ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

# A Database Hadoop Hybrid Approach of Big Data

Rupali Y. Behare [#1], Prof. S.S.Dandge [#2]

M.E. (Student), Department of CSE, Department, PRMIT&R, Badnera , SGB Amravati University, India[1].

Assistant Professor, Department of CSE, Department, PRMIT&R, Badnera, SGB Amravati University, India[2]

**ABSTRACT**: Big data is a popular term used to describe the exponential growth and availability of data, both structured and unstructured. Big data may be important to business and society as the Internet has become. Big Data is so large that it's difficult to process using traditional database and software techniques. Big data analytics refers to the process of collecting, organizing and analysing large sets of data ("big data") to discover patterns and other useful information Systems. Hadoop is based on a simple data model, any data will fit. HDFS designed to hold very large amounts of data (terabytes or petabytes or even zettabytes), and provide high-throughput access to this information. Hadoop Map Reduce is a technique which analysis big data. MapReduce has recently emerged easy programming model. In this work by giving the idea from HDFS (Hadoop Distributed File System) developed distributed system. and find the processing time between Hadoop based system and non Hadoop based system and compare them. Implement efficient algorithm for developing distributed system and map reducing functions.

**KEYWORDS**: machine learning algorithm, Hadoop, HDFS, Map Reduce, UDTF.

## I. INTRODUCTION

Data creation is occurring at an unprecedented rate. In 2010, the world generated over 1ZB of data and by 2014, generated 7ZB of data [1]. Big Data means Data sets whose volume and variety is beyond the ability of commonly used software tools to capture, manage, and process the data within a tolerable elapsed time that is relevant to business. The difficulty can be related to data capture, storage, search, sharing, analytics and visualization etc. Just as this data is generated by people in real time, it can be analyzed in real time by high performance computing networks, thus creating a potential for improved decision-making [6]. Hadoop is the core platform for structuring Big Data, and solves the problem of making it useful for analytics purposes. Hadoop MapReduce is an implementation of the algorithm developed and maintained by the Apache Hadoop project. Map Reduce is a programming model for processing large data sets with parallel distributed algorithm on cluster.

## II. LITERATURE REVIEW

1. Big Data

D.Usha and Aslin Jenil proposed that (2014), Big Data is data whose scale, diversity, and complexity require new architecture, techniques, algorithms, and analytics to manage it and extract value and hidden knowledge from it.

Characteristics of Big Data:

Volume of data-Volume refers to amount of data. Volume of data stored in enterprise repositories have grown from megabytes and gigabytes to petabytes.

Variety of data -Different types of data and sources of data. Data variety exploded from structured and legacy data stored inenterprise repositories to unstructured, semi structured, audio, video, XML etc.

Velocity of data-Velocity refers to the speed of data processing. For time-sensitive processes such as catching fraud, big data must be used as it streams into your enterprise in order to maximize its value.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

2. Hadoop

By Markus Maie, Hadoop is the Apache Software Foundation open source and Java-based implementation of the Map/Reduce framework. Hadoop was created by Doug Cutting, the creator of Apache Lucene2, the widely used text library.

Characteristics of Hadoop

Scalable–As required new nodes can be added without changing the data formats, the way which data is loaded, way the jobs or application are written.

Cost effective– Hadoop brings massively parallel computing to commodity servers. The output is a sizeable decrease in cost which in turn makes it affordable to model all the data.

Flexible– Hadoop is not schema oriented so it can absorb any type of data, structured or not, from number of sources. Data from multiple sources can be joined and aggregated in arbitrary ways enabling deeper analysis than any other system can provide.

Fault tolerant– When a node was lost, the system redirects work to another location of the data and continues processing without missing a beat [2].

Hadoop consists of 2 components:

2.1 HDFS

D.Usha and Aslin Jenil stated that after the data is loaded into clusters in Hadoop it is distributed to all the nodes. The HDFS then splits the data into sets which allow management by individual nodes within the cluster [2].

By Dr. Siddaraju, Sowmya C L, Rashmi K, Rahul , HDFS is designed to run on clustered computing platform. One of the salient features of HDFS is that it is fault-tolerant to a very high degree and cost effective. The system allows for greater and faster access to data of an application which is an advantage for processes that require access to large amount of data [6].

2.2 Map Reduce

By Colin Whit (2012) MapReduce is a technique popularized by Google that distributes the processing of very large multi-structured data files across a large cluster of machines. High performance is achieved by breaking the processing into small units of work that can be run in parallel across the hundreds, potentially thousands, of nodes in the cluster. MapReduce is a programming model, not a programming language, i.e., it is designed to be used by programmers, rather than business users [9].

2.3 Hadoop Architecture

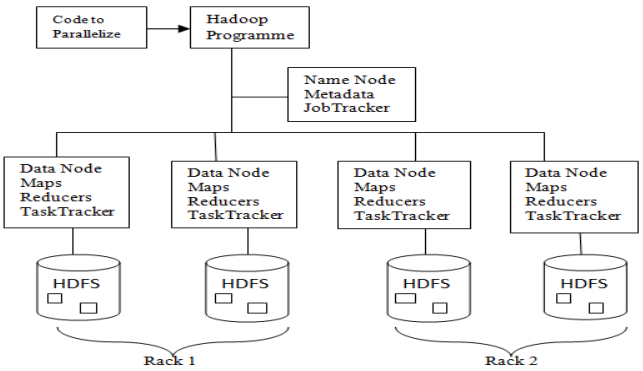



Fig. 2.3 Hadoop Architecture

Figure 2.3 shows the Architecture of Hadoop. HDFS is one of the primary component of Hadoop cluster and it is designed like a Master-slave architecture. The Master (NameNode) manages the file system namespace operations like opening, closing, renaming files and directories and also determines the mapping of blocks to DataNodes along with

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

regulating access to files by clients. Slaves (DataNodes) are responsible for serving read and write request from the clients along with performing block creation, deletion, and replication upon instruction from the Master (NameNode). When a client makes a request of a Hadoop cluster, this request is managed by the JobTracker. The JobTracker, working with the NameNode, distributes work as closely as possible to the data on which it will work. The NameNode is the master of the file system, providing metadata services for data distribution and replication. The JobTracker schedules map and reduce tasks into available slots at one or more TaskTrackers. The TaskTracker working with the DataNode (the slave portions of the distributed file system) to execute map and reduce tasks on data from the DataNode. When the map and reduce tasks are complete, the TaskTracer notifies the JobTracker, which identifies when all tasks are complete and eventually notifies the client of job completion. Master {Jobtracker} is the point of interaction between users and the map/reduce framework. When a map/reduce job is submitted, Jobtracker puts it in a queue of pending jobs and executes them on a first-come/first-served basis and then manages the assignment of map and reduce tasks to the tasktrackers. Slaves {tasktracker} execute tasks upon instruction from the Master {Jobtracker} and also handle data motion between the maps and reduce phases [2].

## III. RELATED WORK

Hadoop is Java framework and here giving the idea from Hadoop, developed distributed system using C#. For that two algorithms mainly used which is Distributed Gradient Descent used for making distributed system and other is iterative parameter mixtures for mapping and reducing that distributed node. Because of distribution of data processing goes very fast and using map reduce programming get easy to perform. So efficiency of system gets increases. Also used data encryption and data decryption algorithm for security of data so original form of data never is seen by user. Also UDTF (user defined table generating function) used for displaying output. UDTF is used to generate more than one row output from one row input. For this project developed website for giving the detail information about particular product. In which user can login and add the information about products. Also Admin can add or delete product data using admin login function. In this, number of system connected in LAN. User details are stored in one system and product details are stored in another system. i.e. both these system are Hadoop based system and combine information about user detail and product information also stored in third system, which is non hadoop based system. And objective is that Calculate the processing time between Hadoop based system and non Hadoop based system.

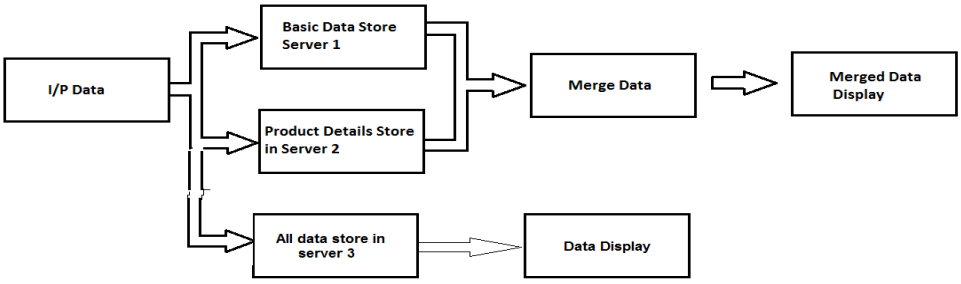



Figure 3.1 Block diagram of implemented system

In block diagram of implemented system used three numbers of server/machines. Where in first server gives the input from user and distribute that data in two systems i.e. in server1 stores the basic information of user also login information and advertisement information and in server two stores the information about services and categories. And in server3 stores the all combine information from server1 and server2. Then by combination of server1 and server2 find the processing time of both machines. Also find processing time of server3.

## IV. PROPOSED ALGORITHM

Algorithm Distributed Gradient Descent and Iterative parameter mixture

Use Iterative parameter mixture and Distributed Gradient Descent algorithm for distribution and mapping of data. Following are the pseudo code for distributed gradient descent and iterative parameter mixture.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

$w = \emptyset$
$t = 0$
**repeat**
  $S = \{D^1, D^2, ..., D^K\}$ — *shuffle training data S into K partitions*
  **for** $j = 1 \ldots K$ **do** {run on mappers.}
    $w_0^j = w_{t-1}$ — *mix parameters for each epoch*
    **for** $i = 1 \ldots |D^j|$ **do** {for each instance, update weights based on gradient}
      $w_i^j = w_{i-1}^j - \gamma^t \nabla F_{D_i^j}\left(w_{i-1}^j\right)$
  **for** $f = 1 \ldots |w|$ **do** {for each feature in the model, aggregate on reducers}
    $w^{t+1}(f) = \frac{1}{K}\sum_{j=1}^{k} w_{|D^j|}^j(f)$ — *calculate average of feature weights*
  $t = t + 1$
**until** *converged*;

Algorithm 1: Pseudo code for Distributed Gradient descent and iterative parameter mixture [7].

Algorithm for Data Encryption.

Algorithm for data encryption and data decryption is used to security of data. so original form of data cannot be displayed by any types of user.

INPUT: Plaintext (StrValue), Key (StrKey).
OUTPUT: Ciphertext (EncryptedData).

1: Add the key to Text (StrKey + StrValue)---> full string (StrFullVlaue).
2: Convert the Previous Text(StrFullVlaue) to ascii code (hexdata).
3: Foreach (byte b in hexdata).
a. Convert the Previous ascii code (hexdata) to binary data (StrChar).
b. Switch (StrChar.Length).
Case 7 ---> StrChar = "0" + StrChar.
Case 6 ---> StrChar = "00" + StrChar.
Case 5 ---> StrChar = "000" + StrChar.
Case 4 ---> StrChar = "0000" + StrChar.
Case 3 ---> StrChar = "00000" + StrChar.
Case 2 ---> StrChar = "000000" + StrChar.
Case 1 ---> StrChar = "0000000" + StrChar.
Case 0 ---> StrChar = "00000000" + StrChar.
c. StrEncrypt += StrChar. (where, StrEncrypt= "")
4: Reverse the Previous Binary Data(StrEncrypt).
5: For i from 0 to StrValue.Length do the following:
a. if (binarybyte.Length == 8).
i.Convert the binary data (StrEncrypt) to ascii code and,
ii.Divide the ascii by 4 □ □ the result(first character) and,
iii.The remainder of the previous □ □ second character.
6: Return (EncryptedData).

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

Algorithm for Data Decryption.

INPUT: Ciphertext (EncryptedData), the Key (StrKey).
OUTPUT: Plaintext (DecryptedData),

1: For (i = 0; i < EncryptedData.Length; i += 2)
a. Get the ascii code of the encrypted text
b. newascii = (EncryptedData[i] * 4) + the next digit(remainder)[i+1].
2: Foreach (byte b in newascii).
a. Convert the Previous ascii code (newascii) to binary data (StrChar).
b. Switch (StrChar.Length).
Case 7 ---> StrChar = "0" + StrChar
Case 6 ---> StrChar = "00" + StrChar.
Case 5 ---> StrChar = "000" + StrChar.
Case 4 ---> StrChar = "0000" + StrChar.
Case 3 ---> StrChar = "00000" + StrChar.
Case 2 ---> StrChar = "000000" + StrChar.
Case 1 ---> StrChar = "0000000" + StrChar.
Case 0 ---> StrChar = "00000000" + StrChar.
c. StrDecrypt += StrChar.
3: Reverse the Previous Binary Data(StrDecrypt).
4: For i from 0 to StrDecrypt.Length do the following:
a. if (binarybyte.Length == 8).
i. Convert the binary data (StrChar) to ascii code (hexdata) and,
ii. Convert the previous ascii code (hexdata) to the text (StrFullVlaue).
5: Remove the key from the text (StrFullVlaue - StrKey) (StrValue).
6: Return (DecryptedData).

## V. EXPERIMENTAL RESULTS

- Requirement Analysis:

For implementation of this system, .Net technology is used. A main part of the .Net technology and structure is the ASP.net set of technologies. These web development technologies are used in the making of Websites and net services working on the .NET infrastructure. ASP.NET was billed by Microsoft from one of their big technologies and web programmers can make use of any encoding language they want to write ASP.NET, from Perl to C Sharp (C#) and of course VB.NET and a few extra language unspoken with the .NET technology.

*1) Hardware Requirements*
Windows XP, RAM – 1GB, Hard Disk - 20GB

*2) Software Requirements*
-.Net Framework
- SQL Server 2005

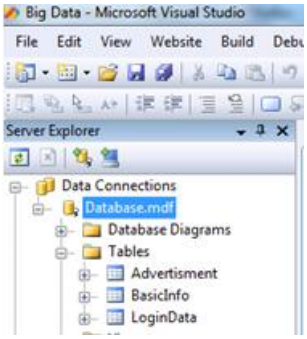
- Result (Screenshots):

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

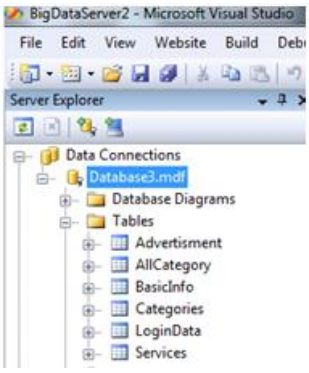



Screenshot 1: Showing number of table stored in server 1

Screenshot 1 shows that number of tables stored in machine 1 which is, table of Advertisement, Basic information about user and login data.


Screenshot 2: Showing number of table stored in server 2

Screenshot 2 shows that number of tables stored in machine 2 which is, table about number Services available, Number of Categories of services and table of details information about categories.




Screenshot 3: Showing number of table stored in server 3

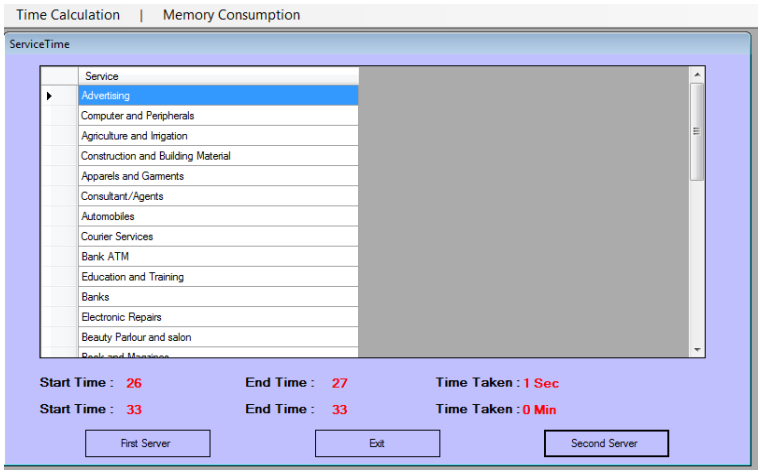
Screenshot 3 shows that number of tables stored in machine 3 which is, table about number Services available, Number of Categories of services and table of details information about categories also table of Advertisement, Basic information about user and login data.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

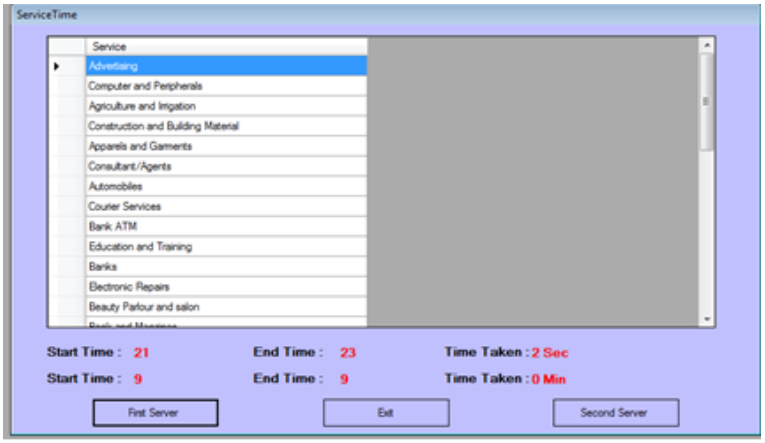
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**




Screenshot 4: Showing processing time required for server 2 which is non Hadoop based

Screenshot 4 shows that processing time required for machine 2 or server 2, which is Hadoop based distributed machine.




Screenshot 5: Showing processing time required for server 1 which is Hadoop based

Screenshot 5 shows that processing time required for machine 1 or server 1 which is non Hadoop based machine.
So, experimental results shows that processing time required for Hadoop based distributed system is less than simple non Hadoop based system.

## VI. CONCLUSION AND FUTURE WORK

In this system developed a distributed system and by comparing the processing time between Hadoop based system and non Hadoop based system, conclude that Hadoop based system required less time. So, efficiency of machine gets increases. Here, used very small network of 3 nodes, as number of nodes increases the complexity will increase. In future, can increase the number of nodes and analyze the performance.

## REFERENCES

1. Sangeeta Bansal, Dr. Ajay Rana" Transitioning from Relational Databases to Big Data," ijarcsse ,Volume 4, Issue 1, January 2014 ISSN: 2277 128X.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 5, May 2015**

2. D.Usha and Aslin Jenil , "A Survey of big data processing in perspective of hadoop and mapreduce", Vol.4, No.2 (April 2014) , E-ISSN 2277 – 4106, P-ISSN 2347 – 5161.
3. Asha T, Shravanthi U.M, Nagashree N, Monika M(2013), "Building Machine Learning Algorithms on Hadoop for Bigdata" Volume 3 No. 2, February, 2013,ISSN 2049-3444.
4. Hadoop,"PoweredbyHadoop," http://wiki.apache.org/hadoop/PoweredBy.
5. Jianqing Fan1, Fang Han and Han Liu, " Challenges of Big Data analysis", National Science Review Advance Access published February, 2014.
6. Dr. Siddaraju, Sowmya C L, Rashmi K, Rahul M ,"Efficient Analysis of Big Data using Map Reduce Framework", Volume 2, Issue 6, June 2014, ISSN 2347-6435.
7 Makoto YUI and Isao KOJIMA" A Database-Hadoop Hybrid Approach toScalable Machine Learning", 2013 IEEE International Congress on Big Data.
8. XIAO DAWEI, "Exploration on Big Data Oriented Data Analyzing and Processing Technology",vol.10, Issue 1,No 1,January 2013,ISSN (Online):1694-0814.
9. Colin White,BI Research(2012),"Map Reduce And Data Scientist",pg no.4,5.

## BIOGRAPHY

**Rupali Y. Behare** is a M.E. student in the Computer Science & Engg. Department, College of PRIT & Research Badnera, SGB Amravati University, India.

**Prof. S.S.Dandge** is a Assistant Professor in the Computer Science & Engg. Department, College of PRIT & Research Badnera, SGB Amravati University, India.