Theoretical Assessment of Big Data And Hadoop Techniques

Sonali Gurjar[1], Nirmala Choudhary[2], Anil Dhankhar[3]
[1]Dept of MCA
[2]Asst. Professor, Dept of MCA
[3]Assoc. Professor, Dept of MCA
[1, 2, 3] RIET Jaipur

Abstract- *We live in on-demand, on-command Digital universe. This data is categories as "Big Data" due to its complete Volume, Variety& Velocity. 'Big Data' describes techniques and technologies used to check large-sized datasets with high-velocity. Big data can be structured, unstructured, semi-structured or heterogeneous in nature. Data is originated from various different sources and can arrive in various rates. Due to its different nature it is stored in distributed file system.Hadoop and HDFS by Apache is widely used for storing and managing Big Data. In order to process these large amounts of data in an inexpensive and efficient way, parallelism is used. Data management, warehousing and analysis systems facesshortage of tools to analyze data. Testing of Big Data is a difficult as it involves large distributed file systems which should be fault tolerant. Traditional DBMS methods like Joins and Indexing is used for classification and clustering of Big Data.*

Keywords- Big Data Analysis, Big Data Management clustering , HDFS

I. INTRODUCTION

Big Data:It is a term that describe the combinations of data sets whose size (volume), complexity (variability), and rate of growth (velocity) make them difficult to be stored, managed, processed, retrieve or analyzed by traditional technologies and tools, that is impossible by using traditional methods such as relational databases .it used the large amount of data and modify these large data sets. For example with the coming of smart technology there is rapid increase in use of mobile phones due to which large amount of data is generated in every second, so it is impossible to manage these large amount of data by using traditional methods hence to solve this problem big data is introduced it have Layered Architecture which can be divided into three layers, including Infrastructure Layer, Computing Layer, and Application Layer from top to bottom.

II. 3 VS OF BIG DATA

(A). DATA VOLUME: volume refers to the amount of data. At present the volume of data stored has increase from megabytes and gigabytes to peta-bytes and is supposed to increase to zeta-bytes in nearby future Volume of data stored in activity repositories have grown from megabytes and gigabytes to petabytes.

(B). DATA VARIETY: it defines different types of data-– text, images video, audio and sources of data. Data being produced is not of single category and not only includes the traditional data but also the semi structured data from various resources like web Pages, Web Log Files it burst from structured and tradition data stored in enterprise repositories to unstructured, semi structured, audio, video, XML etc.

(C). DATA VELOCITY:it refers to the speed of data processing. Some time consuming processes such as catching fraud, big data must be used as it streams into your enterprise in order to maximize its value. It is a concept which deals with the speed of the data coming from various sources. This characteristic is not being limited to the speed of incoming data but also speed at which the data flows and aggregated Velocity

Fig. 1 Architecture

III. PROBLEM ASSOCIATED WITH BIG DATA

In the distributed systems word "Big Data" started become big problem in the late 1990"s due to the impact on world-wide Web and a resulting need to index and query its rapidly boom content. Database technology (including parallel databases) was considered for the task, but was found to be neither suited nor cost-effective for those purposes. The turn of the millennium then brought further challenges as companies began to use information such as the topology of the Web band users" search histories processing provide it uses HDFS useful search results, as well as more effectively-targeted advertising to display alongside and fund those results. Google's technical responses to the challenges of Web scale data management and analysis was simple, by database standards, the layers found in the modern software Data revolution in the world system. To handle the challenge of Web-scale storage, the Google File System (GFS) was created.

GFS gives clients familiar OS-level byte-stream abstraction, but it have extremely large files whose content can span hundreds of machines. For enabling just needs Google developers to process large collections of data by writing two based record management operations, the Hadoop Base store is available as a key-value layer in the Hadoop framework applies to the instances (map) and sorted groups of stack the contents of Hadoop share a common key (reduce) – similar to the sort either be directly accessed and implement by a client of partitioned parallelism utilized in shared-nothing parallel query processing. Driven by very similar requirements, software Developers Yahoo!, Facebook and other large Web companies Taking Google's GFS technical specifications, were the middle layer is Map Reduce system, which applies map operations to the data in division of an HDFS file, sorts and then performs reduce operations on the groups of output data .Processing the challenges of Big Data are the real hurdles.

(A). SIZE:

The first thing with Big Data is its size. Managing large and rapidly increasing volumes of data has been a challenging task for many decades. In the past, this challenge was mitigated by processors getting improves, following Moore's law, to provide us with the resources needed to increasing volumes of data.

(B). HETEROGENEITY:

When humans stored information, a great deal of heterogeneity is comfortably tolerated. In fact richness of natural language can provide valuable depth. machine analysis algorithms expect similar data. In consequence, data must be carefully structured as a first step. Computer systems work most smoothly if they can store multiple items that are all identical.

(C). PRIVACY AND SECURITY :

It is the most sensitive challenges of Big data. The personal information of a person when included with external large data sets, leads to the inference of new facts about that person and it's possible that these kinds of facts may be covert and the person might not want data owner to know about them. Information regarding the people is collected and used in order to add information. Another important conciseness arising would be Social sites where a person taking advantages of the Big data predictive analysis and will be easily identified and treated worse. Big Data increase the chances of certain tagged people to phases from adverse consequences without the ability to fight back

(D.) TECHNICAL CHALLENGES:

With the incoming of new technologies like Cloud computing and Big data it is always calculated that whenever the failure occurs the damage done should be acceptable. Fault-tolerant computing is extremely hard. Thus the main task is to reduce the probability of failure. Two methods which seem to increase the fault tolerance in Big data are as:

First is to divide the whole calculation being done into tasks and assign the tasks to different nodes for calculation.

Second : one node is giventhe work of detecting that nodes are working properly. If something happens that particular task is restarted. But sometimes it's quite possible. The output of the previous calculation of task is the input to the next computation. Thus restarting the whole calculationbecomes hard process.

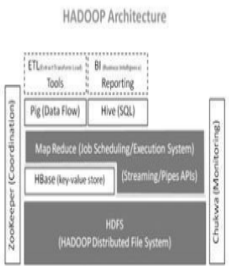

Figure 1.Hadoop Architecture Layers

Figure 2.Hadoop Architecture Tools and usage

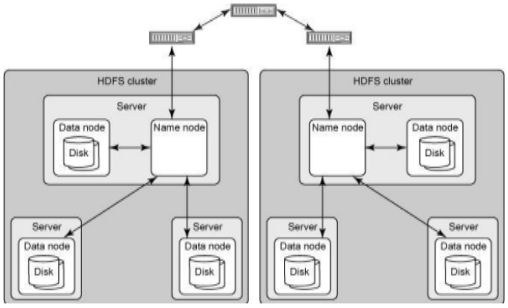

Figure 3. HADOOP Clusters

Fig 2 Hadoop architecture

(A). HDFS :

Hadoop add a fault-tolerance storage system called the Hadoop Distributed File System, or HDFS. HDFS store large amounts of information, scale up incrementally and survive the failure of significant parts of the storage infrastructure without losing data. itgenerates clusters of machines and coordinates work among them. Clusters can be built with inexpensive computers. If one fails, Hadoop continues to operate the cluster by shifting work to the remaining machines in the cluster. HDFS manages storage on the cluster by breaking incoming files into "blocks," and storing each of the blocks always across the pool of servers. It saves three full copies of each file by copying each piece to three different servers.

(B). MAPREDUCE ARCHITECTURE:

The processing stand in the Hadoop is the MapReduce framework. It allows the organization of an operation applied to a huge data set, divide the problem and data, run it in parallel, this can processed on multiple dimensions. In a traditional warehousing In Hadoop, the operations are written as MapReduce. There are lots of higher level languages that make programs easier. The outputs of these jobs can be written back to either HDFS or placed in a traditional warehouse. There are two functions in MapReduce as follows:

Map takes key pairs as input and generates an intermediate set of key.

Reduce function merges all the between values linked with the same intermediate key.

IV. MAP REDUCE WORKING.

We implement the Mapper and Reducer interfaces to show the working:-

I. MAPPER

Mapper maps input value couple to a set of intermediate key/value pairs. Maps are the individual tasks that implement input records into intermediate records. The changeable intermediate records do not need to be the similar type as the input records.

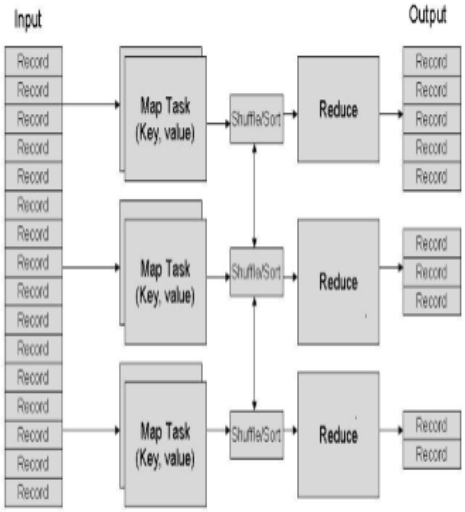

Fig .3 Map reducing components

A given input pair map to zero or many output pairs. The number of maps is used by the total size of the inputs. The right level of parallelism for maps seems to be around 10-100 maps per-node the maps take at least a minute to execute. Example, if we expect 10TB input data and have a block size of 128MB, we'll end up with 82,000 maps .

II. REDUCER

It reduces a set of transitional values which share a key to a smaller set of values. It has 3 primary phases: shuffle, sort and reduce. .

V. MAP REDUCE TECHNIQUES

(A). COMBINING

Combiners provide a general working within the MapReduce framework to decreases the amount of moderate data generated by the mappers. They can be understood as "mini-reducers" that. The combiner's average term counts the documents processed by each map task. This result in decreasing in the number of moderate key-value pairs that need to be shuffled across the network. They cut down the result size of map functions and perform reduce-like function in each

(B). SHUFFLING

Shuffling is the process of mixing the indexes of the files and their keys, so that a different mix of dataset can be obtained. If the dataset is drag, then there are better chances that the resultant query processing will yield near accurate results. We can relate the dragging process with the population generating by crossover in the GA algorithms. The processes are different in but their working is similar.

(C). SHADING

It is a term used to divide the Mappers in the HDFS architecture.it define the groupings or documents which are done so that the MapReduce jobs are done parallel in a distributed environment.

(E). JOINS

It is a RDBMS term; it refers to join two or more discrete datasets to get Cartesian product of data of all the possible combinations. Map Reduce does not have its own Join techniques, but RDBMS techniques are tweaked and used to get the maximum possible combinations. The join techniques which are adopted for Map Reduce are Equi Join, Self Join, Repartition Join and Theta Join .

VI.CONCLUSION

There is a chances for making faster growth in scientific discipline for testing the largest data .The technical tasks are most common across the large variety of application domains, therefore new cost active and faster methods must applied to test the big data. We have entered an generation of Big Data. The paper define the basics of Big Data along with 3 Vs, Volume, Velocity and variety. These technical task must be consign for efficient and fast working of Big Data. The task include not just the obvious issues of scale, but also heterogeneity, lack of structure, error-handling, provenance, and visualization, at all stages of the analysis pipeline from data acquisition to result interpretation. These technical problems are common across a large variety of application domains, and therefore not cost effective to address in the context of one domain alone. The paper tails Hadoop which is an open source software used for processing of Big Data.

REFERENCES

[1] S.Vikram Phaneendra, E.Madhusudhan Reddy,"Big Data- solutions for RDBMS problems- A survey", In 12thIEEE/ IFIP Network Operations & Management Symposium (NOMS 2010) (Osaka, Japan, Apr 19{23 2013).

[2] Aveksa Inc. (2013). Ensuring "Big Data" Security with Identity and Access Management. Waltham, MA: Aveksa.

[3] Hewlett-Packard Development Company. (2012). Big Security for Big Data. L.P.: Hewlett-Packard Development Company.

[4] Kaisler, S., Armour, F., Espinosa, J. A., Money, W. (2013). Big Data: Issues and Challenges Moving Forward. International Confrence on System Sciences (pp. 995-1004). Hawaii: IEEE Computer Soceity.

[5] Katal, A., Wazid, M., Goudar, R. H. (2013). Big Data: Issues, Challenges, Tools and Good Practic

[6] OnurSavas, YalinSagduyu, Julia Deng, and Jason Li,Tactical Big Data Analytics: Challenges, Use Cases and Solutions, Big Data Analytics Workshop in conjunction with ACM Sigmetrics 2013,June 21, 2013.

[7] Kyuseok Shim, MapReduce Algorithms for Big Data Analysis, DNIS 2013, LNCS 7813, pp. 44–48, 2013.