High Dimensional Data Clustering Using Cuckoo Search Optimization Algorithm

[1]Priya Vaijayanthi, [2]Xin-She Yang, [3]Natarajan A M and [4]Raja Murugadoss

[1,3]Department of CSE, Bannari Amman Institute of Technology, Sathyamangalam
[4]Department of Civil Engineering, Bannari Amman Institute of Technology, Sathyamangalam
[2]Department of Mathematics, Cambridge University, UK
Email: [1]priyardoss@gmail.com

Abstract— The amount of data available over Internet and World Wide Web is increasing exponentially. Retrieving data that is more close to user's query effectively and efficiently is a challenging task in Information Retrieval (IR) system. Clustering of Documents is one of the solutions to this. Clustering is the process of partitioning a set of objects in such a way that the objects in same cluster are more similar. The number of possible ways in which the documents can be clustered is enormous and this makes the problem to be a combinatorial optimization problem. Nature inspired algorithms are commanding tools to attack this type of problem. In this paper, an attempt has been made to use Cuckoo Search Optimization (CSO) algorithm to solve the problem of document clustering. The CSO algorithm is experimented with standard benchmark dataset, Classic4 dataset. The quality of solutions generated by CSO algorithm in terms of DB Index was compared with K-means algorithm and Ant Colony Optimization (ACO) algorithm. The results reveal that CSO algorithm is a viable to achieve world class solutions to high dimensional data clustering.

Keywords—document clustering, optimization, cuckoo search, ant colony, meta heuristic

I. INTRODUCTION

Usage of Internet has become a usual activity in business and other walks of life. This is mainly due to the increase in the availability of data in electronic form. With the advent of Internet, World Wide Web (WWW) and decline in the cost of storage devices, electronic storage has become popular. The amount of information available over Internet is growing exponentially. In addition, the amount of documents maintained represents an accumulated knowledge. Information Retrieval (IR) system has become a powerful sub field of information science that concerns storage, access and retrieval of information. Effective and efficient retrieval of documents has become vital. Search engines are common gateways for effective retrieval. They are continuously optimized and improved to serve the users in a better way. Even with the usage of sophisticated ranking algorithms, Search Engines fail to retrieve the documents that closely match with user's query. One way to address this problem is to cluster or group the documents in an order so that navigation becomes easy and efficient in IR systems. Clustering is an unsupervised learning that partitions a set of objects in such a way that the similarity between the objects in same group is more and less between the objects in other groups. If the amount of data that is to be clustered is more, then the number of possible ways in which partition can be done is also enormous. This makes the problem as a combinatorial optimization problem. K-means algorithm is the traditional clustering algorithm. This works good when the size of the data set is small. But it fails to scale with very huge data sets. Nature inspired algorithms (Ant Colony Optimization Algorithm, Artificial Bee Colony Algorithm, Particle Swarm Optimization, Bird Flocking Algorithm, Frog Leaping Algorithm, Genetic Algorithm) are viable tools to solve complex optimization problems.

II. RELATED WORKS

Several researchers tried and proposed many meta heuristic approaches and nature inspired algorithms to solve the problem of document clustering. A "Novel Ant Colony Optimization algorithm for Clustering" was proposed by [He et al. (2006)]. This algorithm uses the basic ACO and constructs a connected graph, where the documents are the vertices and are connected through edges. Later the graph is disconnected which results in clusters. A Document clustering method based on Ant algorithm that was tested over text clustering was proposed by [Lukasz Machnik. (2005)]. The author also suggested that parallelization of the algorithm would bring better results. The phenomena of corpse clustering and larval sorting in ants were proposed in [17]. A modified model proposed in [16] uses a dissimilarity based evaluation of the local density in order to make it suitable for data clustering. They had also introduced the idea of short term memory within each artificial agent. [18] - [19] combined the stochastic principles of clustering by ants with popular K-means algorithm in

order to improve the convergence of ant based algorithms. The proposed algorithm was called as AntClass. [20] - [21] proposed Ant System and ACO which is a meta-heuristic approach based on foraging behavior (a positive feedback) of real world ant species. It is based on pheromone model. [22] developed a new algorithm called "a cluster" to solve unsupervised clustering and the data retrieval problem. The algorithm was tested with text document clustering, [23] presented hybridization of ant system with Fuzzy C-means algorithm (FCM) to determine the number of clusters automatically. In this, ant based algorithm is refined using FCM algorithm. [24] presented a novel clustering algorithm called AntTree for unsupervised learning. [25] proposed an ant based clustering algorithm that was proved to be better than traditional partitioning algorithm when tested over real datasets. [26] proposed a hybrid algorithm that combines ant system with SOM and K-means for cluster analysis. This improves the robustness of traditional algorithm. [27] developed multiple ant colonies approach for data clustering. It involves parallel engagement of several individual ant colonies.

III. PROBLEM FORMULATION

Clustering is a typical unsupervised learning technique for grouping similar data points as in Fig. 1. A clustering algorithm assigns a large number of data points to a smaller number of groups such that data points in the same group share the same properties while, in different groups, they are dissimilar.

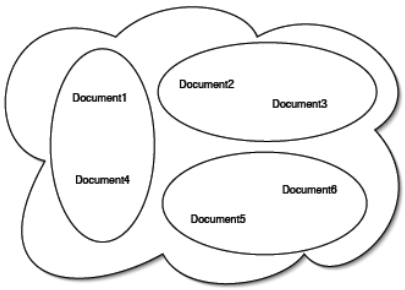

Fig. 1 Conceptual Model of Document Clustering

There are several methods in which a document can be represented. Among them, vector space model is the widely used representation. The documents are pre-processed before their representation. Pre-processing includes stop word removal, stemming and unique word identification. A unique set of words from all the documents of the document corpus 'D' or data set is obtained and this is used to represent each document in the dataset. Thus, a document d_i is represented as a vector as per Eq. (1) where each element in the vector is the weight of the terms in the corpus.

$$d_i = \{w_{i1}, w_{i2} \ldots w_{im}\} \qquad (1)$$

where, w_{ij} is the weight of term 'j' in 'd_i' and 'm' represent the total number of terms in the corpus. The weight of each word or term represents the importance of the word in the document. It is calculated using Eq.(2).

$$w_{ij} = tf_j * idf_j, \text{where} \qquad (2)$$

w_{ij} is the weight of $term_j$ in d_i, tf_j is the term frequency in d_i. idf_j is the inverse document frequency (importance of $term_j$ in other documents in the document corpus). The problem of clustering documents in a document corpus is formulated as optimization problem. The solution is a vector where each component corresponds to the centre of a cluster and is calculated using Eq. (3).

$$C = (c_1, c_2, c_{3 \ldots} c_k) \qquad (3)$$

where

'C' is the solution vector and c_i is the centroid of each cluster. The quality of the solution vector is measured by Davis-Bouldin index (DB index) as in Eq. (4) which is based on internal criterion (i.e. inter- and intra- cluster similarity).

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max\left(\frac{\sigma_i + \sigma_j}{\text{dist}(c_i, c_j)}\right)_{i \neq j} \qquad (4)$$

where, 'k' is the number of clusters, σ_I is the average distance of all the documents in cluster 'i' to its centre 'c_i' and dist (c_i, c_j) is the similarity between c_i and c_j.

The problem of clustering 'n' documents into 'k' clusters is formulated as an optimization problem. This is because of the exponential growth in the solution space with increase in the number of documents. The total number of probable solutions will follow Stirling number as in Eq. (5).

$$= \frac{1}{k!}\sum_{i=1}^{k} (-1)^{k-i} \left(\frac{k}{i}\right) i^n \qquad (5)$$

It is clear that with smaller values for 'n' and 'k' as 10 and 3 respectively, the number of possible partitions comes around 9330. Examining all possible partitions and to identify the global optimum partition is not computationally feasible.

IV. CSO ALGORITHM

Cuckoo search optimization algorithm is a new metaheuristic algorithm, inspired by nature, developed by Yang and Deb in 2009. The algorithm is based on the breeding behavior such as brood parasitism of some species of cuckoos. Cuckoos lay their eggs in communal nests by removing the eggs of host birds thereby increasing the hatching probability of their own eggs. In the basic version of cuckoo search algorithm, there is no possibility of remembering the previously visited solutions. The algorithm begins with a group of randomly generated solutions that are populated in a set of nests. The number of eggs in each nest is fixed and

equal. The number of cuckoos is taken to be equal to the number of nests. Here, egg represents a potential solution. The assumptions of basic Cuckoo search algorithm are adapted in the proposed algorithm also. The assumptions are 1) Each Cuckoo lays one egg at a time and dumps it in a randomly chosen nest. 2) The best nest with high quality eggs are carried in next generation. 3) The number of available host nests is fixed. The quality of eggs is proportionate to the fitness value of the objective function. The best egg in each nest and thus the current best solutions are found accordingly and registered. Each cuckoo egg represents a new solution. This new egg replaces the worst egg in the chosen nest. To generate a new solution, a Levy flight is performed on the current best solution in the nest. Levy flight is a random walk with step lengths taken from Levy distribution. The random walk is a Markov chain where next solution depends on the current solution.

Fig.2 CSO algorithm

V. EXPERIMENTAL RESULTS AND DISCUSSION

As document clustering is high dimensional data, a standard benchmark dataset Classic3 dataset is taken. This dataset has collection of 4 different corpuses, CRAN, CISI, CACM and MED. The details of the dataset are given in Table I. Pre-processing steps like stop word removal and stemming were done. This also helps to reduce the dimension of the solution space. A unique set of terms in the entire set is identified and is used to represent all the documents in the dataset. Totally three experiments were conducted and the details of the experiments are given in Table II. There are more than 5000 terms in the entire data set. So the dimension of the search space is enormous. As the data set taken has four documents on four categories, the number of clusters is taken to be 4. Several trails were conducted to find best suitable values for the parameters of both ACO and CSO algorithm.

Table I. Dataset Details

Dataset	#no of documents
CACM	3204
CISI	1460
CRAN	1400
MED	1033

Table II. Experiment Details

Expt. No	# no of documents	v_a	'N'	'ρ_a'
Expt 1	600	6	10	0.15
Expt 2	800	8	15	0.15
Expt 3	1200	12	15	0.18
Expt 4	2000	20	15	0.20

In Table II, v_a represents the number of abstract ants engaged in ACO algorithm. Other parameters are taken as given in [28]. 'N' represents the number of abstract cuckoos engaged in CSO algorithm. The quality of clusters generated by K-means, ACO and CSO algorithms were evaluated using Equation (4). The results are summarized in Tables III and IV.

Table III. K-means Vs ACO

#no. of documents	DB Index		
	v_a	K-means	ACO
600	6	0.5241	0.5191
800	8	0.7812	0.6299
1200	12	0.8709	0.6105
2000	20	0.8215	0.5923

It is clear from Table III that the quality of clusters generated by ACO algorithm is better than that of K-means algorithm. This is mainly because; the quality of results of K-means algorithm depends on the initially selected cluster centres. Also, K-means algorithm fails to scale with large data sets. At the same time, ACO algorithm gives comparably better results for the same data set. Experiments were conducted by varying the number of ants engaged. From that, we observed that the optimal number of ants is in the range of 10 to 15. The values of the parameters 'g' and 'h' were taken to be 0.5 and 1 respectively. Accordingly the graph is plotted and given in Fig. 3.

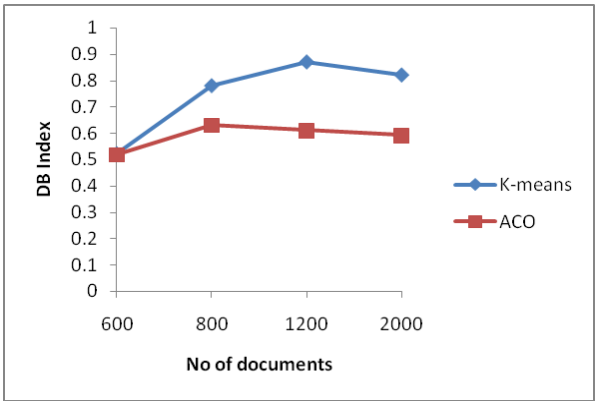

Fig. 3 Cluster Quality: K-means Vs ACO

The same set of experiments was conducted by applying CSO algorithm. The number of abstract cuckoos engaged is varied. It is observed that the optimal number of cuckoos may be between 6 and 15. It is also found that by engaging more number of cuckoos for high dimensions does not have notable impact on the quality of clusters generated. The results of experiments were given in Table IV and graph is plotted accordingly and given in Fig. 4.

Table IV. K-means Vs CSO

#no. of documents	DB Index		
	v_a	K-means	CSO
600	6	0.5241	0.5234
800	8	0.7812	0.5908
1200	12	0.8709	0.5832
2000	20	0.8215	0.5715

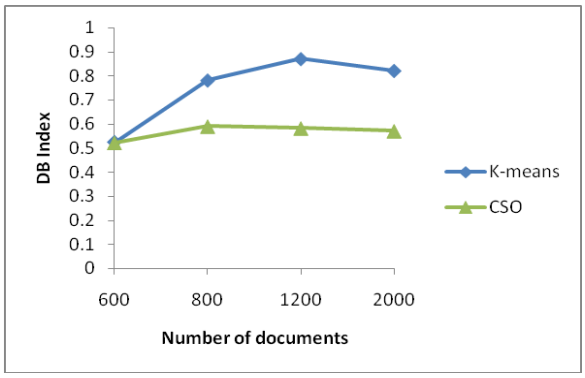

Fig. 4 Cluster Quality: K-means Vs CSO

Based on the experimental results it has been observed that K-means algorithm is suitable for smaller data sets. The quality of clusters generated is closer to the ones generated by ACO and CSO algorithms. But, when compared to these two algorithms, the performance of K-means degrades with larger data sets. This clearly shows that K-means algorithm is not suitable for large and very large data sets. The quality of solutions generated by ACO algorithm is better than K-means algorithm and as good as that of CSO algorithm both for small data set and large data sets. Results generated by ACO and CSO algorithms are given in Table V and accordingly graph was plotted and given in Fig. 5.

Table V. ACO Vs CSO

#no of documents	DB Index	
	ACO	CSO
600	0.5191	0.5234
800	0.6299	0.5908
1200	0.6105	0.5832
2000	0.5923	0.5715

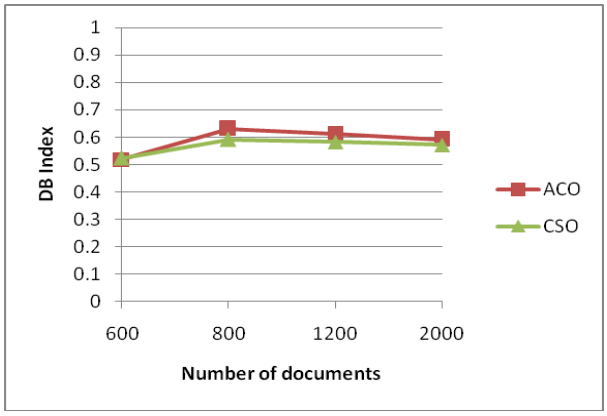

Fig. 5. Cluster Quality: ACO Vs CSO

VI. CONCLUSION

Nature inspired meta heuristic approaches are proved to be dominant techniques to attack combinatorial optimization problems in generating near optimal solutions. Cuckoo Search Optimization algorithm is very new in this list and has been proved to be effective optimization algorithm. In this paper, CSO algorithm has been used to find near good quality solutions to document clustering problem. The results were compared with that generated by Ant colony Optimization algorithm and K-means algorithm. It is observed that the performance of CSO algorithm is as good as ACO algorithm in some cases and marginally better in most of the situations. solutions generated by ACO and CSO algorithms were studied.

REFERENCES

[1] J. A. Bland, "Optimal structural design by ant colony optimization," Engineering Optimization, vol. 33, pp. 425 – 443, 2001.

[2] E. Bonabeau, M. Dorigo and G. Theraulaz, "Swarm intelligence: From Nature to Artificial Systems," New York: Oxford University Press, 1999

[3] M. H. Botee and E. Bonabeau, "Evolving ant colony optimization," Adv. Complex Systems, vol. 1, pp. 149-159, 1998

[4] A. Colorni, M. Dorigo and V. Maniezzo, "An investigation of some properties of an ant algorithm," in 1992 Proc. Parallel Problem Solving from Nature, Amsterdam, Elsevier, pp. 509-520

[5] D. Costa and A. Hertz, "Ants can colour graphs," Journal of Operational Research Society, vol. 48, pp. 295-305, 1997

[6] M. Dorigo, "Ant algorithms solve difficult optimization problems," in 2001 Proc. Advances in Artificial Life: Artificial Life Conf., Springer Verlag, pp. 11-22

[7] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," IEEE Trans. Evol. Comp., vol. 1, pp. 53-66, 1997

[8] M. Dorigo and T. Stuzzle, "An experimental study of the simple ACL algorithm," in 2001 Proc. WSES Evolutionary Computation Conf., WSES-Press International, pp. 253-258

[9] M. Dorigo and T. Stuzzle, "Ant colony optimization", England: MIT Press, 2004

[10] M. Dorigo, G. Di Caro and L. M. Gambardella, "Ant algorithms for discrete optimization," Artificial Life, vol. 5, pp. 137 – 172, 1999

[11] M. Dorigo, V. Maniezzo and A. Colorni, "The ant system: optimization by a colony of cooperating agents," IEEE Transactions on Systems, Man, and Cybernetics-Part-B, vol. 26, no.1, pp. 1-13, 1996

[12] J. Holland, "Concerning efficient adaptive systems," Self – organizing systems, Washington, D. C.: Spartan Books, pp. 215-230, 1962

[13] J. Holland, "Adaptation in natural and artificial systems", Ann Arbor: University of Michigan Press, 1975

[14] Yulan He, Siu Cheung Hui, and Yongxiang Sim, "A novel ant based clustering algorithm for document clustering," Asia Information Retrieval Symposium, pp. 537 – 544, 2006

[15] Lukasz Machnik, "ACO based document clustering method," Technical report , Annales UMCS Informatica AI 3, pp 315-323, 2005

[16] J. L. Deneubourg, S. Gross, N. Franks, A. Sendova, C. Detrain and L. Chretien, "The dynamics of collective sorting: robot like ants and ant like robots," in 1991 Proc. First International Conference on Simulation of Adaptive Behavior: From Animals to Animats, MIT Press: Cambridge, MA, pp. 356-363

[17] E. D. Lumer and B. Faieta, "Diversity and adaptation of populations of clustering ants," in 1994 Proc. Simulation of Adaptive Behavior Conf., pp. 501-508

[18] N. Monmarche, M. Silmane and G. Venturini, "On improving clustering in numerical databases with artificial ants," Advances in Artificial Life, pp. 626-635, 1999

[19] N. Monmarche, "On data clustering with artificial ants," "Data mining with evolutionary algorithms: research directions", AAAI Workshop, AAAI Press, pp. 23-26, 2005

[20] M. Dorigo, E. Bonabeau and G. Theraulaz, "Ant algorithms and stigmergy," Future Generation Computer Systems, vol. 16, no. 8, pp. 851-871, 2000

[21] M. Dorigo, G. Di Caro and L. M. Gambarella, "Ant algorithms for discrete optimization," Artificial Life, vol. 5, no. 3, 137-172, 1999

[22] V. Ramos and J. J. Merelo, "Self organized stigmergic document maps: environment as mechanism for context learning," in 2002 Proc. Evolutionary and Bio-inspired Algorithms Conf., pp. 284-293

[23] P. Kanade and L. O. Hall, "Fuzzy ants as a clustering concepts", in 2003 Proc. North American Fuzzy Information Processing Society Conf., pp. 227-232.

[24] H. Azzag, N. Monmarche, M. Slimane and G. Venturini, "Ant Tree: a new model for clustering with artificial ants," Evolutionary Computation, vol. 4, pp. 2642-2647, 2003

[25] P. S. Shelokar, V. K. Jayaraman and B. D. Kulkarni, "An ant colony algorithm for clustering,"Analytica Chemica Acta, vol.509, no. 2, pp. 187-195, 2004.

[26] S. Chi and C. C. Yang, "Integration of ant colony SOM and K-means for clustering analysis," Knowledge based Intelligent Information and Engineering Systems, LNCS, Springer, vol. 4251, pp. 1-8, 2006

[27] Yan Yang and Mohamed S. Kamel, "An aggregated clustering approach using multi-ant colonies algorithms," Pattern Recognition, vol. 39, no. 7, pp. 665-671, 2006

◈◈◈