I J C T A, 9(26) 2016, pp. 201-207

DPPSOK Algorithm for Document Clustering

M Thangarasu[1] and H Hannah Inbarani[2]

ABSTRACT

Data clustering is a data consideration technique that tolerates objects with related individualities to be clustered together in order to simplify their surplus processing. Numerous clustering algorithms have been proposed by researchers and used in several areas like documentation, marketing, social networks, security etc. The *K*-means algorithm is one of the important clustering algorithms. It is efficient but its performance is very complex for the initialization of clusters. Many solutions have been proposed to address this problem. In this research, a hybrid algorithm is proposed for document clustering. The proposed algorithm is based on *K*-means, PSO and parallel execution approach. It is evaluated on datasets and the results are compared to those obtained by the algorithms *K*-means, PSOK and DPPSOK-means. The experimental results show that the proposed algorithm generates the better result compared to the existing algorithms.

Keywords: Parallel, Clustering, Text, Parallel, PSOK-Means, Distributed.

1. INTRODUCTION

Data mining is the method of analyzing information from the Knowledge Discovery and summarizing into helpful data. It uses mathematical algorithms to part the data and evaluate the same for identifying the probability of future events. Text mining, also referred to as text data mining, roughly corresponding to text analytics, also refers to the process of deriving high-quality information from the text [2]. Clustering is the task of federation of a set of objects in such a way that objects in the same group (called cluster) are more similar to each other than to those in other groups [3]. The notion of a "cluster" cannot be precisely defined, that is one of the reasons for proposing numerous clustering algorithms.

In the year of 2015 [1] Yang Yan Li hui Chen and William-Chandra Tjhi, proposed Fuzzy semi-supervised co-clustering for text documents. This research work proposes a new heuristic semi-supervised fuzzy co-clustering algorithm (SS-HFCR) for categorization of large web documents. Based on this research, in this paper, the hybrid algorithm PSOK with parallel technique is proposed. The parallel environment is helpful to reduce the time complexity of proposed algorithm.

The organization of the rest of this paper is as follows: Section 2 illustrates the related work done in big data environment. Section 3 deals with description and implementation of the PPSOK-means algorithm. Section 4 illustrates the experimental analysis of the proposed model. Proposed work is concluded in Section 5.

2. RELATED WORK ON DOCUMENT CLUSTERING

Jayaraj *et. al.* proposed Correlated concept based dynamic document clustering algorithms for newsgroups and scientific literature. In this research, Term frequency based Maximum Resemblance Document Clustering (TMARDC), Correlated Concept based Maximum Resemblance Document Clustering (CCMARDC) and Correlated Concept based Fast Incremental Clustering Algorithm (CCFICA) algorithms were implemented for clustering the news corpora [2]. In the year of 2008, Sophoin *et. al.* proposed a Novelty-based Clustering

1 Department of Computer Science, Periyar University. *E-mail: thangarasumathan@gmail.com*

2 Department of Computer Science, Periyar University. *E-mail: hhinba@gmail.com*

Method for clustering the On-line Documents [3]. Multi-view document clustering via ensemble method is proposed by Syed *et. al.* [4].

Hassan H Malik and *et. al.* proposed the Hierarchical document clustering using local patterns for clustering the document in an efficient way [5]. In the year of 2013, Wim De *et. al.* proposed the Representations for multi-document event clustering [6] for clustering the news documents. Several researchers proposed various algorithm techniques for clustering the documents [7-8-9-10].

This research focuses on hybrid the PSOK-Means with Parallel Computing using distributed array technique named DPPSOK for improving the speed of document clustering. The detailed description of proposed approach DPPSOK is discussed in the following sequel.

3. IMPLEMENTATION OF DPPSOK-MEANS ALGORITHM

The PSO algorithm is used to spawn initial centroids for *K*-means. This algorithm is proposed for document clustering [16]. The Particle Swarm Optimization (PSO) algorithm is an optimization technique based on population. It can find an optimal solution [16]. The hybrid algorithm DPPSOK includes two modules: the DPPSO module and the *K*-means module. Firstly, the DPPSO module is executed to determine the centroids of the clusters. The algorithm PSO is used at the first step to find the neighboring points of the optimal solution. Then the resulting centroids are used by the *K*-means module in order to refine and generate the optimal solution. Finally these two phases are executed in the distributed and parallel environment.

A. PSO Module

PSO-based *K*-means clustering algorithm [16], recognized as PSOK and *K*-means clustering algorithm is incorporated with PPSO. An enhanced performance can be local finest resolution found so far by the i^{th} particle, while P_g stands for the positional coordinates of the healthy particle found so far in the whole cluster. Once the iterations are completed, most of the particles are projected to converge to a small radius nearby the global optima of the search space. In PSO, an inhabitant of conceptual 'particles' is initialized among random positions X_i and velocities V_i, and function, f, is calculated, using the particle's positional coordinates as input values. In an n-dimensional search space, $X_i = (x_{i1}, x_{i2}, x_{i3}, ..., x_{in})$ and $V_i = (v_{i1}, v_{i2}, v_{i3}, ... v_{in})$: Positions and velocities are adjusted, and the function is estimated with the new Coordinates at every time-step. The essential update equations for the d^{th} dimension of the i^{th} particle in PSO may be specified as

$$V_{id}^{new} = V_{id}^{old} = + c_1, r_1, p_{id}, x_{id} + c_2, r_2, p_{id}, x_{id} \quad (1)$$

$$X_{id}^{new} = X_{id}^{old} = V_{id}^{new} \quad (2)$$

The variables r_1 and r_2 are random positive numbers, drawn from a uniform distribution and classified by an upper limit R_{max}; which is a parameter of the system. In (2), c_1 and c_2 are called acceleration constants whereas w is called inertia weight. P_b is the local finest solution found so far by the i^{th} particle, while P_g correspond to the positional coordinates of the fittest particle found so far in the entire community. Fig. 1 represents the PSOK Algorithm

In the PSOK-means algorithm, the global search of PSO and the rapidity of convergence of *K*-means are combined. The algorithm PSO is used at the first step to find the neighboring of the optimal solution. The obtained result is then used for the *K*-means algorithm, by using a local search to generate the final result.

B. *K*-means Module

K-means algorithm is based on identifying a preliminary amount of groups, and iteratively changing objects between clusters to union. We set an integer k and allocate n data points $D_{is}, \subset R_d$. We aspire to select the group of k centers C, consequently to minimize the potential function [1].

Algorithm PSOK

Input: *D* dataset, *K*-number of clusters,

Output: *K* overlapping clusters of dataset

Step 1: At the first stage, each particle randomly chooses *K* different d vectors from the Dataset as the initial cluster centroids vectors.

Step 2: For each particle:

(a) Assign each vector in the Data set to the closest centroid vector.

(b) Calculate the fitness value

$$f = \frac{\sum_{j=1}^{N_2}(\sum_{j=2}^{p^1} d(O_i M_{ij})}{N_s}$$

(c) Using the velocity and particle position update equations (1) and (2) and generate the next solutions.

Step 3.Repeat step (2) until one of the following termination conditions is satisfied.

(a) The maximum number of iterations is exceeded or

(b) The average change in centroids vectors between iterations is less than a predefined value.

Figure 1: PSOK-Means Algorithm

$$\varphi = \sum_{x \times D_{is}} \min \|x - c\| 2 \tag{3}$$

This algorithm allocates every position to cluster whose center is nearby. The center organizes are the arithmetic mean calculation for every element one by one over all the points in the cluster. Suppose itr is the convergent limit, the pseudo code of *K*-means algorithm presented in Figure 2.

C. DPPSOK Algorithm

A parfor (parallel for) loop is helpful in state that need many loop iterations of a simple calculation. In particular methods such as pcent ( ) and cent ( ) are overloaded and return results that are themselves distributed arrays. It is instructive to note that this is similar to the parallelism based on threads. The parallelism is hidden inside computational methods and functions that are called from otherwise sequential code.

Algorithm K-means (D, k)

Step 1: Let I = MAX VALUE; $j = 1$

Step 2: Select k centers from D_{is}, let $C(0) = c_1^{(j)}, c_2^{(j)}, ..., c_k^{(j)}$

Step 3: While $i > itr$ do

Step 4: Form k clusters by assigning each point in X to its nearest center

Step 5: Find new centers of the k clusters $c_1^{(++j)}\ c_2^{(++j)}, ..., c_k^{(++j)}$

Step 6: $i \leftarrow \varphi = \Sigma_{x \in Dis} \min \|x - c\|2$

Step 7: Output $C(j)$

Figure 2: *K*-means Algorithm

for $i = 1 : k$	parfor $i = 1 : k$
for $j = 1 : p$	for $j = 1 : p$
centroid1 = p cent(i, j);	centroid1 = p cent(i, j);
centroid2 = cent(i, j);	centroid2 = cent(i, j);
$v(i, j) = v(i, j) + c1 * r1 *$	$v(i, j) = v(i, j) + c1 * r1 *$
(centroid1 – centroid2) + $c2*r2*$	(centroid1 – centroid2) + $c2*r2*$
p cent(p best, j) – centroid2);	p cent(p best, j) – centroid2);
centroid2 = (centroid2 + $v(i, j)$);	centroid2 = (centroid2 + $v(i, j)$);
end	end
end	end

Figure 3: The for-loop on the left can be made parallel by Changing for into parfor as was done on the right (*D*-PPSOK).

The advantage of using distributed arrays over threading is the fact that they can be scaled beyond a single multi-core computer. It supports more computational power and larger matrices to work with.

Concerning the growth of the data size and limitation of a single machine a natural solution to believe Parallelism in a distributed computational environment and according to the Fig. 3 the distributed array technique and Parfor are used to convert the normal processing PSOK into DPPSOK.

Our algorithm is summarized as follows:

1. Choose S sub-samples in the initial collection (this number represents the number of PSO particles).
2. Apply the *K*-means on each sub-sample using random initial centroids. We obtain for each sub-sample a set of centroids.
3. Apply PSO having S particles on the how data using the centroids resulting from the previous step.
4. Apply the *K*-means on all data using as initial configuration the results obtained by PSO.

Evaluation of the Solutions

To evaluate the solutions use the formula defined by the equation (4). This function is used as a fitness function for the PSO algorithm and to evaluate the different results. It measures the average distance between the documents of a cluster and its centroid. The smaller this value is, the more the cluster will be compact. So, it can be used to evaluate the quality of the clusters. To calculate the similarity between two documents m_p and m_j, we often use a distance measure. The most used measures are based on the Minkowski formula given by:

$$D_n(m_p, m_j) = \left(\sum_{i=1}^{d_m} \left| m_{i,p} - m_{i,j} \right|^n \right)^{1/n} \tag{4}$$

For $n = 2$, the formula defines the Euclidian distance. For our work we use the normalized Euclidian distance. The normalized Euclidian distance between the documents m_p and m_j is given as follows:

$$D_n(m_p, m_j) = \sqrt{\sum_{k=1}^{d_m} (m_{pk} - m_{ik})^2 / d_m} \tag{5}$$

Where m_p and m_j are two document vectors; d_m is the size of the vectors; m_{pk} and m_{jk} represents respectively the weight values of the k^{th} term in the collection for the documents m_p and m_j.

4. EXPERIMENTAL ANALYSIS

K-means algorithm, PSOK and DPPSOK algorithm, are evaluated based on the four datasets (http:// trec.nist.gov). Since this document collection is large (11 000 documents with 12 621 201 terms), and the results of some researches, say that it is preferred to use the data with low dimensionality for the PSOK-means algorithm, in this research used the sub-categories separately.

For example, algorithm used the category with the subject Banking and Finance to find the clusters: Commercial Banks, Building societies and Insurance Agencies select the documents with low size to reduce the number of terms. For each algorithm, Euclidian distance is used for the similarity measure.

After several executions, the *K*-means is stable after 20 iterations for most of the datasets. The PSO algorithm needs generally 100 iterations to reach a stable solution. Experimental result results show that hybridization of PSO and *K*-means returns the best solution than the previous algorithms, and it stabilizes after 50 iterations.

Two kinds of parameters used for evaluating the algorithm like the parameters that are fixed in the program and they do not change (see Table 1.), and the parameters we change at each test (see table 2.). The choice of the fixed parameters is based on the results of [18]. In order to explore the large data space, we choose to use 50 particles for the PSO algorithm and 25 particles where PSO is used (*K*-means, PSOK, DPPSOK).

Since the number of stamps is related to the number of particles, we used 25 sub stamps.

Table 1
Fixed PSO Parameters

Parameter	*Value*
Inertia factor	0.3
Confident coefficient at its best position.	0.72
Confident coefficient at its neighboring	1.49

Table 2
Parameters of the DPPSOK algorithm

Parameter	*Value*
Number of clusters	5
Number of iteration in DPPSOK	5-50
Number of iteration in *K*-means	10-25
Number of particles	10-25
Dimension of the problem	500-10.000
Size of a stamp	26-65
Number of documents to classify	137-655

The fitness function evaluates the different generated solutions. The best solution is the one which minimizes the value of this function. Table 3 shows the results obtained by using the algorithms *K*-means, PSOK and DPPSOK. The values represent the average of the fitness values for 10 simulations performed separately. The data sets: Dataset 1, Dataset 2, Dataset 3, Dataset 3 and Dataset 4 represent respectively the

document collections: “Banking and Finance”, “Programming Languages”, “Science” and “Sport”. The results show that the DPPSOK-means algorithm generates the lowest values of the fitness function. This means that the clusters generated by this algorithm are the most compact.

Table 3
Performances of the implemented algorithms (fitness values)

	K-means	*PSOK*	*DPPSOK*
Dataset 1	5.093	6.982	4.098
Dataset 2	4.788	4.871	4.021
Dataset 3	7.245	8.632	6.883
Dataset 4	9.09	10.093	7.902

CONCLUSIONS

In this paper, DPPSOK algorithm is proposed for getting efficient document clusters based on parallel technique. According to the experimental analysis, the performance of DPPSOK is better than the *K*-means and PSOK algorithms. Further work on this research might implement a more comprehensive study both on more mixed test datasets, as well as on real time datasets.

REFERENCES

[1] Yang Yan Lihui Chen and William-Chandra Tjhi, “Fuzzy semi-supervised co-clustering for text documents”, Fuzzy Sets and Systems, Vol. 215, 74–89, 2013.

[2] Jayaraj Jayabharathy and Selvadurai Kanmani, “Correlated concept based dynamic document clustering algorithms for newsgroups and scientific literature”, Decision Analytics, Springer, 3-23, 2014.

[3] Sophoin Khy Yoshiharu Ishikawa and Hiroyuki Kitagawa, “A Novelty-based Clustering Method for On-line Documents”, World Wide Web, Springer, 1–37, 2008.

[4] Syed Fawad Hussain, Muhammad Mushtaqand Zahid Halim, “Multi-view document clustering via ensemble method”, Springer, Intelligent Information System, 81–99, 2014.

[5] Hassan H Malik, John R Kender, Dmitriy Fradkin, Fabian Moerchen, “Hierarchical document clustering using local patterns”, Springer, Data Mining Knowlodge Discovery, 153–185,2010.

[6] Wim De Smet, Marie Francine Moens, “Representations for multi-document event clustering”, Springer, Data Mining Knowlodge Discovery, 533–558, 2013.

[7] Argyris Kalogeratos, Aristidis Likas, “Text document clustering using global term context Vectors, Springer, Knowlodge Information System, 455–474, 2012.

[8] Ying Zhao, George Karypis, “Empirical and Theoretical Comparisons of Selected Criterion Functions for Document Clustering”, Springer, Machine Learning, 311–331, 2004.

[9] Huifang Ma, Weizhong Zhao, Zhongzhi Shi, “A nonnegative matrix factorization framework for semi-supervised document clustering with dual constraints”, Springer, Knowledge Information System, 629–651, 2013.

[10] Ali Aïtelhadj, Mohand Boughanem, Mohamed Mezghiche and Fatiha Souam, “Using structural similarity for clustering XMLdocuments”, Knowlodge Information System, Springer, 109–139, 2012.

[11] Illhoi Yoo, Xiaohua Hu, and Il-Yeol Song, “A coherent graph-based semantic clustering and summarization approach for biomedical literature and a new summarizationevaluation method”, BMC Bioinformatics, 1-15, 2007.

[12] Claudio Corsi, Paolo Ferragina and Roberto Marangoni, “The BioPrompt-box: an ontology-based clustering tool for searching in biological databases”, BMC Bioinformatics, 1-9, 2007.

[13] Tony Ohmann and Imad Rahal, “Efficient clustering-based source code plagiarism detection using PIY”, Knowlodge Information System, Springer, 445–472, 2015.

[14] M Thangarasu, H Hannah Inbarani, “Analysis of *K*-means with Multi View Point Similarity and Cosine Similarity Measures for Clustering the Document”, International Journal of Applied Engineering Research, 6672-6675, 2015.

[15] Charu C Aggarwal and ChengXiang Zhai. “A survey of text clustering algorithms”. Mining Text Data,77-128, 2012.

[16] Nadjet Kamel, Imane Ouchen, Karim Baali, “A Sampling PSOKmeans Algorithm for Document Clustering”, Advances in Intelligent Systems and Computing, Springer, 45-54, 2013.

[17] Piotr Luszczek, “Parallel Programming in MATLAB”, International Journal of High Performance Computing Applications”, ACM DL Digital Library, 277-283, 2009.

[18] Patrick J F Groenen, Rudolf Mathar and Willem J Heiser, “The majorization approach to multidimensional scaling for Minkowski distances”, Journal of Classification,Springer, 1995, pp. 3-19

[19] Shi, Y., Eberhart, R C, “Parameter selection in particle swarm optimization”. LNCS, Springer, 591–600, 1998.