Efficient Measures for Improvement and Optimization of Big Data Hive Queries

K. DeviPriya, V. Chandra Kumar

Abstract—Big Data is the term used for representing huge datasets that are not processed using traditional techniques. These days, every sector like Industry, education, hospitals, companies and devices like IOT and Sensors are generating huge amount of data in the order of terabytes, petabytes, yottabytes etc. It is a difficult task to store and analyze the data using traditional RDMS and programming languages. Storing and analyzing big data requires special tools to process the data in an effective manner. Hadoop is one such tool to store and process big data in an efficient way. Map Reduce is a programming model under Hadoop framework which processes the data using < key, value> pairs. But the difficulty level of the programming in the Map Reduce approach, users needs another easy and effective solution to handle big data. Hive is such a solution that processes large amounts the structured data. The environment and commands of hive provide facility to the users easy and flexible query interaction of analyzing big data. But the problem associated with hive is lack of optimization of queries. In this paper, initially, we formulate the approach of map reducer for analyzing big data then discuss problems and difficulty associated with the map reducers. We then discuss implementation and evaluation of Hive queries and then how to create Hive tables with optimized. The results are evaluated using Hadoop Ubuntu Virtual Machine.

Index Terms—Big Data, Bucketing, Hadoop, Hive, Map Reduce, TEZ engine

1 Introduction

The age of Big Data has begun[1]. Data on web servers, Social media, Industries,Bio Informatics ,Medical Sciences etc., increased very quickly and the present technologies are unable to store the data due to storage problems and retrieving useful information from the stored server is also challenge task. These complex data sets may be supporting different formats structured, semi structured or unstructured. Recently industries have been spending millions on big data area to meet the challenges. Apache Hadoop is one tool among the existing technologies to handle big data and it is an open source project maintained by many people around the world. Map Reduce, Pig, Hive are some of the core components in the Hadoop framework.Map Reduce is a batch processing model which analyzes and processes the data in terms of mapper and reducers. Apache Pig is interactive query processing model to process and analyzing the data by writing pig Latin scripts. Hive is interactive query language which is process the data by writing hive queries.

- *K DeviPriya is with the Department of Computer Sceince and Engineering, Aditya Engineering College, Surampalem, AndhraPradesh, E-mail: devipriya.kamparapu@aec.edu.in.*
- *V ChandraKumar. is with the Department of Computer Sceince and Engineering, AIMS Engineering College, Mummidivaram,. E-mail:chandrakumar.vichuri@gmail.com.*

2 Architechture of Hadoop

2.1 Hadoop

Hadoop Files System is desgined with the nature of distributed file system. Hadoop is run on the commodity hardware for the storing of complex datasets in a distributed way. HDFS holds vast amount of data and easily accessible by the applications. The objective of Hadoop is moving processing technique to data instead of moving data to the processing model. The detailed architecture of Hadoop Ecosystem is shown in the below Fig. 1.

HDFS-Hadoop Distributed File System [2] is used for storing the large data set in to different data nodes that are available in the Hadoop cluster. Ha-

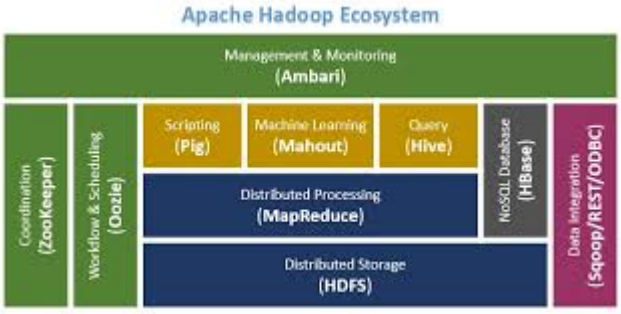

Figure 1 EcoSytem of Apache Hadoop.

doop cluster is a group of machines that are designed to store huge volumes of the data in a distributed environment. It supports the distributed stor-

ing of large datasets in to the data nodes. The controlling of data nodes will be handled by a node considered as Name Node. Name Node is a master node which stores the meta information about the data and known storing information about data.

MAPREDUCE-Map Reduce is a programming model that analyzes and process any type of data in the format of <key, value >pair using mapper and reducer classes. In this model, the Job is divided in to tasks and assigns to the task trackers and controlling, montoring of all these task trackers will be handled by the Job Tracker.

PIG-Pig is a scripting language to support interactive query processing. The language supported by the Pig is Pig Latin. In Pig Latin script Load, Dump, Transform, Store etc., commands are available to analyze and process the data.

MAHOUT-It is an Apache project goal to build scalable machine learning algorithms.

Hive-Hive is an important interactive query processing model that process and handles the data using Hive Query Language. The users who are not familiar with the programming languages writes the Hive queries easily with basic SQL knowledge.

HBase-HBase is a NoSQL database. Apache HBase isa column-oriented key/value data store built to run on top of the Hadoop Distributed File System (HDFS) **Sqoop-**Sqoop is a tool intended for efficient transfer of vastamount of data between Hadoop and Relational database

Ambari-Hadoop management simpler by developing software for provisioning, managing, and monitoring Apache Hadoop clusters

ZooKeeper-is used to manage large clusters in the Hadoop file sytem. It have open source and, distributed nature

Oozie-is used to manage workflow of jobs in the large clusters. It is scheduler that schedule jobs in the apache adoop.

2.2 Characterstics of Big Data

The huge amount of data generated from the different sources is termed big data –is identified mainly by three characteristics:

1. Volume-Volume is considered as quantity of data sets generated from the sources
2. Velocity –Velocity is referred as the speed at which the data is generated.
3. Variety-Variety is represented as the formats of the datasets like structured, semi structured and unstructured. Examples includes here is images, text,video,audio etc.,

2.3 Application Areas of BigData

Big Data is generated from the different application areas. The following are some of the application areas comes under area of the bigdata [3].

1. Geographic Information System-The main Objective of GIS System is better decision about the location. It includes features modifying, managing, collecting, reteriving and sorting of geographical data. Geographical data is very hugeand for analyzing these data Apache Hadoop, Map Reduce, Apache Spark will be needed.
2. Cloud Control System (CCS)CCS manages large amount of traffic hosting, delivery, video streaming etc., generates the Big Data and the efficient processing tool for this area is Apache Hadoop framework.
3. Social Media-FACEBOOKFacebook generates huge amount of the data like post, uploading photos, likes etc., According to statistics Facebook data warehouse has 700TB of data. The efficient processing of these data is possible through Hadoop and Hive.
4. Bio-Informatics [csi-pgno36-7ref] is the study of understanding the molecular mechanism of the life on earth by analyzing Genomic information. Biological data is very huge big data, understanding and analyzing of these data is very difficult and challenge task faced by the researchers. In this area BioPig and Cross bow has been developed for sequence analysis.

3 MAP REDUCE APPROACH

Map Reduce is a framework that provides facility to write the programs for parallel processing in the distributed environment.This approach is divided in to two taks map and reduce.It is possible to write the map function followed by the reduce function.In the configuration settings the number of mappers required to process the data will be decided.The map and reduce function considers the input and out putint the format of <key, value>pair.The following pseudo code describes the data flow from the input of key/value pairs to the list output:

Map (key1, value1) ->list (key2, value2)
Reduce (key2, list (value2)) ->list (value3)

Workflow of MapReduce consists of 5 steps

1. **Splitting** – Splitting of data based on parameters like space,comma, newline etc.,.
2. **Mapping** – Conversion of input<key,value> into another <key,value> format.
3. **Intermediate splitting** – The entire procedure in parallel on different clusters. In order to group them in "Reduce Phase" the similar KEY data should be on same cluster.
4. **Reduce** – In this phase,aggregation operations performed like sum,count,max,min.
5. **Combining** – It is the last phase where all the data is combine together to form a Result

Creation of Mapper and reducer class includes the foolwing
Class MyMapper extends Mapper<LongWritable,Text,Text,IntWritable>
{
Public void map(LongWritable key,Text Value,COntect c)
{
//mapper logic
}
}

Class MyReducer extends Reducer<Text,IntWritable, Text,IntWritable >
{
Public void reduce(LongWritable key,Text Value,COntect c)
{
//reduce logic
}
}
The classes that are required for map reduce programs are available in the Hadoop API.The implementation of map reduce is possible through different programming languages but the challenge of mapreducede is the developer must have good knowledge in the programming concepts which is very difficult task to the normal users and solves the big data problems which are key, value format.

4 HIVE

Hive [4] tool which structures data in to databases using the concepts tables,colums,rows,partitions,bucketing etc. Hive supports primary data types –int,float,double etc., and complex types sturct,unio,map with key and value pair etc., The user uses either CLI or Web GUI or JDBC/ODBC to execute Hive queries. If the user uses CLI or Web GUI for Hive queries then it is directly connected to the Hive driver. If the user uses JDBC/ODBC (JDBC Program) at that movement of the time by using API (Thrift server),it is possible to connect to the hive driver. The Hive driver accepts the Hive queries from the user and sends to the Hadoop Distributed File system.(HDFS).HDFS uses NameNode,DataNode,Job Tracker, Task Tracker for receiving and dividing the work for parallel execution. Meta Store is used for storing the schema of the Hive tables. The detailed architecture of Hive is shown in Fig. 2.

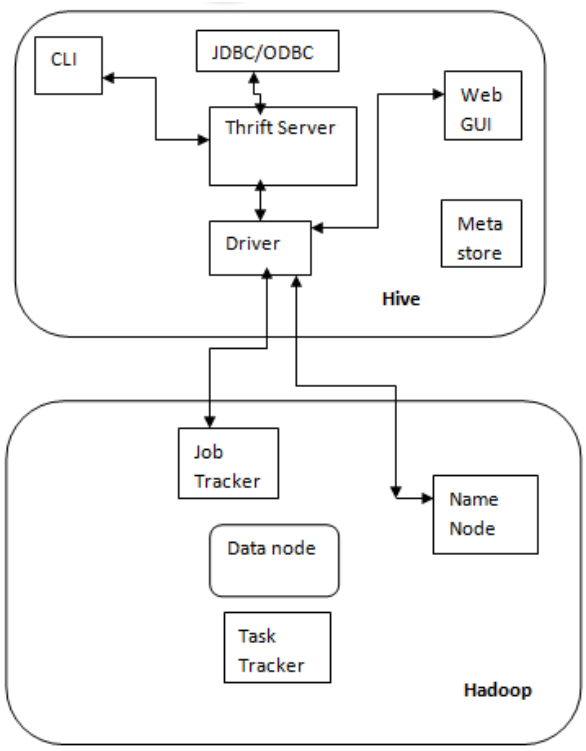

Fig. 2. Components of Hive

Hive is easy and interactive query language,persons who don't know the programming language also can easily writes the Hive queries for anlayzing the big data.The syntax and queries of Hive is similar to SQL.The following examples shows how to stores and reterives data into tables using Hive query language.

Creation of Table using Hive

```
create table if not exists research( reid int, rname string,rarea string, yearofjoining date)
    comment 'research details'
    row format delimited
    fields terminated by '\t'
    lines terminated by '\n'
    stored as textfile;
    .
```

Load Data into research table Using Hive

```
load data local inpath '/client/user/research.txt'
overwrite into table research;
```

Reterive Data from research table Using Hive

```
select * from research where rid=3;
```

The discussed examples show the normal way of storing and reterving the data without applying optimization. Optimization to the Hive increases the performance of queries. The

defaultway of creating tables doesnot provides optimization.

5 OPTIMIZATION OF HIVE QUERIES

For any type of data, performance of the queries becomes an important challenge. The queries which have long run execution on bigdata not only consumes resources of the system that makes the capability of server and application down. So, optimization of the queries becomes an important task.Hive without optimization is useful for the queries that requires scan of the entire table [5]. Even the queries run on the Hive needed limited amount of the data to be analyzed and processed. For this type of requirements users needed some domain knowledge on the attributes of the table and tell it to Hive. This requirement is possible through Partitioning of tables in the Hive. "Partitioning" is a feature that improves the performance of the queries. For some types of the attributes partitioning is not possible, then it is better to implement bucketing or clustering on the colums.The Column values which are same will be stored in one bucket. Bucketing is useful for the Joins operations. The following techniques improve optimization of hive queries and discussed with examples.

5.2 Hive Tables Creation Using Parttiion

Even it is possible to create tables in hive without partionning. Then the queries will scan entire table to get the result.By applying partition the records are stored in the separate folders then the queires fetch only required directories instead of fetching all. The following syntax indicates how to create partions in hive table.

Creation of Partition Table

create external table tablename(columnname datatype,column2 dattype,….)
partitioned by(column datatype)

The following example show creation of researcher table and partitioning the table based on the Researcher Joining Date.

create external table
researcher_info111 (rid smallint, rname string)
partitioned by (rjoindate date)

Loading of Data into Hive Tables through partition wise

load data local inpath
/home/user2/desktop/sample.txt'
overwrite into table r_info
partition (year='2016');

5.3 Hive Tables Creation Using Buketing

Bucketing [6] is another way of decomposing table in to manageable parts. The Bucketing is based on the hash function which improves the query performance. The following examples shows how to create table with bucketing concept

create external table

```
hive> CREATE TABLE ResearhcerInfo (
    >     name string,
    >     city string,
    >         R_id int )
    >     PARTITIONED BY (year  Date)
    >     CLUSTERED BY (R_id) INTO 256 BUCKETS  ;
OK
Time taken: 0.146 seconds
hive>
```

tablename(columnname datatype,column2 dattype,….)
partitioned by(column data type)

create external table
researcher_info (rid smallint, rname string)
partitioned by (rjoindate date)
clustered by (rid)into 256 bucketes.

5.4 ORC format for Storing

The Optimized Row Columnar (ORC) file format provides a an efficient way to store the data in the Hive database. It was designed to overcome the limitations of the other Hive file formats. When Hive is reading, writing, and processing data ORC files improves performance [7]

```
Createtable table_orc (
column1datatype,
column2datatype,
column3type,
column4type
) stored as orc;
```

5.3 TEZ instead of Map Reduce Engine

TEZ engine is more efficient than map reduce for interactive queries. To set the TEZ engine is possible by setting the following property.

set hive.execution.engine=tez

The use of TEZ engine is supporting for interactive queries along with single map phase followed by multiple reduce phases. But in map reduce always reducer reduce require map phase The response time of TEZ is efficient compare to map reduce due to lesser job splitting and HDFS access. In map reduce the task is divided in to more jobs and HDFS accessing is also more. TEZ does not write any temporary results in to HDFS.After completion of all map and reduce tasks only the final result is stored in to the HDFS.Coming to map reduce for map and reduce phases the temporary result is stored in the HDFS which is time consuming process.

The following points summarized points to optimize the Hive Queries
1. Create Hive Tables Using Partitions
2. Create Hive Tables using Bucketing Concept
3. Store the table in ORC format
4. Use TEZ engine instead of Map Reduce engine

6 EXPERIMENTAL SETUP

We implemented Hive queries using Ubuntu virtual machine with Hadoop and Hive. The evaluated results are shown below.

Creation of ResearcherInfo using Partition and Bucketing

Loading of Data

```
hive> LOAD DATA LOCAL INPATH '/home/lalitha2/Desktop/sample.txt'
    > OVERWRITE INTO TABLE Researcher_Info PARTITION (year='2016');
Copying data from file:/home/lalitha2/Desktop/sample.txt
Copying file: file:/home/lalitha2/Desktop/sample.txt
Loading data to table default.researcher_info partition (year=2016)
Partition default.researcher_info{year=2016} stats: [num_files: 1, num_rows: 0,
total_size: 79, raw_data_size: 0]
Table default.researcher_info stats: [num_partitions: 1, num_files: 1, num_rows:
 0, total_size: 79, raw_data_size: 0]
OK
Time taken: 0.79 seconds
hive>
```

```
hive> SHOW PARTITIONS Researcher_Info;
OK
year=2016
year=2017
Time taken: 0.069 seconds, Fetched: 2 row(s)
hive>
```

Show Partions of Researcher_Info

7 CONCLUSION

Hadoop is a framework that handles large amount of different formats of datasets using mapreduce,Pig,Hive etc.,In this paper, initially discussed architecture of Hadoop ,characteristics and application areas of Big Data with supported framework tools.Later,the detailed description and architechture of Hive is described with CLI,Web / GUI,ThriftServer,JDBC / ODBC,Meta Store component.Hive a toplevel Hadoop project that process vast amount of structured data using Hive query language.Hive is useful for the queries which require to scan entire table.In some applications,queries doesnot require to scan the entire table.In this situation ,table creation using partitions and bucketing redcues the scanning time of the tables and improves the efficiency of the queries.In this paper practically implemented creation and insertion of data using partions and bucketing concept. The file formats also affects the effieciency of Hive.ORC file format is an efficient format for Hive storage. After this, described the objective of using TeZ engine instead of Mpa reduce engine is briefly described .Finally the experimental results are evaluated using Ubuntu Virtul Machine.

REFERENCES

[1] Rupender Singh *, "Analyzing Performance of Apache TEZ and Map Reduce with Hadoop multi node cluster on Amazon Cloud" Journal of Big Data (2016) 3:19 DOI 10.1186/s40537-016-0051-

[2] Dr. S.Rama Sree,K.DeviPriya "An Insight of Big Dat Analytics Using Hadoop" article CSI Communications Volume no.40 issue No 8 November 2016

[3] Swati Harinkhere et.al "Big Data Processsing Techinques &Applications :A Technical Review" article CSI Communications Volume no.40 issue No 8 November 2016

[4] Ashish Thusoo et.al "Hive- A Petabyte Scale Data Warehouse Using Hadoop" 2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)Year: 2010

[5] http://hadooptutorial.info/partitioning-in-hive/#Syntax

[6] http://www.hadooptpoint.com/hive-buckets-optimization-techniques/

[7] Yib Huail et.al "Major Technical Advancements in Apache Hive" Proceedings of the 2014 ACM SIGMOD InternationaConference

Mrs. K. Devi Priya (CSI - F8000838) is working as a Senior Assistant professor in the department of Computer Science and Engineering, Aditya Engineering College, Surampalem, Andhra Pradesh. She has 7 years of teaching experience. She taught several subjects such as Web Technologies, Mobile Computing, Database Management Systems, Big Data Analytics, Hadoop and Big Data. Her research interests include Big Data Analytics, Cloud Computing, Network Security. She can be reached at k.devipriya20@gmail.com

Mr V. Chandra Kumar is currently working as Assistant Professor in the department of Computer Science and Engineering, AIMS College, Mummidivaram. He completed M.Tech from JNTU, Kakinada. He has 7 years teaching experience . He taught several subjects webtechnologies, mobile computing, Java, Computer Organization, Computer Graphics etc.,His research interests includes Mobile Computing,Image Processing,Design and Analysis of Algorithms ,Big Data Analytics etc., He can be reached chandrakumar.vichuri@gmail.com