Hierarchical Document Clustering via Optimal Centroid Selection using Adaptive Pillar K-means–Gaussian Firefly Algorithm

Nagalashmi P  
Panacorp Software Solutions  
India  
nagaleship17@gmail.com

ABSTRACT: In recent days, data growth is enormous and tough to handle it, in an organized manner and in apt time. In data mining, wide range of researches are available for managing data effectively. Document clustering is a spirited zone in data mining and here our main objective is to assemble the related documents. In this paper, we generate an algorithm for clustering by means of Adaptive Pillar K-Means and Gaussian Firefly Algorithm. For determining the proper centroid in order to attain the proper clustered documents, Adaptive Pillar K-means algorithm is utilized. Subsequently, Gaussian firefly algorithm is exploited for the optimization process and also for enhancing the precision that results in reducing the sum of squared errors and computational time. Here, the performance of the proposed methodology is compared with various algorithms such as Genetic Algorithm, Ant colony optimization and gravity clustering. The attained results show the performance of the proposed methodology and the simulation results illustrated the betterment in quality with low sum of squared errors.

Keywords: Document Clustering, Pillar K-means, Centroid Selection, Gaussian Firefly Algorithm, Sum of the Squared Errors

Received: 25 August 2018, Revised 4 October 2018, Accepted 16 October 2018

DOI: 10.6025/jic/2019/10/1/1-14

© 2019 DLINE. All Rights Reserved

1. Introduction

Clustering or cluster analysis is one of the main data analysis techniques and deals with the organization of a set of objects in a multidimensional space into cohesive groups, called clusters [1]. Each cluster contains objects that are very similar to each other and very dissimilar to objects in other clusters [2]. Document clustering (or text clustering) is the application of cluster analysis to textual documents. It has applications in automatic document organization, topic extraction and fast information retrieval or filtering [3]. The Web is the largest information repository in the history of mankind. Finding the relevant information on www is not an easy task. Clustering is one of the Data Mining techniques to improve the efficiency in information finding process [4]. Document clustering has become an increasingly important technique for enhancing search engine results, web crawling, unsupervised document organization, and information retrieval or filtering [5]. Clustering is a division of data into groups of similar objects. It can be performed in two different modes, hard and soft. In hard clustering the clusters are disjointed and non-overlapping in nature. In the case of soft clustering, a pattern may belong to any or all of the classes with various membership grades [6].
A cluster consists of objects that are similar between themselves and dissimilar to objects of other groups [7]. In other words, the goal of a good document clustering scheme is to minimize intra-cluster distances between documents, while maximizing inter-cluster distances. A distance measure thus lies at the heart of document clustering [8]. The partition clustering algorithms, such as the family of k-means, are reported performing well on document clustering [9]. Document clustering techniques have been applied in several areas, with the web as one of the most recent and influential [10]. Both general-purpose and text-oriented techniques exist and can be used to cluster a collection of documents in many ways. Usually, if the dataset consists of some clusters then, the majority of the intra-cluster distances will be smaller than the majority of the inter-cluster distances [11].

Web Search Engines certainly return a ranked list of documents that are relevant to the query submitted by the user [12]. For each document, its title, URL and snippet are usually presented. The problem becomes harder if the user cannot guess additional words for restricting his query. One solution to these problems is results clustering which provides a quick overview of the search results. It aims at grouping the results into topics, called clusters, with predictive names (labels), aiding the user to locate quickly documents that [13]. Data clustering techniques can be considered a method for finding similarities in data sets and grouping similar information [14]. The methods of clustering are (a) Hierarchical, (b) Partitioning, (c) Density-based, and (d) Grid-based [15]. K-means is a popular unsupervised learning algorithm that is used in a wide range of applications, such as data mining, because of its simplicity. “K-Means clustering” is a partitioning method [16]. The purpose of k-means clustering is to separate n observations into k clusters so that each observation is assigned to the cluster with the nearest mean [17].

The partitional clustering divides datasets into clusters into a single level. The center-based clustering is one of the most popular partitional clustering approaches and K-means is one type of it [18]. In K-means algorithm, clusters are formed by minimizing the objective function which it is the sum of distance between the center of cluster and each item in the cluster. Even though K-means is a simple method and is efficient for large dataset, it suffers from the local optima problem. Examples of Swarm Intelligent algorithms include the Artificial Bee Colony (ABC) [19], Cuckoo Search Optimization algorithm, Ant Colony Optimization and particle swarm optimization. These types of Swarm Intelligent algorithms have been utilized in text clustering; however, they need to predefine the number of k clusters. The determination of the k number of clusters is considered a problem as a user may not have any knowledge about the dataset prior to clustering [20] [21].

In this research we are concentrating on clustering the documents and finding the initial values of clusters. One of the efficient method used in document clustering is K-means. Clustering depend on the initial values of clustered documents. Initial value of cluster centers is calculated by firefly algorithm so for the process of finding the initial values we proposed the firefly algorithm. The rest of the paper is organized as follows: Section 2 provides the related work in document clustering, while section 3 includes description on the standard K-means and Firefly algorithm and the results and comparison are presented in section 4, and finally conclusion of the work is made in section 5.

2. Related Researches

Moreno Carullo et al. [22] have proposed a heuristic online document clustering model that can be specialized with a variety of text-oriented similarity measures. An experimental evaluation of the proposed model was conducted in the e-commerce domain. Performances were measured using a clustering-oriented metric based on F-Measure and compared with those obtained by other well-known approaches. The obtained results confirmed the validity of the proposed method both for batch scenarios and online scenarios where document collections can grow over time.

Abdolreza Hatamlou [23] has proposed a heuristic algorithm that is inspired by the black hole phenomenon. Similar to other population-based algorithms, the black hole algorithm (BH) started with an initial population of candidate solutions to an optimization problem and an objective function that was calculated for them. At each iteration of the black hole algorithm, the best candidate was selected to be the black hole, which then started pulling other candidates around it, called stars. If a star got too close to the black hole, it would be swallowed by the black hole and was gone forever. In such a case, a new star (candidate solution) was randomly generated and placed in the search space and starts a new search. To evaluate the performance of the black hole algorithm, it was applied to solve the clustering problem, which was a NP-hard problem. The experimental results have shown that the proposed black hole algorithm outperformed other traditional heuristic algorithms for several benchmark datasets.

Daniel Ramage et al. [24] have explored the use of tags in 1) K-means clustering in an extended vector space model that includes
tags as well as page text and 2) a novel generative clustering algorithm based on latent Dirichlet allocation that jointly models text and tags. They evaluated the models by comparing their output to an established web directory. They found that the naive inclusion of tagging data improved cluster quality versus page text alone, but a more principled inclusion can substantially improve the quality of all models with a statistically significant absolute F-score increase of 4%. The generative model outperformed K-means with another 8% F-score increase.

Reynaldo Gil-García and Aurora Pons-Porrata [25] have presented two clustering algorithms called dynamic hierarchical compact and dynamic hierarchical star. Both methods aimed to construct a cluster hierarchy, dealing with dynamic data sets. The first created disjoint hierarchies of clusters, while the second obtained overlapped hierarchies. The experimental results on several benchmark text collections have shown that these methods not only were suitable for producing hierarchical clustering solutions in dynamic environments effectively and efficiently, but also offered hierarchies easier to browse than traditional algorithms. Therefore, they advocated its use for tasks that required dynamic clustering, such as information organization, creation of document taxonomies and hierarchical topic detection.

Gianfranco Chicco [26] has provided an overview of the clustering techniques used to establish suitable customer grouping, included in a general scheme for analyzing electrical load pattern data. The characteristics of the various stages of the customer grouping procedure were illustrated and discussed, providing links to relevant literature references. The specific aspect of assessing the performance of the clustering algorithms for load pattern grouping was then addressed, showing how the parameters used to formulate different clustering methods impact on the clustering validity indicators. It emerged that the clustering methods able to isolate the outliers exhibit the best performance. The implications of this result on the use of the clustering methods for electrical load pattern grouping from the operator’s point of view were discussed.

David Sculley [27] has presented two modifications to the popular k-means clustering algorithm to address the extreme requirements for latency, scalability, and sparsity encountered in user-facing web applications. First, he proposed the use of mini-batch optimization for k-means clustering. This reduced computation cost by orders of magnitude compared to the classic batch algorithm while yielding significantly better solutions than online stochastic gradient descent. Second, he achieved sparsity with projected gradient descent, and gave a fast accurate projection onto the L1-ball.

3. Hybridization of Pillar k-means and Gaussian Firefly algorithm for Clustering Documents

Document clustering plays a crucial role in various fields such as data mining, information retrieval and more. An automatic learning and automated process of grouping documents is termed as Document clustering. Here relevant documents are combined together based on the nominated initial values. Variety of algorithms are exploited for the clustering process, but in this work, we develop an effective hybrid document clustering mentioned as Pillar k-means and Gaussian Firefly Algorithm. In our work, we amalgamate the relevant documents and hence to improve the performance by minimizing the sum of squared errors, time and thereby enhancing the precision. The proposed methodology is depicted in the following figure 1 and the proposed methodology is detailed below.

![Figure 1. Process of Document Clustering](image-url)
In our proposed methodology, initially pillar k means is utilized which is an effective method and that selects optimum centroid for clustering documents. On the other hand, Pillar K-means has a tendency to converge to local optima and depends upon the initial value of cluster centers. In order to improve the efficiency of the methodology, Gaussian Firefly algorithm is utilized. Here each document is assumed as a single firefly and the entire weight of the document is the initial brightness of the firefly.

Here the optimum firefly is opted as the firefly with brightest light intensity (weight) and during iteration this can be updated with brightest rather than the existing firefly. This operation is identified to be a new approach in utilizing the Gaussian firefly in document clustering optimization. The simulation results are demonstrated in order to prove the quality of the proposed methodology in terms of reduced and reduced sum of squared errors (SSE) and improved precision. The proposed methodology is compared with the well known standard clustering algorithms like as genetic algorithm (GA) and ant colony optimization (ACO) algorithm and gravity clustering (GC).

3.1 Clustering
In data mining, clustering and classification are the essential process and here we have considered clustering process. Clustering is the process of combining a group of abstract objects into classes of similar objects. In other words, it is the process of constructing a group of related objects together with the intention of amalgamating analogous objects together in a class and diverse objects in another class. The intention of our research is document clustering and various methods and algorithms are employed for clustering documents. Here we propose an proficient hybrid evolutionary data clustering technique mentioned as Pillar K-Means blending with Gaussian Firefly algorithm. This process is detailed below.

![Flow chart for optimization algorithm](image-url)
As per the above figure, initially, the clusters can be parted with the aid of adaptive pillar K-means algorithm and here the main idea is to determine the initial values of clusters hence we propose an optimization algorithm denoted as Gaussian Firefly Algorithm (GFA). For clustering huge datasets, the pillar K-Means is apposite algorithm and which splits the high dimensional documents into various clusters and each defined by different prototype. Pillar K-means algorithm is utilized here to compute the centroid of document clusters and also to reduce the objective function with numerous iterations, the cluster are updated. With the intention of detecting related document based on the distance and brightness and then blending process is carried out based on the minimal distance.

3.2 Adaptive Pillar K- Means Clustering Algorithm

This algorithm is inspired by the thought process of determining a set of pillars’ locations in order to make a stable house or building. Fig. 2 illustrates the locating of two, three, and four pillars, in order to withstand the pressure distributions of several different roof structures composed of discrete points. It is inspiring that by distributing the pillars as far as possible from each other within the pressure distribution of a roof, the pillars can withstand the roof’s pressure and stabilize a house or building. It considers the pillars which should be located as far as possible from each other to withstand against the pressure distribution of a roof, as number of centroids among the gravity weight of data distribution in the vector space. Therefore, this algorithm designates positions of initial centroids in the farthest accumulated distance between them in the data distribution[]. Applications like data mining exploits pillar K-Means which is a prevalent unsupervised learning method. In terms of complexity, it is simple but its an effective technique exploited in data clustering.

Pillar K-Means is a popular unsupervised learning algorithm that is used in a wide range of applications, such as data mining. Pillar K-means is simple yet an efficient method used in data clustering. The Pillar K-means clustering algorithm is known to be efficient in clustering large data sets. This clustering algorithm is one of the simplest and the best known unsupervised learning algorithms that solve the well-known clustering problem. Since it is very simple and has fast convergence speed, it is widely used in data mining. The Pillar K-Means algorithm aims to partition a set of objects, based on their features, into k clusters, where k is a predefined or user-defined constant. In a general sense, Pillar k-means clustering works by assigning data points to a cluster. To run an iteration of pillar k-means on our dataset, we first randomly initialize k number of points to serve as cluster.

For initialize the Pillar K- means clustering technique divides the high dimensional documents into a number of clusters, each one defined by a prototype. The example of Pillar K-means clustering is shown in figure 2.

![Figure 3. Example diagram for Pillar K-means Clustering](image)

In the above figure 3 suggests five different sets of clusters that have shown clearly. In the above figure such raw documents are shown and in the second approach which separate and partition the documents using the Pillar K-means cluster algorithm. Pillar K means clustering intends to partition n set of objects into k clusters in which each object belongs to the cluster. As a result of this approach we may notice that the k clusters does not change their location. This method produces exactly k different clusters of greatest possible distinction. The objective of Pillar K means clustering is to partition a set of objects as calculated the formula as

\[
P_k = \sum_{j=1}^{k} \sum_{i=1}^{n} \left( dist(e_i, c_j) \right)^2
\]
where $n$ is the number of clusters, $dist(e_i, c_j)$ is the distance between two data sets, $e_i$ is the data point and $c_j$ is the cluster point from their respective cluster.

Algorithm 1: Pillar K-means Algorithm

**Input:** $E = \{e_1, e_2, \ldots, e_n\}$ (set of entities to be clustered)

$\quad k$ (number of clusters)

**Output:** $F_{center i}$ (Optimized Centroids)

Choose $k$ points at random selection

For each $i \in C$ do

$\quad \{e_1, e_2, \ldots, e_n\} \leftarrow$ select random number

$\quad c_j \leftarrow e_j \in E \leftarrow$ random selection

For $n \leftarrow 1$ to $n$

end

For each $e_i \in E$ do

$\quad e_i \leftarrow$ distance $(e_i, c_j) j \in \{1 \ldots k\}$

Stablecluster $(c_i)$;

end

for each $e_i \in E$ do

$\quad$ mindistant $(e_i, c_j) j \in \{1 \ldots k\}$

Finish in which $F_{center i}$ is the solution as optimized initial centroids.

end

Pillar K-means algorithm is to divide a set of data objects into predefined $K$ clusters. Initialization database of the Pillar K-means algorithm which in a database $C$ containing $N$ data objects, the data object is defined as

$$C = x_i \ (1 \leq i \leq N) \quad (2)$$

where $N$ is the total amount of data objects and $x$ is the data object. Perform the Pillar K-means clustering method to group documents of the user with different $k$ values.

$$F_{center i} = \sum_{k=1}^{C_i} P_k \ (P_k \in Cluster \ i) \quad (3)$$

where $F_{center i}$ centroid value for Adaptive Pillar K-means methods and $C_i$ is denoted as the cluster. $F_{center i}$ with highest values is considered and forms a new cluster for each user. The Adaptive Pillar K-means is a simple algorithm that has been adapted to many problem domains and it is good to work for a randomly generated data points. Gaussian firefly which is to optimize and improve the precision also which is used to reduce the sum of square errors and computational time. After complete the process finally, we update the centroid step by step which is explained in below.

3.3 Gaussian Firefly Algorithm (GFA)

Firefly Algorithm, is currently an active focus of research, where several modifications and improvements were recorded within
the past few years. Nature inspired algorithms are among the most powerful algorithms for optimization. One of the optimization algorithm that we chose in our paper is Gaussian Firefly Algorithm.

This proposed system utilizes Adaptive Pillar algorithm, which is very robust and superior for initial clusters optimization. First the Pillar K-means algorithm is to divide a set of data objects into predefined $k$ clusters and centroid that is shown in above part and next approach is an optimization and minimizes the sum of the squared errors (SSE) for clustering. To define a cluster centroids iteration for each cluster we proposed gaussian firefly algorithm (GF). It gradually changes the positions of the $k$ centers through iterative process, until the positions are changed based on GF algorithm, then the criterion function SSE is minimized.

Some modifications are done in general firefly algorithm that is the step length of the variable is changed simultaneously for each user firefly from one feedback session pseudo documents to another pseudo documents to perform clustering process. The clustering is performed based on the behaviour of firefly through association of documents based on the objective function from Pillar K-means clustering document. So all the documents in the search engine results moves from one place to another place cluster to perform best clustering based on user specified objective function. Based on this process entire best results are formed based on the user search goals for each documents. Proposed system GFA clustering methods for documents uses a weight value for each documents and rely on total number of iterations.

\[ W_{iter} = u + \left( \frac{(iter_{max} - iter)^p}{iter_{max}} \right) (v - u) \]  \hspace{1cm} (4)

$W_{iter}$ is the weight value of the documents for each user. The weight of Witer is assigned between two different user documents $u$ and $v$, $\alpha \in [0, 1]$ so $u = 0$ & $v = 1$. $n$ and $iter$ is specified. In equation (1) value is determined by: $n = 10^{(\text{dim})}$ If the value $n$ becomes low and number of iterations performed by system is too low, thus improves the clustering results of each user in web search engine system. In proposed algorithm GFA method is performed based on the random selection of the documents. Random selection of documents follows the procedure of random walking with consecutive selection of random number of documents in the feedback session. The random selection is performed based on the predicted step length value from (1); it finds best cluster pseudo documents for each user via Gaussian distribution is specified in the equation (2).

\[ p = f \left( \frac{u - \mu}{\delta} \right) = \frac{1}{\delta \sqrt{2\pi}} e^{-\frac{(u - \mu)^2}{2\delta^2}} \]  \hspace{1cm} (5)

where is $u$ an error $e$ of the cluster and fitness value firefly $i$.

\[ u = F_{\text{center}}(\text{gbest}) - F_{\text{center}} \]  \hspace{1cm} (6)

where, $\mu$ is mean and $\delta$ is standard deviation which is predefined as $\mu = 0$ and $\delta = 1$, it is associated to probability value is followed by:

\[ \mu_i = \mu_i + \alpha \times (1 - p) \times U(u, v) \]  \hspace{1cm} (7)

That $u, v \in [0, 1]$ behavior, if the new cluster data points are found based on the centroid value, it is selected as cluster data points for each user in the user feedback session. This way of finding best cluster data points in the feedback session is repeated until all the number of feedback session is completed.

4. Experiment Results and Comparison

This section shows the results obtained in the experimental study as well as the analysis based on them. The objective of the proposed model is to partition the document with high accurately and less time complexity using the proposed algorithm. The performance of the proposed algorithm is evaluated in different problems in the field of clustering. Existing some clustering algorithms may fail to find the best partitioning of data. In some clustering algorithms, larger size cluster may not split accurate, so sum of squared error criterion is applied for each iteration for standard clustering. To evaluate the performance of proposed
Algorithm 2: Gaussian Firefly Algorithm (GFA)-based Adaptive pillar K-means clustering

Step 1: Parameters Setup
The parameters include the size of population (N), light absorption coefficient (γ), attractiveness value (β₀), trade-off constant (α) and k clusters.

Step 2: Initialization
Generate N fireflies (Xᵢₙ) as the cluster’s centroids, which can be represented as:

\[ Xᵢₙ = (z_{iₙ₁}, ...., z_{iₙₗ}, ...., z_{iₙₖ}), \]

where denotes the jth cluster’s centroids for the ith firefly (1 ≤ i ≤ N) and (1 ≤ j ≤ k).

Step 3: Fitness Calculation
Calculate the fitness value for each firefly by using Euclidean distance.

Step 4: Attractiveness
In the Gaussian firefly algorithm, it is assumed that the attractiveness of a firefly is determined by its brightness, which is associated with the objective function of the optimization problem. The attractiveness of a firefly can be formulated as follows:

\[ \beta = \beta₀e^{-γrᵢ²}, \]

where \( \beta₀ \) is the attractiveness value at \( r = 0 \), \( γ \) is the light absorption coefficient at the source; and \( rᵢₖ \) is the Euclidean distance between \( xᵢ \) and \( xⱼ \):

\[ rᵢₖ = \left\| xᵢ - xⱼ \right\| = \left( \sum_{k=1}^{d} (xᵢₖ - xⱼₖ)^2 \right)^{1/2}, \]

Step 5: Firefly Movement
The movement of a firefly \( i \), which is attracted to another more attractive firefly \( j \), is determined by:

\[ xᵢ^{new} = xᵢ^{old} + \beta e^{-γrᵢ²} (xⱼ^{old} - xᵢ^{old}) + \alpha \left( rand - \frac{1}{2} \right), \]

where \( \alpha \) determines the random behavior of movement.

Step 6: Euclidean Distance Calculation
Calculate the Euclidean distance of all the \( x \) data vectors to all clusters’ centroids for each firefly.

Step 7: Assign New Cluster
Assign each data vector \( x \) to the closest cluster centroid.

Step 8: New Centroids
Recalculate the cluster centroids vector for each firefly by using:

\[ Cenᵣ^{new} = \frac{1}{nᵣ} \sum xᵣᵢ, \]

where \( Cenᵣ^{new} \) is the new centroid vectors of cluster \( d^{th} \) and \( nᵣ \) is the size of cluster \( d \), \( xᵣᵢ \) is \( i^{th} \) data in cluster \( d \).

Step 9: Check if the Number of Iterations is Satisfied
Stop if the number of iterations is satisfied; otherwise, go back to Step 3.
algorithm for document clustering, the datasets are collected from Machine Learning Laboratory[28]. To demonstrate the effectiveness of the proposed model, four kinds of comparisons are investigated in this research such as genetic algorithm (GA) and ant colony optimization (ACO) algorithm and gravity clustering (GC). By the way comparing to this, our proposed method give minimum SSE in each iteration, the high quality of solution. Our proposed method is implemented in the working platform MATLAB.

In this paper, the experimental images are shown in figure 4 and it shows how the document is clustered in minimum iteration with low SSE and get better solution. In this below their having the iteration level. We can identify that their have an minimum SSE. Moreover, the average numbers of iterations to obtain optimal solutions required by the proposed algorithm are lower than those of the other algorithms.

Figure 4. The experimental of the Pillar K-means with Gaussian firefly algorithm
Sum of the Squared Errors (SSE)

SSE is the sum of the squared errors of all objects in database C. It is more robust, because it minimizes a sum of dissimilarities. After each iteration to check if SSE is decreasing or not in our proposed algorithms. We need the minimum SSE during the iteration. First, the clustering performance strongly depends on the initial clustering centers, since the initial cluster centroids are selected by using the FA algorithm. In the large database it will lead to increment of the number of iterations required for convergence, so that computational complexity will increase and it is hard to obtain accurate clustering results. For this we need a minimum SSE. So for each iteration here SSE will applied for each iteration.

\[ SSE = \sum_{j=1}^{K} \sum_{x \in c_j} \text{dist}(c_j, x) \] (8)

From the formulas above, we can see that in each iteration of proposed algorithm, the cluster centers must be readjusted and all data objects need to be updated again. The updated result shown that the high quality of solution and the better convergence speed. The convergence speed which measures how fast of a sequence will converges.

Accuracy

The accuracy rate depends on the effectiveness of the algorithm in clustering related documents together. The accuracy rate of the proposed algorithm is comparatively better than other algorithms.

\[ \text{Accuracy} = \frac{\text{No of True Positive} + \text{No of True Negai}}{\text{No of True Positive} + \text{False Positive} + \text{False Negative} + \text{True Negative}} \]

In our paper we can defined the accuracy as

\[ \text{Accuracy} = \frac{c.a.d + c.r.d}{\text{total documents}} \]

where \(c.a.d\) is the correctly accepted document and \(c.r.d\) is the correctly rejected document by the clustering algorithm. dac..drc..

4.1 Comparison

To demonstrate the effectiveness of the proposed model, three kinds of comparisons are investigated in this research such as genetic algorithm (GA) and ant colony optimization (ACO) algorithm and gravity clustering (GC)

| Algorithm    | SSE      | Time taken for computation | Accuracy  |
|--------------|----------|----------------------------|-----------|
| ACO          | 9.8158   | 24.0928 sec                | 97.03%    |
| GC           | 7.1635   | 34.6802 sec                | 96.33%    |
| GA           | 8.1092   | 10.4002 sec                | 97.33%    |
| Proposed algorithm | 6.4623 | 4.4326 sec                | 98.78%    |

Table 1. Comparison of SSE, accuracy and time in ACO, GC, GA and Proposed algorithm

By analyzing the values of SSE, accuracy, and time taken for computation, we can come to know that, the proposed technique is efficient than existing system and also improved in performance. After the comparison of the above process, this verifies that the proposed table shows that our proposed algorithm is better when we compared to the existing method. The accuracy of the proposed methodology is 98.78% which is greater than the previous approaches and the time and also SSE is reduced.
Figure 5. The graphical representation of SSE in ACO, GC, GA and Proposed

Figure 6. The graphical representation of time taken for compute in ACO, GC, GA and Proposed

Figure 7. The graphical representation of accuracy in ACO, GC, GA and Proposed
Figures 5, 6 and 7 shows the graphical representation of the time, accuracy and SSE of existing and our proposed method. By analyzing the values of SSE, time and accuracy we can come to know that, the proposed technique is efficient than existing system and also improved in performance. So from the result obtained from this research work we can say that the proposed method is the best method for document clustering that we have proved in performance analysis.

Before going to the part of the conclusion and their remarks of the present work, it becomes very important to study the experimentation work and analysis the results of the methods based on the user search goals. The clustering results of each methods are evaluated based on the following metrics such as Classified Average Precision (CAP), Voted AP (VAP) and average precision (AP) were used. The results of each metrics are evaluated between the clustering methods (ACO, GA, K-Mean and Proposed) with user search Goal along with semantic similarity values.

**Average Precision (AP)**

AP is determined based on the user implicit feedbacks session with user search goals, which is average precision value of documents with ranked sequence.

\[
AP = \frac{1}{N'} \sum_{r=1}^{N} rel(r) \frac{R_r}{r}
\]  

(9)

where \(N'\) is the denoted as the amount of clicked documents, \(r\) is the rank, \(N\) is the total amount of the documents returned during search process, \(rel(r)\) is relevance score value for each document, and \(R_r\) is the total amount of the documents which is most relevant to query.

**Voted Average Precision (VAP)**

VAP is measured based in on the search result for AP class is 1, it is specified as,

\[
VAP = \frac{1}{NC} \sum_{r=1}^{NC} rel(r) \frac{R_r}{r}
\]  

(10)

where \(N\) is the total amount of the documents returned during search process with class label one, \(rel(r)\) is relevance score value for each document, and \(R_r\) is the total amount of the documents which is most relevant to query of document.

**Classified Average Precision (CAP)**

Expand the procedure of VAP via adding Risk and it is specified as CAP,

\[
CAP = VAP \ast (1 - \text{risk}) \gamma
\]  

(11)

where \(\gamma\) is used as the important factor to adjust Risk value on CAP.

![Figure 8. Classified Average Precision (CAP) comparison](image)
Then illustrate the results of the clustering methods (ACO, GA, K and proposed) for CAP and is shown in Figure 8. it shows that proposed GFA achieves higher CAP results than existing methods. shows that proposed algorithm achieves higher CAP results than existing methods. The clustering results are measured via the parameters AP,VAP, CAP and risk for each user search goal inference. The result obtained from this research work we can say that the proposed method is the best method for document clustering that we have proved in performance analysis.

5. Conclusion

In this paper, the documents are clustered with the aid of adaptive pillar $k$ means algorithm along with Gaussian firefly algorithm. The adaptive pillar k-means algorithm is proposed with the intention of selecting proper centroid, because the selection of better centroid succors to attain a better clustering process. Subsequently, for the betterment of clustering process, the Gaussian firefly algorithm is utilized and then the related documents are grouped together. The experiments result on how to select the clustering by means of Adaptive Pillar K-means algorithm and G-FA were carried out and the best tradeoff between the recognition accuracy, time efficiency and speed was achieved. Finally, The method was compared with genetic algorithm, ant colony optimization algorithm and gravity clustering, and the experimental results demonstrated.

References

[1] Saravanan, D., Vengatesh JKN. (2015). Video Content Retrival Using Histogram Clustering Technique. Procedia Computer Science 50. 560-565.
[2] Jain, AK, (2010). Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8) 651-666.
[3] Muflikhah, L., Baharudin, B. (2009). Document clustering using concept space and cosine similarity measurement. In Computer Technology and Development, 2009, ICCTD’09. International Conference IEEE 1. 58-62.
[4] Raut, AB., Bamnote, GR. (2010). Web Document Clustering Using Fuzzy Equivalence Relations. Journal of Emerging Trends in Computing and Information Sciences, CIS Journal 2.
[5] Zheng, HT., Kang, BY., Kim, HG. (2009). Exploiting noun phrases and semantic relationships for text document clustering. Information Sciences, 179 (13) 2249-2262.
[6] Aliguliyev, RM, (2009). Performance evaluation of density-based clustering methods. Information Sciences 179 (20) 3583-3602.
[7] Trappey, AJC., Trappey, CV., Hsu, FC., Hsiao, DW. (2009). A fuzzy ontological knowledge document clustering methodology, Systems, Man, and Cybernetics, Part B: Cybernetics. IEEE Transactions 39 (3) 806-814.
[8] Mahdavi, M., Abolhassani, H. (2009) Harmony K-means algorithm for document clustering. Data Mining and Knowledge Discovery, 18 (3) 370-391.
[9] Luo C., Li ,Y., Chung, SM. (2009). Text document clustering based on neighbors. Data & Knowledge Engineering 68 (11) 1271-1288.
[10] Zhang, T., Tang, YY., Fang, B., Xiang, Y. (2012). Document clustering in correlation similarity measure space. Knowledge and Data Engineering. IEEE Transactions, 24 (6) 1002-1013.
[11] Dash, M., Koot, PW. (2009). Feature selection for clustering. In: Encyclopedia of database systems, 1119-1125.
[12] Brin, S., Page, L. (2012). Reprint of: The anatomy of a large-scale hypertextual web search engine. Computer networks, 56 (18) 3825-3833.
[13] Papadakos, P., Kopidaki, S., ArmenatZoglou, N., Tzitzikas, Y. (2009). Exploratory web searching with dynamic taxonomies and results clustering. In: Research and Advanced Technology for Digital Libraries 106-118.
[14] Pereira, FA., Melicio, R., Mendes, VMF., Figueiredo, J., Quadrado, JC. (2014). Fuzzy Subtractive Clustering Technique applied to Demand Response in a Smart Grid Scope. Procedia Technology 17. 478-486.
[15] Adhau, SP., Moharil, RM., Adhau, PG. (2014). K-Means clustering technique applied to availability of micro hydro power. Sustainable Energy Technologies and Assessments, 8, 191-201.
[16] Nopiah., ZM., Baharin, MN., Razali, N. (2012). An Outcome-based Approach Analysis of a Mathematical Engineering Course
using K-Means Clustering Techniques. *Procedia-Social and Behavioral Sciences* 60: 179-183.

[17] Al-Mohair HK, Saleh JM and Suandi SA (2015) Hybrid Human Skin Detection Using Neural Network and K-Means Clustering Technique. *Applied Soft Computing* 33: 337-347.

[18] He, K., Wen, F., and Sun, J. (2013). K-means hashing: An affinity-preserving quantization method for learning binary compact codes. *In Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference* 2938-2945.

[19] Dervis, K., and Bahriye, A., (2009) A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation*, 214 (1), 108-132.

[20] Krishnasamy, G., Kulkarni, AJ., Paramesran, R., (2014) A hybrid approach for data clustering based on modified cohort intelligence and K-means. *Expert Systems with Applications*, 41 (13), 6009-6016.

[21] Mohammed, AJ., Yusof, Y., Husni, H. (2015) Document Clustering Based on Firefly Algorithm. *Journal of Computer Science* 11(3), 453-465.

[22] Carullo, M., Binaghi, E., Gallo, I. (2009). An online document clustering technique for short web contents. *Pattern Recognition Letters*, 30 (10), 870-876.

[23] Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. *Information sciences*, 222, 175-184.

[24] Ramage, D., Heymann, P., Manning, CD., Garcia-Molina, H. (2009). Clustering the tagged web. *In Proceedings of the Second ACM International Conference on Web Search and Data Mining* 54-63.

[25] Gil-Garcia, R., Aurora, PP. (2010). Dynamic hierarchical algorithms for document clustering. *Pattern Recognition Letters* 31(6), 469-477.

[26] Chicco, G. (2012) Overview and performance assessment of the clustering methods for electrical load pattern grouping. *Energy*, 42 (1), 68-80.

[27] Sculley, D. (2010). Web-scale k-means clustering. *In: Proceedings of the 19th international conference on World wide web* 1177-1178.

[28] Blake, CL., Merz, CJ. (1998). University of California at Irvine Repository of Machine Learning Databases, *http://www.ics.uci.edu/mlearn/MLRepository.html*.

[29] Barakbah, AR., Kiyoki, Y. (2009). A New Approach for Image Segmentation using Pillar-Kmeans Algorithm, *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 3(11), 1897-1902.