A Comparative Study of K Means and Optimized K Means Algorithms

Aditya S. Damodaran[1], Mausumi Goswami(Bhattacharjee)[2], Bipul Syam Purkayastha[3]
[1, 2]Dept. of Computer Science and Engineering, Christ University Faculty of Engineering, Bangalore, India
[3]Professor, Computer Science Department , Assam Central University , Silchar, Assam, India

***Abstract*—Unsupervised Document clustering is a mission critical process. A slight variation in the term frequency of a document can result in an entirely new solution space. The particle swarm optimization technique can be used to iteratively refine the centroid values for each cluster and hence optimize the entire clustering process. This paper seeks to reinforce this idea by providing a comparison between the conventional k-means approach, and a particle swarm optimized approach.**

Keywords— Document Clustering, PSO, K Means, optimization

I. INTRODUCTION

The term 'document clustering' refers to a wide range of techniques that deal with the processing of groups of textual data, called documents, to identify similarities between these documents, and classify them based on this similarity. Some of the sub processes under document clustering are topic extraction, stemming, stop words removal, and filtering. However, the most important sub process of document clustering is the actual clustering process. Several algorithms have been proposed for this step, which uses a term document matrix (a matrix representing the frequencies for each keyword or 'term' in the document collection) to identify centroids, or points that have equal Euclidean distances from all documents in the collection for each cluster, with the number of required clusters being predefined. The most notable of these algorithms are the C-Means and K-Means algorithms. The K-Means algorithm has been known to be an efficient clustering algorithm, but one of the drawbacks of the algorithm is that it is less efficient in cases where the cluster sizes vary. This drawback can be done away with by augmenting and optimizing the algorithm using the particle swarm optimization technique.

II. K-MEANS CLUSTERING

The k-means clustering algorithm is essentially a method of vector quantization, with it's roots in the field of signal processing. It was first proposed by Stuart Lloyd at Bell Labs in 1957 as a pulse-code modulation technique. The algorithm, however, finds extensive use in data mining domains. K-means clustering partitions n observation sets into k clusters, in where an observation set belongs to the cluster with the nearest mean, acting as a prototype of the cluster. This partitions the data space into Voronoi cells.

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2 \quad (1)$$

The k-means problem is computationally difficult and NP-hard; but efficient heuristic algorithms are available that can be used to converge quickly to a local optimum.

III. PARTICLE SWARM OPTIMISATION

Particle swarm optimization is originally attributed to Kennedy, Eberhart and Shi, and meant to simulate the social behavior patterns of flocks of birds or a school of fish. While simulating the same, the researchers behind the algorithm realized that the algorithm was actually performing optimization. It is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. PSO uses a population of candidate solutions to the given problem, called particles. Each particle is associated with velocity and position values. The particles are guided along the search space towards the best known positions, while also being pulled towards their best known local positions. An evaluation function is used to determine the values of positions and also to determine the best positions with each iteration. As and when new best positions are found in each iteration, the particle values are updated. Thus, in a sense, the algorithm actually simulates social behavior. PSO is said to be metaheuristic as it either makes a few or no assumptions at all about the problem being optimized and can search very large candidate solution spaces. It generally finds use on optimization problems that are partially irregular, noisy, change over time, etc.

IV. PSO ALGORITHM

The basic steps followed in a generic PSO algorithm are as follows:

1. For each particle i=1 to S,
   i) Initialize it's position with a uniformly distributed random vector: $x_i \sim U(b_{lo}, b_{up})$, where b_{lo} and b_{up} are the lower and upper boundaries of the search-space.
   ii) Initialize the particle's best known position to its initial position: $p_i \leftarrow x_i$
   iii) If $(f(p_i) < f(g))$ update the swarm's best known position: $g \leftarrow p_i$
   iv) Initialize the particle's velocity: $vi \sim U(-|b_{up}-b_{lo}|, |b_{up}-b_{lo}|)$

2. Repeat for each iteration:
   i) For each particle i=1 to S
      a. Pick random numbers: $r_p, r_g \sim U(0,1)$
      b. For each dimension d = 1, ..., n;
         Update the particle's velocity:
         $v_{i,d} \leftarrow \omega\, v_{i,d} + \varphi_p\, r_p\, (p_{i,d}-x_{i,d}) + \varphi_g\, r_g\, (g_d-x_{i,d})$
      c. Update the particle's position: $x_i \leftarrow x_i + v_i$
      d. If $(f(x_i) < f(p_i))$ do:
         Update the particle's best known position: $p_i \leftarrow x_i$
         If $(f(p_i) < f(g))$ update the swarm's best known position: $g \leftarrow p_i$
3. Now g holds the best found solution.

V. PROPOSED SYSTEM

To compare the efficiencies of both approaches, we must use them both to perform clustering on the same data sets, and then proceed to validate the results. The validation of the resulting clustered output will help us determine whether the PSO approach is economical or not

The implementation of both algorithms and the validation phase are done with the help of MATLAB.

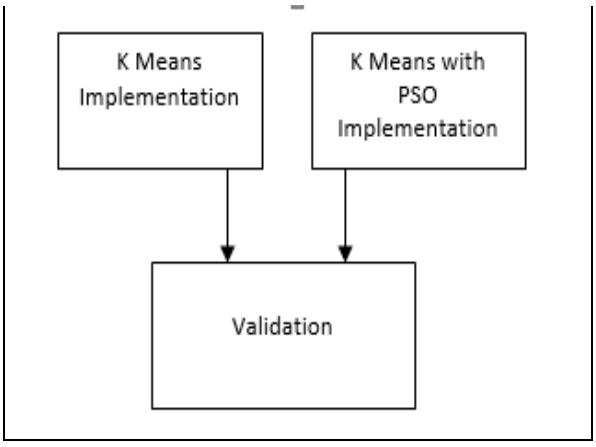

Fig. 1. The Proposed System

In the first implementation, the K Means algorithm is performed on a text document matrix, resulting from preprocessing activities conducted on the provided data set. Centroids are determined, and the documents are classified in the appropriate clusters.

In the second implementation, Particle Swarm Optimization is used to identify the most promising centroid positions for the clustering problem, using the same text document matrix. These centroid positions are fed to the k means algorithm again, along with the initial text document matrix. The k means algorithm directly uses these centroids for computation, rather than trying to locate them on it's own as seen in the k means only approach. It then clusters the given documents according to the centroid positions obtained via the PSO algorithm.

VI. VALIDATION

Cluster validation can be performed by several different methods, like the silhouette value method, calculating entropy and purity values, and the f measure method, to name a few. In this paper, the silhouette plot method is used.

The technique was first described by Peter J. Rousseeuw in 1986, and it provides a succinct graphical representation of how well each object lies within its cluster.

Let us assume that a given data set has been clustered by a clustering algorithm. For each singular unit of data i, let a(i) represent a measure of the average dissimilarity of unit i, with all other data units within it's cluster. b(i) be the lowest average dissimilarity of i to any other cluster to which i does not belong. Now,

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (2)$$

WHICH CAN BE WRITTEN AS:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

Thus, s(i) lies between –1 and 1. Values of s(i) which tend to be closer to 1 imply appropriate clustering, while values that are closer to -1 imply inefficient clustering. A value of 0 indicates that the data unit lies on the border between two clusters.

VII. RESULTS

Both algorithms were executed on a sample data set consisting of 15 documents. The results are shown below.

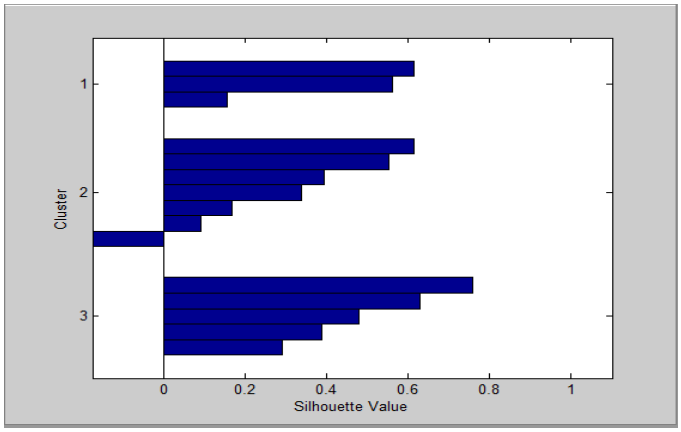

Fig. 2. Silhouette Plot for normal K means

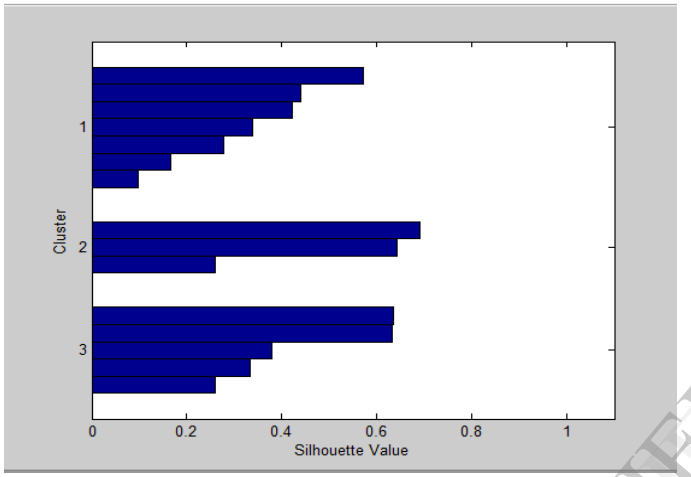

Fig. 3. Silhouette plot for PSO K means

As seen from the silhouette plots shown above, we can observe that the silhouette values for each document are comparatively higher in the particle swarm optimized approach than those of the normal k means clustering approach. This observation gives us a basis to say that the PSO algorithm can indeed increase the efficiency of the k means algorithm.

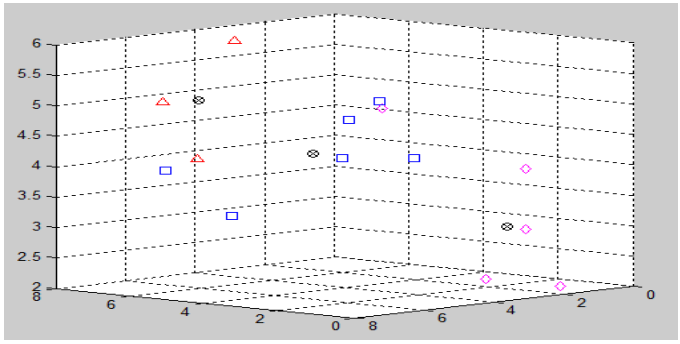

Fig. 4. K Means Clustering

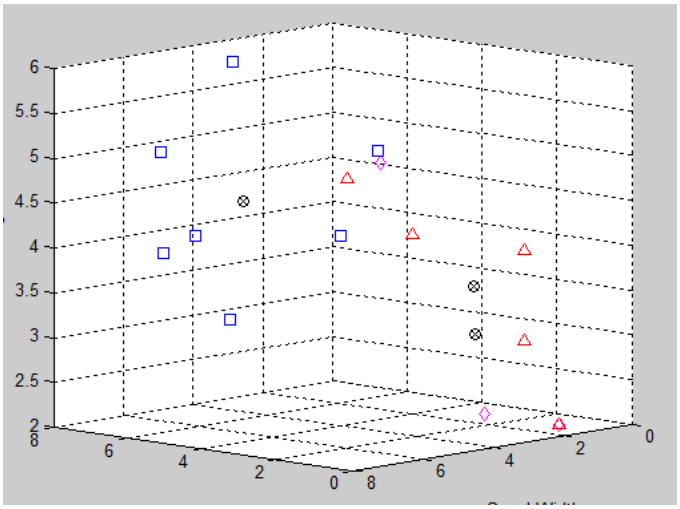

Fig. 5. PSO K Means Clustering

VIII. CONCLUSION

Thus, from the observations of this paper, we can conclude that Particle Swarm Optimization augments the performance of K Means document clustering. The silhouette values for documents that have been clustered using the traditional k means algorithm are less than those of the PSO augmented algorithm.

REFERENCES

[1] Kennedy J., Eberhart R., "Particle Swarm Optimization" Proceedings of IEEE International Conference on Neural Networks IV,1995, pp. 1942–1948

[2] Stuti Karol, Veenu Mangat, Evaluation of text document clustering approach based on particle swarm optimization Springer ,September,2012

[3] Abraham A., Das S., Roy, Swarm Intelligence Algorithms for Data Clustering

[4] Abraham A., Das S., Konar A., Document Clustering using Differential Evolution, IEEE, 2006

[5] Abraham A., Grosan C., Ramos V., Swarm Intelligence in Data Mining, Stud. Comput. Intell., 34, 2006

[6] Shi Y., Eberhart, R.C., "A modified particle swarm optimizer" Proceedings of IEEE International Conference on Evolutionary Computation,1998, pp. 69–73

[7] P. J. Rousseeuw, "Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis," Computational and Applied Mathematics 20,1987, pp. 53–65.