ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2017.0218

CLUSTERING CATEGORICAL DATA USING *k*-MODES BASED ON CUCKOO SEARCH OPTIMIZATION ALGORITHM

K. Lakshmi[1], N. Karthikeyani Visalakshi[2], S. Shanthi[3] and S. Parvathavarthini[4]
[1,3]Department of Computer Applications, Kongu Engineering College, India
[2]Department of Computer Science, NKR Government Arts College for Women, India
[4]Department of Computer Technology, Kongu Engineering College, India

Abstract

Cluster analysis is the unsupervised learning technique that finds the interesting patterns in the data objects without knowing class labels. Most of the real world dataset consists of categorical data. For example, social media analysis may have the categorical data like the gender as male or female. The k-modes clustering algorithm is the most widely used to group the categorical data, because it is easy to implement and efficient to handle the large amount of data. However, due to its random selection of initial centroids, it provides the local optimum solution. There are number of optimization algorithms are developed to obtain global optimum solution. Cuckoo Search algorithm is the population based metaheuristic optimization algorithms to provide the global optimum solution. Methods: In this paper, k-modes clustering algorithm is combined with Cuckoo Search algorithm to obtain the global optimum solution. Results: Experiments are conducted with benchmark datasets and the results are compared with k-modes and Particle Swarm Optimization with k-modes to prove the efficiency of the proposed algorithm.

Keywords:

Cluster Analysis, k-Modes, Cuckoo Search Optimization, Local Optima, Initial Centroids

1. INTRODUCTION

Cluster analysis is one of the techniques used for knowledge discovery in data mining. The main objective of the clustering is to minimize the distance between the data objects within the cluster and maximize the distance between the clusters, i.e. minimize the intra-cluster distance and maximize the inter-cluster distance. Clustering algorithms are generally categorized into two categories: Partitional and Hierarchical. The partitional clustering algorithms, group the dataset based on the pre-defined number clusters. Hierarchical clustering algorithms create a tree like structure of data by merging and splitting criterion.

The *k*-modes clustering algorithm is first proposed and publicly available by Huang in [1] [2]. The *k*-modes clustering algorithm supports the categorical data objects. It selects the initial centroids randomly from the given data objects. Due its randomness in its selection of initial centroids, it provides the local optimum solution. Since, the *k*-Modes algorithm comes from the *k*-means algorithm, it can also be treated as an optimization problem [3].

Recently, there are number of optimization algorithms are introduced to obtain the global optimum solution. Some of the nature-inspired metaheuristic optimization algorithms are Genetic Algorithm, Ant Colony Optimization, Simulated Annealing, Particle Swarm Optimization, Tabu Search, Cat Swarm Optimization, Artificial Bee Colony, Gravitational Search, Firefly Algorithm, Bat Algorithm, Wolf Search Algorithm and Krill Herd.

Cuckoo Search is the recently introduced nature-inspired metaheuristic optimization algorithm by the Yang and Deb in 2009 [4], [5]. This algorithm is based on brood parasitic behaviour of cuckoo species and the levy flight behaviour of flies and birds. Cuckoos are fascinating birds, it makes beautiful sounds and lay their eggs in communal nests and may remove the others eggs to increase the hatching probability of their own eggs. If the host bird discovers the eggs that are not their own, it will either throw the alien eggs or simply abandon the nest and build new nest somewhere.

To overcome the *k*-modes local optimum issue, this paper proposes new clustering algorithm based on Cuckoo Search algorithm is combined with *k*-modes (Cuckoo*k*-modes) to obtain the global optimum solution.

The outline of this paper is as follows: Section 2 describes the related researches in the literature. Section 3 describes the *k*-modes clustering algorithm. Section 4 describes the Cuckoo Search optimization algorithm. Section 5 describes the proposed Cuckoo*k*-modes clustering algorithm. The experimental analysis is discussed in section 6. Section 7 describes the comparison of Particle Swarm Optimization and Cuckoo Search algorithms. Conclusion and future works are provided in section 8.

2. RELATED WORKS

Fuzzy based *k*-modes algorithm is proposed in [6]. The authors compared the hard and soft *k*-modes and *k*-modes versions of clustering algorithms. In hard clustering, each object is assigned to single cluster and in fuzzy clustering, each object belongs to more than one cluster based on cluster membership degree value. Also proposed the fuzzy based *k*-modes clustering algorithm to group the categorical data objects.

Tabu search algorithm is combined with *k*-modes is proposed in [7]. Tabu search algorithm is proposed by Glover in 1997 [8]. Tabu search is the metaheuristic based optimization algorithm that directs the local heuristic search procedure to explore the solution for local optimal problems.

Genetic Algorithm is combined with *k*-modes algorithm is proposed in [9]. It finds the global optimum solution for the given categorical dataset and the crossover operator is replaced with *k*-modes operator. The Genetic Algorithm (GA) is proposed by Holland in 1975 [10]. In GA, the search spaces are encoded in the form of chromosomes. For each population, three genetic operators, i.e. selection, crossover and mutation are applied to current population to obtain new population. A genetic Algorithm is combined with fuzzy *k*-modes clustering algorithm is proposed in [11] to obtain the global optimum solution.

Particle Swarm Optimization (PSO) is combined with *k*-modes clustering algorithm is proposed in [12]. PSO is one of the popular metaheuristic and swarm intelligence based optimization algorithms. Swarm based *k*-modes clustering algorithm uses either roulette or random approach to update each population. A novel PSO based *k*-modes clustering algorithm is proposed in [13]. The categorical data is converted to high-dimensional data by mapping the categorical data into natural numbers. Then, PSO based clustering algorithm called *k-p*-modes algorithm is developed. A PSO based *k*-modes clustering algorithm to retrieve a three dimensional objects is designed in [14].

Artificial Bee Colony (ABC) is combined with *k*-modes is proposed in [15]. In this paper, one step *k*-modes clustering algorithm is executed and then integrate this procedure with the artificial bee colony approach. The ABC algorithm is one of the swarm based metaheuristic optimization algorithm and this algorithm inspired by the foraging behaviour of the bees.

Harmony Search optimization algorithm is combined with Cuckoo search algorithm to enhance the search ability of the cuckoo search (CS) algorithm [16]. In this method, the pitch adjustment operation in harmony search is considered as a mutation operator to speed up the convergence process of the updating the cuckoo search.

Cuckoo Search algorithm is combined with the Fuzzy C-Means clustering algorithm [17]. In this paper, the evaluation strategy is used in the similar components among individuals with fuzzy C-means clustering and then updates the evaluation by clusters. Also, the new solution to the current solution can be improved with greedy strategy.

Cuckoo Search algorithm is hybridized with the *k*-Prototype clustering algorithm [18]. It is the extension of *k* means clustering algorithm and grouping the mixed numeric and categorical datasets. This algorithm calculate the distance for numeric data using *k* means clustering algorithm and categorical data using *k* modes clustering algorithm. Crow Search algorithm is hybridized with the *k* Prototype clustering algorithm to obtain the global optimum solution [19].

There are number of works are available for hybridization of global optimization algorithms with *k*-means, but only few works are available for hybridization of global optimization algorithms with *k*-modes.

3. *k*-MODES CLUSTERING ALGORITHM

The *k*-modes clustering algorithm is an extension of *k*-means clustering algorithm. The *k*-means algorithm is the most widely used centre based partitional clustering algorithm. Huang extends the *k*-means clustering algorithm to *k*-modes clustering algorithm to group the categorical data [1] [2]. The modifications done in the *k*-means are (i) using a simple matching dissimilarity measure for categorical objects, (ii) replacing means of clusters by modes, and (iii) using a frequency-based method to update the modes.

Let X, x_{11}, x_{12},...,x_{nm} be the data set consists of n number of objects with m number of attributes. The main objective of the *k*-modes clustering algorithm is to group the data objects X into K clusters by minimize the cost function Eq.(1).

$$P(W,Q)=\sum_{l=1}^{k}\sum_{i=1}^{n} w_{il} d_{sim}(x_i,q_l) \tag{1}$$

where, w_{il} is an $N\times K$ matrix where each element belongs to 0 or 1. N is the total number data objects and K is the number of clusters. $d_{sim}(x_i,q_l)$ is the simple dissimilarity measure and it is defined in the following Eq.(2).

$$d_{sim}\left(x_i,q_i\right)=\sum_{j=1}^{m}\delta(x_{ij},z_{lj}) \tag{2}$$

where, $\delta(x_{ij}, q_{lj})$ is calculated using the following Eq.(3)

$$\delta(x_{ij},z_{lj})=\begin{cases}1 & if\ x_{ij}=z_{lj}\\ 0 & if\ x_{ij}\neq z_{lj}\end{cases} \tag{3}$$

The *k*-modes clustering algorithm is described in Fig.1.

Input: Data objects *X*, Number of clusters *K*.

Step 1: Randomly select the *K* initial modes from the data objects such that C_j, j = 1,2,…,K

Step 2: Find the matching dissimilarity between the each *K* initial cluster modes and each data objects using the Eq.(2).

Step 3: Evaluate the fitness using the Eq.(1)

Step 4: Find the minimum mode values in each data object i.e. finding the objects nearest to the initial cluster modes.

Step 5: Assign the data objects to the nearest cluster centroid modes.

Step 6: Update the modes by apply the frequency based method on newly formed clusters.

Step 7: Recalculate the similarity between the data objects and the updated modes.

Step 8: Repeat the step 4 and step 5 until no changes in the cluster ship of data objects.

Output: Clustered data objects.

Fig.1. *k*-modes algorithm

The main features of *k*-modes clustering are (i) simple and easy to implement and (ii) handle the large amount data objects efficiently. The main issues are (i) need the number clusters in advance and (ii) handle the categorical data only and (iii) produce the local optimum solutions.

4. CUCKOO SEARCH OPTIMIZATION ALGORITHM

This algorithm is inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds. If a host bird discovers the eggs are not their own, it will either throw these alien eggs away or simply abandon its nest and build a new nest somewhere.

Cuckoo Search is based on three idealized rules: (i) each cuckoo lays one egg at a time, and dumps it in randomly chosen nest (ii) The best nest with high quality eggs will be carried over to the next generations (iii) The number of available nests is fixed, and the egg laid by the cuckoo is discovered by the host bird with a probability pa in [0,1] .

The Cuckoo Search optimization algorithm is described in Fig.2.

```
Input: Population size $N$, maximum iteration $it_{max}$, worst nest $p_a$, Objective function objfn.
Step 1: Randomly generate population with the size of $N$.
Step 2: Evaluate the fitness function using the given objective function.
Step 3: Find minimum fitness and show it is the best nest from the given nests.
Step 4: while $it_{max}$
    i. Generate the new population and also keep the current best nests.
    ii. Evaluate the fitness for new nests
    iii. Remove the nests with worst solutions i.e. based on $p_a$ value
    iv. Evaluate the fitness function using the given objective function for the remaining nests.
    v. Find the minimum fitness as the best nest.
Step 5: Display the bestnest as solution.
Output: Bestnest as the solution.
```

Fig.2. Cuckoo Search algorithm

5. PROPOSED ALGORITHM

The *k*-modes clustering algorithm is easy to implement and efficiently handling large datasets. The main drawback is, it produces the local optimum solutions. To obtain the global optimum solutions, the *k*-modes clustering algorithm is combined with global optimization algorithms. Cuckoo Search algorithm is the metaheuristic global optimization algorithm and combined with *k*-modes to obtain the global optimum solution.

The proposed Cuckoo*k*-modes clustering algorithm is proposed in Fig.3.

```
Input: Data objects $X = x_{11}, x_{12},...,x_{nm}$, Number of clusters $K$, Population size $N$, maximum iteration $it_{max}$, worst nest $p_a$, Tolerance $Tol$, Objective function objfn.
Step 1: Randomly generate population with the size of $N$ from the given data objects.
Step 2: Evaluate the fitness function using the given objective function Eq.(1).
Step 3: Find minimum fitness and show it is the best nest from the given nests.
Step 4: While $it_{max}$
    i. Generate the new population and also keep the current best nests.
    ii. Evaluate the fitness for the new nests
    iii. Remove the nests with worst solutions i.e. based on $p_a$ value
    iv. Evaluate the fitness function using the given objective function for the remaining nests.
    v. Find the minimum fitness as the best nest
Step 5: Display the bestnest as solution.
Step 6: Run the $k$-modes clustering algorithm with bestnest as the centroids until convergence criteria is met.
Output: Clustered data objects.
```

Fig.3. Cuckoo*k*-modes algorithm

6. EXPERIMENTAL RESULTS

The algorithms are implemented using Matlab R2015a on an Intel i5 2.30GHz with 4GB RAM. The *k*-modes, PSO*k*-modes and Cuckoo*k*-modes are executed 20 distinct runs. The algorithm specific parameters value for each algorithm is specified in Table.1. The values for PSO algorithm are suggested in [20] and values for the Cuckoo Search algorithm are suggested in [4], [5].

6.1 DATASETS

To evaluate the performance of proposed Cuckoo*k*-modes algorithm, Soybean, Congressional voting, Car Evaluation, Balance Scale, and Lens benchmark datasets are used. These datasets are collected from UCI machine repository [17].

The first data set is Soybean. This dataset has 47 instances and each instance is described by 35 attributes. Each instance is labeled as one of the four diseases: diaporthe stem rot, charcoal rot, rhizoctonia root rot, phytophthora rot. Except for the phytophthora rot, all other diseases have 10 instances each, phytophthora rot has 17 instances. In this data set 14 attributes have only one category, so we selected 21 attributes for the clustering.

The second data set is Congressional Voting records. This dataset consists of United States congressional voting records. It has 435 instances each of which has 16 binary attributes. It is classified as Republican (168 instances) or Democrat (267 instances).

The third dataset is Car Evaluation. It consists of simple hierarchical decision model and useful for testing constructive induction and structure discovery methods to evaluate the car. This dataset consists of 1728 instances each of which has 6 attributes. Except for name attribute 4 categorical attributes are selected for clustering.

Table.1. Algorithm Specific Parameters

Criteria	***k*-modes**	**PSO*k*-modes**	**Cuckoo*k*-modes**
Iterations	20	100	100
Particles	N/A	15	15
Parameters	N/A	$W = 0.72$ $c_1 = c_2 = 1.49$	$p_a = 0.25$

The fourth dataset is Balance Scale. It contains the Balance scale weight and distance. This dataset consists of 625 instances and 4 attributes. The attributes are the left weight, the left distance, the right weight, and the right distance.

The fifth dataset is Lenses. This dataset specifies the fitting contact lenses. It consists of 24 instances and 4 attributes. There are three classes' namely hard lens, soft lens and no lens.

6.2 MEASURES

In this paper, the clustering results are evaluated using F-measure [18] and Rand Index [19]. The F-Measure is the harmonic mean of the precision and recall coefficients. If the precision is high and recall value is low, this results in a low F-measure. If both precision and recall are low, a low F-measure is obtained. On the other hand, if both are high, a high F-measure value is obtained. F-Measure can be computed using the formula given in Eq.(4),

$$\text{F-Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

Precision is calculated as the number of correct positive predictions divided by the total number of positive predictions. The best precision is 1, whereas the worst is 0. Precision is calculated true positive divided by the sum of false positive and true positive. It is calculated using the Eq.(5),

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

where, *TP* means True Positive and it is the count of actual and predicted values are same. *TN* means True Negative and the actual and predicted values are different and *N* is the total number of objects.

Recall is calculated as the number of correct positive predictions divided by the total number of positives. The best sensitivity is 1.0, whereas the worst is 0.0. It is calculated using the Eq.(6),

$$\text{Recall} = \frac{TP}{N} \tag{6}$$

Rand Index is a measure of the similarity between true labels and predicted labels. It is calculated by using the Eq.(7),

$$\text{Rand Index} = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

The Rand index has a value between 0 and 1, with 0 indicating that the two data clusters do not agree on any pair of points and 1 indicating that the data clusters are exactly the same.

6.3 RESULTS

The objective function values of the datasets for three algorithms are shown in Table.2.

The best values for Soybean for three algorithms are 199. But the worst values are 281, 218 and 217 respectively. The proposed algorithm obtains the lower worst, mean and standard deviation values.

The best values for Voting for three algorithms are 1701. But the worst values are 2482, 1922 and 1849 respectively. The proposed algorithm obtains the lower worst, mean and standard deviation values.

The best values for Car Evaluation for three algorithms are 5446, 5106 and 5093. But the worst values are 5533, 5106 and 5093 respectively. The PSO with *k*-modes obtains the worst, mean and standard deviation values are 5106. The proposed algorithm obtains the worst, mean and standard deviation values are 5093. This value is lower than PSO with *k*-modes.

The best values for Car Evaluation for three algorithms are 1640, 1542 and 1542 respectively. The worst values are 1640, 1542 and 1542 respectively. The mean for three algorithms are 1640, 1542 and 1542. The standard deviation is zero. But the objection value for *k*-modes is higher than PSO*k*-modes and Cuckoo*k*-modes. Also PSO*k*-modes and Cuckoo*k*-modes obtain the same objective function values.

The best values for Lenses for three algorithms are 31, 29 and 27 respectively. The worst values are 31, 29 and 27 respectively. The mean for three algorithms are 31, 29 and 27. The standard deviation is zero. But the objective function value for *k*-modes is higher than PSO*k*-modes and Cuckoo*k*-modes. Also PSO*k*-modes and Cuckoo*k*-modes obtain the same objective function values.

The F-Measure and Rand Index values of each algorithm is shown in Table.3. The F-Measure for Soybean, Voting and Car Evaluation Datasets, the proposed algorithm produces the best value. These values are higher than the *k*-modes and PSO with *k*-modes. The Rand Index for Soybean, Voting and Car Evaluation, Balance Scale and Lenses dataset, the proposed algorithm produces the best value. These values are higher than the *k*-modes and PSO with *k*-modes.

The comparison of F-Measure of datasets for *k*-modes, PSO*k*-modes and Cuckoo *k*-modes are shown in the Fig.4. The comparison of RandIndex for *k*-modes, PSO*k*-modes and Cuckoo*k*-modes are shown in the Fig.5.

Table.2. Objective function values

Dataset	**Criteria**	***k*modes**	**PSO*k*-modes**	**Cuckoo*k*-modes**
Soybean	Best	199	199	199
	Worst	281	218	217
	Mean	205.35	199.95	199.9
	Std. Dev	19.21	4.25	4.02
Voting	Best	1701	1701	1701
	Worst	2482	1922	1849
	Mean	1740.05	1712.05	1708.55
	Std. Dev	174.64	49.42	33.07
Car Evaluation	Best	5446	5106	5093
	Worst	5533	5106	5093
	Mean	5450.35	5106	5093
	Std. Dev	19.45	0	0
Balance Scale	Best	1650	1542	1542
	Worst	1650	1542	1542
	Mean	1650	1542	1542
	Std. Dev	0	0	0
Lens	Best	31	29	27
	Worst	31	29	27
	Mean	31	29	27
	Std. Dev	0	0	0

Table.3. F-Measure and Rand Index of the three Algorithms

Dataset	***k*-modes**	**PSO*k*-modes**	**Cuckoo*k*-modes**
F-Measure			
Soybean	0.7999	0.9066	0.9479
Voting	08557	0.8620	0.8644
Car Evaluation	0.4324	0.4417	0.4452
Balance Scale	0.4809	0.5086	0.5351
Lens	0.5214	0.5908	0.6603
Rand Index			
Soybean	0.8593	0.9265	0.9566
Voting	0.7514	0.7591	0.7627
Car Evaluation	0.4911	0.4955	0.4980
Balance Scale	0.5295	0.5515	0.559
Lens	0.5	0.6051	0.6196

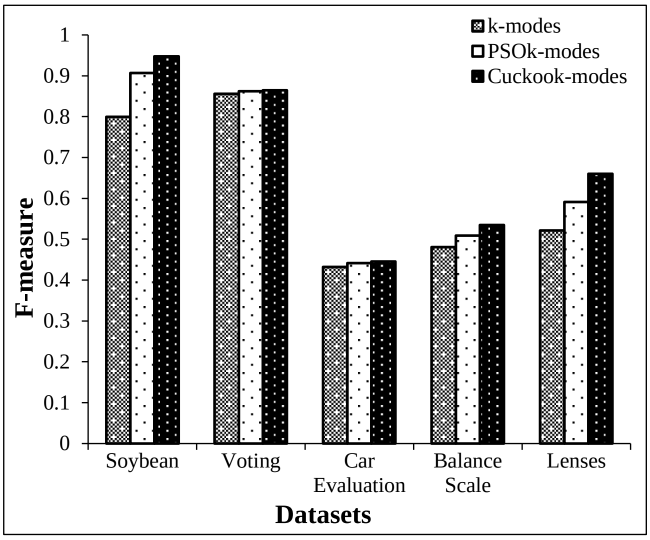

Fig.4. F-measure values of datasets for three algorithms

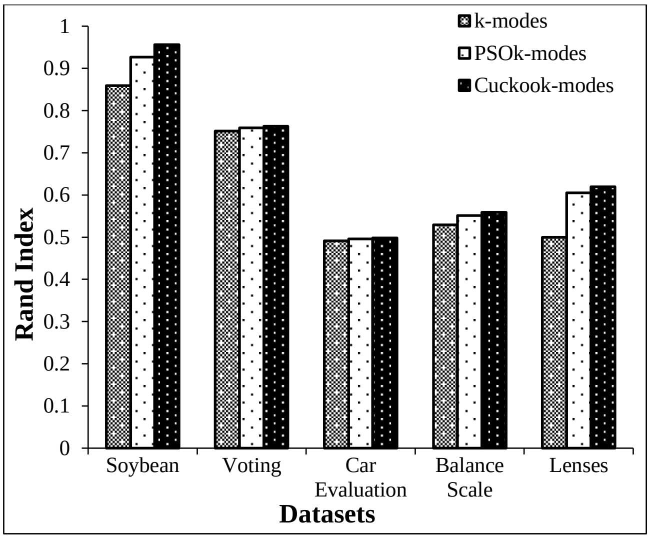

Fig.5. Rand Index values of datasets for three algorithms

7. COMPARISON OF PSO AND CUCKOO SEARCH OPTIMIZATION ALGORITHMS

Both the PSO and Cuckoo Search algorithms are populations based metaheuristic optimization algorithms. All optimization algorithms have individual controlling parameters it is varying from one to another algorithm. Parameter setting is the time consuming task and lagging in setting the proper values for algorithms. In PSO, requires four parameters namely lower and upper bounds of velocity, inertia weight, social learning factor and individual learning factor. In Cuckoo Search, requires only pa parameter. In PSO, have the complexity like need to initialize and check the boundaries of velocity. Also each potential solution is added with randomized velocity to produce the new potential solutions. In Cuckoo Search, does not have the complexity like this. PSO algorithm uses the random walk and Cuckoo Search uses the Levy flight to obtain the new solutions.

8. CONCLUSION

The *k*-modes clustering algorithm most widely used for clustering categorical data. This algorithm is easy to implement and efficiently handling large categorical datasets. In this paper, *k*-modes clustering algorithm is combined with the Cuckoo Search optimization algorithm and proposed the algorithm called Cuckoo*k*-modes algorithm. The motivation behind this work is *k*-modes algorithm selects the initial centroids randomly and produces the local optimum solution and to overcome this problem. The proposed algorithm outperforms than *k*-modes and Particle Swarm Optimization with *k*-modes. In future, extend to dynamically determine the number of clusters and extended with internal validity measures.

REFERENCES

[1] Z. Huang, "A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining", *Proceedings of Data Mining and Knowledge Discovery*, pp. 1-6, 1997.

[2] Z. Huang, "Extensions to the *K*-means Algorithm for Clustering Large Data Sets with Categorical Value", *Data Mining and Knowledge Discovery*, Vol. 2, No. 3, pp. 283-304, 1998.

[3] G. Gan, C. Ma and J. Wu, "*Data Clustering: Theory, Algorithms, and Applications*", Society for Industrial and Applied Mathematics, 2007.

[4] X.S. Yang and S. Deb, "Cuckoo Search via Levy Flights", *Proceedings of IEEE World Congress in Nature and Biologically Inspired Computing*, pp. 210-214, 2009.

[5] X.S. Yang and S. Deb, "Engineering Optimisation by Cuckoo Search", *International Journal of Mathematical Modelling and Numerical Optimisation*, Vol. 1, No. 4, pp. 330-343, 2010

[6] Z. Huang and M.K Ng, "A Fuzzy *K*-Modes Algorithm for Clustering Categorical Data", *IEEE Transactions on Fuzzy Systems*, Vol. 7, No. 4, pp. 446-452, 1999.

[7] M.K. Ng and J.C Wong, "Clustering Categorical Data Sets using Tabu Search Techniques", *Pattern Recognition*, Vol. 35, No. 12, pp. 2783-2790, 2002.

[8] F. Glover and M. Laguna, "*Tabu Search*", Kluwer Academic Publishers, 1997.

[9] G. Gan, Z. Yang and J. Wu, "A Genetic *K*-Modes Algorithm for Clustering Categorical Data", *Proceedings of International Conference on Advanced Data Mining and Applications*, pp. 195-202, 2005

[10] J.H. Holland, "*Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*", MIT press, 1992.

[11] G. Gan, J. Wu and Z. Yang, "A Genetic Fuzzy *K*-Modes Algorithm for Clustering Categorical Data", *Expert Systems with Applications*, Vol. 36, No. 2, pp. 1615-1620, 2009.

[12] H. Izakian, A. Abraham and V. Snasel, "Clustering Categorical Data using a Swarm-based Method", *Proceedings of World Congress on In Nature and Biologically Inspired Computing*, pp. 1720-1724, 2009.

[13] L. Mei and Z. Xiang-Jun, "A Novel PSO *k*-Modes Algorithm for Clustering Categorical Data", *Proceedings of Computer, Informatics, Cybernetics and Applications*, pp. 1395-1402, 2012

[14] X. Zhao and M. Lu, "3D Object Retrieval Based on PSO-*K*-Modes Method", *Multimedia Tools and Applications*, Vol. 8, No. 4, pp. 963-970, 2013.

[15] J. Ji, W. Pang, Y. Zheng, Z. Wang and Z. Ma, "A Novel Artificial Bee Colony based Clustering Algorithm for Categorical Data", *PLOS One*, Vol. 10, No. 5, pp. 1-6, 2015.

[16] G.G. Wang, A.H. Gandomi, X. Zhao and H.C. Chu, "Hybridizing Harmony Search Algorithm with Cuckoo Search for Global Numerical Optimization", *Soft Computing*, Vol. 20, No. 1, pp. 273-85, 2016

[17] L. Yu, Z. Dong, H. Wang and Y. Ding, "The Cuckoo Search Algorithm based on Fuzzy C-Mean Clustering", *Proceedings of 36th Chinese Control Conference*, pp. 2691-2696, 2017

[18] K. Lakshmi, N. Karthikeyani Visalakshi and S. Shanthi. "Cuckoo Search based *K*-Prototype Clustering Algorithm", *Asian Journal of Research in Social Sciences and Humanities*, Vol. 7, No. 2, pp. 300-309, 2017.

[19] K. Lakshmi, N. Karthikeyani Visalakshi, S. Shanthi and S. Parvathavarthini, "Clustering Mixed Datasets using *K*-Prototype Algorithm based on Crow-Search Optimization", *Proceedings of Developments and Trends in Intelligent Technologies and Smart Systems*, pp. 191-197, 2017.

[20] F. Van Den Bergh, "An Analysis of Particle Swarm Optimizers (PSO)", PhD Dissertation, Faculty of Natural and Agricultural Science, University of Pretoria, 2001.

[21] A. Asuncion and D. Newman, "UCI Machine Learning Repository", Available at: http://www.ics.uci.edu/~mlearn/MLRepository.html, Accessed on 2007.

[22] C.J. Van Rijsbergen, "Information Retrieval", PhD Dissertation, Department of Computer Science, University of Glasgow, 1979.

[23] W.M. Rand, "Objective Criteria for the Evaluation of Clustering Methods", *Journal of the American Statistical association*, Vol. 66, No. 336, pp. 846-850, 1971.