A DISTRIBUTED K-NEAREST-NEIGHBOR ALGORITHM FOR TEXT CATEGORIZATION

Suman Sahu[1], Dr.Abha Choubey[2]
[1]M.Tech Student, [2]Associate professor
[1,2]Department of Computer Science and Engineering 1,2Shri Shankaracharya Group of Institution

Abstract*—Text categorization is the application of text mining. Content classification is a supervised learning technique, it plays important role for indexing of document like different applications. Content order has abundant applications, in several fields and for different sorts of information. Numerous issues identified with information stockpiling, administration and recovery can be defined as far as content order. Clustering plays vital part in text mining. K-means clustering is widely used text categorization technique, still more work can be carried out to improve the performance of k-means text classification technique. In this paper we have proposed parallelization of the renowned k-means clustering algorithm. The parallel implementation of k-means uses data parallelism. In this paper we have compared the performance of parallel k-means text classification with sequential k-means with respect to time factor i.e. total time required for content classification and eventually we have calculated the F-measure value by calculating precision and recall which decides what percentage of messages were classified correctly.

***Keywords*— KNN;Recall; F-measure; Precision**

I. INTRODUCTION

Report ordering, spam sifting, populating the various leveled inventories of web assets, archive sort distinguishing proof, computerized exposition reviewing, and classifying newspaper advertisements are a portion of the critical utilization of Text Categorization in the field of science and innovation. It is additionally utilized as a part of the fields of finance, games and entertainment and medicinal sciences. There are several application of text classification as:

- Item classification
- Web Search personalization
- Report sifting for advanced libraries
- Author discovery
- Product surveys characterization
- Investigating general supposition or assumption mining
- Spam email filtering

Naive Bayes and K-Nearest Neighbor calculations, which are supervised learning strategies. Regulated Learning is a procedure in which results are derived from a training set. Training set is one which contains sets of information and classification names to which they have a place with. Trained data is at first ordered by specialists. Once the classification engine is trained, it must have the capacity to classify the test information to its suitable classification. Classification is one of the regulated machines learning procedure. Machine learning is a self-decision framework which is equipped for obtaining and coordinating information continually. This capacity to gain from past encounters, logical perception, and different means, brings about a framework that can perpetually self-enhance to offer expanded proficiency and adequacy.

Text classification method performance can be measured i.e. what % of messages were classified correctly? For example two system giving accuracy as follows:

	Overall accuracy	Accuracy on spam	Accuracy on gen
System 1	95%	99.99%	90%
System 2	95%	90%	99.99%

For above we need to go for Precision and recall are the most collective procedures for estimating an information retrieval (IR) system. Precision is the percentage of returned documents that are targets, whereas recall is the percentage of target documents reverted.

Precision = Good messages kept / All messages kept ……. (1)
Recall =Good messages kept / All good messages…… (2)
F-measure = 1 / (average (1/precision, 1/recall))…… (3)

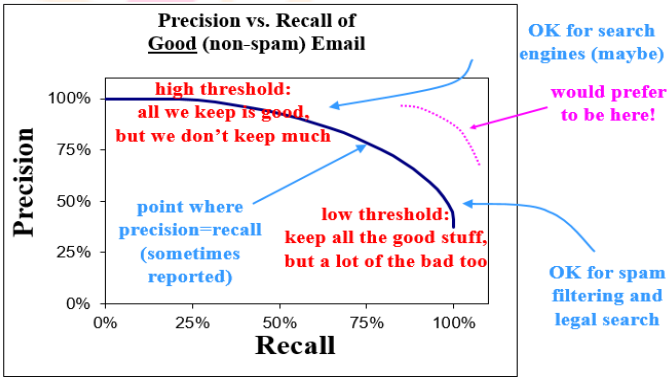

Fig.-1 **Parameters to Measure Performance of Email Text Classification**

Further in next section of this paper we will briefly discuss about several literature and we will provide tabular comparison among some literatures, in section

II. LITERATURE SURVEY

Arman KhadjehNassirtoussiet. al. [Elsevier 2014] concluded that the major systems for market prediction based on online text mining have been reviewed and some of the predominant gaps that exist within them have been identified. The review was conducted on three major aspects, namely: pre-processing, machine learning and the evaluation mechanism; with each breaking down into multiple sub-discussions. It is believed to be the first effort to provide a comprehensive review from a holistic and interdisciplinary point of view. This work intended to accomplish: Firstly, facilitation of integration of research activities from different fields on the topic of market prediction based on online text mining; Secondly, provision of a study-framework to isolate the problem or different aspects of it in order to clarify the path for further improvement; Thirdly, submission of directional and theoretical suggestions for future research.

Li Baoliet. al. [ICCPO 2009] have proposed a modified kNN method. For different classes, according to their distribution in the training set, we use a suitable number of nearest neighbors to predict the class of a test document. Preliminary experiments on Chinese text categorization show that our method is less sensitive to parameter k than the traditional one, and it can properly classify documents belonging to smaller classes with a large k. The method

is promising for some special cases, where estimating the parameter k via cross-validation is not allowed. We plan to experiment our improved method on more different data sets in the future.

EnmeiTuet. al. [Elsevier 2016] proposed a new k nearest-neighbor algorithm, mkNN, to classify nonlinear manifold distributed data as well as traditional Gaussian distributed data, given a very small amount of labeled samples. We also presented an algorithm to attack the problem of high computational cost for classifying online data with mkNN and other transductive algorithms. The superiority of the mkNN has been demonstrated by substantial experiments on both synthetic data sets and real-world data sets. Given the widespread appearance of manifold structures in real-world problems and the popularity of the traditional kNN algorithm, the proposed manifold version kNN shows promising potential for classifying manifold-distributed data.

Johan Bollenet. al. [Elsevier 2010] investigated whether public mood as measured from large-scale collection of tweets posted on twitter.com is correlated or even predictive of DJIA values. Our results show that changes in the public mood state can indeed be tracked from the content of large-scale Twitter feeds by means of rather simple text processing techniques and that such changes respond to a variety of socio-cultural drivers in a highly differentiated manner.

HunnyPahujaet. al. [IJERA 2012] have addressed the problem of optimizing the acoustic feature set by ACO technique for text-independent speaker verification system based on GMM-UBM. In our previous work we have proposed an ACO algorithm for feature selection in GMM-based ASV systems (Nemati et al., 2008). In this paper we propose some modifications to the algorithm and apply it to larger feature vectors containing MFCCs and their delta coefficients, two energies, LPCCs and their delta coefficients. ACO selected the most relevant features among all features in order to increase the performance of our ASV system.

S. No.	Author/Title/Publication	Method Used	F-measure	Discription
1.	Prathima Madadi et.al./Text Categorization Based on Apriori Algorithm's Frequent Itemsets/UNLV Teses 2009 [Ref.-1]	Apriori	92%	Paper was to come to a decision on the model built using frequent itemsets i.e. Can frequent itemsets be efficiently used to perform text categorization
2.	István Pilászy/Text Categorization and Support Vector Machines/Budapest University 2010[Ref.-2]	SVM	-	This paper gives some introduction into text categorization, and describes the common tasks of a TC system.
3.	Thorsten Joachims/Text Categorization with Support Vector Machines: Learning with Many Relevant Features/Springer 2005 [Ref.-3]	SVM (Support Vector Machine)	86.4%	This paper introduces support vector machines for text categorization. It provides both theoretical and empirical evidence that SVMs are very well suited for text categorization. SVMs acknowledge the particular properties of text:(a) high dimensional feature spaces (b) few irrelevant features (dense concept vector)(c) sparse instance vectors.
4.	Hunny Pahuja et. al./Ant colony optimization-based selected features for Textindependent speaker verification/IJERA 2012A [Ref.-4]	Ant colony optimization-based selected features	85%	In this paper, we have addressed the problem of optimizing the acoustic feature set by ACO technique for text-independent speaker verification system based on GMM-UBM and propose some modifications to the algorithm and apply it to larger feature vectors containing MFCCs and their delta coefficients, two energies, LPCCs and their delta coefficients. ACO selected the most relevant features among all features in order to increase the performance of our ASV system.
5.	Li Baoli et. al./An Improved k-Nearest Neighbor Algorithm for Text Categorization/ ICCPOL 2003[Ref.-5]	Improved K-Means	90.05%	Author propose an improved kNN algorithm, which uses different numbers of nearest neighbors for different categories, rather than a fixed number across all categories. More samples (nearest neighbors) will be used for deciding whether a test document should be classified to a category, which has more samples in the training set.

Table 1. Comparison of Literature

III. PROBLEM IDENTIFICATION

After been through several literature we have provided theoretical comparison and mathematical comparison based on F-measure.

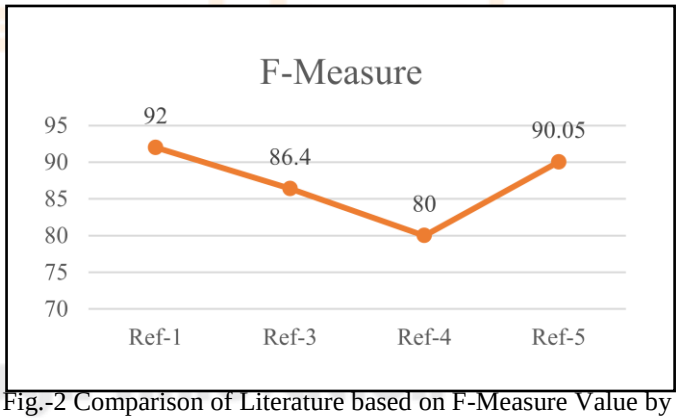

Fig.-2 Comparison of Literature based on F-Measure Value by Referring **Table-1**

From Fig.2 we can say that still need to increase the accuracy of text categorization techniques. In this paper we do emphasis on K-means clustering based content classification technique. K-Means is a commonly used clustering algorithm used for data mining. Clustering is a means of arranging n data points into k clusters where each cluster has maximal similarity as defined by an objective function. Each point may only belong to one cluster, and the union of all clusters contains all n points. Here are some bottleneck of K-means clustering technique:

- K-means has problems when clusters are of differing
  - Sizes
  - Densities

- Non-globular shapes
- Problems with outliers
- Empty clusters
- However, it suffers from major shortcomings that have been a cause for it not being implemented on large datasets. The most important among these are K-means is slow and scales poorly with respect to the time it takes for large number of points.
- The sequential version of k-means algorithm takes lots computational time on computing distances between each one of N data objects and the present K centroids. Then iteratively allot each data objects to the closest cluster.

IV. SOLUTION METHDOLOGY

The sequential enactment of k-means algorithm proceeds a much more complex in computation time for calculating distances between every one of N data objects further in different iteration time complexity increases. In our proposed algorithm we have implemented parallel k-means using JAVA executor service. Proposed algorithm uses data parallelism, we have observe that during iteration updating of centroid are independent henceforth by creating task, task can be assigned to available processors, after completion of task by threadpool java futurelist we return the updated centroid which involves message passing. Distance function we have used cosine distance for calculation distance among vector.

Cosine similarity is a measure [1] of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0° is 1, and it is less than 1 for any other angle. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors at 90° have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in (0,1).

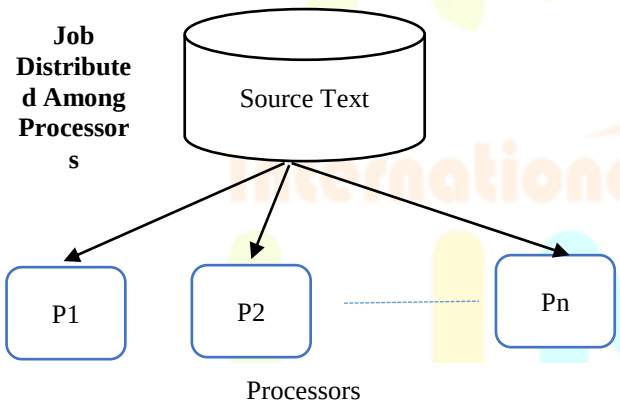

Fig.3- Distribution of task among available processors

Sequential k-means clustering algorithm

Step-1. Select objects randomly. These objects represent initial group centroids k.
Step-2. Assign each object to the group that has the closest centroid.
Step-3. When all objects have been assigned, recalculate the positions of the centroids k.
Step-4. Repeat Steps 2 and 3 until the centroids no longer move.

Proposed Algorithm

Proposed ()

Step-1: Read Source Text //Implementation of Encoder which uses Term Frequency - Inverse Document Frequency (TF-IDF) encoding.
Step-2: Calculate TF-IDF (document)
Step-3: Calculate cosine distance between Vectors.

/** A Clusterer implementation based on Parallel (Executor Service) k-means clustering. param distance the distance metric to use for clustering paramclusteringThreshold the threshold used to determine the number of clusters k paramclusteringIterations the number of iterations to use in k-means clustering*/
Step-4:KMeansClusterer(distance ,clusteringThreshold,Iteration).

Calculate TF-IDF(Document)

Step-1: Construct a term frequency - inverse document frequency encoder. The encoder encodes documents into Vectors with the specified number of features.
Step-2: Encode all documents within the provided DocumentList.
Step-3: Calculate word histogram for the provided document and store in the histogram field. To ensure a constant size histogram Vector the words are first hashed to an integer between 0 and numFeatures - 1. Calculate word histograms for all documents in a DocumentList.
Step-4: Calculate inverse document frequency for the provided DocumentList. The inverse document frequency for a word i is defined as log(N/Ni) where N is the total number documents and Ni is the number of documents where word i occurs. This method requires that the document histogram for each document has already been calculated.
Step-5: Encode all documents within the provided Document List.

KMeansClusterer(distance,clusteringThreshold,Iteration)

// Get Available Number of Processors
Step-1: nrOfProcessors = Runtime.getRuntime().availableProcessors()
Step-2: Create Threadpool as per nrOfProcessors
Step-3: Mark all documents as not being allocated to a cluster
Step-4: Add a cluster to the ClusterList
Step-5: Create Task for Executor Service
Step-6: Executor Service.submit(Task)
Step-7: Stop Executor Service

Task()

Step-1: Allocate any unallocated documents in the provided DocumentList to the nearest cluster in the provided ClusterList. Find cluster whose centroid is closest to a document.
Step-2: Update centroids of all clusters within ClusterList.
Step-3: Clear out documents from within each cluster. Used to cleanup after each clustering iteration.
Step-4: Calculate ratio of average intracluster distance to average intercluster distance. Used to optimize number of clusters k.
returnfuturelist

V. RESULT AND DISCUSSION

In this section, we will evaluate the effectiveness of our proposed algorithm. Here we have presented an experimental valuation using data.As a source dataset for experimental evaluation we have used Google News dataset. Furthermore we have compared the performance of Sequential version of k-means algorithm with our proposed algorithm with rest to two parameters as follows

I. Total Execution Time
II. F-Measure

F-measure is a measure of classification accuracy. We have executed based on three articles as follows.

Snippet of dataset as follows:

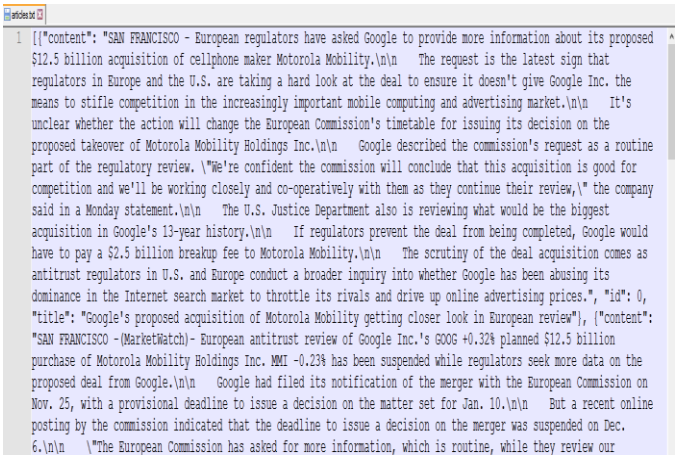

Fig.4 Google News Dataset

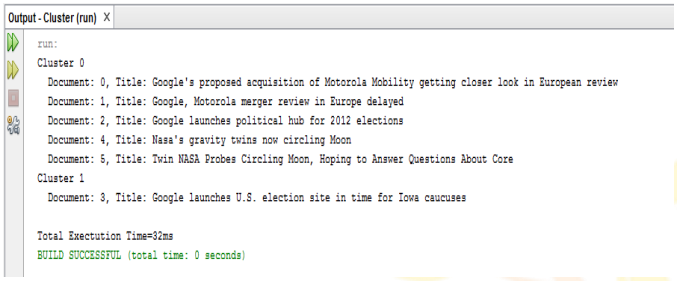

Fig.5 Snippet of outcome of proposed algorithm

	Execution Time		
	article1.txt	**article.txt**	**article2.txt**
Sequential k-means	1375ms	1328ms	1443ms
Proposed	63ms	31ms	80ms

Table 2. Performance comparison

Proposed Algorithm			
	Document in Cluster 0 (Relevant)	Document in Cluster 1 (Relevant)	Total Documents
article1.txt	2	4	6
article.txt	2	5	7
article2.txt	4	6	10
Sequential k-means			
article1.txt	3	3	6
article.txt	4	3	7
article2.txt	5	5	10

Table 3. Document Classification

Depend on the values of Table-3 we can calculate the value of F-Measure.
F-measure= 2[(precision. recall/ (precision + recall)]
Precision= True Positive/ (True Positive + False Positive)
Recall= True Positive/ (True Positive + False Negative)

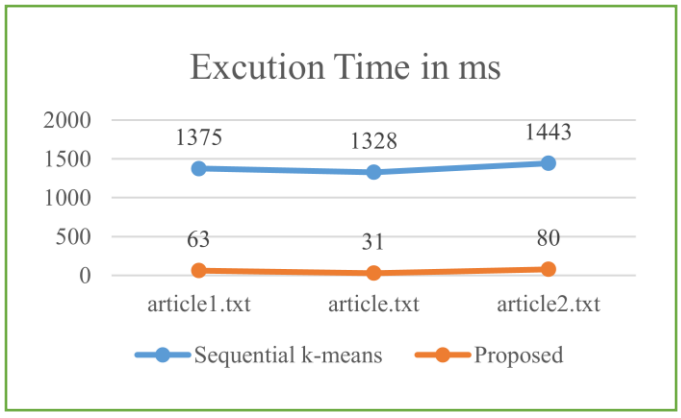

Fig 6. Graph based on Table-2

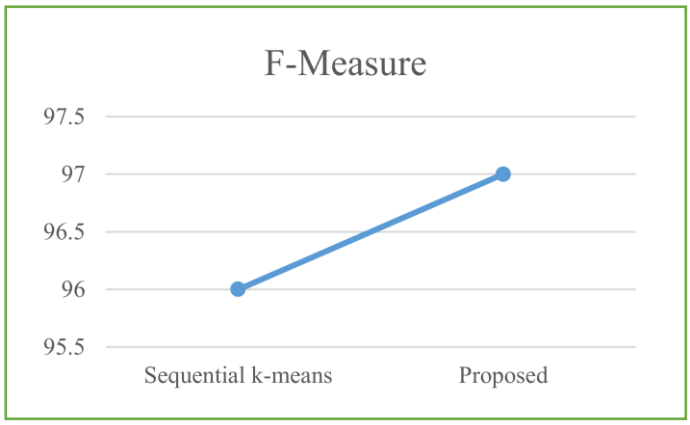

Fig. 7Fig. Graph based on Table-3

VI. CONCLUSION

These days through the sudden development of the Internet and on-line accessible archives, the undertaking of sorting out content information gets to be distinctly one of the primary issues. A noteworthy approach is content classification, the undertaking which tries to consequently allocate archives into their individual classifications.

After implementation of sequential k-means clustering and proposed text classification algorithm based on data we can conclude that with respect to execution time proposed algorithm performs well, accuracy between algorithms are almost same. If we increase the number of available processors proposed algorithm will perform well. In future we can put some more efforts towards increase the accuracy and we can apply parallelism to other well performing clustering algorithm so that we can increase the value of F-Score.

REFERENCES

[1] Alexandra Balahur, Ralf Steinberger, Erik van der Goot, Bruno Pouliquen, MijailKabadjov, pinion Mining on Newspaper Quotations IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies 2009.

[2] Eui-Hong (Sam) Han George Karypis, Vipin Kumar Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification Department of Computer Science and Engineering Army HPC Research Center University of Minnesota 2015.

[3] Johan Bollen1,Huina Mao1,Xiao-Jun Zeng Twitter mood predicts the stock market Elsevier 2010.

[4] EnmeiTua, YaqianZhangb, Lin Zhuc, JieYangd, Nikola KasaboveA Graph-Based Semi-Supervised k Nearest-Neighbor Method for Nonlinear Manifold Distributed Data Classification Elsevier 2016.

[5] Li Baoli, Yu Shiwen, and Lu Qin An Improved k-Nearest Neighbor Algorithm for Text Categorization International Conference on Computer Processing of Oriental Languages, Shenyang, China, 2003.

[6] Aditya ChainuluKaramcheti A Comparative study on text categorization UNLV Teses, Dissertations, Professional Papers, and Capstones 2010.

[7] HunnyPahuja, JitenderChhabra, Ajay Khokhar Ant colony optimization-based selected features for Textindependent speaker verification International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 Vol. 2, Issue 3, May-Jun 2012, pp.1466-1476.

[8] Arman KhadjehNassirtoussi , Saeed Aghabozorgi , Teh Ying Wah, David Chek Ling Ngo Text mining for market prediction: A systematic review Elsevier 2014.