlan, "Induction of Decision Trees. Mach. Learn," 1986.

[8] L. Rokach, and O. Maimon, "Top-down induction of decision trees classifiers-a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 35, no. 4, pp. 476-487, 2005.

[9] J. A. Sonquist, E. L. Baker, and J. N. Morgan, *Searching for structure*: Survey Research Center, Inst. for Social Research, University of Michigan, 1971.

[10] J. N. Morgan, and R. C. Messenger, "THAID, a sequential analysis program for the analysis of nominal scale dependent variables," 1973.

[11] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 29, no. 2, pp. 119-127, 1980.

[12] H. Almuallim, and T. G. Dietterich, "Learning boolean concepts in the presence of many irrelevant features," *Artificial intelligence*, vol. 69, no. 1-2, pp. 279-305, 1994.

[13] J. C. Schlimmer, "Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning." pp. 284-290.

[14] P. Langley, and S. Sage, "Oblivious decision trees and abstract cases." pp. 113-117.

[15] R. Kohavi, and D. Sommerfield, "Targeting Business Users with Decision Table Classifiers." pp. 249-253.

[16] M. Last, O. Maimon, and E. Minkov, "Improving stability of decision trees," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 02, pp. 145-159, 2002.

[17] A. Gogna, and A. Tayal, "Metaheuristics: review and application," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 25, no. 4, pp. 503-526, 2013.

[18] R. C. Barros, M. P. Basgalupp, A. C. De Carvalho, and A. A. Freitas, "A survey of evolutionary algorithms for decision-tree induction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 3, pp. 291-312, 2011.

[19] J. Kozak, *Decision tree and ensemble learning based on ant colony optimization*: Springer, 2019.

[20] Z. Beheshti, and S. M. H. Shamsuddin, "A review of population-based meta-heuristic algorithms," *Int. J. Adv. Soft Comput. Appl*, vol. 5, no. 1, pp. 1-35, 2013.

[21] I. Fister Jr, X.-S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization," *arXiv preprint arXiv:1307.4186*, 2013.

[22] E. Bonabeau, M. Dorigo, D. d. R. D. F. Marco, G. Theraulaz, and G. Théraulaz, *Swarm intelligence: from natural to artificial systems*: Oxford university press, 1999.

[23] J. Kennedy, R. Eberhart, and Y. Shi, "Swarm Intelligence, 1st Morgan Kaufmann," *San Francisco, USA*, 2001.

[24] G. Beni, and J. Wang, "Swarm intelligence in cellular robotic systems," *Robots and biological systems: towards a new bionics?*, pp. 703-712: Springer, 1993.

[25] U. Boryczka, and J. Kozak, "Enhancing the effectiveness of ant colony decision tree algorithms by co-learning," *Applied Soft Computing*, vol. 30, pp. 166-178, 2015.

[26] R. S. Parpinelli, H. S. Lopes, and A. A. Freitas, "An ant colony algorithm for classification rule discovery," *Data mining: A heuristic approach*, pp. 191-208: IGI Global, 2002.

[27] R. S. Parpinelli, H. S. Lopes, and A. A. Freitas, "Data mining with an ant colony optimization algorithm," *IEEE transactions on evolutionary computation*, vol. 6, no. 4, pp. 321-332, 2002.

[28] W. Jiang, Y. Xu, and Y. Xu, "A novel data mining method based on ant colony algorithm." pp. 284-291.

[29] Z. Wang, and B. Feng, "Classification rule mining with an improved ant colony algorithm." pp. 357-367.

[30] B. Liu, H. A. Abbas, and B. McKay, "Classification rule discovery with ant colony optimization." pp. 83-88.

[31] D. Martens, M. De Backer, R. Haesen, B. Baesens, and T. Holvoet, "Ants constructing

rule-based classifiers," *Swarm intelligence in data mining*, pp. 21-43: Springer, 2006.

[32] D. Martens, M. De Backer, R. Haesen, J. Vanthienen, M. Snoeck, and B. Baesens, "Classification with ant colony optimization," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 5, pp. 651-665, 2007.

[33] K. Thangavel, and P. Jaganathan, "Rule mining algorithm with a new ant colony optimization algorithm." pp. 135-140.

[34] K. M. Salama, and A. M. Abdelbar, "Extensions to the Ant-Miner classification rule discovery algorithm." pp. 167-178.

[35] K. M. Salama, A. M. Abdelbar, and A. A. Freitas, "Multiple pheromone types and other extensions to the Ant-Miner classification rule discovery algorithm," *Swarm Intelligence*, vol. 5, no. 3-4, pp. 149-182, 2011.

[36] J. Kozak, "Evolutionary Computing Techniques in Data Mining," *Decision Tree and Ensemble Learning Based on Ant Colony Optimization*, pp. 29-44: Springer, 2019.

[37] U. Boryczka, and J. Kozak, "Ant colony decision trees–a new method for constructing decision trees based on ant colony optimization." pp. 373-382.

[38] U. Boryczka, and J. Kozak, "An adaptive discretization in the ACDT algorithm for continuous attributes." pp. 475-484.

[39] F. E. Otero, A. A. Freitas, and C. G. Johnson, "Inducing decision trees with an ant colony optimization algorithm," *Applied Soft Computing*, vol. 12, no. 11, pp. 3615-3626, 2012.

[40] T. Sousa, A. Silva, and A. Neves, "Particle swarm based data mining algorithms for classification tasks," *Parallel computing*, vol. 30, no. 5-6, pp. 767-783, 2004.

[41] N. Holden, and A. A. Freitas, "A hybrid PSO/ACO algorithm for discovering classification rules in data mining," *Journal of Artificial evolution and Applications*, vol. 2008, 2008.

[42] N. P. Holden, and A. A. Freitas, "A hybrid PSO/ACO algorithm for classification." pp. 2745-2750.

[43] M. Çelik, D. Karaboğa, and F. Köylü, "Artificial bee colony data miner (abc-miner)." pp. 96-100.

[44] K.-H. Chen, K.-J. Wang, K.-M. Wang, and M.-A. Angelia, "Applying particle swarm optimization-based decision tree classifier for cancer classification on gene expression data," *Applied Soft Computing*, vol. 24, pp. 773-780, 2014.

[45] I. Bida, and S. Aouat, "A New Approach Based on Bat Algorithm for Inducing Optimal Decision Trees Classifiers." pp. 631-640.

[46] U. Boryczka, B. Probierz, and J. Kozak, "An ant colony optimization algorithm for an automatic categorization of emails." pp. 583-592.

[47] U. Boryczka, B. Probierz, and J. Kozak, "A new algorithm to categorize e-mail messages to folders with social networks analysis," *Computational Collective Intelligence*, pp. 89-98: Springer, 2015.

[48] U. Boryczka, B. Probierz, and J. Kozak, "Automatic categorization of email into folders by ant colony decision tree and social networks," *Intelligent Decision Technologies 2016*, pp. 71-81: Springer, 2016.

[49] J. Kozak, and P. Juszczuk, "Association ACDT as a tool for discovering the financial data rules." pp. 241-246.

[50] N. Holden, and A. A. Freitas, "Web page classification with an ant colony algorithm." pp. 1092-1102.

[51] M. Durgadevi, and R. Kalpana, "Medical distress prediction based on classification rule discovery using ant-miner algorithm." pp. 88-92.

[52] R. Kuo, and C. Shih, "Association rule mining through the ant colony system for National Health Insurance Research Database in Taiwan," *Computers & Mathematics with Applications*, vol. 54, no. 11-12, pp. 1303-1318, 2007.

[53] K. Robbins, W. Zhang, J. Bertrand, and R. Rekaya, "The ant colony algorithm for feature selection in high-dimension gene expression data for disease classification," *Mathematical Medicine and Biology: a Journal of the IMA*, vol. 24, no. 4, pp. 413-426, 2007.

[54] G. Schaefer, "Gene expression analysis based on ant colony optimisation classification," *International Journal of Rough Sets and Data Analysis (IJRSDA)*, vol. 3, no. 3, pp. 51-59, 2016.