DOI: http://dx.doi.org/10.26483/ijarcs.v9i1.5090

ISSN No. 0976-5697

Volume 9, No. 1, January-February 2018

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

GENERATION OF A HYBRID CLUSTERING ALGORITHM FOR BIG DATA

Deepak Ahlawat
PhD Research Scholar, MMU, Sadopur
Ambala, Haryana, India

Dr.Deepali Gupta
HOD CSE, MMU, Sadopur
Ambala, Haryana, India

Abstract: In this paper, a Hybrid Algorithm for clustering big data is proposed which is based on Rank Similarity. Rank Similarity is calculated by taking the sum of both Cosine and Gaussian Similarity. Proposed Technique is compared with the existing technique which is based on Cosine Similarity only. Comparison is done by taking parameters precision, recall, F-Measure, and accuracy. Results are evaluated on Java Netbeans 8.2.

Keywords: Cosine Similarity, Gaussian Similarity, Rank Similarity.

1. INTRODUCTION

1.1 Big Data

As stated by IBM, with pervasive handheld devices, communication of machine-to-machine, online/mobile social networks, 2.5 quintillion bytes of data is created every day from the last two years. It became tough for the users to store, capture, manage, analyse, share, and visualize with related data and processing tools. Because of this, big data concept has been proposed.

The capability for data generation has never been enormous and powerful since the development of the IT (Information Technology) in the late 19 century. As another example, dated on October 4, 2012, first presidential debate between President Obama and Prime Minister Mitt Romney has debated all these tweets and triggered more than 10 million tweets in two hours and generates the discussion at the specific moment, in fact, reveals the public interest with the discussion on Medicare and vouchers. However, the term 'big data' is still vague. As shown in Wikipedia, Big Data is a data set that contains all the terms of any, large and complex data, difficult to use traditional data processing applications for processing. Widely accepted definition belongs to IDC: 'big data technology describes a new generation of technology and architecture, that aims to achieve high-speed capture, discovery and / or economic analysis to extract value from large amounts of data' to explore the use of large and exceptional value data that must increase the risk of security privacy. For example, 'Amazon' monitor user's shopping preferences. Facebook also seems to attract all the information, as well as our social relationships. Mobile operators not only know to whom the person is talking but the availability of someone to the user. The promising values are in sighted to the one that analyses and the signs depict the further surge in another's storage, re-usage and gathering of the personal data. If the age of the Internet threat to security and privacy, then the era of big data will endanger them. Before moving ahead for what big data is, a moment is required to look at the below diagram by Hewlett-Packard:

Fig.1. Amount of Data Volume

1.2 Clustering

Grouping of data in different sets or classes or in clusters is known as the Clustering. The data which is placed in one cluster is similar to other data in that cluster; also this data is dissimilar to data present in other clusters. Dissimilarities can be calculated according to various attributes.There are various distance measures which describe the dissimilarity in the various data objects. These dissimilarity attributes are then used to construct a Dissimilarity matrix. Clustering of data is useful in various fields like, data mining, statistics, biology, and machine learning.In literature, numerous clustering algorithms are discussed. Every algorithm has its own pros and cons; also they find there use differently in different situations [1]:

Typically clustering algorithms are categorized in the following categories:

1. Partitioning Methods.
2. Density-Based Methods.
3. Hierarchical Methods.
4. Grid-Based Methods.
5. Supervised and Unsupervised Learning Based Methods.

In this paper, basically two important (Partitioning and Density-Based) Methods are exploited to do the clustering.

Partitioning Methods: Suppose there is a database containing n objects or data tuples and the task is to divide these data objects into different clusters, say, K clusters, where k≤ n. Then, the partitioning method is used to do there clustering according to the dissimilarity between various data objects. The objects which are similar are in same group and which are dissimilar are placed in different groups. There are some essential requirements which should be met by the clustering algorithm, these are: (1) the cluster must not be empty, i.e., every cluster should contain at least one data object, and (2) no data object is shared among clusters.k-Means and k-Medoids, are two well-known partitioning clustering methods [2].

Density-Based Methods: Partitioning methods discussed above are used to divide the data into clusters that are mainly of spherical shape. But some time there is a need to cluster the objects in arbitrarily shapes. In these situations, the notion of *density* is used to create the clusters (that may not be of spherical shapes). The idea behind the density-basedmethods is that to add the data objects in a given cluster until its density exceeds some threshold. The concept of neighbourhood similarity is also taken into account. The clusters resulting from density-based methods may be of arbitrarily shapes [3, 4].

2. GENERATION OF CLUSTERS USING COSINE SIMILARITY

2.1 Term Frequency.The TF is a text statistical-based technique which has been widely used in many search engines and information retrieval systems. Assume that there is a collection of 500 documents and the task is to compute the similarity between two given documents (or a document and a query). The following describes the steps of acquiring the similarity value [5, 6]:

1. Document pre-processing steps
   - Tokenization: A document is treated as a string (or bag of words), and then partitioned into a list of tokens.
   - Removing stop words/ Stemming word: Stop words are frequently occurring, insignificant words. This step eliminates the stop words.
2. Document representation
   - Generate the Index terms and then represent them as N-dimensional vector in term space.
3. Computing Term weights
   - Compute the Term Frequency.
   - Do Term Frequency weighting.

After the 3 steps stated above, measure the similarity between two documents:

The cosine similarity can be calculated by measuring the cosine of the angle between two document vectors

$$\text{Cosine Similarity, } s(x, y) = \frac{x^t . y}{||x||||y||} \qquad (1)$$

where, $x^t$is a transposition of vector x, ||x|| is the Euclidean norm of vector x, ||y|| is the Euclidean norm of vector y, and s is the cosine of the angle between vectors x and y [7].

Using the code:

```
//
cosineSim=(double)trw.getCosineMetric(TermWeights,
TermWeights1);
doublemSim=Alpha*(lfv/7)+(1-Alpha)*cosineSim;
DecimalFormatnewFormat = new
DecimalFormat("#.####");
cosineSim = Double.valueOf(newFormat.format(mSim));
returncosineSim;
//
```

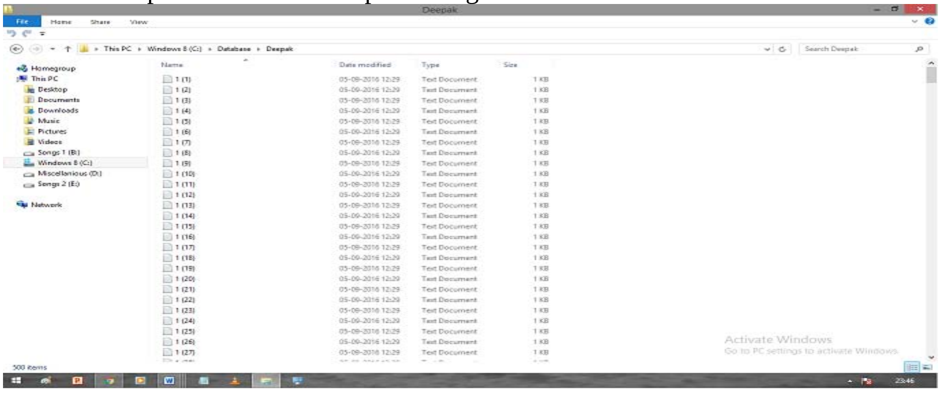
2.2 Cluster Formation

There are numerous clustering algorithms occur in exploration but centroid selection based clustering k-mean algorithm is general because of its simplicity for execution and competence to harvest good results. It is a dividing based methodology which divides dataset into pre-defined k partition known as clusters which have minimum intra cluster distance. K-mean algorithm is based on partition and it will work according no of clusters k given at the time of input. Algorithm arranges all the given objects into k partitions and each partition is known as separate cluster. It is simple and straight forward in nature. In the proposed approach, the clusters are created using *centroid* selection. Initially evaluate the centroid for the different clusters, and then add further documents in the clusters by choosing the cluster whose centroid is nearest to the document.In every step documents are placed in the different clusters and the formation of clusters is done. For each step, there is also a need to calculate the *error function*. If new centroids provide lower error function value then the new centroid will be kept and movement will be continued in same direction otherwise if value of error function is higher than previous then the movement direction will change. Continuing the process till end of all the documents will result in the formation of Clusters.

In this paper, algorithm for Cluster formation using only cosine similarity as measure is called *First Technique* and Cluster formation using Hybrid Clustering Algorithm is called *Second Technique*.

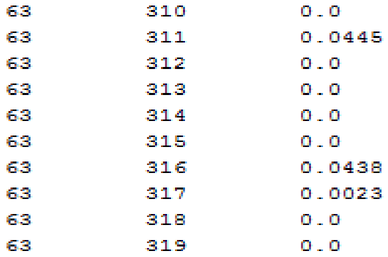
2.3 Results Evaluated in First Technique

Atotal of 500 text files have been uploaded till now for processing



Name	Date modified	Type	Size
1 (1)	05-09-2016 12:29	Text Document	1 KB
1 (2)	05-09-2016 12:29	Text Document	1 KB
1 (3)	05-09-2016 12:29	Text Document	1 KB
1 (4)	05-09-2016 12:29	Text Document	1 KB
1 (5)	05-09-2016 12:29	Text Document	1 KB
1 (6)	05-09-2016 12:29	Text Document	1 KB
1 (7)	05-09-2016 12:29	Text Document	1 KB
1 (8)	05-09-2016 12:29	Text Document	1 KB
1 (9)	05-09-2016 12:29	Text Document	1 KB
1 (10)	05-09-2016 12:29	Text Document	1 KB
1 (11)	05-09-2016 12:29	Text Document	1 KB
1 (12)	05-09-2016 12:29	Text Document	1 KB
1 (13)	05-09-2016 12:29	Text Document	1 KB
1 (14)	05-09-2016 12:29	Text Document	1 KB
1 (15)	05-09-2016 12:29	Text Document	1 KB
1 (16)	05-09-2016 12:29	Text Document	1 KB
1 (17)	05-09-2016 12:29	Text Document	1 KB
1 (18)	05-09-2016 12:29	Text Document	1 KB
1 (19)	05-09-2016 12:29	Text Document	1 KB
1 (20)	05-09-2016 12:29	Text Document	1 KB
1 (21)	05-09-2016 12:29	Text Document	1 KB
1 (22)	05-09-2016 12:29	Text Document	1 KB
1 (23)	05-09-2016 12:29	Text Document	1 KB
1 (24)	05-09-2016 12:29	Text Document	1 KB
1 (25)	05-09-2016 12:29	Text Document	1 KB
1 (26)	05-09-2016 12:29	Text Document	1 KB
1 (27)	05-09-2016 12:29	Text Document	1 KB

Fig.2. Representing unstructured documents



```
63          310          0.0
63          311          0.0445
63          312          0.0
63          313          0.0
63          314          0.0
63          315          0.0
63          316          0.0438
63          317          0.0023
63          318          0.0
63          319          0.0
```

Fig.3. Represents the connection of main document to connecting document with its cosine similarity

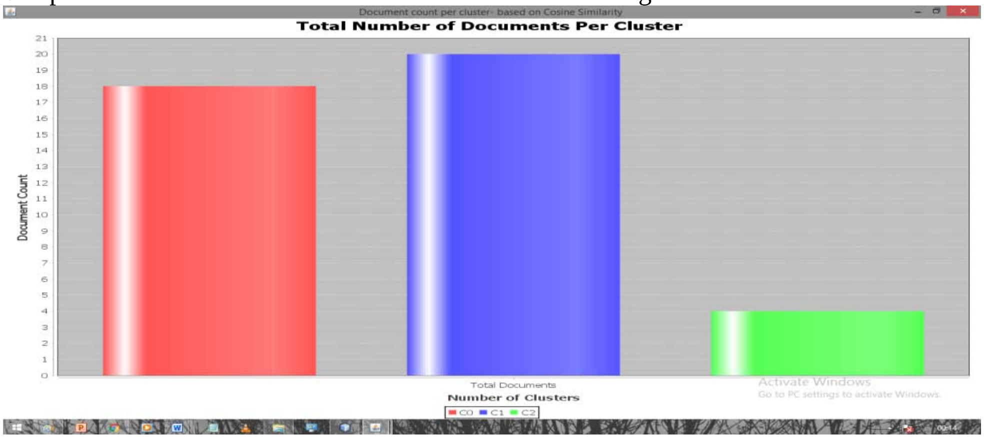
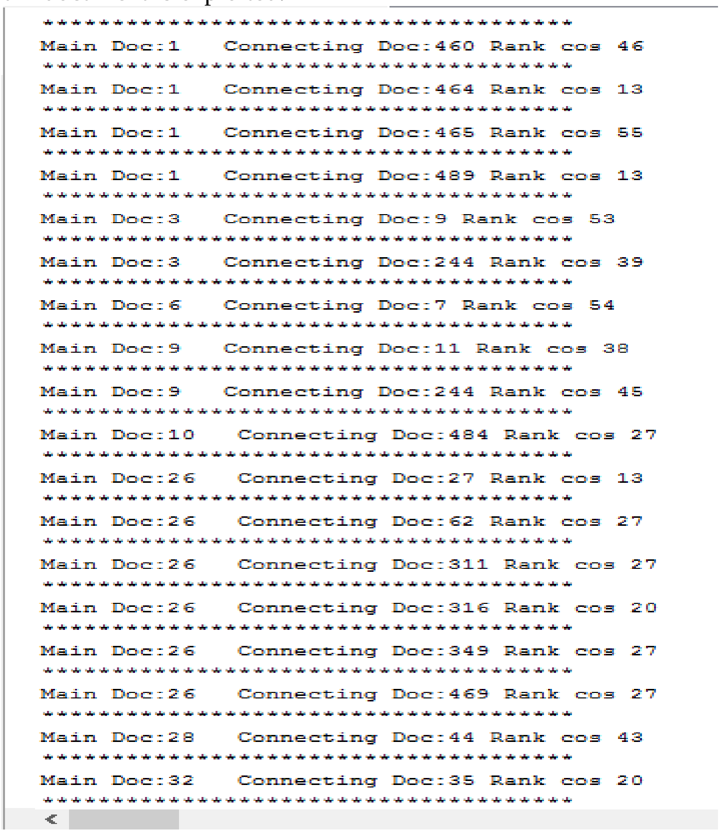


Fig.4. Output representing clusters using First Technique

3. GENERATION OF HYBRID ALGORITHM

Hybrid Algorithm makes use of Rank Similarity, i.e., sum of Cosine similarity and Gaussian similarity [8]. Concept of neighbour reachable from the main document is exploited.

3.1 Results Evaluated in Second Technique



```
**************************************
Main Doc:1   Connecting Doc:460 Rank cos 46
**************************************
Main Doc:1   Connecting Doc:464 Rank cos 13
**************************************
Main Doc:1   Connecting Doc:465 Rank cos 55
**************************************
Main Doc:1   Connecting Doc:489 Rank cos 13
**************************************
Main Doc:3   Connecting Doc:9 Rank cos 53
**************************************
Main Doc:3   Connecting Doc:244 Rank cos 39
**************************************
Main Doc:6   Connecting Doc:7 Rank cos 54
**************************************
Main Doc:9   Connecting Doc:11 Rank cos 38
**************************************
Main Doc:9   Connecting Doc:244 Rank cos 45
**************************************
Main Doc:10   Connecting Doc:484 Rank cos 27
**************************************
Main Doc:26   Connecting Doc:27 Rank cos 13
**************************************
Main Doc:26   Connecting Doc:62 Rank cos 27
**************************************
Main Doc:26   Connecting Doc:311 Rank cos 27
**************************************
Main Doc:26   Connecting Doc:316 Rank cos 20
**************************************
Main Doc:26   Connecting Doc:349 Rank cos 27
**************************************
Main Doc:26   Connecting Doc:469 Rank cos 27
**************************************
Main Doc:28   Connecting Doc:44 Rank cos 43
**************************************
Main Doc:32   Connecting Doc:35 Rank cos 20
**************************************
```

Fig.5. Represents the total rank after the combination of (Cosine Similarity and Gaussian Similarity = Rank Based Similarity)

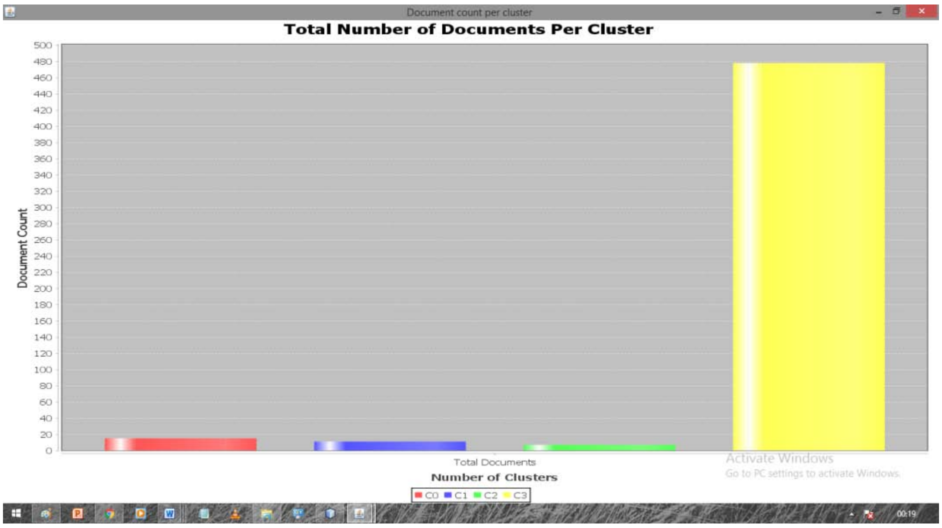
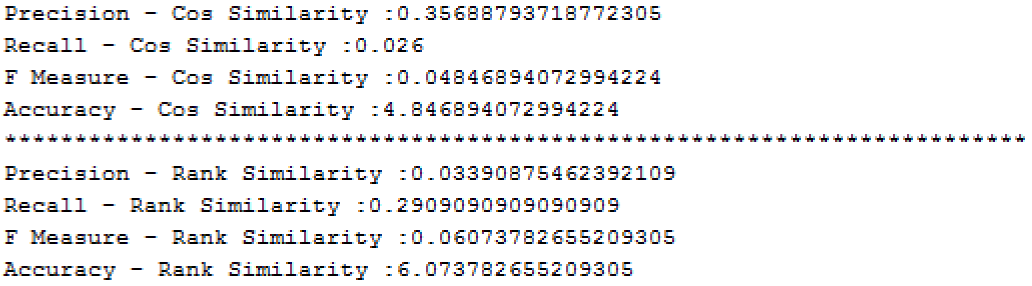


Fig.6. Output representing clusters using Second Technique

3.2 Comparison of Two Techniques



```
Precision - Cos Similarity :0.35688793718772305
Recall - Cos Similarity :0.026
F Measure - Cos Similarity :0.04846894072994224
Accuracy - Cos Similarity :4.846894072994224
***************************************************************************
Precision - Rank Similarity :0.03390875462392109
Recall - Rank Similarity :0.2909090909090909
F Measure - Rank Similarity :0.06073782655209305
Accuracy - Rank Similarity :6.073782655209305
```

Fig.7. Represents the comparison of Cosine Similarity and Combinational Rank Similarity

Where,

$$\text{Precision} = \text{totallink(true link)/totaldocs} \tag{2}$$

$$\text{Recall} = \text{lastlink(last left)/totaldocs} \tag{3}$$

$$\text{F Measure} = \text{(2*precision*recall)/(precision+recall)} \tag{4}$$

$$\text{Accuracy} = \text{f measure*100} \tag{5}$$

4. CONCLUSION

Fig.7. shows that the Precision is reduced in Hybrid Algorithm but the other parameters viz., Recall, F Measure are improved and ultimately results in the more accurate algorithm. The accuracy is increased from 4.84 % to 6.07 %.

REFERENCES

1. Huang A.: Similarity Measures for Text Document Clustering. NZCSRSC 2008, pp. 49-56, Christchurch, New Zealand, April (2008).
2. Han, J. and Kamber, M.: Data Mining: Concepts and Techniques. Elsevier (2006).
3. Deshmukh, H.S. and Ramteke, P.L.: Comparing the Techniques of Cluster Analysis for Big Data. International Journal of Advanced Research in Computer Engineering & Technology, vol. 4 no. 12, pp. 4339-4343, December (2015).
4. Zahid et al., “Fuzzy clustering based on K-nearest-neighbours rule”, *Fuzzy Sets and Systems* 120.2 (2001): 239-247.
5. Satyasree, K.P.N.V. and Murthy, J.V.R.: Clustering Based on Cosine Similarity Measure. International Journal of Engineering Science & Advanced Technology, vol. 2, no. 3, pp. 508-512, May-June (2012).
6. Umamaheswari, U. and Rajesh, K.: Text Clustering Using Cosine Similarity and Matrix Factorization. International Journal of Research in Computer and Communication Technology, vol. 3, no. 10, pp. 1343- 1347, October (2014).
7. Abbas, O.A.: Comparison between Data Clustering Algorithms. The International Arab Journal of Information Technology, vol.5, no. 3, pp. 320-325, July (2008).
8. Radhakrishna, V., Sriniwas, C., and Gururao, C.V.: A Modified Gaussian Similarity Measure for Clustering Software Components and Documents. ISDOC 2014, Lisbon, Portugal, pp. 99-114, May (2014).