aurum

AURUM MÜHENDİSLİK SİSTEMLERİ VE MİMARLIK DERGİSİ
AURUM JOURNAL OF ENGINEERING SYSTEMS AND ARCHITECTURE

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

USING FEATURE SELECTION AND ACO ALGORITHM FOR OPTIMIZING SMART CLASSROOM

Dhuha Abdulameer Abd Ali Abd Ali ABD ALI[1*], Hasan Hüseyin BALIK [2],
[1]Department of Information Technology, Altınbaş University, Türkiye, 203721013@ogr.altinbas.edu.tr
(https://orcid.org/0000-0002-4184-7633)
[2]Department of Computer Engineering, Istanbul Aydin University, Türkiye, balik@aydin.edu.tr
(https://orcid.org/ 0000-0003-3022-100X)

Received: 22.09.2022 Research Article
Accepted: 01.12.2022
*Corresponding author

Abstract
The smart education had a huge impact on learning and teaching, so it must be effective and highly efficient. An efficient smart campus or smart classroom will make the learning more and more easily, the students could learn and give the best activities. In addition, the teachers will be able to make right decisions. To achieve this goal, the smart classroom's conditions must be ideal. Since ACO (ant colony optimization algorithm) is a meta heuristic algorithm, in this paper, it is found that ACO, in conjunction with a machine learning classifier, was an effective method used in feature selection for selecting best features from an intelligent campus data set to create an environment that is conducive to academic success and student learning, such as (humidity and temperature), lighting and sound pressure levels, wind direction, and raw rainfall amounts (among other variables). In this contribution to get the most accurate results, the ACO algorithm was combined with a logistic regression classifier that was used to select the best features. The accuracy of the proposed model was 0.927438624 and 0.898268071 for two sets of data back to the School of Design and Environment 4, building located at the National University of Singapore.
Keywords: Smart classroom, feature selection, ACO algorithm, logistic regression , genetic algorithm

AKILLI SINIFI OPTİMİZE ETMEK İÇİN ÖZELLİK SEÇİMİ VE ACO ALGORITMASI

Öz
Akıllı eğitimin öğrenme ve öğretme üzerinde büyük bir etkisi vardır, bu nedenle etkili ve yüksek verimli olmalıdır. Verimli bir akıllı kampüs veya akıllı sınıf, öğrenmeyi daha kolay hale getirecek, öğrenciler öğrenebilecek ve en iyi etkinlikleri verebileceklerdir. Ayrıca öğretmenler de doğru kararlar verebileceklerdir. Bu hedefe ulaşmak için akıllı sınıfın koşullarının ideal olması gerekir. ACO (karınca kolonisi optimizasyon algoritması) bir meta sezgisel algoritma olduğundan, bu makalede, ACO'nun bir makine öğrenimi sınıflandırıcısı ile birlikte, akademik başarıya ve öğrenci öğrenimine elverişli bir ortam oluşturmak için akıllı bir kampüs veri setinden (nem ve sıcaklık), aydınlatma ve ses basıncı seviyeleri, rüzgar yönü ve ham yağış miktarları (diğer değişkenlerin yanı sıra) gibi en iyi özellikleri seçmek için özellik seçiminde kullanılan etkili bir yöntem olduğu bulunmuştur. En doğru sonuçları elde etmek için bu katkıda, ACO algoritması en iyi özellikleri seçmek için kullanılan bir lojistik regresyon sınıflandırıcısı ile birleştirilmiştir. Önerilen modelin doğruluğu, Singapur Ulusal Üniversitesi'nde bulunan Tasarım ve Çevre Okulu 4 binasına ait iki veri seti için 0,927438624 ve 0,898268071 olmuştur.
Anahtar Kelimeler: Akıllı sınıf, özellik seçimi, ACO algoritması, lojistik regresyon, genetik algoritma

1. Introduction

In keeping with the flow of history, a wide range of applications have emerged for smart technology including smart environments, agriculture, health care, cities, and education. In smart educational field at all levels, from kindergarten to high school, digital technologies have had an enormous impact on the education sector, and many educational applications have been developed to make improvement the teaching or the learning experience for students. Innovative educational institutions are embracing the concept of a "smart campus," which integrates smart technologies with physical infrastructure to enhance student services, faculty decision-making, and overall campus sustainability (Min-Allah et al., 2020). Distance education systems or intelligent environments with a wide range of hardware and software modules have been referred to as "smart campuses" (Gligorić et al., 2012). Connect the dots between class and teacher, assist the instructor in teaching more effectively, improve the quality of life in the classroom so that teaching and learning can flourish, and so on are some of best practices for creating a high-tech school setting (Saini & Goel, 2019). The data set of smart campus that had been collected need to the treatment for getting the best features by using the feature selection (Shardlow, 2016). One of the most crucial elements that can affect the classification accuracy rate is feature selection. If the dataset has numerous features, the dimension of the room will be huge, unclean, and inferior. the degree of categorization accuracy. a strong and effective utilizing feature selection might help you get rid of loud, irrelevant, and duplicate data (Venkatesh & Anuradha, 2019). The dimensionality of data has a significant impact on the performance of classification algorithms (Kumar & Minz, 2014). The performance of a learning method (such as a classifier) degrades as the number of characteristics in the dataset grows larger. By filtering out superfluous, irrelevant, and noisy features, Feature Selection (FS) is a vital preprocessing step that can boost the accuracy and speed of classification algorithms (Li et al., 2017a). Along with improving the efficiency of classification algorithms, FS also helps to cut down on the amount of time spent in front of a computer. It is important to think about how to search for the optimal feature subset and how to determine the goodness of a subset while developing an FS algorithm (Li et al., 2017b). There are two primary models that have been frequently utilized in the FS literature to evaluate the quality of a feature subset, and these are the filter and wrapper models. Feature dependencies and inter feature connections play a significant role in filter evaluation. The filter method (Kashef & Nezamabadi-pour, 2015) is represented by techniques like the F-score, and Principal Component Analysis (PCA). If a learning algorithm (like classification) is to be used in the wrapper approach's evaluation, its efficacy must be considered. Wrapper approaches are better suited when the performance of a specific learning algorithm is the aim, but their optimization process is slower than filter approaches because a learning algorithm is used to evaluate each feature subset.

Finding the optimal feature subset is the second component of FS techniques. Exhaustive (complete), random, and heuristic search techniques are the three primary options (Khaire & Dhanalakshmi, 2019). On the one hand, a comprehensive search approach will typically locate all conceivable feature subsets, rank them in order of quality, and finally choose the most advantageous one. It's easy to see that this approach is difficult and time-consuming. In a strict sense, we should construct and assess 2N subsets if we have a dataset with N features. Therefore, this approach is not viable when working with massive datasets. However, random search is another option for exploring the features subsets, albeit it may exhaust just as many possibilities as the exhaustive search before hitting the mark. Heuristic search is the third approach that can be taken when dealing with the FS problem. This method falls somewhere in the middle of full and random searches. The search begins with a randomly generated solution and is then directed towards the desired subset by means of a heuristic value. Heuristic algorithms are designed to discover a (near-)optimal solution quickly (Joseph Manoj et al., 2019). There are currently far too many proposed approaches to FS, but two that have garnered significant interest include population-based optimization algorithms like the genetic algorithm (GA)-based method and the ant colony optimization (ACO)-based method. These techniques aim to improve upon earlier iterations by using what was learned. Natural selection serves as the basis for genetic algorithms, which are optimization methods. For navigation, it used genetically inspired procedures (Mafarja et al., 2018)). Dorigo and Caro, in the early 1990s, presented a meta-heuristic optimization method inspired by ant behavior (1999). Swarm intelligence, of which ACO is a subset, is a relatively recent advancement in the realm of artificial intelligence (SI). Computing swarm intelligence (CSI) is a formal term for algorithmic models of the problem-solving behaviour that emerges from the interaction of cooperative agents (Kashef & Nezamabadi-pour, 2015). Swarm intelligence, in its more formal definition, is the quality of a system in which coherent functional global patterns arise because of the aggregate behaviours of unsophisticated agents interacting locally with their environment (Akhtar, 2019). The cooperative effort of a colony is the primary explanation for the intelligent behavior of social insects like ants and bees, where an individual can only execute simple tasks. The cooperative nature of ant colonies served as inspiration for the ACO algorithm. Ants are unable to see, but they can navigate to and from their nest using chemical molecules called pheromone that they leave behind as they travel (Al Salami, 2009). There are wide range of measures including sound pressure level, relative humidity, air temperature, illuminance, supply air pressure, supply air temperature, and many others had been provided in smart classroom. However, these measures could go large in both quantity and variety, which makes processing them in a reasonable amount of time almost impossible. Besides, not all these readings have the same level of importance as some of them weigh more than the others when it comes to analysing sensor data for smart classrooms. The importance of reconsidering the data that comes from sensors in smart classrooms by selecting only the features that are expected to be more important than the other features in our calculations had been presented in this paper. Selecting important features from a larger set of features had helped reduce the number of dimensions and hence fighting the curse of dimensionality issue. As such, in this study the using of ant colony optimization (ACO), along with another popular classifier had, which is Logistic Regression (LR), to build a hybrid model had been proposed. The proposed hybrid model, which was built based on a series of experiments on two sensor datasets, ultimately helped to select the appropriate set of features in smart classroom environments. This paper had been organized as follows section 2 presented a brief of ant colony optimization, section 3 for the proposed method and section 4 for present the results.

1.1 Ant Colony Optimization

Hard combinatorial optimization solution can be found using the metaheuristic known as ACO, short for "ant colony optimization," which is a meta heuristic algorithm (Hassib et al., 2019). As part of his doctoral thesis, Marco Dorigo developed and published the Ant System (AS), a trio of ACO algorithms. Dorigo and Gambardella (1996) created the first ant algorithm to solve the traveling salesman issue. Each city was modelled as a state in the proposed algorithm, and the connections between them were viewed as transitions. A pheromone was left on the interconnected routes to stress their significance in the overall answer (Müller & Bonilha, 2022). The problem's heuristic function was set up as the inverse of the distance between the cities. they found that the distance between the cities governed the transition rule and how much pheromone waste is left on the walkways. For this reason, the algorithm keeps track of which cities have already been visited and keeps records of which routes to those cities. A route is calculated at the beginning of each iteration. After determining the strength of a solution, ants will then lay pheromone along the route(s) they have traversed (Dorigo & Stützle, 2019). Because of Ant's success, (Dorigo et al., 1999) created ACO. In addition to the ant systems' solution construction and pheromone update modules, ACO employs pheromone evaporation to forget unsuccessful solutions. Dorigo and Gambardella (1997) experimented with different pheromone update functions. In addition to the pheromone update at the end of every epoch, they also used a local pheromone update. A rank-based ant system was developed by Bullnheimer, Hartl, and Strauss (1996).

This system ranks ants according to the quality of their solutions and displays them in descending order (Kashef & Nezamabadi-pour, 2015). ACO-based problem-solving algorithms typically produce search space with nodes (called states) and develops a method to discover a path to a solution (Kashef & Nezamabadi-pour, 2015). A group of simple computational agents cooperate and exchange information via synthetic pheromone trails to form the artificial ant colony (Hossain et al., 2022). The Ant System was an early example of this type of algorithm, which was demonstrated with a well-known application as a basis (Akhtar, 2019). Initialization and a main loop comprise the bulk of ACO. The main loop is set to run for a specified number of iterations by the user before being terminated (Meena et al., 2012). The artificial ant colony is run by a group of simple computational agents that cooperate and use artificial pheromone trails to communicate with one another. Two elements govern the ants' movement (Rozveh & Kamarposhti, 2018)

1. Specific to the problem at hand, heuristic knowledge is be known ahead of time before the algorithm is executed. It's a metric for choose to transition from state Si to state Sj if given the choice.

2. Artificial pheromone trails, in which information about past solutions is left as "pheromones" in the search space.

Many scientific, industrial, and practical issues have benefited from the use of ant algorithms. Other NP-hard problems, like the Quadratic Assignment problem (QAP) and the Job-Shop Scheduling (JSP) problem, have been applied to Ant algorithms, and all the trials have demonstrated its robustness. After that, ant algorithms were heavily utilized to solve the shortest path for the data network routing problem, which is a highly dynamic and complex problem. Vehicle routing, graph colouring, and set covering issues were among of the first targets for the development of ant algorithms (Deng et al., 2019). The procedure of ACO algorithm for combinatorial optimization problems is (Müller & Bonilha, 2021)

- Initialization
- while (termination condition not met) do
- Construct Ant Solutions
- Apply Local Search
- Global Update Pheromones
- end

Main is invoked after initialization of parameters and pheromone trails. It's important to remember that there are three primary phases to a loop. In the first step, the ants build their own answers to the challenge. Subject to consideration, skewed by the pheromone data, and perhaps the available heuristic information. After the ants have solved the problem, there is a chance to refine the results. Localized looking around. Pheromone trails are modified just before the beginning of the next cycle to simulate what it's like for ants to look for something (Abi et al., 2020).

1. 1.1 Proposed ACO algorithm with feature selection (methodology)

The proposed study is based on four steps. These are.

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

1- Firstly, data had been taken from this source [https://github.com/zeynepduygutekler/robod]. Data included 5 rooms. The data that we adopted are two sets (room 1, room 5) because of the big difference between them in the number of features and records, table1 shows that

Table 1. The date obtained from https://github.com/zeynepduygutekler/robod

Item	Room	Records	Features
1	Room 1	8338	28
5	Room 5	13536	36

Features were {timestamp, VOC, sound_ pressure_ level, indoor_ relative_ humidity, air_ temperature, illuminance, pm2.5, indoor_co2, Wi-Fi_ connected_ devices, ceiling_ fan_ energy, lighting_ energy, plug_ load energy, chilled_ water_ energy, FCU_ fan_ energy, temp _setpoint, FCU_ fan speed, supply air_ pressure, supply _air_ temperature, barometric_ pressure, dry bulb_ temp, global_ horizontal_ solar_ radiation, wind_ direction, wind_ speed, outdoor_co2, rainfall_ raw, outdoor_ relative_ humidity, occupant_ presence, occupant_ count}.

2- Data had been pre-processed by three steps

a- Imputation: Using substituted values to fill in for missing data (e.g., 0, mean, max, min, etc.). Data imputation, in which zero is substituted for missing values, was used in this research.

b- Normalization: This is the process of organizing the data so that it appears consistent across all fields and records. Increasing data cohesion means better data quality. Min-max normalization had been used to normalize the data (where 0 represents the lowest value and 1 represents the highest value).

c- Data transformation: data transformation had been used to convert table data into a matrix format (i.e., NumPy array format) before using it in subsequent model building steps to make it more suitable for those operations.

3- A typical procedure after data preprocessing is to inject the data into a model. However, for lowering the data's dimensionality, feature selection must be used first. All possible subsets of the input features are examined. The algorithm assesses the feature subsets' performance and measures the classification outcomes using an evaluation function and the related reduced feature space.

The main phases of the proposed feature selection model are as follows:

I. Initialization: Ant populations, the density of pheromone trails connected to any feature, and the maximum number of allowed iterations are all considered.

II. Formulating and testing solutions. Then, visit the features where each ant could build an entire solution, and assign ants to them at random.

III. Evaluation of subsets of data: This phase includes sorting subsets according to the classifier performance and their length. After that, the best subset is chosen.

IV. Examining the stopping criteria: If the current phase's maximum number of repetitions is reached, the procedure is aborted; otherwise, it continues.

V. Maintaining accurate pheromone concentrations requires that all ants simultaneously deposit pheromone onto the graph, a process that occurs during this phase. In the end, the top ant is given free rein to spread its pheromones across the network's nodes.

VI. create new ants. During this phase, ants are wiped out and new ones are created.

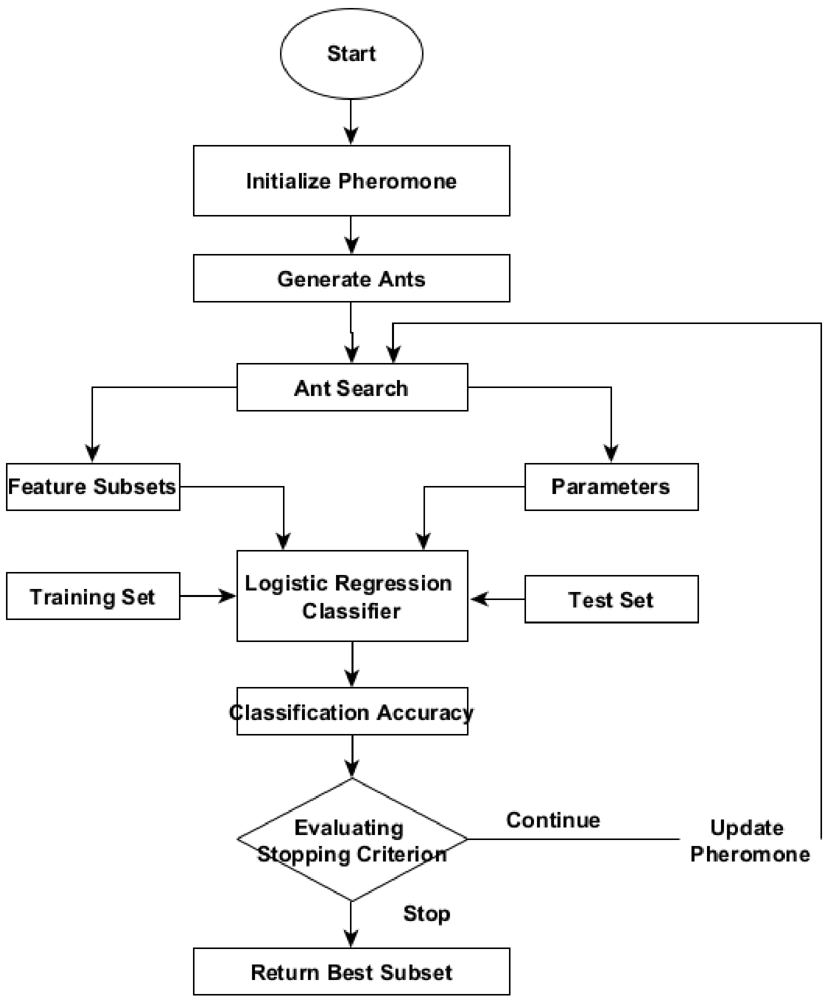

Figure 1: The proposed ACO algorithm with logistic regression classifier

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

The ACO algorithm's ability had been used to perform hyperparameter search and determine which features are best for a given task to integrate it into our solution to reduce the feature vector and increase 'classification accuracy'. It is, however, classification accuracy that determines whether a solution is good. Each solution's classification accuracy will be determined by the fitness function's return value. Higher numbers indicate a more effective solution. A machine learning classifier had been needed that is trained on the feature elements returned by each solution to return the classification accuracy. Parents will be able to choose the best options for their children by evaluating each solution's fitness value. These parents are then put into the mating pool to produce offspring that will form the basis of the next generation's population.

The ACO algorithm's machine learning classifiers are tested for this purpose (logistic regression). Under these conditions, our experiments had been conducted.

Table 2: ACO Experimental settings.

I	# Experiments	# Ants	# Epochs	Alpha	Beta	Lambda	P_1	P_2
	5	20	40	1	0.2	0.005	0.75	0.1

The goal of ‘Alpha’ and ‘Beta’ parameters is to provide a relative importance of learning (i.e., pheromone) and heuristics. The goal of ‘lambda’ is to modify the objective function to minimize while optimizing accuracy, the number of characteristics selected. In practice, higher values of ‘lambda’ would indicate more dominance of the number of features on the objective functions. ‘P1’ and ‘P2’, (1-(p1+p2)) are the probabilities with which method #1, method #2, and method #3 of the heuristics are chosen.

The 'timestamp' (the time at which the observation was made) and 'occupant count' were removed from the data as part of the preprocessing (number of occupants, if any). As for the rest of the features, away with 'temp' and "off coil' temperature setpoints" had been done. Because their values remained constant throughout the entire set of observations, they were omitted. As a result, their existence was no longer required for the label feature to function properly. When no one is in the room, occupant presence is set to '0', and when someone is present, it is set to '1'.

Apart from selecting features, zero replacement had been utilized for missing values, min-max normalization, and data transformation, in which table data was transformed into a matrix format (i.e., NumPy array format) because subsequent processes required the data to be in a matrix format. Using a machine learning algorithm, Figure 2 shows the accuracy results for both the Room 1 and the Room 5 datasets.

AURUM MÜHENDİSLİK SİSTEMLERİ VE MİMARLIK DERGİSİ
AURUM JOURNAL OF ENGINEERING SYSTEMS AND ARCHITECTURE

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

Figure 2:_ The accuracy results for both the Room 1 and the Room 5 datasets

2. CONCLUSIONS

In this paper, we compared our proposed method (ACO algorithm and logistic regression classifier) for feature selection in smart classroom environments to two other metaheuristic algorithms, namely Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). For this, we'll use logistic regression with a classifier trained on machine learning. To compare, the PSO algorithm had a population size of 10, while the number of generations and mutation rates for genetic algorithms were 0.8 and 2, respectively. A total of 80% of the dataset was used for training and only 20% was used for testing. Room 1 and Room 5 datasets were used for implementation, and the results are as follows

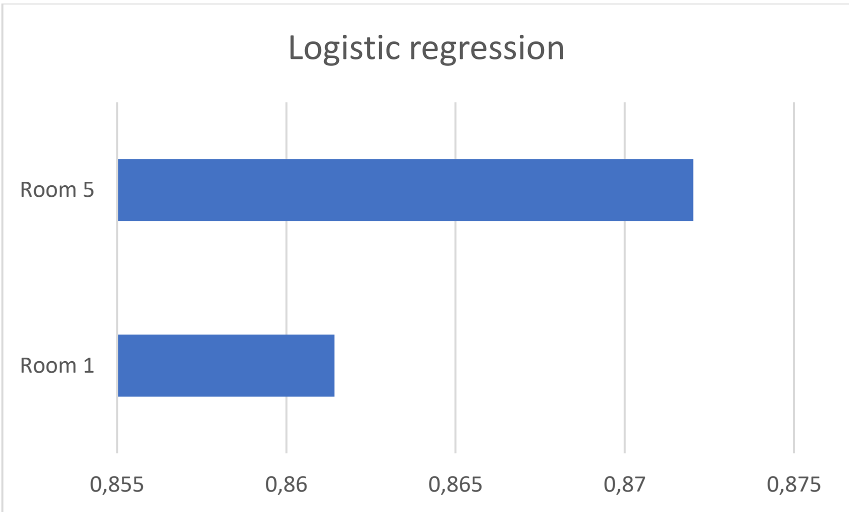

Figure 3: Performance evaluation of the genetic algorithm

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

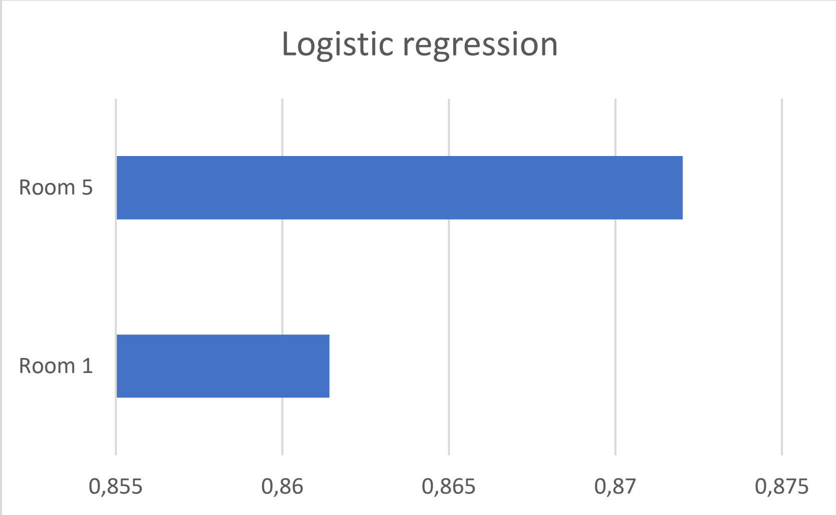

Figure 4: Performance evaluation of the particle swarm optimization algorithm

The results of our model, when compared to those of other metaheuristic algorithms, are conclusive in this study.

REFERENCES

Abi, S., Benhala, B., & Bouyghf, H. (2020). A Hybrid DE-ACO Algorithm for the Global Optimization. In 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS) (pp. 1-6). IEEE.

Akhtar, A. (2019). Evolution of Ant Colony Optimization Algorithm - A Brief Literature Review. arXiv preprint arXiv:1908.08007.

Al Salami, N. M. A. (2009). Ant Colony Optimization Algorithm. UbiCC Journal, 4(3), 823-826.

Deng, W., Xu, J., & Zhao, H. (2019). An Improved Ant Colony Optimization Algorithm Based on Hybrid Strategies for Scheduling Problem. IEEE access, 7, 20281-20292.

Dorigo, M., & Stützle, T. (2019). Ant Colony Optimization: Overview and Recent Advances. International Series in Operations Research and Management Science, 272, 311-351.

Gligorić, N., Uzelac, A., & Krco, S. (2012). Smart Classroom: Real-Time Feedback on Lecture Quality. In 2012 IEEE International Conference on Pervasive Computing and Communications Workshops (pp. 391-394). IEEE.

Hassib, E. M., El-Desouky, A. I., El-Kenawy, E. M., & El-Ghamrawy, S. M. (2019). An Imbalanced Big Data Mining Framework for Improving Optimization Algorithms Performance. IEEE Access, 7, 170774-170795.

Hossain, S. K. M., Ema, S. A., & Sohn, H. (2022). Rule-Based Classification Based on Ant Colony Optimization: A Comprehensive Review. Applied Computational Intelligence and Soft Computing, 2022.

Joseph Manoj, R., Anto Praveena, M. D., & Vijayakumar, K. (2019). An ACO-ANN Based Feature Selection Algorithm for Big Data. Cluster Computing, 22(2), 3953-3960.

Kashef, S., & Nezamabadi-pour, H. (2014). An Advanced ACO Algorithm for Feature Subset Selection. Neurocomputing. http://dx.doi.org/10.1016/j.neucom.2014.06.067

Khaire, U. M., & Dhanalakshmi, R. (2019). Stability of Feature Selection Algorithm: A Review. Journal of King Saud University-Computer and Information Sciences.

Kumar, V., & Minz, S. (2014). Feature Selection: A Literature Review. SmartCR, 4(3), 211-229.

Cilt 7, Sayı 1 | Yaz 2023
Volume 7, No 1 | Summer 2023, 109-118

Li, J., Tang, J., & Liu, H. (2017a). Reconstruction-Based Unsupervised Feature Selection: An Embedded Approach. In IJCAI (pp. 2159-2165).

Li, J., et al. (2017b). Feature Selection: A Data Perspective. ACM Computing Surveys (CSUR), 50(6), 1-45.

Mafarja, M., Jarrar, R., Ahmad, S., & Abusnaina, A. A. (2018). Feature Selection Using Binary Particle Swarm Optimization with Time Varying Inertia Weight Strategies. In Proceedings of the 2nd International Conference on Future Networks and Distributed Systems (pp. 1-9).

Meena, M. J., Chandran, K. R., Karthik, A., & Samuel, A. V. (2012). An Enhanced ACO Algorithm to Select Features for Text Categorization and Its Parallelization. Expert Systems with Applications, 39(5), 5861-5871.

Min-Allah, N., & Alrashed, S. (2020). Smart Campus—A Sketch. Sustainable Cities and Society, 59, 102231.

Müller, F. M., & Bonilha, I. S. (2021). Hyper-Heuristic Based on ACO and Local Search for Dynamic Optimization Problems. Algorithms, 15(1), 9.

Rozveh, R. S., & Kamarposhti, M. A. (2018). Determining Optimal Capacity Of Distributed Generation Units In Multiple Energy Conversion Centers Considering Load Uncertainty Using Aco And Pso Algorithms. University Politehnica Of Bucharest Scientific Bulletin Series C-Electrical Engineering And Computer Science, 80(3), 153-170.

Saini, M. K., & Goel, N. (2019). How Smart Are Smart Classrooms? A Review of Smart Classroom Technologies. ACM Computing Surveys, 52(6), 1-28.

Shardlow, M. (2016). An Analysis of Feature Selection Techniques. The University of Manchester, 1(2016), 1-7.

Venkatesh, B., & Anuradha, J. (2019). A Review of Feature Selection and Its Methods. Cybernetics and Information Technologies, 19(1), 3-26.