Improved Cuckoo Search Algorithm for Document Clustering

Saida Ishak Boushaki, Nadjet Kamel, Omar Bendjeghaba

Improved Cuckoo Search Algorithm for Document Clustering

Saida Ishak Boushaki [1], Nadjet Kamel [2], and Omar Bendjeghaba [3]

[1] LRIA (USTHB) and University of Boumerdes, Algeria
saida_2005_compte@yahoo.fr
[2] LRIA (USTHB) and University of Ferhat Abas Setif, Algeria
nkamel@usthb.dz
[3] LREEI (UMBB) and University of Boumerdes, Algeria
benomar75@yahoo.fr

Abstract. Efficient document clustering plays an important role in organizing and browsing the information in the World Wide Web. K-means is the most popular clustering algorithms, due to its simplicity and efficiency. However, it may be trapped in local minimum which leads to poor results. Recently, cuckoo search based clustering has proved to reach interesting results. By against, the number of iterations can increase dramatically due to its slowness convergence. In this paper, we propose an improved cuckoo search clustering algorithm in order to overcome the weakness of the conventional cuckoo search clustering. In this algorithm, the global search procedure is enhanced by a local search method. The experiments tests on four text document datasets and one standard dataset extracted from well known collections show the effectiveness and the robustness of the proposed algorithm to improve significantly the clustering quality in term of fitness function, f-measure and purity.

Keywords. Document Clustering, Vector Space Model, Cuckoo Search, Cosine Similarity, F-measure, Purity, Metaheuristic, Optimization.

1 Introduction

The high advance of the internet has led to exponential growth of the amount of information available in the World Wide Web (WWW). Consequently, exploring the data and finding the relevant information on the web became hard tasks. Over the past decades, many approaches have been developed in order to manage and organize efficiently this large set of documents. For this purpose, clustering is the well known method used by the scientific community dealing by the datamining. It is unsupervised technique [1] [2], that extract hidden structural characteristics in the data and gathering the highly similar objects in the same group, whereas segregates dissimilar objects in different ones. Due to the importance task of the clustering technique, it has been applied in variety engineering and field like image segmentation, pattern recognition and gene-expression. The algorithms of clustering are divided into different categories: hierarchical clustering algorithms, nominal data clustering, density based clustering, cohonen networks and partitioning relocation clustering. The last category of clustering contains algorithms with linear time complexity. This makes the partitional algorithms more suitable for web clustering. One of the most famous partitional algorithms is the K-means [3] due to its simplicity and efficiency. However, this algorithm may give a poor results this is due to its random initialization and local exploration, which leads to a local minimum. Actually, nature inspired algorithms cope the shortcoming of a local solution by a global one [4]. One of the most recent metaheuristic algorithms is cuckoo search (CS) optimization [5] [6]. It is based on the interesting breeding behaviour such as brood parasitism of certain species of cuckoos and typical characteristics of Lévy flights. The results of experiment comparison show that the cuckoo search algorithm outperform the most famous metaheuristics [7] [8] [9].

In order to improve the clustering result, and inspired from the hybrid algorithm proposed in [10], we propose in this paper a new algorithm for document clustering, based on CS. In this algorithm, CS is enhanced by additional functions. Which make it superior to conventional CS in term of fitness, convergence speed and external quality.

The remaining of this paper is organized as follows: in section 2, we present most recent metaheuristics algorithms proposed for web document clustering. In section 3, the formal definitions of document clustering are presented. In section 4, we present the fundamental steps of a cuckoo search algorithm for the clustering problem. The improved cuckoo search adapted for document clustering is presented in section 5. Numerical experimentation and results are provided in Section 6. Finally, the conclusion and future work are drawn in Section 7.

2 Related Works

Document clustering based on nature inspired algorithms is an active research field. In 2013, Kamel et al. [11] overcome the weakness of K-means in the initial seed by a hybrid algorithm based on K-means, PSO and Sampling algorithms for document clustering. Leticia Cagnina et al. [12] have presented an improved version of the discrete particle swarm optimization (PSO) algorithm. This version includes a different representation of particles, a more efficient evaluation of the function to be optimized and some modifications in the mutation operator. In 2014, Wei Song et al. proposed a fuzzy control genetic algorithm (GA) in conjunction with a novel hybrid semantic similarity measure for document clustering. It outperforms the conventional GA [13]. In 2013, A novel document clustering algorithm based on ant colony optimization algorithm was proposed by Kayvan Azaryuon and Babak Fakhar. It improves the standard ants clustering algorithm efficiency by making ant movements purposeful, and on the other hand, by changing the rules of ant movement [14]. S. Siamala Devi et al. have used the hybrid K-means with harmony search (HS) to do the comparison between the concept called coverage factor and the concept factorization method for document clustering problem [15]. The experimental results show that factorization produces better results. Recently, the experimental results of [7] shown that the cuckoo search (CS) clustering achieves best results compared to the well known and recent algorithms: K-means, particle swarm optimization, gravitational search algorithm, the big bang–big crunch algorithm and the black hole algorithm. More recently, in our previous work, we have proposed a new hybrid algorithm for document clustering based on CS and K-means [10]. This new hybrid algorithm outperforms the CS and K-means in term of fitness and external quality.

3 Formal Definitions

Let S be a set of n objects $O_1, O_2,..,O_n$, each object is defined in multi dimensional space. Clustering S into k clusters means dividing it into k groups or clusters $C_1, C_2,..,C_k$, such that:

$$\begin{cases} C_i \neq \{\ \} & \text{for i} = 1,..,\text{k} \\ C_i \cap C_j = \{\ \} & \text{for i} = 1,..,\text{k, j} = 1,..,\text{k and i} \neq \text{j} \\ C_1 \cup C_2 \cup .. \cup C_K = S & \end{cases} \quad (1)$$

In addition, the objects in the same cluster are similar and the objects in different clusters are dissimilar. This property is proportional to the quality of the clustering.

In our case, data are documents. They are represented by using the vector space model (VSM) [10] [16].

The cosine distance is the most used and the best one for document clustering [17]. Given two documents d_i and d_j represented by two vectors v_i and v_j, respectively, the cosine distance is given by the following formula:

$$\cos(d_i, d_j) = \frac{v_i^t v_j}{|v_i||v_j|} \quad (2)$$

Where $|v_i|$ is the norm of the vector v_i

To evaluate the quality of clustering results, we have used two external quality indexes: the famous F-measure and Purity [2] [10].

4 Cuckoo Search Clustering Algorithm

For solving the clustering problem, the standard cuckoo search algorithm is adapted to reach the centroids of the clusters that optimize predefined fitness function. We have used the fitness function presented in [18]. The goal of this function is to find the solution that maximizes the similarity between each document and the centroid of the cluster that is assigned to. This objective function is given by the following formula:

$$\text{Fitness}\quad \text{Maximize}\ \sum_{i=1}^{k}\sum\nolimits_{d_l\in C_i}\cos\left(d_l, c_i\right) \tag{3}$$

Where: k is the number of clusters and $\cos\left(d_l, c_i\right)$ is the cosine distance between the document d_l and the nearest centroid c_i of the cluster C_i.

The cuckoo search clustering algorithm (CSCA) is given by the following steps [7] [10]:

```
1. Generate randomly the initial population of nb_nest host nests;
2. Calculate the fitness of these solutions and find the best solution;
3. While (t < Max_Iter) or (stop criterion);
   (a) Generate nb_nest new solutions with the cuckoo search;
   (b) Calculate the fitness of the new solutions;
   (c) Compare the new solutions with the old solutions, if the new solution is better
       than the old one, replace the old solution by the new one ;
   (d) Generate a fraction ($p_a$) of new solutions to replace the worse nests;
   (e) Compare these solutions with the old solutions. If the new solution is better
       than the old solution, replace the old solution by the new one;
   (f) Find the best solution;
4. End while;
5. Print the best nest and fitness;
```

5 Improved Cuckoo Search Clustering Algorithm

Cuckoo search clustering algorithm can achieve the best global solution compared to most other metaheuristics. Usually, this global solution is obtained after huge number of iterations due to the slow convergence of the algorithm. It is obvious that in CSCA, the research area is explored using the standard cuckoo function [19]. In the present work, we propose to perform after each new solution generated by the standard cuckoo function an auxiliary local research in the research area in order to improve the solution. If this local search finds a solution that is better than the existing one, then it will be replaced by the new reached one.

For each current solution (host nest), the local search procedure exploits this one by calculating the gravity center of each cluster using the equation (4). Thus, the solutions are replaced only if their new fitness is better. The pseudo code of the new improved cuckoo search clustering algorithm (ICSCA) is presented in Fig. 1.

$$gc_i = \frac{1}{n_i}\sum\nolimits_{\forall d_l\in C_i} d_l \tag{4}$$

Where gc_i is the gravity center of the cluster C_i, d_l denotes the document that belong to the cluster C_i and n_i is the number of documents in cluster C_i.

```
Begin
1. Set the initial parameters:
 - $p_a$ (the probability of worse nests)
 - nb_nest (the number of host nest is the population size)
 - k (number of clusters)
 - Max_Iter (the maximum number of iterations)
2. Generate randomly the initial population of nb_nest host
   nests;
3. For each solution change the empty clusters
4. Calculate the fitness of each solution using the
   equation(3)and find the best nest;
5. While (t < Max_Iter) or (stop criterion)
 5.1 Generate nb_nest new solutions using the standard
     cuckoo search function;
 5.2 For each new solution change the empty clusters;
 5.3 Calculate the fitness of each new solution using the
     equation (3);
 5.4 For each solution compare the new solutions with the
     old solutions, if the new solution is better than the
     old one, replace the old solution by the new one ;
 5.5 Generate nb_nest new solutions by calculating the
     gravity center of each cluster using equation (4)
 5.6 For each new solution change the empty clusters;
 5.7 Calculate the fitness of each new solution using the
     equation (3);
 5.8 For each solution compare the new solutions with the
     old solutions, if the new solution is better than the
     old one, replace the old solution by the new one;
 5.9 Generate a fraction ($p_a$) of new solutions to replace
     the worse nests;
 5.10 For each new solution change the empty clusters;
 5.11 Calculate the fitness of each new generated solution
      using the equation (3);
 5.12 Compare the new solutions with the old solutions, if
      the new solution is better than the old one, replace
      the old solution by the new one ;
 5.13 Find the best solution;
End while;
6. Print the best nest and fitness;
End
```

Fig. 1. ICSCA procedure

To illustrate this idea, we give an example. In Fig. 2, we have three clusters and we can see that the objects are more similar to their gravity center than to the centroid generated by the standard cuckoo function.

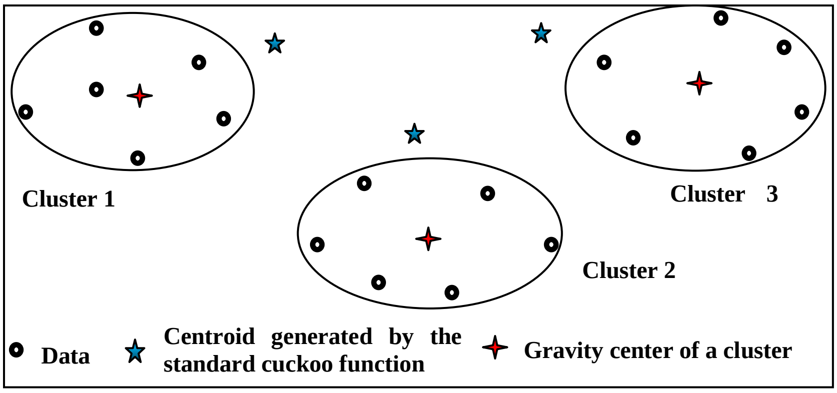
Cluster 1
Cluster 3
Cluster 2
Data
Centroid generated by the standard cuckoo function
Gravity center of a cluster

Fig. 2. Example of local search

To illustrate this idea, we give an example. In Fig. 2, we have three clusters and we can see that the objects are more similar to their gravity center than to the centroid generated by the standard cuckoo function.

We should notice that another primary function must be performed after each new generated solution. The main goal of this function is to ensure that there is no empty cluster. The simple way for doing this is to replace the empty cluster by a random one.

6 Experiments and Results

In order to test the efficiency of each auxiliary function added to the standard cuckoo search clustering algorithm, we compare between three algorithms: standard cuckoo search clustering algorithm (CSCA), standard cuckoo search algorithm augmented by the change empty cluster function (CSDC+CEC) and the improved cuckoo search document clustering (ICSCA) enhanced by the local search procedure and the change empty cluster function.

6.1 Datasets

Two kinds of datasets are used in the whole of experiments: four text document datasets and one standard dataset. The text document datasets are extracted from two well known collections: Classic3 [20] and Text REtrieval Conference (TREC) collections [21]. The description detail of text document datasets is given in Table 1, where the standard dataset is obtained from the famous UCI Machine Learning Repository. The description detail of standard dataset is given in Table 2.

Table 1. Summary of text document datasets

datasets	Number of documents	Number of terms	Classes description	Number of groups
Classic300	300	5471	100, 100, 100	3
Classic400	400	6205	100, 100, 200	3
Tr23	204	5833	6, 11, 15, 36, 45, 91	6
Tr12	313	5805	9, 29, 29, 30, 34, 35, 54, 93	8

Table 2. Description of standard dataset

datasets	Number of instances	Number of attributes	Classes description	Number of groups
Iris	150	4	50, 50, 50	3

6.2 Related Parameters

For the purpose of comparison, the number of iterations is fixed to 100 iterations for the text datasets and only 20 iterations for the Iris standard dataset. We note that for all runs, the probability of worse nests was set to 0.25, while the population size was set to 10. The cosine distance is used as similarity measure for all experiment tests.

6.3 Results and Comparisons

The three algorithms: (CSCA), (CSDC+CEC) and (ICSCA) are compared for the different datasets in term of best fitness value and two external validity indexes (F-measure and purity). In Table 3 we present the best fitness value of the three algorithms for each datasets.

As we can see from this table, the ICSCA can reach the best results in comparison with the CSCA and CSCA+CEC. In addition, the CSCA+CEC is better than CSCA and the gap between them is proportional to the number of clusters. As the number of cluster increases, more than the gap increases.

Table 3. Best fitness value

Datasets	CSCA	CSCA+CEC	ICSCA
Classic300	28.0540	28.3145	56.3282
Classic400	36.7592	36.8108	70.5629
Tr23	29.1706	59.4068	88.5799
Tr12	35.7372	41.5007	93.4783
Iris	149.6669	149.7503	149.8383

For each datasets, the convergence behaviors in term of fitness function obtained by the different algorithms are illustrated in Fig. 3, Fig. 4, Fig. 5, Fig. 6 and Fig. 7.

From these figures, it is clear that the ICSCA can reach the best results in a few iterations number for all datasets. Also, we should notice that the gap between graph variation obtained by the CSCA, and CSCA+CEC algorithms is proportional to the number of cluster. In fact, they are close to each other for Classic300 dataset and Classic400. This is due to the small probability of empty cluster. However, for the Tr12 dataset the gap is more significant due to the big number of clusters.

From Fig. 7, it is obvious that the proposed algorithm speed up the convergence behavior of fitness function. Thus, the cosine distance is accurate for the clustering of the standard dataset.

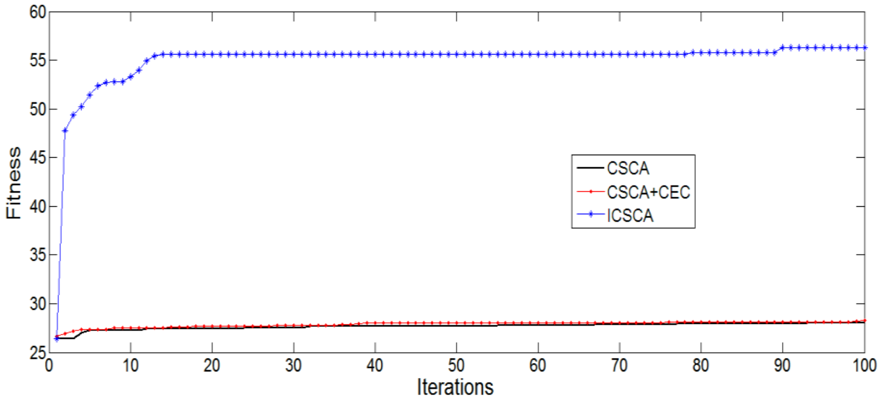

Fig. 3. Graph variation of fitness function of Classic300

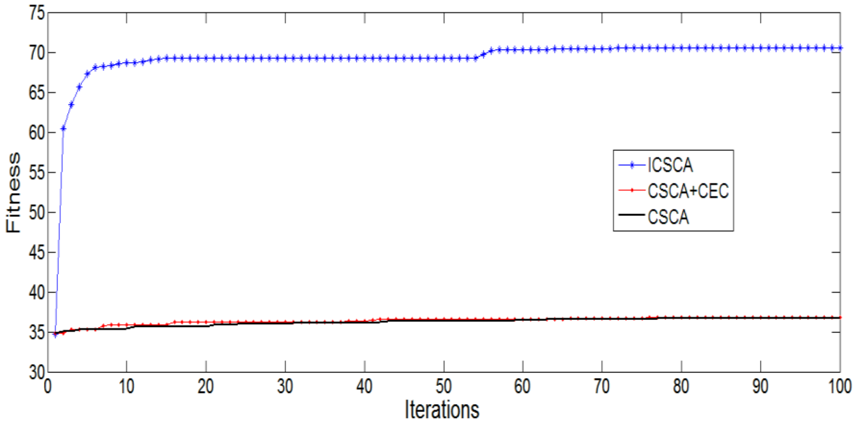

Fig. 3. Graph variation of fitness function of Classic400

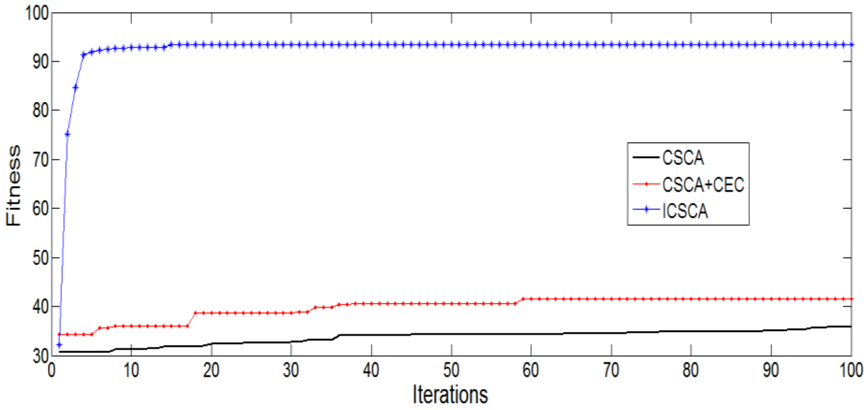

Fig. 4. Graph variation of fitness function of Tr12

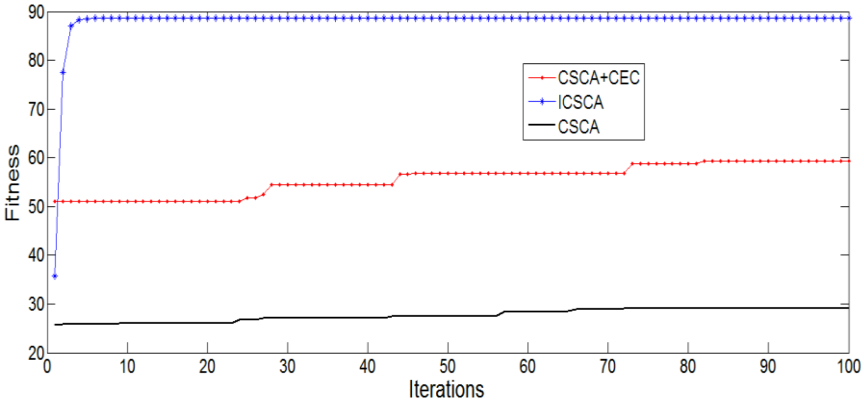

Fig. 5. Graph variation of fitness function of Tr23

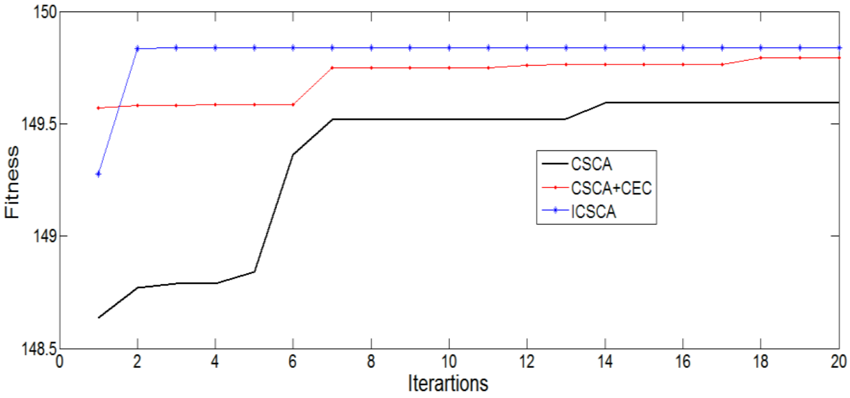

Fig. 6. Graph variation of fitness function of Iris

The recorded F-measure and purity by the different algorithms for each dataset is given in Table 4. and Table 5. From these tables, it is clear that the proposed algorithm can improve significantly the quality of the clustering results.

Table 4. F-measure comparison of CSCA, CSCA+CEC and ICSCA on the differents datasets.

Datasets	CSCA	CSCA+CEC	ICSCA
Clasic300	0.3800	0.4160	0.7728
Classic400	0.4109	0.4308	0.6878
Tr23	0.3997	0.4636	0.5476
Tr12	0.2851	0.4187	0.6017
Iris	0.7778	0.9131	0.9666

Table 5. Purity comparison of CSCA, CSCA+CEC and ICSCA on the differents datasets.

Datasets	CSDC	CSDC+CCN	ICSDC
Classic300	0.3854	0.3895	0.7863
Classic400	0.3933	0.4071	0.6468
Tr23	0.3138	0.4672	0.4710
Tr12	0.3923	0.4865	0.6532
Iris	0.6667	0.9158	0.9697

The calculated percents that ICSCA improve upon the CSCA+CEC, in terms of fitness function (CPF), f-measure (CPFM) and purity (CPP) is presented in Table 6. It can be stated from this table that the proposed ICSCA is more effective than the CSCA+CEC.

Table 6. Percents improvements of ICSCA improve upon the CSCA+CEC

Datasets	CPF(%)	CPFM(%)	CPP (%)
Classic300	0.4973	0.4616	0.5046
Classic400	0.4783	0.3736	0.3705
Tr23	0.3293	0.1533	0.0080
Tr12	0.2552	0.3041	0.2552
Iris	0,0022	0,0553	0,0555

7 Conclusion

The paper presents an improved cuckoo search clustering algorithm (ICSCA). The novelty of the proposed algorithm is to enhance the conventional cuckoo search clustering by a local search procedure. The experiment results show that the proposed

ICSCA is more robust than the CSCA+CEC and CSCA, in term of fitness value, f-measure and purity, when applied on four well known text document dataset and Iris standard dataset. Furthermore, the percent improvement of ICSCA upon the CSCA+CEC is significant.

The proposed ICSCA can also speed up significantly the convergence behavior when applied on Iris standard dataset. Therefore, the cosine distance is accurate for the clustering of the standard dataset. Finally, as future work, we plan to extend the proposed approach for the incremental document clustering.

References

1. Jain, A. K., Murty, M. N. and Flynn, P. J.: Data clustering: a review. ACM Computing Surveys (CSUR), Volume 31 Issue 3, Pages 264-323, Sept. 1999.
2. Dipak, . P., Mukesh, Z.: A Review on Web Pages Clustering Techniques. In: Proceedings of the International Conferences, NeCOM. WeST. WiMoN, Communications in Computer and Information Science, July 15-17 2011, Chennai India. D. C. WYLD, M. WOZNIAK, N. CHAKI: Springer Berlin Heidelberg, vol. 197, p. 700-710, 2011. doi: 10.1007/978-3-642-22543-7_72.
3. Huang, X., Su, W.: An Improved K-means Clustering Algorithm. Journal of Networks, Jan 2014, Vol 9, No 1 (2014), 161-167, doi:10.4304/jnw.9.01.161-167.
4. Hruschka, E. R, Campello, R. J. G. B, Freitas, A. et al.: A Survey of Evolutionary Algorithms for Clustering. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, March 2009, Volume 39 Issue 2, p. 133-155. doi: 10.1109/TSMCC.2008.2007252.
5. Yang, X.-S., Deb, S. Cuckoo Search via Levy Flights. In: Proceedings of World Congress on Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, 9-11 Dec. 2009, Coimbatore, IEEE Publications, pp. 210–214, doi: 10.1109/NABIC.2009.5393690.
6. YANG, X.-S., Deb, S.: Engineering Optimisation by Cuckoo Search. International Journal of Mathematical Modelling and Numerical Optimisation, September 2010, Volume 1, Number 4/2010, pp. 330–343, , doi: 10.1504/IJMMNO.2010.03543.
7. Ishak Boushaki, S., Kamel, N. and Bendjeghaba, O.: A New Algorithm for Data Clustering Based on Cuckoo Search Optimization. In: J. S. PAN, P. KRÖMER and V. SNÁŠEL. Genetic and Evolutionary Computing, Proceedings of the Seventh International Conference on Genetic and Evolutionary Computing, ICGEC 2013, August 25 - 27, 2013 - Prague, Czech Republic, Advances in Intelligent Systems and Computing, Springer International Publishing, 2014 , vol. 238, p. 55-64. doi: 10.1007/978-3-319-01796-9_6.
8. Civicioglu, P., Besdok, E.: A Conceptual Comparison of the Cuckoo-search, Particle Swarm Optimization, Differential Evolution and Artificial Bee Colony Algorithms. Artificial Intelligence Review, Springer Netherlands, 2013, Volume 39, Issue 4, pp. 315-346 , doi: 10.1007/s10462-011-9276-0.
9. Civicioglu, P., Besdok, E.: Comparative Analysis of the Cuckoo Search Algorithm. In: YANG, X.-S. (ed.), Cuckoo Search and Firefly Algorithm, Theory and Applications, Studies in Computational Intelligence, Springer International Publishing Switzerland, 2014, Series Volume 516, pp. 85-113, doi: 10.1007/978-3-319-02141-6_5.
10. Ishak Boushaki, S., Kamel, N. and Benjeghaba, O.: A New Hybrid Algorithm for Document Clustering Based on Cuckoo Search and K-means. In: T. HERAWAN, R.

GHAZALI and M. MAT DERIS. Recent Advances on Soft Computing and Data Mining, Proceedings of The First International Conference on Soft Computing and Data Mining (SCDM-2014) Universiti Tun Hussein Onn Malaysia, Johor, MalaysiaJune 16th-18th, 2014, Advances in Intelligence Systems and computing, Springer International Publishing Switzerland, 2014, Volume 287, p. 59-68. doi: 10.1007/978-3-319-07692-8_6.

11. Kamel, N., Ouchen, I. and Baali, K.: A Sampling-PSO-K-means Algorithm for Document Clustering. In: J. S. PAN, P. KRÖMER and V. SNÁŠEL. Genetic and Evolutionary Computing, Proceedings of the Seventh International Conference on Genetic and Evolutionary Computing, ICGEC 2013, August 25 - 27, 2013 - Prague, Czech Republic, Advances in Intelligent Systems and Computing, Springer International Publishing, 2014 , vol. 238, p. 45-54. Doi: 10.1007/978-3-319-01796-9_5.
12. Cagnina, L., Errecalde, Ingaramo, M., D., et al. An Efficient Particle Swarm Optimization Approach to Cluster Short Texts. Information Sciences. Volume 265, 1 May 2014, Pages 36–49. doi: 10.1016/j.ins.2013.12.010.
13. Song, W., Zhen liang, J. and Cheol Park, S.: Fuzzy Control GA with a Novel Hybrid Semantic Similarity Strategy for Text Clustering. Information Sciences, 20 July 2014, Volume 273, p. 156-170. doi: 10.1016/j.ins.2014.03.024.
14. Azaryuon, K. and Fakhar, B.: A Novel Document Clustering Algorithm Based on Ant Colony Optimization Algorithm, Journal of mathematics and computer Science Vol.7 , pp. 171-180, 2013.
15. Devi S, S. and Shanmugam, Dr. A.: Hybridization of K-means and Harmony Search Method for Text Clustering Using Concept Factorization. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 3 Issue 8, August 2014.
16. Salton, G., Wong, A., Yang, C.S. A Vector Space Model for Automatic Indexing. Communications of the ACM, 18(11), pp. 613-620, 1975. doi: 10.1145/361219.361220.
17. Anna Huang.: Similarity Measures for Text Document Clustering”, NZCSRSC 2008, April 2008, Christchurch, New Zealand.
18. Zhao, Y. and Karypis, G.: Empirical and Theoretical Comparisons of Selected Criterion Functions for Document Clustering. Machine Learning, 55, 311–331, 2004, 2004 Kluwer Academic Publishers. Manufactured in The Netherlands.
19. Xing, B. and Gao, W.-J:. Cuckoo Inspired Algorithms. In: Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms, Intelligent Systems Reference Library, Part II, Chapter 7, Springer International Publishing Switzerland, 2014, Series Volume 62, pp. 105-121, doi: 10.1007/978-3-319-03404-1_7.
20. Classic3 and Classic4 DataSets, Tunali, Volkan, http://www.dataminingresearch.com/index.php/2010/09/classic3-classic4-datasets/
21. “Text retrival conference TREC.” [Online]. Available: http://trec.nist.gov/