Abstract—Big data processing is currently becoming increasingly important in modern era due to the continuous growth of data generated by various fields such as particle physics, human genomics, earth observations etc. However the efficiency of processing large-scale data on modern infrastructure is not clear. Machine Learning (ML) concepts play a major role in data analysis. The traditional practice has restriction that only subset of large set of data has to be taken for analysis as size of data grows in exponentially. Hadoop is one such framework that offers distributed storage and parallel data processing.

In this project we propose to build a combined clustering and classification model that run on Hadoop to process Big data. We try to optimize the performance of Big data analysis by integrating clustering and classification concepts with map reduce paradigm of Hadoop architecture.

Keywords—Machine Learning, Big data, Hadoop, Mapreduce, Data mining, Parallelism.

I. INTRODUCTION

Widespread success of machine learning is in addressing an even broader range of tasks for actionable business insights and optimizations. But amount of data being produced today is so much that, how to compute, transfer and store these huge data is a prominent challenge which will bring great impact on traditional architectures and methods of computation, networking and storage. Hadoop an open source framework developed by Dough cutting[2], is built on top of the Hadoop distributed file system(HDFS) and Mapreduce paradigm[1] provides a parallelization framework which has generated considerable claim for its ease-of-use, scalability and fault tolerance.

The value of Big data to an organization falls into two categories: analytical use, and enabling new products. Big data analytics can reveal insights hidden previously by data too costly to process, such as peer influence among customers, revealed by analyzing shoppers’ transactions, social and geographical data. Being able to process every item of data in reasonable time removes the troublesome need for sampling and promotes an investigative approach to data, in contrast to the somewhat static nature of running predetermined reports. The emergence of Big data into the enterprise brings with it a necessary counterpart: agility.

Successfully exploiting the value in Big data requires experimentation and exploration. Whether creating new products or looking for ways to gain competitive advantage. Thus there is need for a proper method or an algorithm to process Big data. Machine Learning can be used to exploit Big data, so that the information retrieved is useful and meaningful.

The input data to the proposed system is the simulated real time tweet messages of twitter user. The input data is edited since the input data has no age and gender specification as we need it for classification. The system is generalized one which works for all type of application that include text document as input

II. PROBLEM STATEMENT

To propose a combined model of clustering and classification on Hadoop environment that optimizes the analysis of Big data for text mining using the parallel programming Mapreduce architecture.

Input:
Set of data.

Output:
- Classified output.
- Categorized data.
- Graph in terms of histograms.

Objectives:
- To develop a combined model using clustering and classification concepts.
- To integrate the developed model within parallel programming architecture of Mapreduce.
- Performance optimization of Big data analysis with the proposed approach.

III. LITERATURE SURVEY

During the literature survey, the existing methodologies are studied and the needs of development for efficient platform have been identified. We have also studied relevant tools, techniques and their effective usage for our objectives.
Literature survey is the documentation of a comprehensive review of the published and unpublished work; this forms the secondary source of data in the area of specific interest to the design. The sources of literature survey could be books, journals, newspapers, government publications. The main goal of a literature survey is to gather a basis for the work and to show that the designer is familiar with the existing literature and research of the topic.

Machine Learning [3] has enormous application in the field of data mining. As it is extensively used for large sets of data, it is necessary to parallelize the algorithms to run them with less time. There are many parallel programming languages such as orca, occam, ABCL, SNOW, MPI and PALOG, but none of these approaches make it obvious how to parallelize a particular algorithm.

Jin and Agrawal [5] give a general machine learning programming approaches but only for shared memory machines. This does not fit the architecture of cellular or grid type multiprocessors where cores have local cache, even if it can be dynamically reallocated.

There was also a general and exact technique for parallel programming of large class of machine learning for multi core processors based on a Mapreduce paradigm. Again that was a way to achieve speedup in multi-core system and this cannot be used on single core system. Using Hadoop, Machine learning can be parallelized on single core systems achieving a linear speedup in execution and performance.

Machine learning playing an important application in data mining. People are interested to work with big set of data. As we seen that Big data consist of unstructured format. The existing storage system has problem that it can’t give reliable storage for this unstructured data. Example, the Relational database cannot provide reliable storage. So in our proposed system we are using tool “Hadoop” to manage and provide reliable storage to Big data[2].

A. HADOOP

Apache Hadoop is powerful open source software that addresses the problem of Big data. For the same purpose to manage large set of data over cluster of machines we are using this tool.

Two major components of Hadoop are:
1. Hadoop Distributed File System (HDFS) for distributed storage,
2. Mapreduce for parallel processing.

Thus, Hadoop offers a reliable and scalable environment for distributed computing, which involves many that houses huge data upon which necessary computing need to be carried out. Hadoop includes a fault tolerance storage system called the Hadoop distributed file system.

HDFS manages storage on the cluster by breaking incoming files into pieces called “blocks” and storing each of blocks r stores three complete copies of each file by copying each piece to three different machines.

In Hadoop Distributed File System (HDFS)
- Data is organized into files and directories.
- Files are divided into uniform sized blocks (default 128MB).
- Replication for performance and fault tolerance.
- HDFS exposes block placement so that computation can be migrated to data.

B. Similarity Measures

Before clustering, a similarity/distance measure must be determined. The measure reflects the degree of closeness or separation of the target objects and should correspond to the characteristics that are believed to distinguish the clusters embedded in the data. In many cases, these characteristics are dependent on the data or the problem context at hand, and there is no measure that is universally best for all kinds of clustering problems.

Euclidean Distance is a standard metric for geometrical problems. It is the ordinary distance between two points. Different similarity measures gives different values based on application. Since we are analysing for text mining, the input text document similarity is measured using cosine similarity. Cosine similarity is one of the most popular similarity measure applied to text documents, such as in numerous information retrieval applications and clustering too.

Given two documents $v_a$ and $v_b$, their cosine similarity is:

$$SIM_C(v_a, v_b) = \frac{\overrightarrow{v_a} \cdot \overrightarrow{v_b}}{|\overrightarrow{v_a}| \times |\overrightarrow{v_b}|}$$

Where $v_a$ and $v_b$ are m-dimensional vectors over the term set $T = \{t_1, \ldots, t_m\}$. Each dimension represents a term with its weights in the document, which is non-negative. As result, the cosine similarity is non-negative and bounded between [0,1].
IV. PROPOSED SYSTEM

In the proposed system we propose to build a combined clustering and classification model that runs on Hadoop to process Big data. We try to optimize the performance of Big data analysis by integrating clustering and classification with map reduce paradigm of Hadoop architecture. Fig 1 shows data flow diagram of the proposed system. Input is an actor containing text documents and final output presented in graph. The intermediate processes run over Hadoop which are written in java in Mapreduce form to parallelize the computation in cluster of machines. The details of each processes explained in part A and B and C. Using Mapreduce paradigm on Hadoop framework.

Machine learning can be parallelized on single core system achieving a linear speedup in execution and performance.

Let D = \{d1, . . . , dn\} be a set of documents and T = \{t1, . . . ,tm\} the set of distinct terms occurring in D. We discuss more precisely what we mean by “terms” below: for the moment just assume they are words. A document is then represented as a m-dimensional vector $\vec{t}_d$. Let $tf(d, t)$ denote the frequency of term $t \in T$ in document $d \in D$. Then vector representation of document D is:

$$\vec{t}_d = (tf(d, t1), \ldots , tf(d, tm))$$

Figure 2: Angle Between Documents

With documents presented as vectors, we measure the degree of similarity of two documents as the correlation between their corresponding vectors, which can be further quantified as the cosine of the angle between the two vectors. Figure 2 shows the angle in two-dimensional space but in practice the document space usually has tens and thousands of dimensions.

B. Clustering

In proposed system the input text documents are clustered based on measuring the cosine similarity between documents. In order to find the similarity we are providing a reference document for comparing with the input documents, each input text documents are compared with reference document to cluster them based on cosine value. Since objective is for large set of text data, input text documents are pushed into Hadoop distributed file system and stored in terms of key value pair. Here reference document contains desired text or key words based on which we want to cluster the input documents.

Providing reference document as input to this combined model it will use the word count algorithm and find the bag of words in the reference document. These words are considered as key words for comparison test for other input documents. Next to cluster the input documents the strategy is that only the desired key words are made to search among each input documents, the key word is searched line by line and whenever the similar term occurs it will keep adding counts. Now the output will be bag of terms or key words with their counts.

These terms of each individual documents with their counts as weights considered as a m-dimensional vector, and named documents as d1,d2,\ldots, dm  and stored at intermediate storage.
Each documents d1,d2,...,dm containing the desired terms that are searched with their counts. Now each documents d1,d2,...,dm are compared with reference document for similarity measure using cosine similarity measure between two documents.

The clustering is done as matched cluster and non-matched cluster based on the cosine similarity measure values obtained between reference document and documents d1,d2,...dm. A threshold is fixed for clustering such that the cosine measure values that fall under 0.3 (cosine value lies between 0 t0 1) are grouped as non-matched cluster and the above the threshold value are grouped as matching cluster. We can change the threshold value as we desire.

C. Classification

Classification is also a primary object of this paper. To get final categorized information classification is done for the matched cluster that we have obtained. Classification here is done on age group under they fall and the gender. The strategy here is that age values and key words male or female are searched for the matched clustered document in their respective input set. The value that obtained is represented in terms of graphs. We can infer from this graph that the similar behavior people falls under which age group in terms of percentage and how many percentage people are male and female. The final categorized information is in visual format.

V. EXPERIMENTAL ANALYSIS

This combined model is tested and analyzed for synthetic input set containing the tweets from twitter. Here one user’s tweets considered as single text document and we have edited these tweets documents with age value and gender for the use of classification with visual representation with graph. The reference document we have taken is a text document containing discussion about mobile phone technology trend. For experiment we considered five documents as input for this combined model. From the reference document we have selected desired key words that we want to search in input text documents for grouping. Using simple word count algorithm gives all terms in reference document and we will select only the desired key words or all the key words.

Tools and Techniques:

- An open-source framework – Hadoop.
- Mapreduce- a programming model/software framework.
- Clustering and classification algorithms.
- Graph in terms of histograms.

Next the key words are made to search in each input documents. The output of each document contains the key words with their count in that particular document. Now considering these keywords and count value the cosine similarity for these individual documents with reference document is calculated.

Given two documents \( \rightarrow \) and \( \rightarrow \), their cosine similarity is

\[
SIMC(\vec{t_a}, \vec{t_b}) = \frac{\vec{t_a} \cdot \vec{t_b}}{|\vec{t_a}| \times |\vec{t_b}|}
\]

Where \( \rightarrow \) and \( \rightarrow \) are m-dimensional vectors over the term set \( T = \{ t_1, \ldots, t_m \} \). Each dimension represents a term with its weights in the document, which is non-negative. As result, the cosine similarity is non-negative and bounded between \([0,1]\). Here in proposed system \( \rightarrow \) is the reference document and \( \rightarrow \) corresponds to input documents T1, T2, T3, T4, T5 as shown in table 1 below.

The resulting cosine value is between 0 to 1. Towards 1 is most similar. We have fixed a threshold value of 0.3 so that the cosine value that falls above 0.3 is considered as matching cluster.

<table>
<thead>
<tr>
<th>Reference document</th>
<th>Input documents</th>
<th>Cosine value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ref.txt</td>
<td>T1.txt</td>
<td>0.6666712111</td>
</tr>
<tr>
<td>Ref.txt</td>
<td>T2.txt</td>
<td>0.1221113214</td>
</tr>
<tr>
<td>Ref.txt</td>
<td>T3.txt</td>
<td>0.9999999999</td>
</tr>
<tr>
<td>Ref.txt</td>
<td>T4.txt</td>
<td>0.3245612222</td>
</tr>
<tr>
<td>Ref.txt</td>
<td>T5.txt</td>
<td>0.5643323311</td>
</tr>
</tbody>
</table>

From Table 1 we can notice that documents T1.txt, T3.txt, T5.txt are above threshold value 0.3. So they fall under matching cluster. Now the classification model will be applied on the matching cluster, it will search for age group that we provide to search for and represent in the graph as histogram in percentage.
VI. CONCLUSION

In this project we propose a combined clustering and classification model in parallel fashion using a distributed framework like Hadoop. The resulting model optimizes the analysis of Big data by speeding up the processing of Big data. The proposed system studies the performance and efficiency of running Mapreduce based parallel machine learning applications on Hadoop cluster platform.

We applied clustering before applying classification on input data in order to avoid clumsy result of classification. We observed that the model successfully works for Big data by speeding up the computation for large data sets. The proposed system is generalized one. It works for all kind of real time input that should be in rows so that it will count the words and recognize words. The reference document will be of your desired key words related to your application. Proposed system may fit in the applications like Terrorist related conversation on social network, in cyber-crime detection and Discussion forum analysis.

REFERENCES


