Big Data Analytics using Hadoop Map Reduce Framework and Data Migration Process

Payal M. Bante
Department of Computer
Pimpri Chinchwad College of Engineering
Pune, India
Payalbante93@gmail.com

Dr. K. Rajeswari
Department of Computer
Pimpri Chinchwad College of engineering
Pune, India
Raji.pccoe@gmail.com

***Abstract*—Database are increasing in tremendous speed, volume (terabyte to petabyte), and types (variety of Data) becoming more complex. Managing such big data has turn out to be the comprehensive challenge. To conquer this problem, Migration of Data from MySQL to NoSQL and bigdata processing performs through a programming concept identified as Hadoop MapReduce. This paper provides methodology for Migration of data from relation to NoSQL (MongoDB) database and bigdata analytics using Hadoop Map Reduce framework above Hadoop Distributed File System (HDFS). Also Sqoop is used for migrating data from Relational Database to Hadoop for analytics process. Hive is introduced for migrate analyzed data from Hadoop to MongoDB. All Experiments are performed on Four different datasets- Loan Database, Connect 4 dataset, Lenses dataset and player tennis dataset form UCI Repository and KAGGLE repository.**

Keywords—MySQL, NoSQL, MongoDB, HDFS, Hadoop, Big Data, Analytics, Hive, Sqoop.

I. INTRODUCTION

Bigdata is a well-known term use to explain structured and unstructured data. Bigdata might be essential to industry and civilization as the Internet has suit. Bigdata is so huge that it's tricky to procedure using conventional database and software techniques. Additional data may guide to further perfect analyse. Further precise analyse may guide to additional definite decision making. And improved results can represent bigger functioning efficiencies, cost dropping and compact risk. Analyzing bigdata is most challenging for researchers that require particular analyzing procedures. Bigdata analytics [5] is the method of investigative bigdata to discover unseen patterns, unrevealed associations and other practical in sequence that can be used to build improved choice. Bigdata analytics consigns to the procedure of assembling, systematize and analyzing great sets of data ("big data") to determine patterns and former helpful information. Not just bigdata analytics facilitate to recognize the information restricted inside the data, but it will in addition facilitate recognize the data that is on the whole imperative to the production and outlook business choice. Bigdata analysts essentially want the facts that come from analyzing the data. Hadoop [11] is stands on a straightforward data model; any kind of data is robust. HDFS intended to embrace exceedingly huge quantity of data, and make available high-throughput admission to this information. Sqoop[10] (SQL-to-Hadoop) is bigdata tools that offer the ability to remove data on or after non-Hadoop data stores, alter the data addicted to a figure functional by Hadoop, and then fill the data inserted in HDFS. This procedure is called ETL [10], (Extract, Transform, and Load). Hive is a data warehousing infrastructure assembles over Hadoop [11]. It supplies an SQL vernacular, called Hive Query Language (HQL) for querying data accumulates in a Hadoop cluster. Similar to all SQL vernaculars in extensive employ, it does not completely be traditional to individual exacting amendment of the ANSISQL standard. It is possibly neighbouring to MySQL vernacular, but with important dissimilarity. This paper proposes a Methodology for Migrating Relational Database to NoSQL Data base and Big Data Analytics on Migrated data using C4.5 Hadoop Map Reduce Algorithm. Sqoop Connector is used for Migration process from Relational database to Hadoop. Hadoop distributed file system is used for Storage. C4.5 [2] Hadoop Map Reduce algorithm is used for processing on migrated unstructured data. Hive [11] is used to store aggregated processing data in MongoDB [2]. This paper also provides implementation with big dataset for given methodology.

II. RELATED WORK

In recent few decades huge amount of growth is observed on internet (social sites). Today's existing globe is united by social networking websites. Such sites encompass enormous sizes of databases. Given reassess in this part demonstrates that not several approaches that offer tools or utilities for database adaptation. Leo. Rocha in [1] proposed argues Migration and Mapping module in which Framework is additional resourceful than using only MySQL also NoSQL Layer is a clarification appropriate to handle great quantity of data. In [4] Ying-Ti Liao gives Data adapter converting among SQL along with NoSQL database Data Adapter, BT mode, BD mode and DA mode which offers A lithe tic and greatly modularized data adapter. Reins query flow during database transformation. Ayman Lotfy in [3] describes an A core layer clarification to maintain ACID properties meant for NoSQL DB's by using ACID Properties Four Phase Commit Protocol which gives Concurrent transactions Increase Scalability throughput. Mohmad Hesham El-Deeb Khaled in [5] reviews and describes group of decision tree classification algorithms. Wei Dai in [2] intend at giving - classification, characteristics

978-1-5386-4008-1/17/$31.00

and assessment of NoSQL DB's within Bigdata Analytics. C4.5 Algorithm is time Consuming Minimize Communication Cost. Though here no any available efficacy converting NoSQL DB's into SQL DB's, Sanobar Khan [6] attempts to utilize NoSQL DB [6] Attempts to utilize NoSQL DB to restore the relational database. It highlights boosting equipment of NoSQL DB for and most popular example is MongoDB, and it makes featured comparison among MySQL and NoSQL which shows how and why NoSQL is preferred over MySQL. Here is no any tool or utility is available which can convert SQL Database into NoSQL. As NoSQL is a new trend in today's technical era, here is no software or tool has considered for the desired adaptation. On the whole of the utilities residential were developed earlier for adapt one structure of SQL DB to an added SQL database

III. .PROPOSED SYSTEM

In proposed system methodology to migrate data from Relational database to unstructured database (Hadoop) by using SQOOP [11] is given. Hadoop is use for storage as well as processing while relational and NoSQL databases are use only for storage hence Hadoop is introduced. Migrated data is store in HDFS (Hadoop Distributed File system). Big Data analytics is performed on these migrated unstructured data on HDFS using Map-Reduce Framework [2]. Here C4.5 Algorithm is used for data analysis. After successful big data analytics data analyzed data will be migrated to MongoDB (NoSQL database) using Hive by creating JSON objects because in MongoDB Data can be store only in the form of JSON objects. Flow of proposed system is given in Figure 1.

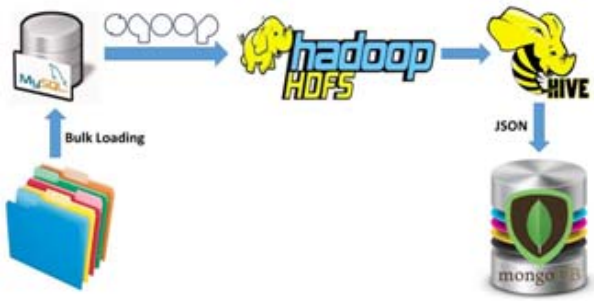

Figure 1: Proposed System

A. SQOOP

Sqoop is a tool intended for economically move mass data in a distributed style among relational databases such as MySQL to HDFS and vice versa. Sqoop bring in the data starting RDBMS to HDFS, transform data in Hadoop MapReduce, and afterwards export the data reverse into an RDBMS. It automates mainly of this procedure, Sqoop utilize MapReduce for import-export process which offers parallel operation and fault-tolerance. Figure 2 gives the architecture of Sqoop in which import-export method is specified.

B. HADOOP 2.X

Hadoop2 architecture is given in Figure 3. Hadoop2 [11]

Architecture has mainly 2 set of daemons.

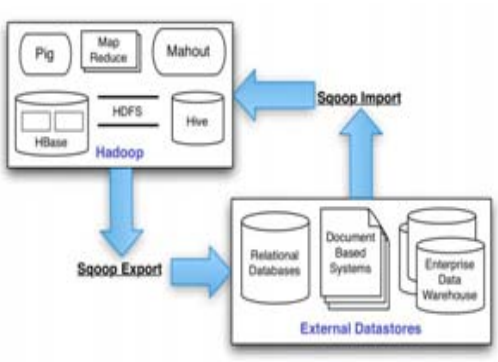

Figure 2 Architecture of SQOOP

- HDFS 2.x Daemons: Hadoop 2.x [11] permit numerous Name-Nodes for HDFS alliance. Recent Architecture consent to HDFS High accessibility manner in which it be able to contain dynamic and Stand by Name-Nodes.
- MapReduce 2.x Daemons (YARN): MapReduce2 have replaced older daemon procedure Job Tracker and Task Tracker with YARN [11] mechanism Resource-Manager and Node- Manager correspondingly. These two mechanisms are accountable for implementing dispersed data multiplication trade in Hadoop2.x

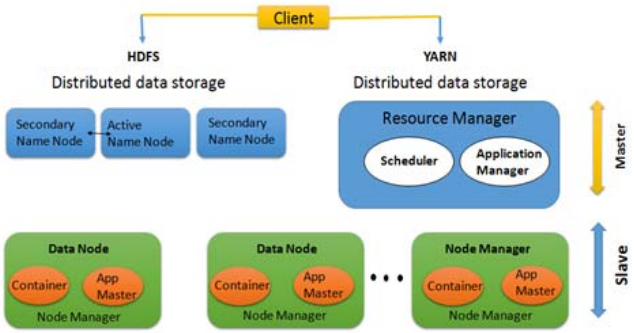

Figure 3: Architecture of Hadoop 2.X

1. Resource Manager: These daemon lopes on master node. It is liable for accomplishment job yield from client and schedules it on top of cluster; examine operation jobs on cluster and assigning suitable assets on the slave-node[2]. It communicates with Node-Manager on the slave node to trail the reserve consumption. It utilize two former process named as Application-Manager and scheduler.

2. Node Manager: This process runs on the slave nodes. It is accountable for synchronize with Resource Manager for task scheduling and follow the source consumption on the slave node. It also rumour the resource consumption back to the Resource Manager. It uses former daemon process similar to Application Master and Container for MapReduce task scheduling and execution on the slave node [2].

C. HIVE

Hive is a data warehousing infrastructure for Hadoop [2]. The key task is to offer data summarization, question and examination. It chains analysis of huge datasets accumulates in HDFS plus on the Amazon-S3 file system. The finest division of HIVE is that it chains SQL-Like admittance to structured data which is identified as HiveQL (or HQL) as well as big data analysis with the help of Map Reduce [2]. Hive is not built to get a quick response to queries but it it is built for data mining applications. Data mining applications can take from several minutes to several hours to analysis the data and HIVE is primarily used there. Architecture of hive is given in Figure 4.

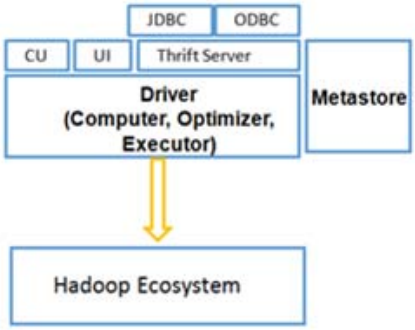

Figure 4: Architecture of HIVE

IV. DESIGN AND IMPLEMENTATION

A. C4.5 Algorithm

C4.5 [4] is standard algorithm, which induce the classification rule in the form of decision trees. As an expansion of ID3 [5] the evasive criteria to choose split attribute in C4.5 algorithm is information gain ratio. As an alternative to use information gain ratio in ID3, it avoids the bias of selecting many attributes with other values in C4.5. Map-Reduce Function Mapper function consist of Key- value pair and checks whether this instance belongs to Current Node or not while Reduce function counts number of occurrences of combination of (index and its value and class Label) and prints count against it. Steps in MapReduce in implementation of C4.5 algorithm are as follows:

1. Current Node is assumed for splitting.
2. Map (key, value){
3. Check whether this instance belongs to Current Node or not.
4. For all uncovered attributes it outputs index and its value and class label of instance. }
5. Reduce (key, value) {
6. Count number of incidence of grouping of and gives count against it. }

C4.5 [2] is a algorithm based classification rules which is demonstrate in the decision tree. It is an modified version of ID3 [5], information gain ratio is the evasion technique of choosing ripping attributes in C4.5. As a replacement for information gain ratio ID3 is used. Key aspect of information gain ratio is that it ignores the prejudice decision of picking attributes with numerous former values.

B. Mathematical Model

Mathematical model for C4.5 with information gain ratio and split value is as follows. The entropy [2] of given attribute value is computed as:

Entropy (S) = -P (S, j) x log P (S, j)

Consequently, the information gain by a preparation dataset is the subset of T persuade by S , and T(s, v) , is subset of T with attribute value S has a value of v .

Thus, the information gain ratio [2] of attribute S is:

Gain Ratio (S, T) = Gain (S, T) / SplitInfo (S, T) .

SplitInfo value [2] is estimated as:

SplitInfo (S, T) = -|Ts, v| / |Ts | x log |Ts, v| / |Ts |

To assemble a general choice tree first we need to consider preparing informational index, on all property in then given preparing informational index we will concern an estimation work with the reason for locate a best split trait. After traits are get hold of split and built up, the case space is parcelled excited about various parts. Preparing example among each parcel they all be in the correct place to one just class, ensuing to that the calculation finishes up. If not the part procedure will be recursively performed in foresight of the entire segment is committed to the indistinguishable given class. At extreme when a choice tree is delivered, arrangement principles can be produced effectively, a short time later use for order of new class marks.

V. EXPERIMENTAL RESULTS

This Section evaluates the performance of implementation of proposed methodology i.e. big data analytics on migrated data into Hadoop using C4.5 algorithm and Hadoop map reduce frame work and migration from Relational to NoSQL (MongoDB).

In this Experiment Hadoop cluster deployed on single PC with 2.20 GHz dual-core CPU, 4G RAM and 400G hard disk. Each core is used as a Hadoop node, and thus it contains one node. On physical core, both a HDFS and MapReduce nodes are deployed. Each HDFS contains Resource Manager and node manager. Each Resource manager contains scheduler and application manager respectively Name node on node manager. For the Implementation we have used Tennis Dataset from UCI with total six attributes and seven million instances. Because of Massive dataset we are using 230 MB of data for each Node. Size of Each data node is 245 MB.

A. Performance on Single Node

In subsection provides Graphical Representation for Different datasets, multiple instances and comparison between Original C4.5 Algorithm and C4.5 Map Reduce. There are two

main conditions, first, the larger the dataset is, the more time consuming it is to build the decision tree. Second, the execution time of our MapReduce based algorithm is much less than the original C4.5 algorithm as the size of dataset increases. Therefore, it is proved that our proposed method outperforms the sequential version even on a single node environment.

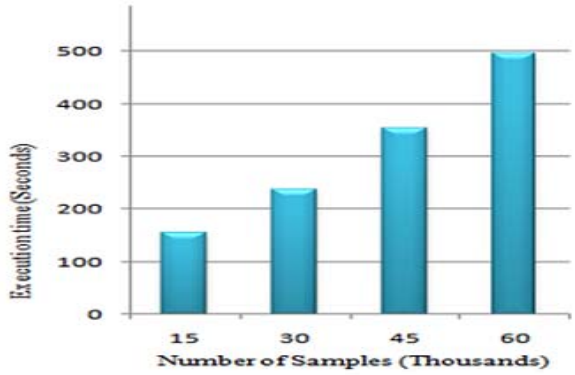

Figure 5: Performance on single node for Multiple Instances

Figure 5 shows Execution Time for Different Size of Sample Datasets. Given graph shows:

1. The larger the training dataset it use, the more cost of execution time;

2. If enough nodes are leveraged, even the size of dataset is big; the performance can be close to the optimal one.

With this graph it shows the Flexibility of system. When different Types of datasets is use with multiple instances all kind of datasets can be processed, classified, Migrated and analysed successfully.

Form graphical interpretation shows that As number of instances increases, time increases but it is less as compare to the original C4.5 algorithm when it use Hadoop Map Reduce framework it takes less time because of parallel processing.

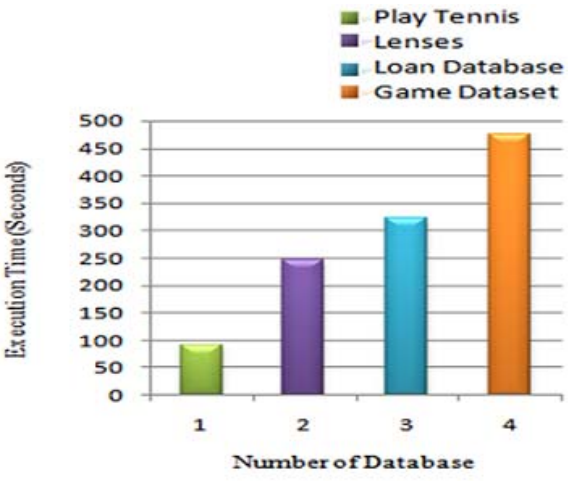

Figure 6: Performance on single node for Difference Datasets

Figure 6 illustrates the results, which shows the following observations. First, the larger the dataset is, the more time consuming it is to build the decision tree. Second, the execution time of our MapReduce based algorithm is much less than the original C4.5 algorithm as the size of dataset increases. Therefore, it is proved that our proposed method outperforms the sequential version even on a single node environment.

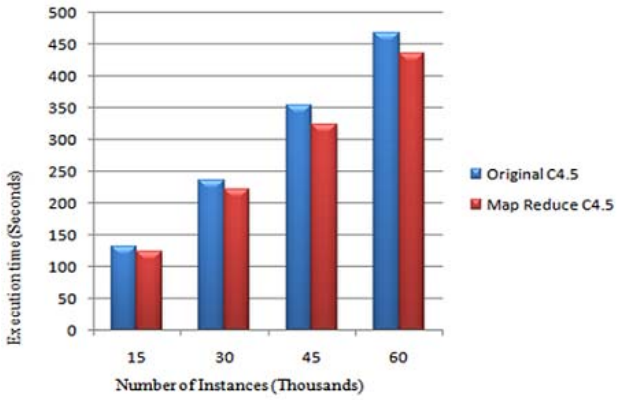

Figure 7: Performance on single node with Original C4.5 and Map Reduce (Accuracy)

Figure 7 Shows that Original C4.5 algorithm takes more time as compare to the Map Reduce C4.5 algorithm on single node. This system is flexible for all type of databases as given in Figure 29 there are total four different datasets. Accuracy is calculated for original C4.5 algorithm and Map Reduce C4.5 algorithm on single node.

VI. CONCLUSION

In this paper, a Methodology for Bigdata analytics with Map-Reduce Framework and migration is given. The motivation behind this is more and more growth of cloud computing and bigdata, traditional decision tree algorithms can't robust any longer. Such as, when the volume of training data raises, the method of creating decision trees can be extremely time overwhelming. Moreover, the amount of dataset enlarges, the algorithm contains a high rate on I/O operation and data can't vigorous in memory. To resolve above problems, we thus offer C4.5 based on MapReduce. Using Sqoop Data is migrated from Relational Database to Hadoop. After successful completion of data analytics data is migrated from Hadoop to MongoDB using Hive. In order to evaluate the efficiency of our method, we have conducted experiments on a Tennis massive dataset from UCI. The empirical outcome specifies that our MapReduce implementation exhibit time efficiency. In future works, Experiment will perform on Multiple Nodes (Hadoop cluster) and find Scalability for single and multiple nodes.

References

[1] Leo. Rocha ,Fernando Vale, Elder Cirilo, "A Framework for Migrating Relational Datasets to NoSQL ", ICCS

International Conference On Computational Science (Science Direct), Vol. 21, No. 5,pp. 2013-2024, 2015

[2] Wei Dai,Wei Ji*2 "A MapReduce Implementation of C4.5 Decision Tree Algorithm ", International Journal of Database Theory and Application Vol.7, No.1 , pp.49-60, 2014

[3] Ying-Ti Liao Jiazheng Zhoua Chi-aung Lua, "Data adapter for querying and transformation between SQL and NoSQL database" , Future Generation Computer Systems 65 (Science Direct), vol. 10, No.8, pp 54-67, 2011

[4] Ayman E. Lotfy ,Ahmed I. Saleh,Haitham A., "A middle layer solution to support ACID properties for NoSQL databases" ,Journal of King Saud Computer and Information Sciences vol. 28, pp. 133145,2016

[5] Mohmad,Hesham Mohamed ,El-Deeb Khaled Badran, "Suite of decision tree-based classication algorithms on cancer gene expression data",Egyptian Informatics Journal vol 12, pp. 7382, 2011

[6] R. GAIOSO, F. LUCENA, and J. SILVA. "Integrate: Infrastructure para integracao de fontes dedados heterogeneas." Master Thesis, Federal University of Goias. Informatic Institute, 2007

[7] Seth Gilbert and Nancy Lynch. "Brewers conjecture and the feasibility of consistent, available, partition-tolerant web services". SIGACT News, vol. 33 no.2 pp. 5159, June 2002.

[8] Roberto Ierusalimschy, Luiz Henrique de Figueiredo, and Waldemar Celes Filho. Luaan, "Extensible extension language" Softw. Pract. Exper., vol. 26 no. 6 pp.635652, June 1996.

[9] R. P. Padhy, M. R. Patra, and S. C. Satapathy. Rdbms to nosql, "Reviewing some next-generation non-relational databases.", International of Advanced Engineering Science and Technologies, vol. 11 no.1, September 2011.

[10] J. Han and M. Kamber, Data Mining Concepts and Techniques,Elevier,2011

[11]www.hadoopApache.com