ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 5, Special Issue 11, May 2016

International Conference on Innovations in Engineering and Technology (ICIET'16)

6th & 7th April 2016

Organized by

K.L.N. College of Engineering and Technology, Madurai, Tamil Nadu, India

Hadoop Based Map Reducing In Social Networks

R.Anandha Murugan, G.R.Daksha, R.Divya

Dept. of Computer Science Engineering, K.L.N. College of Engineering, Madurai, Tamil Nadu, India

ABSTRACT: Social Media Networks (SMNs) are playing increasingly prominent role in people's daily life. Dealing with big data in Computational Social Networks may require powerful machines, big storage and high bandwidth which may seem beyond the capacity. Due to tremendous amount of information to identify the communities in big data scenarios has become a challenge. On previous work map equation many mining mechanism are performed. Thus to present different indexing mechanisms such as Mongo and Lucene based Indexing. Mongo is good for storing data. Lucene is a low RAM consumption and has a stable system. By comparing those on the Micro Blog Posts (MBPs) collected from popular Online Social Networks sites (OSN) in India. Upload the indexing mechanism onto the Hadoop Distributed File System (HDFS) and analyze the Result.

KEYWORDS: Hadoop, Mongodb, Lucene indexing, HDFS, Map Reducing.

1. INTRODUCTION

To Carryout big data research on OSNs using Commodity computers in small environment. **Map Reduce** is the center concept of Hadoop. It is the programming paradigm that allows massive scalability across hundreds and thousands of servers. Hadoop is the most popular tool for handling Big Data which is open source. It implements the concepts of big data.**HDFS** used for distributed storage and distributed processing of big data. Processing these data with Two **Indexing Mechanisms** such as *Lucene and Mongo*. Designing an experiment to text the efficiency and accuracy of these Statistical analyses. Examining how fast each method responding to user queries, as well as how fast the method carryout Statistical Analysis and returning the results. Comparing accuracy of each method.

II. LITERATURE SURVEY

[1] **Dr. Arup Barman et al.(2015)** Proposed a Big Data has the fueling capacity for the transformation of world to the digital world. The word "Big Data "is itself misleading one. It gives us an illusion as if data after certain size is called as big data. But the "Big Data" is defined in terms of system capacity only and its value changes depending upon the capacity of a system or organization .The data which is beyond storage capacity and which can't be processed by system is simply referred to as Big data. Big data is definitely disruptive, potentially transformational .TheConsensus is clear: big data brings disruption that can revolutionize business. The author describes the most relevant information regarding BIG DATA and aims to highlight the uses of BIG DATA and its salient features and most importantly its explosive power to revolutionary change in every field of life especially in the field of Human Resource management.

[2] Jonathan et al.(2012) Proposed a ***Hadoop Map Reduce is a large scale***, open source softwareFramework dedicated to scalable, distributed, and data-intensiveComputing. The framework breaks up large data into smaller parallelizable chunks and handles scheduling.

- Maps each piece to an intermediate value.
- Reduces intermediate values to a solution.
- User-specified partition and combiner options.
- Fault tolerant, reliable, and supports thousands of nodes and petabytes of data.
- If you can rewrite algorithms into Maps and Reduces,
- and your problem can be broken up into small pieces

- Solvable in parallel, then Hadoop's ***MapReduce is theWay to go for a distributed problem solving approach to large datasets.***
- Tried and tested in production.
- Many implementation options.

Thus to present the design and evaluation of a data awarecache framework that requires minimum change to theoriginal MapReduce programming model for provisioningincremental processing for Big data applications using theMapReduce model.

[3]Emad A Mohammed et al.(2014) Proposed a the emergence of massive datasets in a clinical setting presents both challenges and opportunities in data storage and analysis. This so called "big data" challenges traditional analytic tools and will increasingly require novel solutions adapted from other fields. Advances in information and communication technology present the most viable solutions to big data analysis in terms of efficiency and scalability. It isvital those big data solutions are multithreaded and that data access approaches be precisely tailored to large volumes of semi-structured/unstructured data. The MapReduce programming framework uses two tasks common in functional programming: Map and Reduce. MapReduce is a new parallel processing framework and Hadoop is its open-source implementation on a single computing node or onClusters. Compared with existing parallel processing paradigms
(e.g. grid computing and graphical processing unit (GPU)), MapReduce and Hadoop have two advantages:

- Fault-tolerant storage resulting in reliable data processing by replicating the computing tasks, and cloning the data chunks on different computing nodes across the computing cluster.
- High-throughput data processing via a batch processing framework and the Hadoop distributed file system (HDFS). Data are stored in the HDFS and made available to the slave nodes for computation.

In this paper, to review the existing applications of the MapReduce programming framework and its implementation platform Hadoop in clinical big data and related medical health informatics fields. The usage of MapReduce and Hadoop on a distributed system represents a significant advance in clinical big data processing and utilization, and opens up new opportunities in the emerging era of big data analytics. The objective of this paper is to summarize the state-of-the-art efforts in clinical big data analytics and highlight what might be needed to enhance the outcomes of clinical big data analytics tools. This paper is concluded by summarizing the potential usage of the MapReduce programming framework and Hadoop
Platform to process huge volumes of clinical data in medical health informatics related fields.

[4] **Songchang Jin et al.(2015)**Proposed a Social media networks are playing increasingly prominent role in people's daily life. Community structure is one of the salient features of social media network and has been applied to practical applications, such as recommendation system and network marketing, with the rapid expansion of social media size and surge of tremendous amount of information, howto identify the communities in big data scenarios has become a challenge. Based on our previous work and the map equation (an equation from information theory for community mining),develop a novel distributed community structure mining framework. In the framework,

- To propose a new link information update method to try to avoid data writing related operations and try to speed up the process.
- To use the local information from the nodes and their neighbors, instead of the pagerank, to calculate the probability distribution of the nodes.
- To exclude the network partitioning process from our previous work and try to run the map equation directly on Map Reduce.

III. PROPOSED WORKS

3.1 Map-Reducing

Map Reduce is a programming model and an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster.Conceptually similar approaches have been very well known since 1995 with the Message Passing Interface standard having reduce and scatter operations.A Map Reduce program is composed of a **Map()** procedure (method) that performs filtering and sorting (such as sorting students by first name into queues, one queue for each name) and a **Reduce()** method that performs a summary operation (such as counting the number of students in each queue, yielding name frequencies). The "Map ReduceSystem" (also called "infrastructure" or "framework") orchestrates the processing by marshalling the distributed servers, running the various tasks in parallel, managing all communications and data transfers between the various parts of the system, and providing for redundancy and tolerance. The model is inspired by the map and reduce functions commonly used in functional programming, although their purpose in the Map Reduce framework is not the same as in their original forms. The key contributions of the Map Reduce framework are not the actual map and reduce functions, but the scalability and fault-tolerance achieved for a variety of applications by optimizing the execution engine once. As such, a single-threaded implementation of Map Reduce will usually not be faster than a traditional (non-Map Reduce) implementation, any gains are usually only seen with multi-threaded implementations. The use of this model is beneficial only when the optimized distributed shuffle operation (which reduces network communication cost) and fault

tolerance features of the Map Reduce framework come into play. Optimizing the communication cost is essential to a good Map Reduce algorithm.
Map Reduce libraries have been written in many programming languages, with different levels of optimization. A popular open-source implementation that has support for distributed shuffles is part of Apache Hadoop. The name Map Reduce originally referred to the proprietary Google technology, but has since been genericized. By 2014, Google was no longer using Map Reduce as their primary Big Data processing model, and development on Apache Mahout had moved on to more capable and less disk-oriented mechanisms that incorporated full map and reduce capabilities.

3.2 Mongo DB

To handle unstructured MBPs in large quantity, it needs a high-performance, steady, and flexible database system. Because MBPs are unstructured, chose Mongo DB for this purpose, which is a common choice for storing unstructured data. Different MBPs from different sources use different data structures. With Mongo DB, can store data in different data structures in the same collection. This makes it convenient to manage the data. Moreover, Mongo DB is a database scheme with high performance on the operations of both read and write, which meets our need of intensive writing and querying MBPs.Mongo DB provides a built-in regular expression searching method. Give the regular expression the text to match, Mongo DB returns all the records that match the regular expression. Then, the system will go overall the MBPs and count MBPs for each statistical feature such as gender, location and sentiment. This search method, however, is inefficient and does not meet our needs.

3.3 Lucene-based indexing

Apache Lucene is a library for a high-performance, full-featured text search engine, which performs indexing and searching using Low RAM, and generates an index file with *reasonable* size. To customize the Lucene core and built a data retrieval system that solved the problem of memory blowup. Our system consisted of two parts:

- A real-time index server and
- A search server.

The real-time index server refreshes the database frequently and indexes dynamically the newly collected MBPs on the fly. The search server returns the MBPs that contain all the keywords entered by the user.

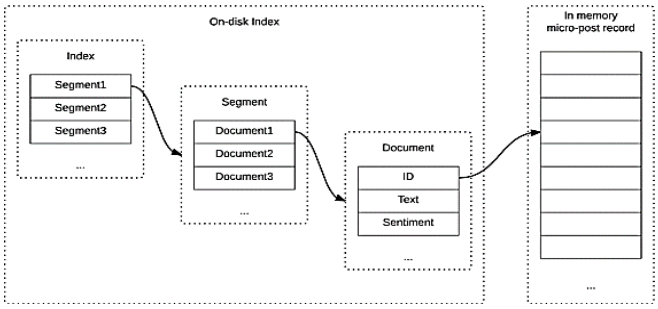

Figure 1Lucene-based index structure.

These two parts may work simultaneously without conflicting each other, and so the search server can return realtime posts the system collects. For convenience, call a person who authors or retransmits an MBP a sender. For each MBP, that will include in one string its sender's location and gender, the posting time, and the OSN this MBP is transmitting within. This string is used as the indexing key of the MBP.

Examining how fast each method responding to user queries, as well as how fast the method carryout statistical analysis and returning the results. Comparing accuracy of each method.

IV. STASTICAL ANALYSIS

In addition to returning the MBPs, the Micro blog Processing Toolkit also returns a number of statistical results of these MBPs, including the proportion of genders, the provinces of the senders, the sources of the MBPs, and the distribution of the posting time. Moreover, hot words (i.e., words with high frequencies) are generated while returning the MBPs.
To carry out statistical analysis on the MBPs which have been collected, and group them according to the following three attributes:

- Posting time
- Sender's gender
- Sender's location.

Using the built-in regular expressions of Mongo DB, it only need to traverse all the MBPs the database returned and then carry out the statistical analysis. This process, however, is extremely time consuming. To customize our own statistical features. As mentioned earlier, for each MBP, include in the index its sender's location, gender, and posting time in one string. To set this string as the key of the MBP. This means that for two MBPs posted during the same hour, at the same location, and by the senders of the same gender, they will share the same key. When indexing the MBP, include this key in the index. When searching for the data, first carry out thegroup search by the key. After obtaining the groups with the same key, traverse each group and perform the statistical analysis in the group.The trending graph, gender distribution, location distribution, and source OSNs can be easily calculated by a simple traversal of the MBPs in each group. However, the hot words and the top senders cannot be obtained this way since each MBP may have its unique top words. Instead, MPT traverses all the MBPs contained in each group and adds all the keywords and senders to two different maps. It then analyzes these two maps to obtain the list of hot words and the list of top senders.

4.1 Experiments

To design an experiment to test the efficiency and accuracy of the two methods for data retrieval and statistical analysis mentioned in the previous sections. For convenience, refer to these methods of Mongo DB's built-in regular expressions and Lucene-based indexing as, respectively, Mongo and Lucene. The MBPs collected in 1 Hour of data. (The day was randomly chosen) from Pinterest and Ello as the data set for comparing the two different methods. There were about 4.3 million MBPs in this 1-day collection. All experiments were executed on a commodity computer with a quad-core CPU and 16-GB RAM.In practice, the system is designed to handle the posts of multiple days with volume much larger than the data used in the experiment. For a dataset of very large scale, our system will slice the dataset into multiple subsets. In particular, for a dataset that contains posts in multiple days, the system will slice the dataset into multiple subsets such that each subset contains the posts of the same day. Thus, our experiment does indicate the fundamental performance. Hence repeat the experiment on datasets from different dates with similar results.

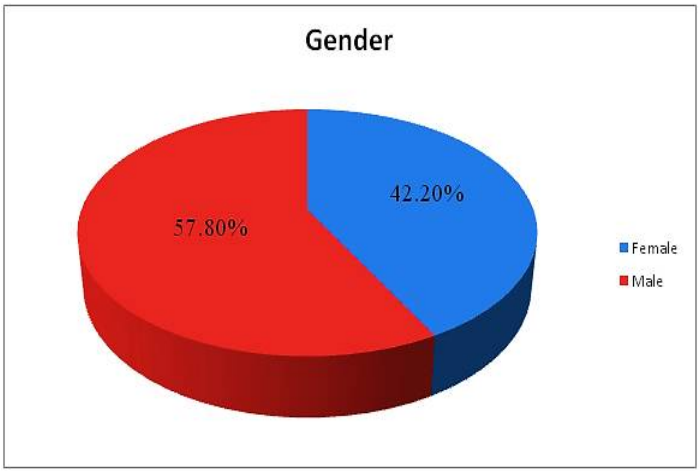

Figure 2: Gender Distribution

The experiment consisted of two parts. In the first part is toexamine how fast each method responded to user queries, as well as how fast the method carried out statistical analysis and returned the results. In the second part, compared the accuracy of each method. To run the experiment, we pre-counted all the keyword phrases in the data set and randomly selected a number of keyword phrases as our testing phrases, such that these phrases weremade up of two or three keywords and appeared in the test data set formore than 100 times. For each phrase selected and executed a query with each method. Recording the time that each method incurred to respond to the query and finished up the statistical analysis. Mongo and Lucene tested separately. For each test, repeating the process twice and calculated the average time.
The results are shown in Figures 3.

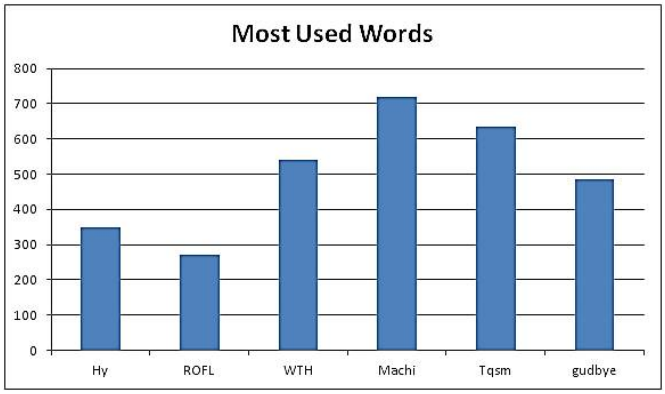

Figure 3. Frequently Searched Words.

Thus can see that the accuracy of each method differs from each other, where Mongo is 100% accurate. In other words, its precision and recall rates are both equal to 1. Next word may miss some MBPs. Lucene, have the worst precision and recall rates, where the number of returned MBPs is usually larger than the actual number that contains the phrase. This is caused by the Lucene indexing structure, where all theMBPs that contain a subset of the search keywords are returned. For example, if phrase X is made up of words A *and* B, when querying X, Lucene will return all the posts that contain A *or* B. So, the returning result usually has a larger-than-actual count.

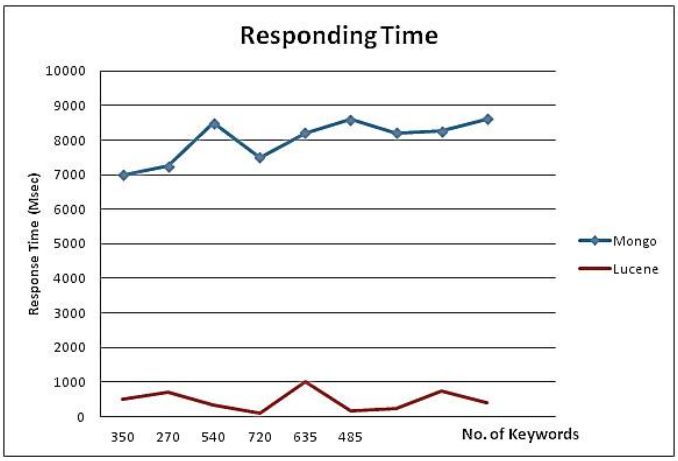

Figure 4: Responding Time

Method	Precision	Recall
MONGO	1	1
LUCENE	0.53	0.81

Table 1: Experimental results on precision and recall rates

Table 1 show the precision and recall rates of the search results using the two differentmethods. It can see from the table that Mongo performs well on accuracy. Lucene suffers precision losscompared to the other two methods. But it offers better performance in the recall valuethan Next word, which means that it may miss fewerMBPs than Next word.

V. CONCLUSION

From our experiment, each method has its pros and cons. In particular,

(i) Mongo provides the regular expression searching method with the perfect accuracy. But it is too slow to meet the needs of real-time querying and stat analysis.

(ii) Lucene, on the other hand, has the fastest responding time andthe fastest statistical analysis time, but it incurs low accuracy. This method is good forbuilding a fast search engine but may not meet the requirement of high accuracy.

REFERENCES

[1] Dr. Arup Barmanand Mr. Husain Ahmed (2015). Big Data in Human Resource Management -Developing Research Context, 19p.Research Gate. North America Research Gate Corporation.
[2] Jonathan and Magnusson (2012). Social Network Analysis Utilizing Big Data Technology .71p*Social Network Analysis Utilizing Big Data Technology*. Uppsala University, Sweden.
[3] Emad A Mohammed, Behrouz H Far1and Christopher Naugler(2014). Applications of the Map Reduce programming framework to clinical big data analysis: current landscape and future trends. 23p, *BioData Mining*. BioMed Central. University of Calgary, Calgary, AB, Canada.
[4] Songchang Jin1 · Wangqun Lin2 · Hong Yin1 · Shuqiang Yang1 · Aiping Li1 ·Bo Deng2 (2015).
Community structure mining in big data social media networks with MapReduce.12p, *Cluster Comput.* Springer Science+Business Media New York
[5] Borthakur, D.: HDFS architecture guide, HADOOPAPACHEPROJECT.http://hadoop.apache.org/common/docs/current/hdfs_design (2008)
[6] Jin, S., Yu, P., Li, S., Yang, S.: A parallel community structure mining method in big social networks, mathematical problems in engineering, (in Press) http://downloads.hindawi.com/journals/ mpe/aip/934301 (2014)
[7] Apache HDFS. Available athttp://hadoop.apache.org/hdfs
[8] Pinterest API Available in Mashape https://www.mashape.com/pinteresttojson
[9] MongoDB .Available at https://docs.mongodb.org/v3.0/installation
[10] Apache Lucene. Available at https://lucene.apache.org/pylucene/install.html