A Comparitive Study on Document Clustering Techniques

Mr. Ashams Mathew(Author)
Department of Computer Science and Engineering,
Christ University Faculty of Engineering, Kanmanike, Bangalore, India,

Mrs. Mausumi Goswami(Guide)
Department of Computer Science and Engineering,
Christ University Faculty of Engineering, Kanmanike, Bangalore, India,

***Abstract*— Information retrieval is the twenty first century's biggest challenge as the information being generated daily accounts for billions of bytes. The retrieval process is the costliest process to tackle as it involves taking the right information from the right place and delivering for the needed ones within a specified time limit. Another major factor involving in this process is the time. Getting the correct information within a stipulated time is of a matter of concern. There lies the importance of clustering the web documents. Not only for information retrieval, but for a variety of processes can be done by using the clustering. Clustering is used to group a set of objects based on a specific set of criteria. A variety of clustering techniques is being used, but which among gives the optimal result with maximum performance matters. As the sizes of the documents as well as the number of documents are increasing day by day, performance really matters in the case of document clustering. This paper lights upon a comparative study on the various methods of document clustering using the background knowledge and without using the background knowledge. This paper aims at evaluating the document clustering by setting up tools to collect the web documents, pre-process the documents, cluster the documents based on a set of algorithms and evaluate each of the methods. The major clustering methods used for the comparison is the K-means clustering, Particle Swarm Optimization Clustering and other varieties. As said earlier the increasing number of documents contributes to a big data problem. This paper also investigates the significance of hadoop architecture in the document clustering with various parallel algorithms implemented on the same. This will give a clear idea on the techniques and trends in the document clustering and signifies the importance of each of the methods and finds the best one.**

Keywords*— *document clustering, k-means, particle swarm optimization, parallel

I. INTRODUCTION

Today, the world is run by the information. The amount of information being generated counts to millions of terabytes per day. And the internet based companies are trying to utilize the maximum amount of data for analyzing and grouping. The importance of clustering lies in the middle of this scenario. The problem of categorizing the data based on their similarity is of very high importance for various criticality applications, such as transcriptomics, sequence analysis, human genetic clustering, medical imaging, market research, social network analysis, crime analysis, petroleum geology etc. The information explosion has drastically increased the need for better information retrieval mechanisms from raw data, saving time and money. Clustering is the art of grouping a number of data objects together which are similar to one another by some measure. This is a common technique used for statistical data analysis, which is a major part of data mining. It has been used in the fields such as, machine learning, pattern recognition, information retrieval etc. For all these reasons, the document cluster domain is worth studying and analyzing. This paper highlights the importance and need for clustering method or techniques for various applications. This paper also describes upon the existing algorithms and its performance on document clustering problem. The two major algorithms that will be discussed later are the K-means algorithm and the Particle swarm optimization (PSO) algorithm. Both are of very much importance as they have their own advantages and disadvantages. The paper aims at giving a clear idea on the clustering algorithms, how they work and the performance of each, for the text analysis domain. This also gives us a conclusion, which algorithm to use to get better results with high efficiency also. This comparative study can be helpful for the real time problem domain, which involves clustering for categorizing data. The rest of the chapters are arranged as, chapter 2 will give s brief idea about the document pre-processing stages and processes involving the same. Chapter 3 deals about the Vector Space Model that will be generated from the pre-processed clusters. It gives a clear picture on the types of the model and the uses of each as well. The fourth and the fifth chapter signify the K-Means algorithm and PSO algorithm. Sixth chapter gives idea about the proposed method with the motive for the modification.

II. DOCUMENT PREPROCESSING

Document Pre-processing is the process of incorporating a new document into the information retrieval system. The major goal in the document preprocessing system is to represent the documents in terms of space and time requirements effectively. It should also maintain good retrieval performance as well. It is a complex process such that, it leads to the representation of each document by a selected set of index terms. The document preprocessing includes mainly five stages.

A. Lexical Analysis
B. Stopwords Removal
C. Stemming

A. *Lexical Analysis*
The objective of lexical analysis is to determine the words present in a document. Lexical analysis separates an input alphabet into word characters (letters A to Z) and word separators (space, newline, tab). In the process the usually the digits are being ignored, but the numbers like telephone numbers are identified as words. Punctuation marks are treated as word separators.

B. *Stop Words Removal*
The objective of the method is to filter out the words that occurs most in the documents. Such words have no value in retrieval process. These words are referred to as stop words. They include pronouns (it, them, you, I,…), articles (an, a, the,..), prepositions (in, on, of,....), conjunctions (but, and, or, if,...), etc. The stop words list may include several hundreds of words. Stop words removal improves the indexing size of the structures.

C. *Stemming*
Stemming removes all the variants of a word with the single stem of the word. The variants which will be stemmed include plurals, 'ing'-forms, third person suffixes, past tense suffixes, etc.
As an example, words such as connected, connecting, connection all relates to a single word 'connect'. Stemming improves the storage and search efficiency, since fewer terms are stored.

III. VECTOR SPACE MODEL GENERATION

Vector space model is an algebraic model [8], which is being used to represent text documents as vectors of indexes. Vector space model is in indexing, information retrieval, and relevancy ranking. In the model, the documents are represented as vectors.

$D_j = (w_1,j,w_2,j,w_3,j,\ldots)$

$q = (w_1,q,w_2,q,\ldots..,w_t,q)$

Vector space model is constructed as a matrix with rows as documents and columns as terms. Each dimension or the column corresponds to a separate term. . Several different ways of computing these values have been developed. There are three model developed based on this as well, they are the

1. Binary model
2. Term Frequency (TF) Model
3. Term Frequency-Inverse Document Frequency (TF-IDF) Model

A. *Binary Model*
If a term occurs in the document, its value in the vector is represented by integer one. If it does not occur in the document, its value is represented as zero.

B. *Term Frequency (TF) Model*
The model stores a term frequency into the feature vector. This is computed from an occurrence count of a term in a document, normalized by a number of all terms in a document.

C. *Term Frequency-Inverse Document Frequency (TF-IDF) Model*
The term-specific weights in the document vectors are products of local and global parameters. A document may be represented as

$D_d=[W_{1,d},W_{2,d},W_{3,d},\ldots\ldots,W_{t,d}]$

Where

$W_{t,d}= tf_{t,d} . \log |D| / | \{d' E D | t E d' \} |$

$tf_{t,d}$ is the term frequency of the term 't' in the document 'd' (local parameter)

The usage of term depends on the application. Mostly terms are keywords, single words, or longer phrases. If single words are determined as the terms, the dimensionality of the vector is the number of words in the documents.

IV. K-MEANS ALGORITHM

A. *Overview*

K Means is an unsupervised clustering algorithm that is popular for cluster analysis in data mining. It partitions 'n' data vectors into 'k' clusters in which each data vector belongs to the cluster with nearest mean[4]. This is typically an NP-hard problem. Given a set of documents (d1, d2, …, dn), where each document is a m-dimensional real vector, k means clustering aims to partition the n documents into k sets ($k \leq n$) $C = \{C1, C2, \ldots, Ck\}$ so as to minimize the within-cluster sum of squares (WCSS):

B. *Algortihm*

K Means algorithm clusters a group of data vectors into a predefined number of clusters. It starts with random initial cluster centroids and keeps reassigning the data objects in the dataset to cluster centroids based on the similarity between the data object and the cluster centroid. A convergence criterion is needed to stop the reassignment procedure, which will usually be either the number of iterations, or centroid does not change after certain number of iterations.

Algorithm k-means (D, K)

1. Choose K data points as the initial centroid.
2. Compute the distance from d, belongs to D, to each centroid and assign d to the closest centroid.
3. Re-calculate the centroid again using the current cluster memberships and go back to Step2 until the stopping criterion is met.

C. *Advantages*

K-Means has got good geometry and statistical significance in the numeric attribute. It is also less sensitive to order. It has good effect on the convex cluster and can run in parallel.

D. *Drawbacks*

The major drawback is that the user should give the number of cluster in advance. K-means is also unable to processes categorical attribute data and is sensitive to the

isolated points. One of the major setbacks is that it cannot discover clusters which are having great size differences and non-spherical clusters. Mostly, K-means results fall into local optimal solution and are unable to obtain the global optimal solution.

V. BASIC PARTICLE SWARM OPTIMIZATION

A. *Overview*

Particle swarm optimization is a computational method that optimizes a problem by trying to improve a candidate solution iteratively, in regards to a given measure. It improves a problem by having a group of candidate solutions and moving these particles around in the space according to simple mathematical formulae over the particle's position and velocity. The local best known position of the particle is said to influence the particles movement, but it is also guided towards the best position found by other particles. This moves the swarm to the best solution. PSO is originally attributed to Kennedy, Eberhart and Shi[1][2] and was first intended for simulating social behavior[3] as a stylized representation of the movement of organisms in a bird flock or fish school. The algorithm was simplified and it was observed to be performing optimization.

B. *Algorithm*

A basic variant of the PSO algorithm works by having a swarm of particles. These particles are moved around in the search-space according to some formulae. The movements of the particles are guided by two factors. First the particles own best known position in the search-space and second the entire swarm's best known position. When improved positions are being discovered these will be updated as the new guiding measure. The process is repeated to get a satisfactory solution.

The Algorithm works as follows

Step1: Initialize the swarm position and velocity vectors and other factors

Step 2: Choose k random document vectors from the collection and store it as initial cluster centroids

Step 3: For each particle:

- Assign the closest document vector to the centroid
- Calculate the fitness function based on the parameters
- Use the velocity and local best known position to update the best known position to generate a solution

Step 4: Repeat Step3 until termination condition is reached

C. *Advantages*

- PSO is based on swarm intelligence.
- PSO is having no overlapping and mutation calculation. During the development of several generations, only the most optimist particle can transmit information onto the other particles, and the speed of the searching is very fast.
- PSO involves very simple calculation. It occupies bigger optimization ability and can be completed easily. The proposed Method

VI. PROPOSED METHOD

While looking at the above given points, it is clear that every clustering algorithm has their own advantages and disadvantages. According to Merwe's research [5], if PSO is given enough time, it could generate very compact clustering results than the K-means algorithm. But when it comes to performance and time constraints, K-means wins the cup, even though clustering results tend to swing around. Clustering large amount of documents require more iteration stages in the case of PSO algorithm, when compared to K-means which requires less iterations. For a dataset of 1000 documents, PSO took around 600 iterations to reach on to an optimal solution, while in the case of K-Means, it took only around 30 to 40 iterations[5][6] to converge to an optimal solution, which is local.

The major drawback comes in term of execution time. PSO generates better results than the K-means for the same dataset, PSO it taking more execution time. To avoid this problem, we propose a method to combine the PSO and the K-means. Both the global optimum and local optimum is achieved through this. Initially, PSO algorithm is run for about 50 to 100 iterations, which is a short period. The result of the PSO algorithm, which is a centroid matrix, is given as the initial centroid vector input to the K-means algorithm. Then the K-means will be run for the same amount of iteration to generate the same results throughout.

VII. EXPERIMENTAL RESULTS

A. *Datasets*

We have used the reuters21578 data which contains 21,578 documents, which are grouped as 21 datasets. Each datasets contains around 500 to 100 documents. We have taken a single dataset that contains 502 documents for comparison purpose. The documents are preprocessed to remove the stop words like a, an, the, it etc. and stemming to remove the 'ing' form is also done.

B. *Experimental Setup*

Both the K-Means and PSO algorithms are run using the reuters dataset. The results are noted down and tabulated.

	Clustering Algorithms		
Execution laps	**K-Means**	**PSO**	**PSO+K-Means**
Execution lap1	**0.0182**	**0.0082**	**0.0093**
Execution lap2	**0.0175**	**0.0082**	**0.0093**
Execution lap3	**0.018**	**0.0082**	**0.0093**
Execution lap4	**0.0168**	**0.0082**	**0.0093**
Execution lap5	**0.0175**	**0.0082**	**0.0093**
Execution lap6	**0.0179**	**0.0082**	**0.0093**
Execution lap7	**0.0169**	**0.0082**	**0.0093**
Execution lap8	**0.0168**	**0.0082**	**0.0093**
Execution lap9	**0.0175**	**0.0082**	**0.0093**
Execution lap10	**0.0179**	**0.0082**	**0.0093**

Table 1: implementation results for PSO and K-Means.

PSO Clustering algorithm was run for 500 iterations individually and the K-Means algorithm was run for 30 iterations individually. In the combination of PSO and K-

Means, PSO has been run for 50 iterations and K-Means for 10 Iterations.

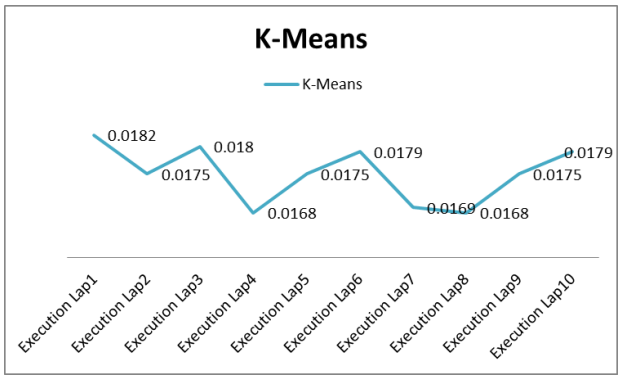

Fig. 1 graph plotted with the results of K-Means run alone

From the figure, its clear that K-Means result swings around and doesnot sticks to an optimal solution. All the values are local optimal solutions only.

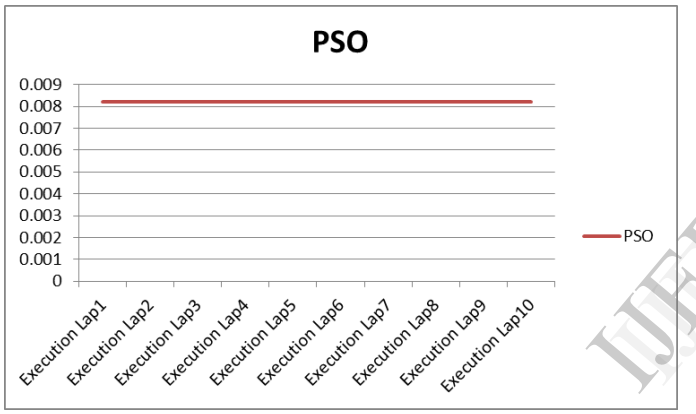

Fig. 2 graph plotted with PSO run alone.

The above figure idicates the constant flow of optimal solution for each exectuion laps. Each execution lap contains the iterations specified. For every execution laps with iteration of 500 each, PSO gives a constant global optimal result.

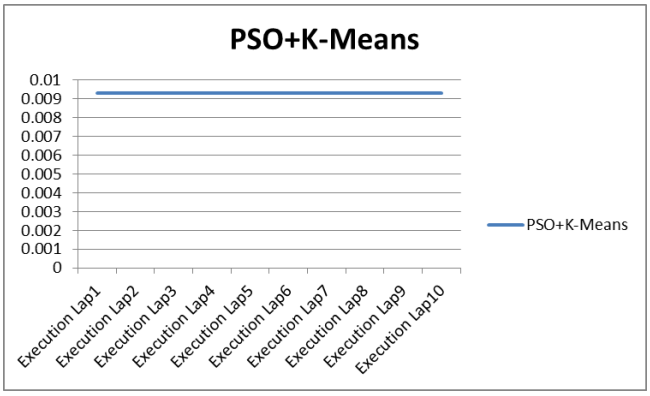

Fig. 3 graph plotted with the results of PSO+K-Means

The above graph shows the rsult for reuters dataset run using thecombination of both PSO and K-Means. First PSO was run for 50 iterations and then using the result K-Means was run for 10 iterations. The same has been executed for consecutive 10 laps to get the same result.

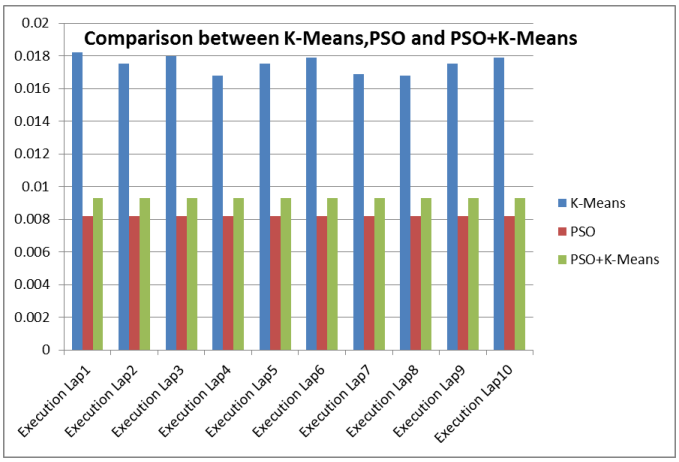

Fig. 4 graph plotted with the comparison results of the three algorithms

VIII. CONCLUSION

From the all the Fig. 4 it is clear that PSO outperforms the traditional clustering algorithm, in the case of generating an optimal solution. But when considering, the number of iterations and the time taken to effectively calculate the same has gone far too much for the PSO than the K-Means. And from the graph we can evaluate that the proposed method gives a better solution, when compared to the traditional K-Means. The best part is that the solution is not changing for the execution loops for K-Means, as its initial centroid is being taken from the PSO output. But still the final optimal solution of K-Means+PSO seems to be not as good as the original PSO since it is not doing the required amount of iterations. But when considering the execution time into the picture, PSO+K-Means is far better when compared to original PSO and excellent than original K-Means, because it gives an optimal solution considering both global and local values, with short execution time.

IX. FUTURE WORK

The Future work involves finding a better method to work with larger dataset. Since the amount of time it takes for a small dataset, like 1000 documents was large, future work involves trying new mechanisms and algorithms to work in parallel to achieve higher performance on large datasets as well.

REFERENCES

[1] Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks IV. pp. 1942–1948. doi:10.1109/ICNN.1995.488968.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

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

[3] Kennedy, J. (1997). "The particle swarm: social adaptation of knowledge". Proceedings of IEEE International Conference on Evolutionary Computation. pp. 303–308.

[4] MacQueen, J. B. (1967). "Some Methods for classification and Analysis of Multivariate Observations". Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1. University of California Press. pp. 281–297. MR 0214227. Zbl 0214.46201. Retrieved 2009-04-07.

[5] Merwe V. D. and Engelbrecht, A. P., 2003. Data clustering using particle swarm optimization. Proceedings of IEEE Congress on Evolutionary Computation 2003 (CEC 2003), Canbella, Australia. pp. 215-220.

[6] Carlisle, A. and Dozier, G., 2001. An Off-The- Shelf PSO, Proceedings of the 2001 Workshop on Particle Swarm Optimization, pp. 1-6, Indianapolis,IN

[7] Shi, Y. H., Eberhart, R. C., 1998. Parameter Selection in Particle Swarm Optimization, The 7th Annual Conference on Evolutionary Programming,SanDiego,CA.

[8] Kennedy J., Eberhart R. C. and Shi Y., 2001. Swarm Intelligence, MorganKaufmann,NewYork.

[9] Everitt, B., 1980. Cluster Analysis. 2nd Edition.Halsted Press, NewYork.