Unstructured Data Clustering Using Hybrid K-Means And Grasshopper Optimization Algorithm (Kmeans-GOA)

Vikash Kumar Sharma, Ravindra Patel

Abstract: K-Means is the well known straight forward partitioning clustering method which is used to divide a group of data points into predefined number of clusters. It has been effectively solved several real world clustering problems. Though, the sensitive nature of the selection of initial cluster centroid and convergence to the local optima is major drawbacks of K-Means algorithm. Several optimization techniques are performed to improve the quality of clustering. Here, a hybrid KMeans-Grasshopper Optimization Algorithm (KMeans-GOA) is proposed to perform optimal clustering and enhanced the efficiency of clustering method. The efficiency of the proposed algorithm KMeans–GOA is analyzed against the efficiency of K-Means, KMeans-PSO and KMeans-ALO clustering algorithms on the basis of various performance factors. Results and statistical analysis are evaluated on six datasets. The evaluated outcomes show that the hybrid KMeans-GOA has been provided better efficiency than K-Means, KMeans-PSO and KMeans-ALO based on sum of average of intracluster distance, F-Measure, Purity Index and Standard Deviation.

Index Terms: Clustering, K-Means, Grasshopper Optimization Algorithm, Purity Index, Statistical Analysis.

1 INTRODUCTION

Data Clustering is a partitioning technique of data elements into several clusters so that data elements in same cluster have maximum similarity; however, they are very different elements in the other clusters on the basis of cluster`s attributes. General method for all clustering techniques is to obtain cluster`s center (centroids) that will characterize all clusters. Several clustering techniques have been implemented and are categorized from various prospects like partitioning methods, hierarchical methods, density-based methods [7], and grid-based methods [7]. Additionally data set can be defined as numeric or categorical. Intrinsic statistical characteristics of numeric data are to be oppressed to logically describe distance function among data elements. While categorical data is to be copied from quantitative and qualitative data and explanations are sprightly obtained from the counts. Textual data can be distributed in clusters by using a model Textual Virtual Schema Model (TVSM) combining the three steps. Firstly the unstructured data [19] is extracted from data sources and convert into structured data. In second step clustering is performed on structured data and in last step similarity of documents is evaluated to improve the accuracy of query [22]. K-Means is an easy and straight forward partitioning algorithm in which each data element allocates to the cluster having minimum distance from cluster centroid. The major disadvantage of the K-means technique is that the outcomes of cluster is very susceptible to the choice of centroids of the primary cluster and may congregate to the local optima.

- *Vikash Kumar Sharma is currently Research scholar, pursuing Ph.D. in Computer Science & Engineering in Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, India. E-mail: vikas.16phdsharma@gmail.com*
- *Ravindra Patel is currently Assistant Professor in Computer Science & Engineering Department, in University Institute of Technology, RGPV, Bhopal, India. E-mail: ravindra@rgtu.net*

If primary centroids are updated then K-Means obtains various outcomes. The K-Means is applied to clinical documents containing medical treatment information for several diseases. Initially data are to be pre-processed to improve the quality of clustering then K-Means is applied to data to extract the symptoms and medication from clinical data [5]. The K-Means is one of the best clustering algorithms for clustering the small datasets but for large datasets [7] the algorithm can be scalable to the K-Means Hadoop Map Reduce (KM-HMR), which implies the Map Reduce implementation of K-means. So the quality of clustering is also enhanced with higher intra-cluster and lower inter-cluster distances, higher efficiency and lower execution time for large datasets [3]. Another application of K-Means categorizes the patent documents into several useful groups. This patent document is to be related to the green tea, which is divided into groups using data preparation and statistical data analysis [20]. In recent years, many researchers developed several bio inspired algorithms to improve the performance of clustering methods. Teaching-Learning-Based Optimization (TLBO) is one of the optimization techniques which are developed for automatic clustering of huge unstructured data sets into optimal clusters. TLBO is divided into two parts, first is teacher phase (learns from a teacher) and the second is the learner phase (learns from learner`s interfacing). It is a population based technique to provide automatic clustering of real life data [9]. Learning and optimal clustering is also performed by using another meta heuristic algorithm Bees Algorithm with Memory Scheme (BAMS) in which a memory system is added to the bees algorithm for calculating the similarities between present and previously visited sites. Swarm behaviour of bees has been efficiently used to optimize the cluster heads and memory scheme is used to store the intermediate results of clustering deciding the similarities among data elements. The outcomes indicate that the BAMS gives better performance than BA in terms of intra cluster distances [12]. Another type of data is health records of patients, which is complex and unstable nature. This type of data is analyzed the diseases and health risks by using Weighted Principle Component Analysis (WPCA) and obtained into clusters using Improved BAT Algorithm (IBAT). This combination is implemented on high dimensional dataset of disease related to liver and heart with

higher accuracy and quality. The results show higher efficiency and lower computation time of IBAT compares to the BAT algorithm [21]. A combination of fuzzy clustering algorithm (FCM) and multi objective genetic algorithm (NSGA-II) is proposed to perform clustering at runtime it means number of clusters is decided at runtime not predefined [24]. The popular K-Means algorithm is merging with nature inspired simulated annealing to form Simulated Annealing K-Means (SAKM) to enhance the efficiency of clustering to minimize the energy consumption and searching strength and applied on artificial and real life datasets [16]. A multi objective optimization based on simulated annealing is also introduced for machine learning and natural language processing for clustering on medical datasets and compares the performance against simulated annealing clustering [2]. Cuckoo Search Optimization (CSO) algorithm is another nature inspired optimization technique applied to high dimensional large datasets over internet. It performs high speed clustering with intelligent operators and local and global search but a number of assumptions is a basic drawback of CSO [14]. An improved version of cuckoo optimization, Extended Cuckoo Optimization Algorithm (ECOA), is implemented to improve the operators from classical version to chaotic form. Selection of eggs is done by using fitness and migration of cuckoos is obtained with several deviation degrees [10]. The results are evaluated and compared with a cuckoo optimization algorithm (COA), K-Means and Ant Colony Optimization (ALO) to represent the higher efficiency and performance of ECOA [10]. A new meta-heuristic technique Whale Clustering Optimization Algorithm (WCOA) is introduced to perform clustering the huge datasets on the basis of humpback whales swarm foraging nature [8]. It is implemented on real datasets and artificial datasets and compared the outcomes with well known algorithms like Particle swarm optimization (PSO), Genetic Algorithm (GA), K-Means, Artificial Bee Colony (ABC), and Differential Evolution based clustering (DE). When whales are searching and encircling prey, then selection of the best agents is difficult in WOA. So a multi swarm WOA (MSWOA) is introduced to find the best agents in two parts exploration and exploitation. The performances of MSWOA are analyzed in terms of intra and inter cluster distances which show the better efficiency against the WOA [15]. Particle swarm optimization (PSO) [18] is a more popular nature inspired technique using the social activities of bird flocking [6]. Each particle in PSO is associated with local best (pbest) and global best (gbest) values which stores the fitness of particle. PSO is combined with K-Means to overcome the problem of local optima and enhance the performance of clustering to provide optimal cluster`s centers [11]. PSO has been used cost function to guide the search agents of swarm in the direction of the best solution. This cost function further improved for multiple objectives and analyzed against other evolutionary and non evolutionary algorithms in terms of Dunn index, inter and intra cluster distances [13]. The multi objective clustering is performed by IMCPSO (Improved Multi-objective Clustering framework using PSO) to provide local optima free solution in continuous search space. The searching efficiency of clustering is also improved and leader selection strategy is evaluated by Pareto set analysis [4]. PSO is also combined with other bio inspired Fractional Genetic algorithm (FG) to form FGPSO to enhance the optimal strength of clustering for high dimensional datasets. The efficiency of FGPSO is evaluated based on accuracy, execution time, DB-Index, XB-Index and Sym-Index. The results indicate better performance and efficiency of FGPSO against K-Means, PSO and GA [1]. The local optima and slow convergence are basic problems in optimized clustering. These problems can be eliminated by using Seed Disperser Ant Algorithm (SDAA) with K-Means to improve quality of clustering. In SDAA male ants produce new advanced colonies continued to generate the global minimum and local optima. The outcomes indicate better performance of SDAA based on fitness values [23]. Another hybrid combination of a meta heuristic algorithm Ant Lion Optimization and K-Means (KMeans-ALO) is introduced based on ant`s behaviour. It provides both the exploration for global search and exploitation for local search and also maintain the higher convergence and random search agent`s selection. The results show that KMeans-ALO gives better performance as compared to the K-Means, DBSCAN, KMeans-PSO, and KMeans-FA in terms of intracluster distance, and F-Measure [17 In above literature review, several clustering techniques are implemented on multiple datasets to improve the information retrieval. Various nature inspired algorithms like PSO, GA, ALO etc are explained to perform optimal clustering. But none of algorithms will resolve all optimization problems. So we implemented a novel hybrid KMeans- Grasshopper Optimization Algorithm (KMeans-GOA) inspired by grasshoppers. It provides high exploitation to converge the global optimum and exploration to search in global space. It obtains the fast convergence speed to get global optima quickly in free space searching. Proposed KMeans-GOA is implemented and results are compared with other data clustering algorithm K-Means, KMeans-PSO and KMeans-ALO in terms of intracluster distance, purity index, F-Measure and standard deviation. The residue of the paper is prepared as follows. Section2 explains the essential preliminaries and methodology which provides the depiction about proposed methodology, dataset and performance factors for measuring the efficiency of clustering algorithms. Section 3 shows the outcomes generated from the experiment of the proposed algorithm. Section 4 provides conclusion of the work.

2 PRELIMINARIES AND METHODOLOGY

2.1 Proposed Hybrid KMeans-GOA Clustering Algorithm

K-Means clustering is a popular partitioning algorithm based on certain establishment of analysis of variances. It is divided the complete dataset into K clusters by (1).

$$D = \sum_{u=1}^{K} \sum_{v=1}^{|DS|} d^2(C_u - X_v) \quad (1)$$

Where $d^2(C_u - X_v)$ =square of $L_{x,y}$ Euclidean $K_{x,y}$ distance from u^{th} centroid of cluster to v^{th} data point, $|DS|$ = size of a data set. The K-Means allocates each data point to the cluster, which is close to the centroid (the average of all data points in a cluster) of cluster. Then it calculates the similarity between data points and centroid and reallocating the data points of dataset to the cluster centroid. The reallocation of data points to the centroid of clusters will continue until a convergence condition is found. Grasshopper Optimization Algorithm (GOA) is a bio-inspired optimization based on behaviour of grasshopper swarms. The Grasshopper is an insect which is helpful for enhancing the cultivation and

harvest creation level. They combine with swarm of creatures and combination is found in both nymph and adulthood. The millions groups of nymph grasshopper`s travel like rolling cylinder and consume plants in their route. The adult grasshoppers appear warm in environment and move quickly over long distances for searching a food. The search process is divided into two steps exploration in which search agents moved rapidly and exploitation in which search agents moves locally. After performing of K-Means clustering and getting the k clusters, GOA is initiated to discover optimal cluster centroid from the existing clusters, which minimize the intracluster distances to generate the best results.
Start

Step1. GOA is performed mathematically by initializing population, generation, function, and position of each grasshopper (indicating clusters in dataset), and search agents (indicating the data points in data sets) to arbitrarily value.

Step2: The arithmetical model applied to imitate the swarming activities of grasshoppers is shown as follows,

$$LO_x = SI_x + GF_x + WA_x \quad (2)$$

Where, LO_x = location of the x[th] grasshopper, SI_x = social interface, GF_x = gravitational force on the x[th] grasshopper, WA_x = wind advection.

Step 3: To enhance the randomness of the eq. 2 can be updated as $LO_x = n_1\ SI_x + n_2\ GF_x + n_3\ WA_x$ where n1, n2, and n3 are arbitrary numbers in [0, 1].

$$SI_x = \sum_{\substack{y=1 \\ y\neq x}}^{N} si(d_{xy})\,\hat{d}_{xy} \quad (3)$$

Where, d_{xy} = distance between the x[th] and y[th] grasshopper, evaluated as $d_{xy}=|lo_y - lo_x|$, si is a function to illustrate the power of social interface $\hat{d}_{xy} = \frac{lo_y - lo_x}{d_{xy}}$ is a unit vector from the x[th] to y[th] grasshopper.

Step 4: The si function, which illustrates the power of social interface, is evaluated by using eq. 4.

$$si(r) = I_a e^{\frac{-r}{\lambda}} - e^{-r} \quad (4)$$

Where, I_a = attraction intensity and λ = attractive length magnitude. The function si indicates the social interaction like attraction and repulsion of grasshoppers.
Step 5: The gravitational force GF_x is calculated by using eq. 5.

$$GF_x = g\,\hat{e}_g \quad (5)$$

Where, g=universal gravitational constant and $\hat{e}_g$ = unit vector in the direction of the center of earth.

Step 6: The wind advection WA_x is evaluated by using eq. 6.

$$WA_x = \alpha\,\hat{e}_w \quad (6)$$

Where α = constant drift and $\hat{e}_w$ = unit vector towards the wind.

Step 7: Substituting the values of SI_x, GF_x, and WA_x in eq. (2), this equation can be extended as follows.

$$LO_x = \sum_{\substack{y=1 \\ y\neq x}}^{N} si(|lo_y - lo_x|)\frac{lo_y - lo_x}{d_{xy}} - g\,\hat{e}_g + \alpha\,\hat{e}_w \quad (7)$$

$si(r) = I_a e^{\frac{-r}{\lambda}} - e^{-r}$ Where , N shows number of grasshoppers. The exploitation of swarm is in free space. Yet, eq. (7) is utilized to simulate the interface between grasshoppers.

$$LO_x = c\left(\sum_{\substack{y=1 \\ y\neq x}}^{N} c\frac{UB_d - LB_d}{2} si(|lo_y{}^d - lo_x{}^d|)\frac{lo_y - lo_x}{d_{xy}}\right) + \hat{T}_d \quad (8)$$

Where, UB_d = upper bound and LB_d = lower bound in the D[th] dimensions, $\hat{T}_d$ = Value of the D[th] dimension in the target. Yet, we do not use gravitational force (no GF component) and suppose that the wind advection (WA component) is always in the direction of a target $\hat{T}_d$. Adaptive parameter c is used twice in eq. 8 in which first c indicates the inertial weight and balances the exploration and exploitation of complete swarm around the target and second c is used to decrease the attraction, comfort and repulsive zone of grasshoppers. Eq. 8 is used to find the next location of grasshoppers depending upon the current location.

$$c = c_{\max} - \frac{c_{\max} - c_{\min}}{N_r} \quad (9)$$

Where $c_{\max}$ is the maximum value, $c_{\min}$ is the minimum value, and N_r is the maximum number of repetitions. The location of the best target is updated by using eq. 8 and distances between grasshoppers are normalized in each repetition and c is evaluated by using eq. 9. Location updating is performed until the contentment of an end condition and finally returned the best global optimal location and fitness of the best target in a search space. The efficiency of K-Means clustering method is enhanced by using hybrid KMeans-GOA. Initially K-Means algorithm is used to perform clustering of giving datasets into specified clusters on the basis of minimum Euclidean distance. After that the optimized cluster centroids are evaluated for all clusters by using GOA algorithm. In GOA, each cluster is initialized as grasshopper and every data points are initialized as search agents. Therefore, each grasshopper is updating their location on the basis of fitness value to minimize the sum of average of intracluster distances. For each cluster grasshopper evaluates its best location to find optimal centroid. The flow chart of proposed KMeans-GOA is represented in Fig. 1.

Hybrid KMeans-GOA Clustering Algorithm

INPUT: N_r =number of repetitions, K =Number of clusters, P_T = Total population of grasshoppers, N_G =number of grasshoppers, N_{sa} =number of search agents, DS =dataset, N_I =number of instances in dataset, N_A = number of attributes in dataset.

OUTPUT: cluster centroids (T_{sa}) with optimum value.

Start	*Number of Operations*
Choose K arbitrary points as cluster centroid	
WHILE an end condition is not contented	*(* N_r *+1)*
FOR every data point	N_r **(* P_T *+1)*
Calculate Euclidean distance of every data point from the centroids	$N_r * P_T * K$
Allocate the data point to minimum distance cluster	$N_r * K$
END FOR	
Calculate the mean of entire data points in every cluster	$N_r * K$
Allocate the mean values as the new centroids	$N_r * K$
END WHILE	
Return K clusters	
FOR every cluster	K *+1*
Initialize the location of grasshopper Swarm lo_x *(x=1,2,3,.....n)*	
Initialize $c_{\max}$, $c_{\min}$, *and* N_r.	
Evaluate the fitness of every search agent	$K * P_T$
T_{sa} *=best search agent*	$K * N_{sa}$
WHILE (i < N_r *)*	K **(* N_r *+1)*
Modify c by using eq. 9.	$K * N_r$
FOR every search agent	$N_r * K$ **(* N_G *+1)*
Distance between grasshoppers is to be normalized	$N_r * K * N_G$
Modify the location of present search agent by using eq. 8.	$N_r * K * N_G$
If present search agent moves outside the boundaries, then carry it back	$N_r * K * N_G$
END FOR	
Modify T_{sa} *if better location found i=i+1*	$K * N_r$
END WHILE	
Return T_{sa}	K
END FOR	
Choose T_{sa} *as the new cluster centroid*	*1*
STOP	

2.2 Pre-processing of Data sets

The documents of an unstructured dataset are performed pre-processing before clustering. Pre-processing helps to improve the quality of clustering by removing the unnecessary information and to convert the datasets from unstructured to structured format. The pre-processing performs following

steps:

Tokenization provides labeling of each word by a token in documents. Stemming performs alteration where a base word is altered to the several forms of that word. E.g. perform is used in place of 'performs', 'performed', and 'performing'. Stop word elimination Stop words are commonly used prepositions, articles and pronouns not playing any role in clustering, therefore eliminated from documents. After that pre-processed documents of dataset are represented as vectors (numerical values) by using term frequency-inverse document frequency (tf-idf) value. Term frequency is defined as a ratio of number of repetitions (frequency) of a word to whole number of words in a document and inverse document frequency is evaluated as log of the ratio of whole number of documents to number the documents holding that word. These two metrics are multiplied to calculate the tf-idf matrix value. This tf-idf value matrix is represented the numerical values of attributes of documents of dataset.

2.3 Datasets

The proposed algorithms (KMeans-GOA) are performed on 6 different datasets which are obtained from the UCI machine learning repository [27]. The datasets are glass, immunotherapy, soybean, cryotherapy, and heart and breast cancer. All the instances of glass dataset defining the glass types are distributed into 7 classes. Immunotherapy datasets can be divided into 2 classes explaining positive or negative results of patients using immunotherapy. All the instances of soybean datasets are categorized into 4 classes defining the types of soybean. Cryotherapy datasets can be distributed into 2 classes providing the information about positive or negative patient`s results using cryotherapy. Heart datasets are divided into 4 classes describing the heart disease. All the instances of breast cancer datasets are distributed into 2 classes explaining the positive or negative results of patients having cancer. A brief explanation about all the datasets represented in table 1.

Table1: *A brief explanation of the Data Sets Used*

Sr. No.	*Data Set*	*No. of Instances*	*No. of Attributes*	*No. of Clusters*
1	*Glass*	*214*	*10*	*7*
2	*Immunotherapy*	*90*	*7*	*2*
3	*Soybean*	*47*	*35*	*4*
4	*Cryotherapy*	*90*	*7*	*2*
5	*Heart*	*297*	*14*	*4*
6	*Breast Cancer*	*116*	*10*	*2*

2.4 Performance Factors

The efficiency of proposed algorithm KMeans-GOA is analyzed on the basis of several performance factors like intracluster distances, Purity index, F-Measure and Standard deviation.

2.4.1 Intracluster Distances

The intracluster distance is defined as the average distance between data points in the same cluster. It should be minimized for optimum efficiency of clustering. It is evaluated by centroid diameter method in which calculated the average distance between the centroid and all data points of same cluster. The process is repeated for all clusters and then an average of intracluster distances of all clusters can be evaluated.

2.4.2 Purity Index

Purity is the accuracy of clustering method that performs most frequent classification of data points. It means all elements of

a single class can be frequently assigned to a single cluster. Purity and Purity Index (PI) are calculated by using eq. 10 & 11.

$$Purity(CL_v) = \frac{\max(|CL_{uv}|)}{|CL_v|}$$

(10)

$$PI = \sum_{v=1}^{k} \frac{\left(|CL_v| Purity(CL_v)\right)}{|DS|}$$

(11)

Where, k= number of clusters, $|CL_v|$ = size of v^{th} cluster, $|DS|$ =size of a dataset, and $|CL_{uv}|$ = number of data points of class u assigned to cluster v. PI value is obtained from the range 0 to 1 and high purity means PI value is obtained closer to 1.

2.4.3 F-Measure

F-Measure is evaluated by using the term precision and recall for information retrieval. Precision and recall are calculated by using eq. 12 & 13 then F-Measure is calculated by using eq. 14 & eq. 15.

$$precision(u,v) = \frac{|CL_{uv}|}{|CL_v|}$$

(12)

$$recall(u,v) = \frac{|CL_{uv}|}{|CL_u|}$$

(13)

$$F(u,v) = \frac{2.precision(u,v).recall(u,v)}{precision(u,v) + recall(u,v)}$$

(14)

$$F = \sum_{u=1}^{k} \frac{|CL_u|}{|DS|} \max\{F(u,v)\}$$

(15)

Where, k, $|CL_v|$, $|DS|$ and $|CL_{uv}|$ are defined in purity index section and $|CL_u|$ = size of u^{th} class.

2.4.4 Standard Deviation

Standard deviation (SD) is a statistical property that defines the tight clustering of data around the mean. It is smaller for optimal clustering and provides the most reliable mean value of clusters. It is evaluated by using the eq. 16.

$$SD = \sqrt{\frac{\sum(d - \bar{d})}{|DS|}} \quad (16)$$

Where, d=each value in dataset, $\bar{d}$ =mean of all values in dataset.

3 RESULT AND ANALYSIS

All the algorithms described in the preliminaries and methodology part is implemented by Intel(R) Core(TM) i3-3110M, 2.40 GHz, 2 GB RAM on a windows 8 platform using MatlabR2018a on system. The experimental outcomes for Purity index, F-measure, Standard Deviation and intracluster distance, are evaluated on entire 6 datasets described in the preliminaries and methodology part represented in table 2 to table 7. The outcomes are calculated separately over 20 runs for 100, 500 and 1000 repetitions. For glass dataset the hybrid algorithm KMeans-GOA provides minimum value of 0.000384 for 100 repetitions, 0.000388 for 500 repetitions and 0.000370 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.74 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.79 for 100, 500 and 1000 repetitions for purity index and minimum value of 0.00009 for 100 repetitions, 0.0000977 for 500 repetitions and 0.0001077 for 1000 repetitions for standard deviation. The outcomes tabulates in Table 2 are evaluated over 214 data points of glass dataset. For immunotherapy dataset the hybrid algorithm KMeans-GOA provides minimum value of 0.000438 for 100 repetitions, 0.000455 for 500 repetitions and 0.000443 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.89 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.92 for 100, 500 and 1000 repetitions for purity index. KMeans-GOA provides negligible value of 0.0001818 for 100 repetitions, 0.000075 for 500 repetitions and 0.000177 for 1000 repetitions for standard deviation where K-Means and KMeans-ALO provide value of zero for 100, 500 and 1000 repetitions for standard deviation. The outcomes tabulates in Table 3 are evaluated over 90 data points of Immunotherapy dataset. For soybean dataset the hybrid algorithm KMeans-GOA provides minimum value of 0.000288 for 100 repetitions, 0.000277 for 500 repetitions and 0.000231 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.93 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.95 for 100, 500 and 1000 repetitions for purity index. KMeans-GOA provides minimum value of 0.000042 for 100 repetitions, 0.000055 for 500 repetitions and 0.000044 for 1000 repetitions for standard deviation. The outcomes tabulates in Table 4 are evaluated over 47 data points of soybean dataset. For cryotherapy dataset the hybrid algorithm KMeans-GOA provides minimum value of 0.000476 for 100 repetitions, 0.000470 for 500 repetitions and 0.000456 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.91 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.96 for 100, 500 and 1000 repetitions for purity index. KMeans-GOA provides negligible value of 0.0001261 for 100 repetitions, 0.0001280 for 500 repetitions and 0.0001328 for 1000 repetitions for standard deviation where K-Means and KMeans-ALO provide value of zero for 100, 500 and 1000 repetitions for standard deviation. The outcomes tabulates in Table 5 are evaluated over 90 data points of cryotherapy dataset. For Heart dataset the hybrid algorithm KMeans-GOA provides minimum value of 0.000644 for 100 repetitions, 0.000644 for 500 repetitions and 0.000478 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.86 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.90 for 100, 500 and 1000 repetitions for purity index. KMeans-GOA provides minimum value of 0.0001261 for 100 repetitions, 0.0001231 for 500 repetitions and 0.0003131 for 1000 repetitions for standard deviation. The outcomes tabulates in Table 6 are evaluated over 297 data points of heart dataset. For Breast Cancer dataset the hybrid

algorithm KMeans-GOA provides minimum value of 0.000650 for 100 repetitions, 0.000661 for 500 repetitions and 0.000686 for 1000 repetitions for sum of average of intracluster distances. KMeans-GOA also provides the maximum value of 0.93 for 100, 500 and 1000 repetitions for F-measure and the maximum value of 0.96 for 100, 500 and 1000 repetitions for purity index. KMeans-GOA provides negligible value of 0.0002269 for 100 repetitions, 0.0002195 for 500 repetitions and 0.0002074 for 1000 repetitions for standard deviation where K-Means and KMeans-ALO provide value of zero for 100, 500 and 1000 repetitions for standard deviation. The outcomes tabulates in Table 7 are evaluated over 116 data points of Breast cancer dataset.

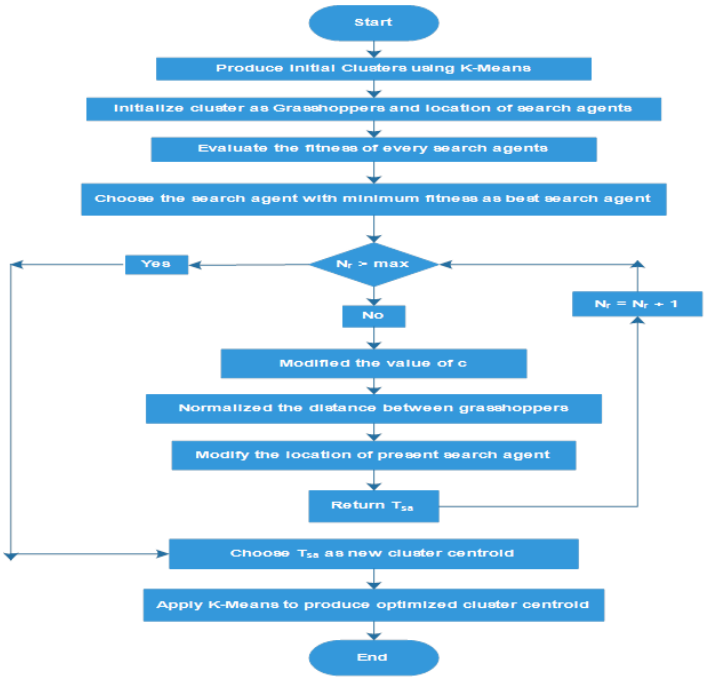

***Figure 1**. Flow Chart of Hybrid KMeans-GOA algorithm*

***Table2:** Outcomes of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA Algorithms for Glass Data set for 100, 500 and 1000 repetitions*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity Index
		Best Value	Average Value	Worst Value			
K-Means	100	0.5700	0.7622	1.0928	0.66	0.2644355	0.73
KMeans-PSO	100	0.5612	0.7459	0.9321	0.69	0.1854505	0.75
KMeans-ALO	100	0.0560	0.0640	0.0784	0.71	0.0113513	0.76
KMeans-GOA	100	0.000384	0.000467	0.000577	0.74	0.00009	0.79
K-Means	500	0.5817	0.7668	1.1118	0.66	0.2690393	0.73
KMeans-PSO	500	0.5630	0.7370	0.9655	0.69	0.2018640	0.75
KMeans-ALO	500	0.0589	0.0680	0.0803	0.71	0.0107398	0.76
KMeans-GOA	500	0.000388	0.000429	0.000574	0.74	0.00000977	0.79
K-Means	1000	0.5826	0.7308	0.9726	0.66	0.1968631	0.73
KMean	100	0.5575	0.7204	0.927	0.69	0.185	0.75
s-PSO	0			2		2838	
KMeans-ALO	1000	0.0548	0.0661	0.0757	0.71	0.0104615	0.76
KMeans-GOA	1000	0.000370	0.000425	0.000578	0.74	0.0001077	0.79

***Table3:** Outcomes of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA Algorithms for Immunotherapy Data set for 100, 500 and 1000 repetitions*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity index
		Best Value	Average Value	Worst Value			
K-Means	100	58.2632	58.2632	58.2632	0.86	0	0.88
KMeans-PSO	100	0.3981	0.4866	0.6884	0.87	0.148789	0.89
KMeans-ALO	100	0.0615	0.0615	0.0615	0.88	0	0.91
KMeans-GOA	100	0.000438	0.000701	0.000787	0.89	0.0001818	0.92
K-Means	500	58.2632	58.2632	58.2632	0.86	0	0.88
KMeans-PSO	500	0.3981	0.4866	0.6884	0.87	0.148789	0.89
KMeans-ALO	500	0.0615	0.0615	0.0615	0.88	0	0.91
KMeans-GOA	500	0.000455	0.000501	0.000603	0.89	0.000075	0.92
K-Means	1000	58.2632	58.2632	58.2632	0.86	0	0.88
KMeans-PSO	1000	0.3981	0.4866	0.6884	0.87	0.148789	0.89
KMeans-ALO	1000	0.0615	0.0615	0.0615	0.88	0	0.91
KMeans-GOA	1000	0.000443	0.000702	0.000782	0.89	0.000177	0.92

***Table4:** Outcomes of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA Algorithms for Soybean Data set for 100, 500 and 1000 repetitions*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity index
		Best Value	Average Value	Worst Value			
K-Means	100	0.2392	0.2960	0.4030	0.87	0.0831721	0.89
KMeans-PSO	100	0.1775	0.3543	0.5234	0.88	0.1729642	0.90
KMeans-ALO	100	0.0261	0.0304	0.0344	0.90	0.0041509	0.92
KMeans-GOA	100	0.000288	0.000308	0.000370	0.93	0.000042	0.95

K-Means	500	0.2475	0.3292	0.3737	0.87	0.0640072	0.89
KMeans-PSO	500	0.1775	0.3956	0.5311	0.88	0.1784006	0.90
KMeans-ALO	500	0.0241	0.0300	0.0344	0.90	0.0051681	0.92
KMeans-GOA	500	0.000277	0.000300	0.000383	0.93	0.000055	0.95
K-Means	1000	0.2475	0.3292	0.3958	0.87	0.0742780	0.89
KMeans-PSO	1000	0.1600	0.3325	0.4937	0.88	0.1668818	0.90
KMeans-ALO	1000	0.0241	0.0278	0.0351	0.90	0.0055973	0.92
KMeans-GOA	1000	0.000231	0.000280	0.000319	0.93	0.000044	0.95

TABLE5: *OUTCOMES OF K-MEANS, KMEANS-PSO, KMEANS-ALO AND KMEANS-GOA ALGORITHMS FOR CRYOTHERAPY DATA SET FOR 100, 500 AND 1000 REPETITIONS*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity index
		Best Value	Average Value	Worst Value			
K-Means	100	28.6100	28.6100	28.6100	0.77	0	0.83
KMeans-PSO	100	0.2688	0.3265	0.3853	0.86	0.0582508	0.91
KMeans-ALO	100	0.0693	0.0693	0.0693	0.87	0	0.92
KMeans-GOA	100	0.000476	0.000492	0.000702	0.91	0.0001261	0.96
K-Means	500	28.6100	28.6100	28.6100	0.77	0	0.83
KMeans-PSO	500	0.2688	0.3853	0.5522	0.86	0.1424449	0.91
KMeans-ALO	500	0.0693	0.0693	0.0693	0.87	0	0.92
KMeans-GOA	500	0.000470	0.000492	0.000702	0.91	0.0001280	0.96
K-Means	1000	28.6100	28.6100	28.6100	0.77	0	0.83
KMeans-PSO	1000	0.2688	0.3853	0.5522	0.86	0.1424449	0.91
KMeans-ALO	1000	0.0693	0.0693	0.0693	0.87	0	0.92
KMeans-GOA	1000	0.000456	0.000492	0.000702	0.91	0.0001328	0.96

TABLE6: *OUTCOMES OF K-MEANS, KMEANS-PSO, KMEANS-ALO AND KMEANS-GOA ALGORITHMS FOR HEART DATA SET FOR 100, 500 AND 1000 REPETITIONS*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity index
		Best Value	Average Value	Worst Value			
K-Means	100	16.2993	17.4666	19.1326	0.75	0.4239460	0.79
KMeans-PSO	100	0.6839	0.7424	0.7647	0.77	0.0417296	0.82
KMeans-ALO	100	0.0552	0.0564	0.0575	0.81	0.0011503	0.85
KMeans-GOA	100	0.000644	0.000827	0.000886	0.86	0.0001261	0.90
K-Means	500	14.4647	16.4666	16.9962	0.75	0.335201	0.79
KMeans-PSO	500	0.6900	0.7068	0.7647	0.77	0.0391891	0.82
KMeans-ALO	500	0.0552	0.0570	0.0586	0.81	0.001701	0.85
KMeans-GOA	500	0.000644	0.000827	0.000886	0.86	0.0001261	0.90
K-Means	1000	15.1237	16.4963	19.1087	0.75	0.243883	0.79
KMeans-PSO	1000	0.6768	0.7456	0.8133	0.77	0.0682507	0.82
KMeans-ALO	1000	0.0552	0.0571	0.0587	0.81	0.0017521	0.85
KMeans-GOA	1000	0.000478	0.000726	0.0011	0.86	0.0003131	0.90

TABLE7: *OUTCOMES OF K-MEANS, KMEANS-PSO, KMEANS-ALO AND KMEANS-GOA ALGORITHMS FOR BREAST CANCER DATA SET FOR 100, 500 AND 1000 REPETITIONS*

Algorithms	Repetitions	Intracluster Distance			F-Measure	Standard Deviation	Purity index
		Best Value	Average Value	Worst Value			
K-Means	100	139.1469	139.1469	139.1469	0.85	0	0.87
KMeans-PSO	100	0.3227	0.4475	0.5824	0.86	0.1298827	0.88
KMeans-ALO	100	0.0482	0.0482	0.0482	0.88	0	0.91
KMeans-GOA	100	0.000650	0.000824	0.0011	0.93	0.0002269	0.96
K-Means	500	139.1469	139.1469	139.1469	0.85	0	0.87
KMeans-PSO	500	0.2655	0.4475	0.5824	0.86	0.1590322	0.88
KMeans-ALO	500	0.0482	0.0482	0.0482	0.88	0	0.91
KMeans-GOA	500	0.000661	0.000879	0.0011	0.93	0.0002195	0.96
K-Means	1000	139.1469	139.1469	139.1469	0.85	0	0.87
KMeans-PSO	1000	0.2655	0.3493	0.5824	0.86	0.1642070	0.88
KMeans-ALO	1000	0.0482	0.0482	0.0482	0.88	0	0.91
KMeans-GOA	1000	0.000686	0.000869	0.0011	0.93	0.0002074	0.96

TABLE8: *AVERAGE RANKING OF K-MEANS, KMEANS-PSO, KMEANS-ALO AND KMEANS-GOA ALGORITHMS BASED ON AVERAGE OF SUM OF INTRACLUSTER DISTANCE*

Data Set	K-Means	KMeans-PSO	KMeans-ALO	KMeans-GOA
Glass	0.5826 (4)	0.5575 (3)	0.0548 (2)	0.000370 (1)
Immunotherapy	58.2632(4)	0.3981 (3)	0.0615 (2)	0.000443 (1)
Soybean	0.2475 (4)	0.1600 (3)	0.0241 (2)	0.000231 (1)
Cryotherapy	28.6100 (4)	0.2688 (3)	0.0693 (2)	0.000456 (1)
Heart	15.1237 (4)	0.6768 (3)	0.0552 (2)	0.000478 (1)
Breast Cancer	139.1469 (4)	0.2655 (3)	0.0482 (2)	0.000686 (1)
Average Rank (AR_y)	4	3	2	1

Ranks are allotted to all algorithms on the basis of their performance in terms of intracluster distance ranged from 1 to

4. Rank 1 is allotted to the algorithm having minimum measure of average of sum of intracluster distance and rank 4 is allotted to the algorithm having maximum measure of average of sum of intracluster distance. Each algorithm is assigned a rank (in brackets) consequently for all datasets (Table 8). The Average Rank AR_y of the y^{th} algorithm is evaluated using Eq. 17.

$$AR_y = \frac{Addition\ of\ total\ Ranks\ generated\ by\ y^{th}\ algorithm}{Total\ Number\ of\ Datasets} \quad (17)$$

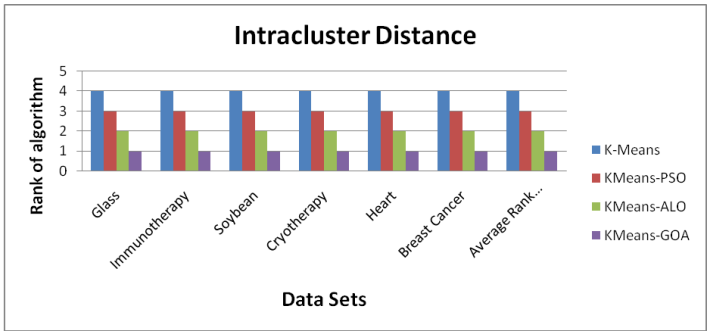

***Figure 2**. Analysis of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA of intracluster distances for all 6 datasets based on average rank of the algorithms*

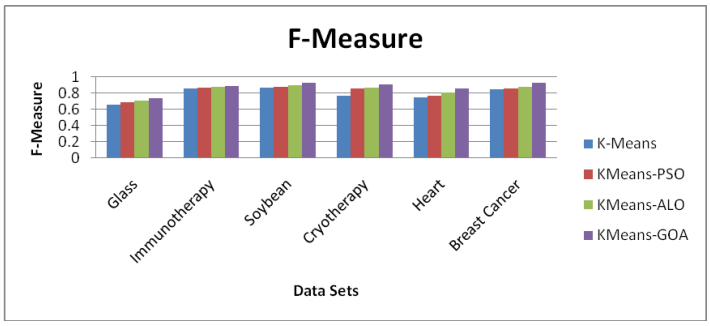

***Figure3.** Analysis of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA of F-Measures for all 6 datasets*

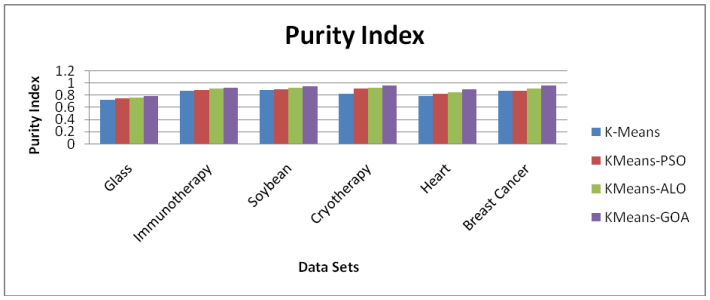

***Figure4.** Analysis of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA of Purity Index for all 6 datasets*

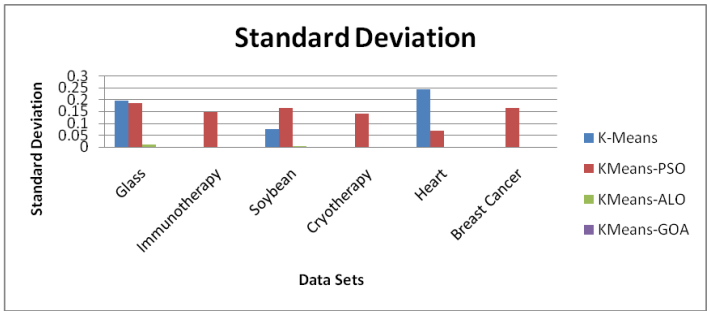

***Figure5.** Analysis of K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA of Standard Deviation for all 6 datasets*

The outcomes in Table 2 to Table 8 represent that KMeans-GOA gives better results (minimum intracluster distance, maximum F-measure, maximum purity index and minimum standard deviation) as compared to KMeans, Kmeans-PSO and KMeans-ALO algorithms. Figure 2 to Figure 5 shows the comparative analysis of the K-Means, KMeans-PSO, KMeans-ALO and KMeans-GOA for all 6 datasets on the basis of intracluster distance, F-Measure, Purity Index and Standard Deviation. The KMeans-GOA provides high quality exploration and local optimum prevention to improve the grasshopper`s repulsive rate for enhancing the search space comprehensively. Therefore KMeans-GOA gives better performance than other algorithms measured by rapidly converging for global optimum.

3.1 Time Complexity of KMeans-GOA Clustering Algorithm

It is a runtime complexity to perform the clustering algorithm depending on the input values to the algorithm. The evaluation of time complexity of KMeans-GOA, input values are taken as follows: N_r =number of repetitions, K =Number of clusters, P_T = Total population of grasshoppers, N_G =number of grasshoppers, N_{sa} =number of search agents, DS =dataset, N_I =number of instances in dataset, N_A = number of attributes in dataset. The cost to perform each step is supposed to 1 unit. Then the total number of operations for KMeans-GOA is calculated from an algorithm (section 2.1).

Total number of operations performed = $(N_r+1) + N_r*(P_T+1) + N_r*P_T*K + N_r*K + N_r*K + N_r*K + K+1 + K*P_T + K*N_{sa} + K*(N_r+1) + K*N_r + N_r*K*(N_G+1) + N_r*K*N_G + N_r*K*N_G + N_r*K*N_G + K*N_r + K+1$

Total number of operations performed= $N_r*P_T*K + 4N_r*K*N_G + 7N_r*K + K*P_T + N_r*P_T + K*N_{sa} + 3K + 2N_r + 3$ (18)

There is N_I*N_A extra operations are performed to access the N_I instances and N_A attributes of datasets. Therefore eq. 18 can be modified to calculate the total cost of performance of the KMeans-GOA and obtained cost function (CF) eq. 19.

CF = $N_r * P_T * K * N_I * N_A + 4\ N_r * K * N_G + 7 N_r * K + K * P_T * N_I * N_A + N_r * P_T + K * N_{sa} + 3K + 2N_r + 3$ (19)

It is to be assumed that all input parameters are equal in eq. 19 for worst case complexity, then eq. 20 is obtained.

$$CF = n^5 + n^4 + 4n^3 + 9n^2 + 5n + 3 \quad (20)$$

The worst case time complexity of K-Means is $O(n^2)$, KMeans-ALO is $O(n^5)$ and KMeans-GOA is $O(n^5)$. It means all K-Means, KMeans-ALO and KMeans-GOA are solvable in polynomial time.

3.2 Assessment of Statistical Analysis

A Statistical test is applied to evaluate the continuation of consequence dissimilarities among the efficiency of the clustering algorithms. In this work a non parametric Friedman test has been used to find dissimilarities among the set of ordinal relevant variables. The null hypothesis H_0 is described that all the clustering algorithms have equal efficiency.The Friedman test equation (FM) is represented by eq. 21.

$$FM = \frac{12 N_d}{K_a(K_a+1)}\left[\sum_{y=1}^{4}(AR_y)^2 - \frac{K_a(K_a+1)^2}{4}\right] \quad (21)$$

Here N_d = the number of datasets, K_a = number of used clustering algorithms and AR_y = Average rank of clustering algorithms.

The critical value of FM for 6 datasets (N_d = 6) and 4 algorithms (K_a = 4) is 6.40 [25] for level of confidence α =0.10. If the estimated FM value is larger than the critical value, then the null hypothesis is to be rejected otherwise accepted. The value of FM is 18 which is calculated by using eq. 21 for 6 datasets (N_d = 6) and 4 algorithms (K_a = 4) and α =0.10. It shows that the estimated FM is larger than critical FM, so null hypothesis is rejected. Therefore, all the clustering algorithms have not equal efficiency, it can be concluded. After that Holm`s process is applied as the post hoc test. In this test the performance of proposed algorithm KMeans-GOA is evaluated and compared statistically with other algorithms. To perform the test, eq. 22 is used to calculate the z value and on the basis of z value, probability p is obtained from the normal distribution table [26]. After that the p_x value is compared with $\alpha/(K_a - x)$ (table 9).

$$z = \frac{AR_x - AR_y}{S_e} \quad (22)$$

Where, $S_e = \sqrt{\frac{K_a(K_a+1)}{6N_d}}$ (23)

TABLE9: *RESULTS GENERATED FROM HOLM`S PROCESS*

X	Algorithms	z-value	p-value	$\alpha/(K_a - x)$	Hypothesis
1	K-Means	-3.9195	0.00005	0.03333	Rejected
2	KMeans-PSO	-2.6831	0.00368	0.05	Rejected
3	KMeans-ALO	-1.3416	0.09012	0.10	Rejected

Table 9 represents that the p_x value is lesser than $\alpha/(K_a - x)$ so hypothesis is rejected for all cases. Therefore, it can be concluded that the performance and efficiency of KMeans-GOA is better than K-Means, KMeans-PSO and KMeans-ALO algorithms.

4 CONCLUSION

K-Means provides fast, high quality, easily implemented, and efficient organized clustering. Though, the sensitive nature of the selection of initial cluster centroid and convergence to the local optima is major disadvantages of KMeans algorithm. Here, an optimization technique Grasshopper Optimization Algorithm (GOA) is combined with K-Means to improve the performance of clustering. The proposed KMeans-GOA has performed experiments on 6 multiple datasets and the efficiency of the algorithm are analyzed based on several performance factors like intracluster distance, purity index, F-measure and standard deviation. The experimental results represent that the proposed KMeans-GOA is performed better quality of clustering than other clustering algorithm to provide minimum sum of average of intracluster distance, maximum purity index, maximum F-Measure and minimum standard deviation as compared to the K-Means, KMeans-PSO and KMeans-ALO clustering algorithm. The Friedman test indicates the continuation of considerable differences among Kmeans, Kmeans-PSO, and Kmeans-ALO. Moreover, Holm`s test illustrates that Kmeans-GOA obtains greater efficiency than K-Means, KMeans-PSO, and Kmeans-ALO. The statistical analysis uses the value of 0.10 for level of confidence, hence the outcomes of proposed KMeans-GOA obtains 90% accuracy.

5 REFERENCES

[1] K, and M. K. Nair, “FGPSO - A Novel Algorithm for Multi Objective Data Clustering”, WSEAS Transactions on Computers, Vol. 17, pp-1-9, 2018.

[2] Ekbal, S. Saha, D. Moll´a and K Ravikumar, “Multi-Objective Optimization for Clustering of Medical Publications”, In Proceedings of Australasian Language Technology Association Workshop, pp-53-61, 2013.

[3] Sreedhar, N. Kasiviswanath and P. C. Reddy, “Clustering large datasets using K-means modified inter and intra clustering (KM-I2C) in Hadoop”, Journal of Big Data, Vol. 27, pp-1-19, 2017.

[4] Gong, H. Chen, W. He, and Z. Zhang, “Improved multi-objective clustering algorithm using particle swarm optimization”, PLOS ONE, pp-1-19, 2017. (https://doi.org/10.1371/journal.pone.0188815)

[5] Naaz, D. Sharma, D. Sirisha, and V. M, “Enhanced K-Means Clustering Approach for Health Care Analysis Using Clinical Documents”, International Journal of Pharmaceutical and Clinical Research (IJPCR), Vol. 8, No. 1, pp-60-64, 2016.

[6] G. A. Kiran, M. Puri and S. S. Suresh, "PSO-Enabled Privacy Preservation of Data Clustering", Indian Journal of Science and Technology, Vol. 10, No. 11, pp-1-10, 2017. (DOI: 10.17485/ijst/2017/v10i11/89318)

[7] H. Bangui, M. Ge, and B. Buhnova, "Exploring Big Data Clustering Algorithms for Internet of Things Applications", In Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security (IoTBDS), pp-269-276, 2018.

[8] J. Nasiri and F. M. Khiyabani, "A whale optimization algorithm (WOA) approach for clustering", Cogent Mathematics & Statistics, Vol. 5, pp-1-13, 2018. (https://doi.org/10.1080/25742558.2018.1483565)

[9] M. R. Murty, A. Naik, J. V. R. Murthy, P. V. G. D. Prasad Reddy, S. C. Satapathy, and K. Parvathi, "Automatic Clustering Using Teaching Learning Based Optimization", Applied Mathematics, Vol. 5, pp-1202-1211, 2014. (http://dx.doi.org/10.4236/am.2014.58111)

[10] M. Lashkari and M. H. Moattar, "Improved COA with Chaotic Initialization and Intelligent Migration for Data Clustering", Journal of AI and Data Mining (JAIDM), Vol. 5, No 2, pp-293-305, 2017.

[11] M. Lashkari and A. Rostami, "Extended PSO Algorithm For Improvement Problems K-Means Clustering Algorithm", International Journal of Managing Information Technology (IJMIT), Vol.6, No.3, pp-1-13, 2014.

[12] M. A. Nemmich, F. Debbat, and M. Slimane, "A Data Clustering Approach Using Bees Algorithm with a Memory Scheme", Springer Nature Switzerland AG, pp-261–270, 2019. (https://doi.org/10.1007/978-3-319-98352-3_28)

[13] P. Nerurkar, A. Shirke, M. Chandane, and S. Bhiru, "A Novel Heuristic for Evolutionary Clustering "6th International Conference on Smart Computing and Communications, ICSCC, Kurukshetra, India, pp-780-789, 2017.

[14] P. Vaijayanthi, X. S. Yang, N. A M and R. Murugadoss, "High Dimensional Data Clustering Using Cuckoo Search Optimization Algorithm", International Journal of Advanced Computer Engineering and Communication Technology (IJACECT), Vol. 3, Issue 3, pp-1-5, 2014.

[15] R. K. Saidala, and N. Devarakonda, "Multi-Swarm Whale Optimization Algorithm for Data Clustering Problems using Multiple Cooperative Strategies", I.J. Intelligent Systems and Applications, MECS, Vol. 8, pp-36-53, 2018.

[16] S. Bandyopadhyay, U. Maulik and M. K. Pakhira, "Clustering Using Simulated Annealing With Probabilistic Redistribution", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 15, No. 2, pp-269-285, 2001.

[17] S. K. Majhi, and S. Biswal, "Optimal cluster analysis using hybrid K-Means and Ant Lion Optimizer", Karbala International Journal of Modern Science, pp-347-360, 2018.

[18] S. Karol, and V. Mangat, "Evaluation of text document clustering approach based on particle swarm optimization", Cent. Eur. J. Comp. Sci., Vol. 3, No. 2, pp-69-90, 2013.

[19] V. Prasad, S. Madhusudanan and S. Jaganathan, "uCLUST-a new algorithm for clustering unstructured data", ARPN Journal of Engineering and Applied Sciences, Vol. 10, No. 5, pp-1-11, 2015.

[20] T. Shanie, J. Suprijadi, and Zulhanif, "Text Grouping in Patent Analysis using Adaptive K-Means Clustering Algorithm", Statistics and its Applications, pp-1-10, 2017. (doi: 10.1063/1.4979457)

[21] V. Shanu, and S.Vydehi, "Optimal and Fast Health Data Clustering Using Hybrid Meta Heuristic Algorithm", International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), Vol. 5, Issue 7, pp-13339-13347, 2017. (DOI: 10.15680/IJIRCCE.2017. 0507069)

[22] W. M. S. Yafooz, "Model of Textual Data Linking and Clustering in Relational Databases", Research Journal of Information Technology (RJIT), pp-1-12, 2016.

[23] W. L. Chang, J. Kanesan, A. J. Kulkarni, and H. Ramiah, "Data clustering using seed disperser ant algorithm", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 25, pp-4522 -4532, 2017.

[24] Z. Dong, H. Jia, and M. Liu, "An Adaptive Multiobjective Genetic Algorithm with Fuzzy c-Means for Automatic Data Clustering", Hindawi Mathematical Problems in Engineering, pp-1-13, 2108. (https://doi.org/10.1155/2018/6123874)

[25] F Distribution Table, 2018 Mar 18. Retrieved from http://www.socr.ucla.edu/applets.dir/f_table.html

[26] Normal Distribution Table. Retrieved from http://math.arizona.edu/~rsims/ma464/standardnormaltable.pdf.