A brief Study of Big Data Analytics using Apache Pig and Hadoop Distributed File System

Ms. Sarika Rathi

Faculty of Computer Engineering in MGM's Polytechnic

***Abstract*:-In today's generation of Information Technology, Dog Cutting has achieved new concept for Big Data. The term 'Big Data' best explains advanced techniques and technologies to capture, store, distribute, manage and analyze petabyte - or terabytes data with high-velocity, volume and variety. To analyze this structured, unstructured or semi-structured data in huge amount "Hadoop" is the only solution. Data is generated from multiple different sources and can arrive in the system at various rates to process such data efficiently. Parallel technique of processing data is used. In Big Data Hadoop Distributed File System is very popular. It gives a framework for storing data in a distributed environment also contains some set of tools to retrieve process.**

***Index Term* :- BigData, DataNode, Hadoop, Hbase, HDFS, Hive, JobTracker,MapReduce, MasterNode, NameNode, NOSQL,Oozie, Pig, PigLatin, SecondaryNode, Sqoop, TaskTracker, Zookeeper.**

I.INTRODUCTION

Big Data is a key word that denotes to structured or unstructured or semi structured data with combinations of 3Vs. The impression of using Hadoop technology on Big Data is transforming approach towards analysis, maintaince and generation of enormous data .Now a days all IT companies are accepting various scripting platforms with Hadoop technology to reduce time for studying vast data. Apache's Pig is an important component of Hadoop system which reduces the coding and analyzing time for Big Data. Big data is collection of complex and large data sets, which include information, may be produced by multiple services. The main task here is to combine multiple data from multiple systems. This paper's first part explains the concept of BigData, second part explains concept of Hadoop architecture using Map Reduce function and third part explains Apache's Pig execution environment. This paper is a brief study to learn Apache's Pig and HDFS.

II. BIG DATA

Big data is a gathering of very large amount of datasets. It consists of structured data as relational data, semi structured data as XML data & unstructured data as Word, PDF, Text, Media logs. Combination of all these contains a huge amount of information. Big data is collection of complex and large data sets, which include information, may be produced by multiple services such as Black Box Data, Social media, Stock exchange, Search engine, sensors used for climate information, digital pictures, traffic, software logs etc.Big data at startcharacterized as combination of 3V's:1st V indicates gigantic Volume of data , 2nd V represents Velocity of processing huge data and 3rd V denotes broad variety of different types of data.

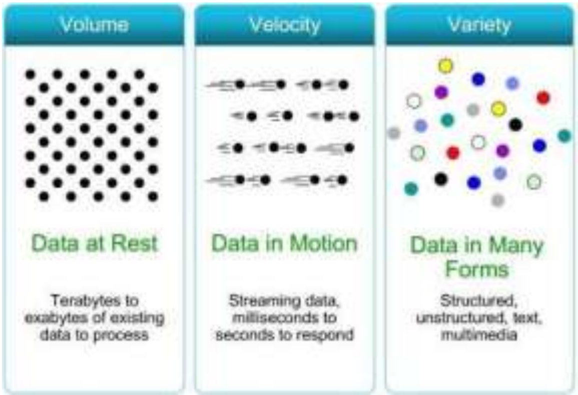
Volume
Velocity
Variety
Data at Rest
Data in Motion
Data in Many Forms
Terabytes to exabytes of existing data to process
Streaming data, milliseconds to seconds to respond
Structured, unstructured, text, multimedia

Fig. 1 Enlighten concept of 3V's

A. Tasks for Big Data are

1. Hard to work using most relational database management systems.
2. Challenges to find insights in new and emerging types of data.
3. Absence of resource availability.

B. Is analysis of big data is useful for an organization

Efficient studies of Big Data give a lot to business benefit so that organizations will able to understand which region to focus means what is important and which areas are less important. Big data analysis supplies some early key indicator that can prevent the company from a huge loss or help in generous a great opportunity with open hands! A precise analysis of Big Data helps in decision making! For instance, nowadays people rely so much on Facebook ,Twitter all social media sites, all online shopee such as Amazon, Snapdeal, Flipcart before buying any product or service. Big data helps to sort out data and generates such reports very fast using Hadoop HDFS.

C. Why we need Hadoop

On a daily basis a large amount of unstructured data is getting dumped into our machines. Major dispute is not to store bulky data sets in our systems but to get back and analyze such big data in the organizations, which data present on different machines in different locations. In such situation we require necessity for Hadoop. Hadoop has skill to analyze the data nearby in different machines at different locations very rapidly also in very less cost of use. It applies idea of MapReduce which permit it to divide query into small parts and process them in analogous, this concept is also recognized as parallel computing.

III. HADOOP ARCHITECTURE

Apache developed an open source framework for distributed data processing on huge volume of data sets and named it as Hadoop. Hadoop was developed for working out bulky amount of data in distributed computing environment and storage of data in multiple data nodes. Hadoop cluster is an unusual type of computational cluster projected for storing and analyzing massive amount of unstructured data in a distributed computing atmosphere. These clusters can also run on low cost commodity computers.

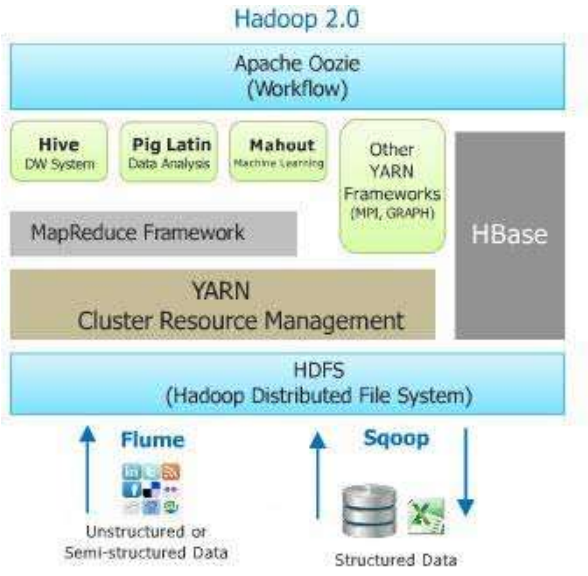

Fig. 2 shows components of Hadoop Ecosystem

In Hadoop file is split out in different blocks, each block then assign to different data node. Sizeof blocks varies for each architecture. For Hadoop default block size is 64MB, for Cloudera default block size is 128MB and for Mapr default block size is 256MB. Block is the smallest quantity of data that can be read or written. Files in HDFS are broken down into block-sized chunks, which are stored as self-determining units. HDFS blocks are large as compared to disk blocks; mostly to reduce the cost of seeks. If a particular file in HDFS containing 50MB data then HDFS block will be consumed by an HDFS block is 50MB and 14 MB will be free to store something else. It is the MasterNode that does data allocation in an efficient manner.

Suppose we have a special file stored in a system but due to some technical reason file gets destroyed. Then there is no chance of receiving data back present in that file. HDFS works with commodity hardware (systems with average configurations) that has high chances of getting crashed any time. To avoid such situations Hadoop, gives special characteristic name as fault tolerance, in this when we store a file, it automatically gets replicated at two other location(machines)also. So even if one or two of the systems collapse, the file is still available on the third system. Hence, there is no chance of losing the data. This replication factor helps us to attain the feature of Hadoop called Fault Tolerant. Blocks provide fault tolerance and availability, to ensure against corrupted blocks, disk and machine failure, each block is replicated to a small number of physically separate machines (typically three). If a block becomes unavailable, a copy can be read from another location. Depending upon the block size, once the data is stored, HDFS will keep on storing the last part of the data which will say where the next part of the data will be and can do indexing of blocks according to do so.

When multiple client contacts to NameNode to open particular file for writing, the NameNode grant a lease to the client to create that particular file. When second client tries to open same file for writing, the NameNode will observe lease for the particular file is already granted to another client, and not open request for the second client

A. Use of HDFS application for large data

HDFS is more appropriate for huge amount of data sets in a single file as compare to little amount of data spread across multiple files, this is so happen because NameNode is a very expensive high performance system, so it is not practical to occupy the space in the Namenode by unnecessary amount of metadata that is generated for multiple small files. So, when there is a large amount of data in a single file, name node will occupy less space. So we prefer HDFS for large amount of data.

Two major layers of Hadoopare :Hadoop Distributed File System (HDFS) and Map Reduce layers.

B Hadoop Distributed File System (HDFS)

HDFS contains master slave architecture. Internally an HDFS , a file, is split in one or more blocks called chunks are stored in a set of data nodes. The NameNode executes file systems namespace operations like opening, closing and renaming files and directories. It also determines mapping of blocks to DataNodes. Datanodes are the slaves which are arrange on each machine and provide the actual storage. The DataNode are responsible for serving read write requests fromfile system's client. The DataNodes also perform block creation, deletion and replication upon instruction from NameNode. Namenodeneeds to be a high-availability machine as total HDFS system relay on NameNode.

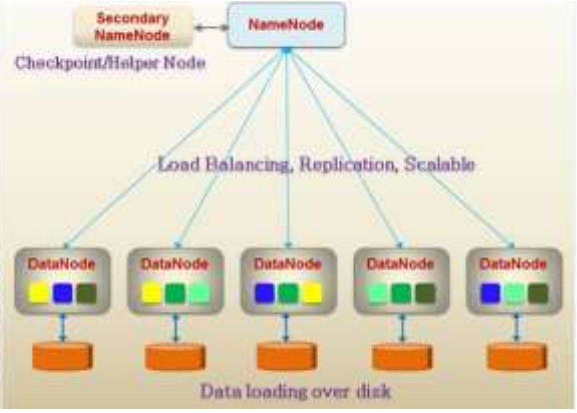

Fig. 3 Work flow for HDFS

An HDFS consists of single NameNode, which consists of file's metadata, keeps track of all file system related information such as which section of file is saved in which part of cluster, last access time for files and user permissions like which user have access to file. NameNode is a part of master node. Copy of NameNode is placed in secondary NameNode which is nothing but 2's compliment of node. The secondary NameNode continuously reads the data from RAM of the NameNode and writes it into the hard disk or the file system. It is not aalternate to the NameNode, so if the Namenode fails, the entire Hadoop system goes down.

Assistance to master node is JobTracker. Slave node is nothing but DataNode which send a signal after every 3 seconds to master node. Assistance to slave node is task tracker. If NameNode does not have any data then it is not a part of cluster.

When Hadoop produced 100 tasks for a job and one of the task failed. In such case it will restart the task again on some other TaskTracker and only if the task fails more than four times (the default setting and can be changed) times then it will kill the job.

If a node come into view to be executing slow, the master node can redundantly carry out another instance of the same task and first output will be taken, process is Speculative execution.

i. Concept of JobTracker

Job tracker is a daemon which runs on NameNode for submitting and tracking MapReduce jobs in Hadoop. It allocates tasks to the different Task Tracker. In a Hadoop cluster, there is only one JobTracker, but many TaskTrackers. JobTracker is the single point of failure for Hadoop and MapReduce Service. If the job tracker goes down all the running jobs are halted. It receives heartbeat from TaskTracker based on which JobTracker decides whether the assigned task is completed or not.

ii.Concept of TaskTracker

TaskTracker is also a daemon that runs on DataNodes. TaskTrackers handle the execution of individual tasks on slave node. When a client submits a job, the JobTracker will initialize the job and divide the work and assign them to different TaskTrackers to perform MapReduce tasks. While performing this action, the TaskTracker will be simultaneously communicating with JobTracker by sending heartbeat. If the JobTracker does not receive heartbeat from TaskTracker within specified time, then it will assume that TaskTracker has crashed and assign that task to another task tracker in the cluster.

C. Map Reduce

Map Reduce is a data processing component. It is 2nd layer of Hadoop placed above HDFS. It consists of two sub functions Map function and Reduce Function. Determination of Map function is to perform filtering, sorting and mapping of the data. Reduce function utilized for performing summary operation, processing intermediate data. MapReduce phase create result into single Key. Map function splits data into chunks do processing, post to the reduce function combines data and yields same key in the form of results.

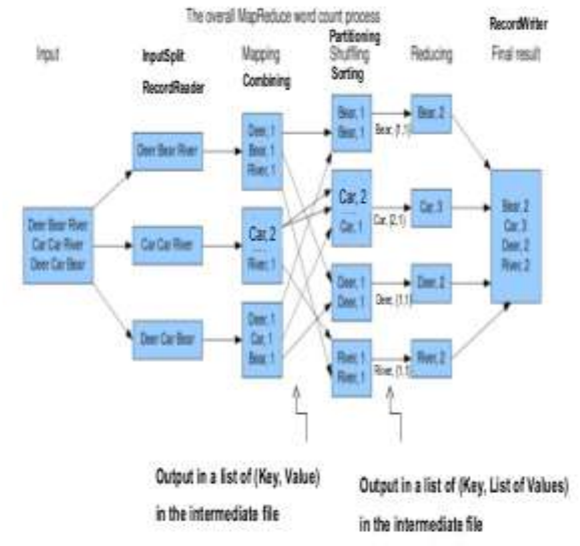

Fig. 4 Work flow of MapReduce function

In this input is taken that input is broken with the help delimeter as new line, each line is then again split into words with help of delimeter as space and assign numbering to each word this step is sometimes called as mapping next step is shuffling of words with minimum shuffling requirements next to that counting of number of words refer it as reducing and last step is final result by coming all lines in reducing.

IV. APACHE PIG

Apache's Pig standsfor a scripting platform developed by Yahoo in 2006 especially to create and execute MapReduce jobs on dataset. Later, shifted into Apache foundation which workings on data flow language in 2007. Apache's Pig platform was used to analyze the huge data sets. Pig can function on compound data structures;still it can have levels of nesting. Pig generates a mechanism which executes data in analogous to evaluation of data on Hadoop. Apache Pig runs on Hadoop byusing Map Reduce for data processing and also Hadoop Distributed File System (HDFS). Apache's Pig bestcommonlyaccepted in following cases by - web search platform, log processing, unstructured data, data set processing, replicated data.

Apache Pig fulfills Pig Latin scripts which users have written into a sequence of one or several Map Reduce. Pig Latin does not have if statement or for loop since it solitary emphasis on execution of data flow. Apache Pig was understood by distributed environment utilized by Map Reduce given those useful results on large data sets.

A. An Execution Phase

In Apache Pig two selections for execution location first is local location and second is distributed location. In localenvironment all files are installed and run from your local host and local file system. There is no need of Hadoop or HDFS. This mode is generally good for testing when we do not have a full distributed Hadoop environment. MapReduce mode is where we load or process data that exists in the Hadoop File System (HDFS) using Apache Pig. In this, whenever we execute the Pig Latin statements to process the data, a MapReduce job is invoked in the back-end to perform a particular operation on the data that exists in the HDFS. To start Pig in local mode command line interpreter bypassing –x local. To start pig in distributed environment by passing – x mapreduce. By default if we have not given any choice it automatically starts in distributed i.emapreduce environment.

PigLatin is collection of statements; each statement can be an operation or a command. To load data from a file use the LOAD operation with a file name as an argument. After LOAD it does not directly show output again we have to debug it. For debugging we have different options first pass command Dump to see result to terminal, second pass command to Describe to review the schema of relation, third pass Explain to view logical, physical or map reduce plan to compute relation and last is Illustrate to view step by step execution of series of statements. The data model of Pig is fully nested.

Relation is the outermost structure of the Pig Latin data model. A Relation is bag of tuples. The relation in PigLatin unordered i.e it does not maintain any sequence. Bag is a collection of tuples {(purvesh,12),(prerna ,15)}. A tuple is an ordered set of fields (purvesh, 30). A field is a piece of data. One more thing that is Map is a set of key value pairs and are separated by '#' ['name'#'prerna', 'age'#15]

Apache Pig scripts executed in interactive mode using Grunt shell, batch mode can be executed in single file using .pig extension, and embedded mode can be executed by using User Defined Functions(UDF) in programming language.

To enter in the Grunt shell in any mode i.e. Local or MapReduce we have to give command as

$ ./ pig – x local or $ ./ pig –x mapreducewhen we enter such command our prompt changes with grunt>

We can give command to load file in Grunt shell as grunt> student= LOAD '/home/cloudera/bda/student ' using PigStorgae(',') as (id : int, name :charrarray);

grunt> Dump student;

We can transfer path from Local to Mapreduce and vice versa. For that simply we can pass on command prompt as
grunt >hadoopfs - mkdair/bda ;
creates directory to Hadoop HDFS. Then give command as hadoop fs – copyFromLocal; we can use put command to put file form local to HDFS. When we want to exit from Grunt just press ctrl +d.

We can say that Pig is an significant component of Hadoop system, as it utilize concept of Hadoop MapReduce and HDFS. Pig Latin is a scripting language used on Pig platform center of attention is data flow where as other language center of attention is control flow.

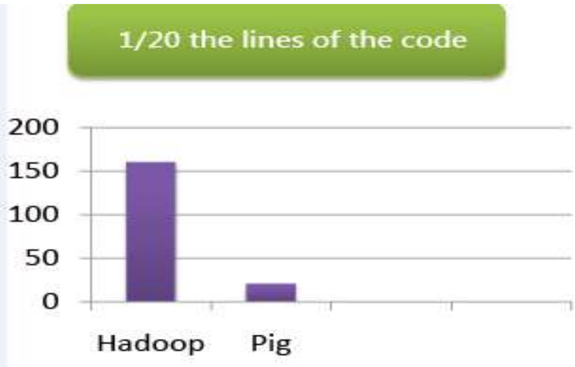

Fig. 5 Graphical representation indicates that Pig requires only 1/20th lines of the code as compared to Hadoop .

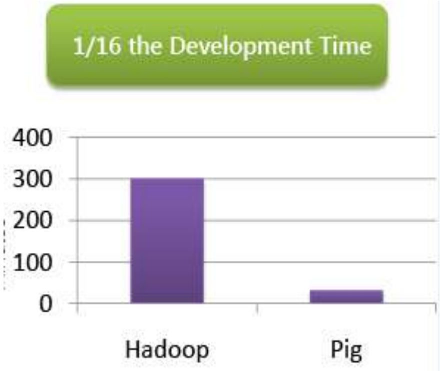

Fig. 6 Graphical representation specify that Pig requires 1/16th time for progress as compare to normal Hadoop.

After Pig new techniques came in marker are HIVE is a data ware house environment in Hadoop framework, uses concept of HQL high query language develop by Facebook. Next is SQOOP is related with databases in HDFS, it has two major functions first is Sqoop Import used to import data from RDBMS and other databases to Hadoop and second is Sqoop Export used to export data from HDFS to other databases. Next is FLUME used to import streaming data collected by live and stored directly to the storage. Next is OOZIE which is a ecosystem used for two functions first is define workflows is nothing but set dependencies and second is scheduling workflows involves scheduling based on internal and external. ZOOKEEPIE is gateway between ecosystems and Hadoop storage generally used to create connection between distributed system and maintain logs. HBASE is one of the NOSQL i.e not only SQLdatabase generally used for large online transactions. It is a distributed column base data built on top of Hadoop .

V. CONCLUSION

Big data contains huge data in size, analysis and structuring is big challenge in front of researchers. This paper best describes concept of 3 Vs as volume, Velocity and Variety. Implementation of Hadoop on Bid Data gives solution for Big Data i.e how it is manageable by reducing our time and space. PigLatin is one of the most suitable procedural dataflow scripting language for analyzing and structing data. Feature of Pig are Ease of Programming, UDF's Handle all kinds of Data. This paper best describes the concept of Hadoop an open source, also concept of HDFS and MapReduce function of Hadoop. Also it best explain difference of traditional RDBMS and Hadoop for Big Data.

REFERENCES

[1] Kedar Dixit Workshop on "Big Data" using Hadoop Technology done in Jawaharlal Engineering college Aurangabad from 11th Jan to 13th Jan 2017.

[2] An introduction to Pig – StratApps <www.startapps.net/intro-pig.php>

[3] an introduction to Pig - < https://www.tutorialspoint.com/apache_pig>

[4] Kranti Bansal and Priyanka Chawla "a study of Bid Data Analysis Using Apache Pig", international journal for IJCTA pp 8665-8672

[5]Sanjeev Dhawan, Sanjay Rathee "big data analytics using HAdoop components like Pig and Hive" Americal International Journal of Research in Science, Technology, Enginering & Mathematics. Pp 88-93 March- May 2013

[6] Concept of MapReduce form www.saphanatutorial/com/mapreduce

Sarika R. Rathi , Computer Department , MGM's Polytechnic College, Aurangabad,India,8087624333/7798615500.