An Improved K-means Algorithm Using Modified Cosine Distance Measure for Document Clustering Using Mahout with Hadoop

Lokesh Sahu
Dept. Of Information Technology
National Institute of Technology Karnataka
Surathkal, Karnataka, India
lokesh.sahu33@gmail.com

Mr. Biju R. Mohan
Asst. Prof., Dept. Of Information Technology
National Institute of Technology Karnataka
Surathkal, Karnataka, India
biju@nitk.ac.in

***Abstract*— In this paper, we have proposed a novel K-means algorithm with modified Cosine Distance Measure for clustering of large datasets like Wikipedia latest articles and Reuters dataset. We are customizing Cosine Distance Measure for computing similarity between objects for improving cluster quality. Our method will calculate the similarity between objects by Cosine Distance Measure and then try to bring distance more closer by squaring the distance if it is between 0 to 0.5 else increase it. It will result in minimum Intra-cluster and maximizes Inter-cluster distance value. We are measuring cluster quality in term of Inter and Intra-cluster distances, good Feature weighting such as TF-IDF, Cluster Size and Top terms of the clusters. We have compared K-means algorithm by Cosine and modified Cosine Distance measure by setting performance metric such as Inter-cluster and Intra-cluster distances, Cluster size, Execution time etc. Our experimental result shows in minimizing Intra-cluster by 0.016% and maximizing Inter-cluster distance by 0.012%, reducing the cluster size by 1.5% and reducing sequence file size by 4%, that will result in good cluster quality.**

Keywords—Document Clustering; K-means; Hadoop; Mahout.

I. INTRODUCTION

In recent years automatic document clustering has played an important role in many fields like information retrieval, data mining, etc. Clustering is a technique of alliance the similar form of object based on different classes, grouping of facts will form a cluster of similar kind objects [3]. So it is important that cluster data is as much as similar as possible, that will result a good cluster. Cluster quality heavily depends on the likeness of items in a cluster. Similarity of document can be measure using a distance measure so the apposite distance measure will result in the quality of the cluster [8]. Distance measure played a significant role in text clustering. Many distance measures have been proposed like Euclidean, Squared Euclidean, Cosine, Manhattan, Jaccard etc. K-means with Jaccard distance measure proposed better result [2]. K-means with Euclidean distance measure performs well up to moderately high dimension approximately 10 to 20 [6]. It has been seen that cosine distance measure produces better results for text clustering than Euclidean measure [7]. Clustering is the most essential type of unsupervised learning problem as every other problem of this class, it concentrates on formation of a structure from the pool of uncategorized records. Desirably, the resulting clusters should represent a set of significant classes. These classes should not be of the same kind, but have similar data. Distance measure is having a vital role in many clustering algorithm like in k-means, fuzzy k-means, LDA (Latent Dirichlet Allocation) etc. [5].

Document clustering is a way to organize facts without demanding prior acquaintance about its classification [5]. Samples within a cluster should have high similarity, but are very divergent to samples in other clusters. This similarity has to be measured between objects in a vector space. Here data are visualized as a point in vector space where points are represented as vectors. Distance between points in a vector space is the similarity measure for clustering. Distance is represented as Inter-cluster and Intra-cluster distance. Cluster quality heavily depends on Intercluster and Intracluster distance. Intracluster distance should be minimized as possible as and Intercluster distance is as far as possible between clusters that result in a good cluster quality.

Traditional methods for document clustering found in literature deals with structured data sets. They were lesser in size like in KB, MB etc. So that they could be executed in a singleton machine without using Hadoop cluster [1]. Nowadays, data size is increasing by GB's every minute. In the document clustering arena, this approach is not realistic, as actual data size can be large and noisy. For example, Wikipedia dataset is a collection of articles with TB's of data. The daily expansion of WWW generates massive amounts of data.

To overcome the above problem we are performing computation in a distributed manner using Hadoop. Cloud Computing like Hadoop is a new computational model deals with enormous data in distributed manner [3]. Hadoop is a framework for performing computation over large dataset, documents clustering includes 1000 of document that need a scalable framework that can process this kind of unstructured data [8].

K-means algorithm has been seen better for text clustering [3], so among many clustering algorithms we have chosen K-means of clustering of large dataset like Wikipedia latest articles. K-means is the most important clustering [1] and using with Cosine measure. K-means perform efficiently over big data, such as Facebook data, Wikipedia articles, Twitter data etc. K-means algorithm with Cosine distance measure and K-means with modified Cosine measure is accomplished for comparing the result in terms of Inter and Intra Cluster distance, Cluster Size, Execution time of an algorithm etc.

The core idea behind our method is to use modified Cosine Distance measure with K-means algorithm to minimize the

Intra-Cluster and Inter-Cluster distances, improving feature weighting using TF-IDF for large document over Mahout with Hadoop. Our algorithm addressing new challenge of dealing with clustering of big data set in a distributed manner with Hadoop.

The key contribution of the proposed work is that to the best of our knowledge, we are the first to propose a customize Cosine Distance measure with K-means algorithm for big datasets using Mahout Framework over Hadoop.

The rest of this paper is organized as follows: Section II describes basic concepts; Section III discusses our proposed modified Cosine Distance measure; Section IV deals with stepwise experimental work and Section V is the experimental result and analysis, that will compare K-means with Cosine and Modified Cosine Distance measure in terms of Inter-Cluster and Intra-Cluster distance, Cluster Size, Sequence file size etc.

II. BASIC DEFINITIONS

A. Document Clustering

Document clustering is a technique that helps in organization of a vast number of documents [4]. Document clustering is a way to organize facts without demanding prior acquaintance about its classification [5]. It is kinds of unsupervised learning, where classes are unidentified unlike classification and similar kind data is grouped into clusters according to their similarity measure. The resulting cluster will signify a meaningful category. The outcome can be used as a basis for document classification. Document categorization can be used for fast information retrieval and data mining. Documents with similar sets of words may be about the same topic. Cluster points are as similar as will represent the documents are much similar in the cluster. The purpose of document clustering is to minimize intra-cluster distance and maximize the inter-cluster distance.

B. Hadoop

Hadoop is an open source software framework which allows distributed processing of big data set in a cluster of machines. It supports data intensive distributed applications on large clusters of commodity hardware. This framework is intended to be scalable, which allow the user to add number of nodes in the cluster as per the application demand. Hadoop uses a programming model called Map Reduce. Map-Reduce projected by Google, is a programming paradigm and an associated enactment in a distributed manner for big data processing. The first step of this programming model is that Map function in input data is applied in parallel to, performing the grouping actions and in the second step, a Reduce function is applied in parallel to each group produced in the first step, to perform the last accumulation of data. Hadoop uses a file system called Hadoop Distributed File System (HDFS), which generates replicas of data slabs and distributes them on computer nodes throughout a cluster to assist reliability. Hadoop single node contains different components for computation like name-node, a data-node, and job-tracker and task- tracker [9].

C. Mahout

Mahout is an Apache open source Software project with the aim of developing a suite of machine learning libraries scalable to large numbers of data objects [8]. Mahout having a pool of libraries for clustering, classification, and other machine learning algorithms intended to execute in Hadoop framework. It contains algorithms for classification, clustering, collaborative filtering, etc. It uses Map Reduce paradigm using Hadoop. It is used as an appropriate solution to solve machine learning problems. Mahout development still in progress which can be used mainly in, to make a recommender, document classifiers, data clustering etc. It has explored many clustering algorithm like Kmeans, Fuzzy Kmeans, canopy clustering, Dirichlet clustering, LDA clustering etc. [5].

D. Distance Measure

Distance measure is the similarity standards for the clustering algorithm. Distance measure has a vital role in clustering that is considered as similarity measure among data in a real space. Distance measure includes for any clustering algorithms is an Inter-cluster and Intra-cluster distance.

Inter Cluster Distance

Inter-Cluster Distance can be defined as the distance between all pairs of centroids of different clusters.

Intra Cluster Distance

Intra-Cluster distance is the distance between members of a cluster. And it will be minimized as compared to Intercluster distance.

E. K-Means Algorithm

The well-known K-means algorithm steps are as follows:

1. Arbitrarily select ‘c’ cluster.
2. Set the convergence threshold value by which centroids don’t move more than this distance, no further iterations are done and clustering process get stopped.
3. Calculate the distance using Cosine and modified Cosine measure between each data point and cluster center.
4. Assign data points to its nearest cluster center.
5. Recalculate the new cluster centroids till convergence threshold using

$$v_i = \left(1/C_i\right) \cdot \sum_{j=1}^{c_i} (b_j) \quad (1)$$

   Where C_i is the number of data points in i^{th} cluster, v_i is the new cluster center for i^{th} cluster and b_j are set of data points.
6. Recalculate the distance between each data point to new cluster center and assign them into a particular cluster.
7. If no data point was reassigned or process goes till the convergence threshold value otherwise repeat step 3.

III. MODIFIED COSINE DISTANCE MEASURE

Cosine similarity deals with similarity among two vector product that calculates the cosine angle between them. The cosine of 0° is 1, cosine of 90° is 0, so similarity values lie from -1 to 1. Two vectors with the same direction have a Cosine similarity of 1, two vectors at 90° will have similarity of 0 and two vectors diametrically contrary to each other have a similarity

of -1. Cosine measure deals with orientation not with magnitude [7]. Cosine similarity is mainly used in a real positive space, where the outcome is nearly confined by [-1, 1]. Cosine similarity will generate a metric that says how related are two documents by looking at the angle instead of magnitude .Given two vectors, A and B, the cosine similarity, cosθ function, is represented using a dot product and magnitude as.

$$\cos\theta = \frac{a.b}{\|a\|.\|b\|} = \frac{\sum_{i=1}^{n} a_i * b_i}{\sqrt{\sum_{i=1}^{n}(a_i)^2} * \sqrt{\sum_{i=1}^{n}(b_i)^2}} \qquad (2)$$

The above cosine distance will give us the similarity index between [-1, 1]. In the same manner we are calculating the Cosine similarity with above given formula. Here cosine is the distance measure criteria, so cosine range [-1, 1] will be distance D.

Here the modified Cosine distance measure step are as follows:

1. Calculate the cosine similarity form above given formula.
2. Get the distance D between vector a_1 and b_1
3. $$D = 1 - \frac{\sum_{i=1}^{n} a_i * b_i}{\sqrt{\sum_{i=1}^{n}(a_i)^2} * \sqrt{\sum_{i=1}^{n}(b_i)^2}} \qquad (3)$$
4. If (D < 0.5)
   Then
   Return $((1-D) * (D * D) + D * \sqrt{D})$
   Otherwise
   Return $\sqrt{D}$
5. Exit

After calculating the distance using standard Cosine distance measure, try to bring distance more closely between two points by squaring the Cosine distance value if the value of D between 0 to 0.5, otherwise increases the distance. Here Intracluster distance is reducing because we are squaring distance so closer points will be in one cluster. Thus Intra-cluster distance will reduce and Inter-cluster distance will increase, that result in good cluster quality.

IV. Experimental work

A. Dataset Used

We have used the following datasets for our experiments:

- Wikipedia Dump article Dataset of size 142 MB and 1.64 GB.

B. Sequence File Conversion Phase

Hadoop is a framework that admits the data in the form of (Key, Value) pair that stores in Sequence file. So firstly we need to convert the data into sequence file format. Mahout with Hadoop also admits the data in the sequence file directory.

C. Seq2Sparse File Generation Phase

Sequence file to Seq2Sparse converts sequence file directory data into vector format, which can be represented in the Vector Space Model (VSM). Here we are setting maxDfPercent to 99 for removing junk words and stop words. That will reduce sequence file size to some extent. Seq2Sparse will accept sequence file as input and generates a vector using a weighting factor like TF or TF-IDF. TF-IDF (Term Frequency-Inverse Document Frequency) scheme which is an improvement over the TF vectorization process because of sparse data, a term can occur multiple times in the dataset which will increase the vector size. For more frequently occurred terms, their IDF value will be small. For calculating IDF we have to first calculate DF (document frequency). DF is the number of documents the word occurred in.

$$IDF_i = (1 \,/\, DF_i) \qquad (4)$$

The TF-IDF weight W_i for a word

$$W_i = TF_i * \log(N \,/\, DF_i) \qquad (5)$$

D. Running K-means Algorithm with the Cosine Measure

Running k-means algorithm by providing a seq2sparse vector file with TF-IDF (Term Frequency-Inverse document frequency) scheme. Run K-means algorithm with Cosine distance measure with setting different k values and over different number of iteration. And analysis the result. Generally k values must be set based on our dataset [2].

V. Experimental result and analysis

An experiment performed over a Hadoop cluster of 4 nodes having high configuration machine of Intel CORE i7 processor, 8 GB of RAM, 1 TB HDD on Linux Ubuntu-12.04 system with the latest release 0.9 of Mahout Framework [8].

Time Taken for stop word filtering, pruning, stemming and converting the Wikipedia dump xml file to the Hadoop format sequence file is presented in Table I**.**

TABLE I. Sequence file conversion phase

Wikipedia Dataset Size (MB)	Conversion Time to Sequence File (ms)	Size of Sequence File after conversion (MB)
142	52817	135
1987	291635	1495

Stemming and filtering of data reduces the file size to some extent. After conversion of Data file to Sequence file, the next step is to convert the Sequence file to Seq2sparse vector file. This will generate a dictionary file, frequency file, TF-vectors, TF-IDF vector, tokenized document and word count. The result of K-means algorithm in terms of time taken to perform clustering and cluster size with Cosine distance measure and modified cosine distance measure over sequence file size of 135 MB in Table II and Table III.

Time taken by K-means using Cosine measure over K-means using modified Cosine measure is more because after getting similarity between two points, we are comparing the distance that will take some more time to perform clustering.

The cluster size generated after performing clustering in the case of K-means with modified Cosine measure is

drastically lesser than K-means with the Cosine measure as shown in Table-III.

TABLE II. EXECUTION TIME COMPARISONS OF K-MEANS USING COSINE AND MODIFIED DISTANCE MEASURE

Number of Iteration	Number Of Clusters	K-means with Cosine measure Execution Time (ms)	K-means with Modified Cosine Distance Measure Execution Time (ms)
5	5	66871	70881
10	10	73539	84583
15	15	102781	131510
20	20	139151	144056

TABLE III. CLUSTER SIZE COMPANIONS USING K-MEANS GENERATED CLUSTER SIZE

Number Of Iteration	Number Of Clusters	K-means with Cosine Measure Cluster size (MB)	K-means with modified Cosine Measure Cluster size (MB)
5	5	7.86	7.48
10	10	12.25	10.79
15	15	12.42	11.6
20	20	14.86	13.41

TABLE IV. INTER CLUSTER AND INTRA CLUSTER DISTANCES

Distance Measure	Using Cosine Distance Measure	Using Modified Cosine Distance Measure
Maximum Intercluster Distance	0.9145672	0.9345475
Minimum Intracluster Distance	0.3202463	0.3043536

Here the above result in Table-IV designate us that K-means algorithm with modified cosine distance measure have changed in Inter and Intra-cluster distance. Inter-cluster distance with modified Cosine measure is increasing while Intra-cluster distance is decreasing as compared to K-means with cosine distance measure.

The below graph in Fig.1 is the graphical representation of the above result in terms of K-means algorithm execution time, Cluster size, Inter and Intra cluster distances with Cosine and Modified Cosine Distance Measure is shown in Fig.1. Inter and Intra Cluster distance are scaled to (10^5) for representation of the graph. The x-axis represents data range.

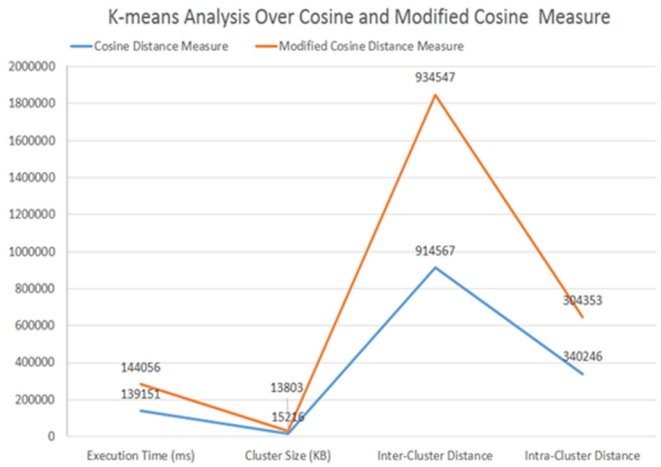

Fig. 1. K-means Analysis

The Cluster Dump utility of Mahout Framework is used to dump the data from Hadoop Cluster by providing a sequence file with terms dictionary file. Here we are dumping top 10 words from Hadoop Cluster in Table-V.

TABLE V. TOP TERMS COMPARISON OF GENERATED CLUSTERS

Top Terms Extracted from K-means Cluster over Modified Cosine Distance Measure		Top Terms Extracted from K-means Cluster over Cosine Distance Measure	
Software	=> 5.97949428	Redirect	=> 2.16725134
Computer	=> 5.93466374	R	=> 0.67895879
Hardware	=> 4.69221210	Camel case	=> 0.64630522
Memory	=> 4.47613143	From	=> 0.42000661
CPU	=> 4.45458618	Capitalization	=> 0.39866744
Ref	=> 4.41775604	Other	=> 0.12245041
Bit	=> 4.27088135	Sovereign	=> 0.09689256
System	=> 4.14908052	Hemingway	=> 0.09649930
Interface	=> 4.00604956	Plural	=> 0.09411977
Code	=> 3.92028052	Misspelling	=> 0.08279604

The above dumped result in Table-V, from cluster generated using K-means using Cosine measure is not clearly specifying any kind of relation between them. Also, this result is contains most frequent words like “from”, “other” etc. that is insignificant. But cluster generated using K-means using modified Cosine measure having significant result than K-means with cosine distance measure, it postulates the result specifically about the topic called “Computer” machine.

VI. CONCLUSION

In literature we have found that distance measure has a vital role in clustering. In this paper we have proposed K-means with customize Cosine distance measure and we found that K-means algorithm with our proposed Cosine distance measure produce better result than k-means over traditional Cosine distance measure. Cluster quality can be measure with Inter and Intra cluster distance. Result shown is improvement in Inter-cluster and Intra-cluster distances.

Result shown is improvement in Inter-cluster and Intra-cluster distances. We have seen better result in terms of cluster size, Inter and Intra cluster distances and top words of cluster. So here we can conclude that modified Cosine measure having good cluster quality for large dataset over Mahout and Hadoop. Our algorithm drawback is that time taken to perform clustering using K-means with modified Cosine distance measure is more than K-means with Cosine measure. This drawback can be overcome by having more number of Hadoop nodes.

REFERENCES

[1] Esteves, R.M.; Chunming Rong, "Using Mahout for Clustering Wikipedia's Latest Articles: A Comparison between K-means and Fuzzy C-means in the Cloud," Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on, vol., no., pp.565, 569, Nov. 29 2011-Dec. 1 2011.

[2] Shameem, M-U-S., and R. Ferdous. "An efficient k-means algorithm integrated with Jaccard distance measure for document clustering." In *Internet, 2009. AH-ICI 2009. First Asian Himalayas International Conference on*, pp. 1-6. IEEE, 2009.

[3] Esteves, Rui Maximo, Rui Pais, and Chunming Rong. "K-means clustering in the cloud--a Mahout test." In *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*, pp. 514-519. IEEE, 2011.

[4] Yang, Hongwei. "A document clustering algorithm for web search engine retrieval system." In *e-Education, e-Business, e-Management, and e-Learning, 2010. IC4E'10. International Conference on*, pp. 383-386. IEEE, 2010.

[5] Berkhin, Pavel. "A survey of clustering data mining techniques." In *Grouping multidimensional data*, pp. 25-71. Springer Berlin Heidelberg, 2006.

[6] Kanungo, Tapas, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. "An efficient k-means clustering algorithm: Analysis and implementation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24, no. 7 (2002): 881-892.

[7] Strehl, Alexander, Joydeep Ghosh, and Raymond Mooney. "Impact of similarity measures on web-page clustering." In *Workshop on Artificial Intelligence for Web Search (AAAI 2000)*, pp. 58-64. 2000.

[8] Anil, Robin, Ted Dunning, and Ellen Friedman. *Mahout in action*. "Manning", 2011, pp. 115-210.

[9] White, Tom. *Hadoop: The definitive guide*. "O'Reilly Media, Inc.", 2012.