ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

EFFICIENT INDEXING AND QUERYING OF BIG DATA IN HDFS

Deepika.D,Pugazhmathi.D
Department of Computer Science and Engineering
A.R.J College of Engineering and Technology
Mannargudi
deepijoy674@gmail.com,pugalmathiarj@gmail.com

Abstract-The data mining is a computation process of discover large sets of data's. It extract information from dataset and transform into understandable structure for further user. Normally database process the structured format of data but unstructured data cannot perform by database. So the data mining process perform an unstructured data. The previous system of data mining is complex to process a large volume of data and cannot process with unstructured data. Because it perform a serial processing. So the data mining concept not efficient perform of large volume of data and increase the time frame process. So the proposed of paper using the Hadoop concept that perform the HACE theorem. First phase is collection of data set from large data set in a parallel processing manner and perform the indexing on collected data. The final phase using the MAP reduce is using to search a data collection in shortest path way from a collection of servers by depending on user search information and using indexing techniques to create common file system called HDFS file system using searchable related information by user searching data. So the proposed HACE theorem is perform large volume ofdata processing model and mining & analysis, security, privacy consideration.

Key terms*:*MapReduce,Big data,Hadoop,HDFS,HACE theorem.

IINTRODUCTION

A)OVERVIEW OF BIG DATA

a)What is Big Data? Big Data is a collection of large datasets that cannot be processed using traditional computing techniques. It is not a single technique or a tool, rather it involves many areas of business and technology.

b)What Comes Under Big Data? Big data involves the data produced by different devices and applications. Given below are some of the fields that come under the umbrella of Big Data.

- Black Box Data**:** It is a component of helicopter, airplanes, and jets, etc. It captures voices of the flight crew, recordings of microphones and earphones, and the performance information of the aircraft.
- Social Media Data**:** Social media such as Facebook and Twitter hold information and the views posted by millions of people across the globe.
- Stock Exchange Data**:** The stock exchange data holds information about the 'buy' and 'sell' decisions made on a share of different companies made by the customers.
- Power Grid Data**:** The power grid data holds information consumed by a particular node with respect to a base station.
- Transport Data**:** Transport data includes model, capacity, distance and availability of a vehicle.
- Search Engine Data**:** Search engines retrieve lots of data from different databases.

c) Big Data Technologies Big data technologies are important in providing more accurate analysis, which may lead to more concrete decision-making resulting in greater operational efficiencies, cost reductions, and reduced risks for the business.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

- To harness the power of big data, you would require an infrastructure that can manage and process huge volumes of structured and unstructured data in real-time and can protect data privacy and security.

d)Operational Big Data

These include systems like Mongo DB that provide operational capabilities for real-time, interactive workloads where data is primarily captured and stored. NoSQL Big Data systems are designed to take advantage of new cloud computing architectures that have emerged over the past decade to allow massive computations to be run inexpensively and efficiently[5]. This makes operational big data workloads much easier to manage, cheaper, and faster to implement. Some NoSQL systems can provide insights into patterns and trends based on real-time data with minimal coding and without the need for data scientists and additional infrastructure.

e) Analytical Big Data These includes systems like Massively Parallel Processing (MPP) database systems and MapReduce that provide analytical capabilities for retrospective and complex analysis that may touch most or all of the data. The data level is increasing and big data's are arrive in day by day. This paradigm entails harnessing large numbers of low-end processors working in parallel to solve a computing problem. In effect, this suggests constructing a data center by lining up a large number of low-end servers instead of deploying a smaller set of high-end servers. With this rise of interest in clusters has come a proliferation of tools for programming them. The rise of Big Data applications where data collection has grown tremendously and is beyond the ability of commonly used software tools to capture, manage, and process within a "tolerable elapsed time."

B) HDFS AND MAP REDUCE

a)Map Reduce MapReduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multiterabyte data-sets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The MapReduce program runs on Hadoop[14],[15] which is an Apache open-source framework.

b)Hadoop Distributed File System TheHadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets. Apart from the above-mentioned two core components, Hadoop framework also includes the following two modules:

- Hadoop Common**:** These are Java libraries and utilities required by other Hadoop modules.
- Hadoop YARN: This is a framework for job scheduling and cluster resource management.

c)How Does Hadoop Work? It is quite expensive to build bigger servers with heavy configurations that handle large scale processing, but as an alternative, you can tie together many commodity computers with single-CPU, as a single functional distributed system and practically, the clustered machines can read the dataset in parallel and provide a much higher. Moreover, it is cheaper than one high-end server. So this is the first motivational factor behind using Hadoop that it runs across clustered and low-cost machines. Hadoop runs code across a cluster of computers. This process includes the following core tasks that Hadoop performs:

- Data is initially divided into directories and files. Files are divided into uniform sized blocks of 128M and 64M (preferably 128M).
- These files are then distributed across various cluster nodes for further processing.
- HDFS, being on top of the local file system, supervises the processing.
- Blocks are replicated for handling hardware failure.
- Checking that the code was executed successfully.
- Performing the sort that takes place between the map and reduce stages.
- Sending the sorted data to a certain computer.
- Writing the debugging logs for each job.

1.2.4 Advantages of Hadoop

- Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatic distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
- Hadoop does not rely on hardware to provide fault-tolerance and high availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer.
- Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.

Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based.

d)Algorithm of Map Reduce

- Generally MapReduce paradigm is based on sending the computer to where the data resides!
- MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

- Map stage: The map or mapper job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.
- Reduce stage: This stage is the combination of the **Shuffle** stage and the **Reduce** stage. The Reducer's job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.
- During a MapReduce job, Hadoop[14],[15] sends the Map and Reduce tasks to the appropriate servers in the cluster.
- The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.
- Most of the computing takes place on nodes with data on local disks that reduces the network traffic.
- After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.

C)OBJECTIVES

This article presents a HACE theorem that characterizes the features of the Big Data revolution, and proposes a Big Data processing model, from the data mining perspective.The Data mining process perform an unstructured data. So the this paper using the Hadoop concept that perform the HACE theorem.

IISYSTEM ANALYSIS

A)PROBLEM DESCRIPTION:

Now a days the data search is a vital role in internet. The previous system to search a complex data or big data is challenging role in data mining. The system can't mining a data from a large volume of data sets. Because it perform a serial processing. The proposed of paper using HACE theorem that perform a Hadoop base processing of Big data. It perform the parallel processing of data at a time from large data sets. The system using Map reduce concept that perform the retrieve information from large data sets in shortest path ways

B)PROPOSED SYSTEM:

The proposed of paper using HACE theorem that perform a Hadoop base processing of Big data. It perform the parallel processing of data at a time from large data sets. The system using Map reduce concept that perform the retrieve information from large data sets in shortest path ways. So the system can also determine the complex data sets and provide fast retrieval result to users and more security, privacy of data.

a)Advantages

- It processing the Big data.
- It reduce the time consuming.
- Retrieval result is accurate.
- Provide more privacy, security of data.
- Efficient to retrieve data from complex data sets.

IIISYSTEM DESIGN

A)SYSTEM ARCHITECTURE

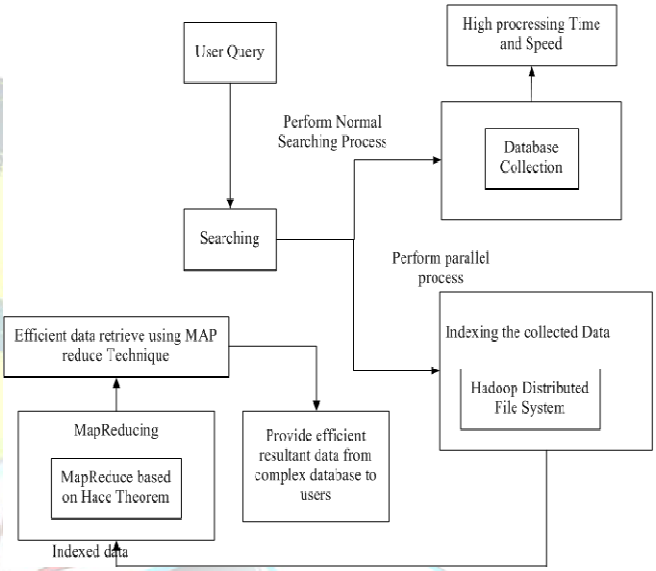

Figure 1 System Architecture

B)MODULE DESCRIPTION

a) Network Initialization: Data mining is an interdisciplinary subfield. It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence database systems. The overall goal of the data mining process is to extract information from a dataset. Before search the data from database by user to first initialize the networks. Suppose the user search something data on a complex database it retrieve the result from collection of related data's from server. That means the user searchable related data are collect the number of data set from variety of servers when the network is initialized.

b) Indexing perform on collected data: The collection of related data set in user searchable information is not a particular format of data. This data's are contain a variety of formats. Users search the data in variety of computer system that contain a different platforms. For example search a text file means it seems to be a pdf, word, txt

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

format of files. So all file formats are not support in every platform. So the indexing technique is using HDFS file system that perform common format of file is supported in each system of users.

c) Map reduce in Hadoop: Map Reduce is a programming model and an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster. The Hadoop is fast retrieval result of data to a user query, even if the data is big data or un-structured data. The map reduce perform parallel processing that is collect user searchable query information from a large complex data base in simultaneously. The map reduce using shortest path of information retrieval from a collection of server of a data set.

d)Using HACE theorem: The map reduce technique is involve shortest path of information retrieval that are depending on the user searchable query data. The map reduce using parallel processing that receive information from a large server of data in simultaneously. The indexing technique perform to create a common file format that is called HDFS file system. The user searchable collected information is convert to common file system that support each platform of a system. So the HACE theorem is using the Map reduce and Indexing techniques that performed to retrieve efficient result to user.

IVALGORITHM AND TECHNIQUES

A) PROPOSED SYSTEM ALGORITHM DETAILS

a)HACE Theorem-Heterogeneous, **A**utonomous, **C**omplex, **E**volving

The HACE theorem suggests that the key characteristics of the Big Data are

1) huge with heterogeneous and diverse data sources,

2) autonomous with distributed and decentralized control, and

3) complex and evolving in data and knowledge associations.

To support Big Data mining, high-performance computing platforms are required, which impose systematic designs to unleash the full power of the Big Data.

b)ALGORITHM

- **algorithm** ford-fulkerson**is**
- **input:** Graph G with flow capacity c,
- source node s,
- sink node t
-
- *(Note that $f_{(u,v)}$ is the flow from node u to node v, and $c_{(u,v)}$ is the flow capacity from node u to node v)*
-
- **for each** edge (u, v) **in**G_E**do**
- $f_{(u, v)} \leftarrow 0$
- $f_{(v, u)} \leftarrow 0$
-
- **while** there exists a path p from s to t**in** the residual network G_f**do**
- let c_f be the flow capacity of the residual network G_f
- $c_f(p) \leftarrow \min\{c_f(u, v) \mid (u, v)$ **in**$p\}$
- **for each** edge (u, v) **in**p**do**
- $f_{(u, v)} \leftarrow f_{(u, v)} + c_f(p)$
- $f_{(v, u)} \leftarrow -f_{(u, v)}$
-
- **return***f*

c)HACE THEOREM EXPLANATION

Hace Theorem is used to model the characteristics of the big data .Big data -information includes of big, heterogeneous, autonomous, and decentralized manipulate desires to discover to the complex and dynamic relationship between data. These traits make it an severe challenge for locating useful advantage from the large knowledge. In a native sense, it will probably imagined that a blind man is making an attempt to measurement up a huge elephant for you to be the colossal knowledge in this context. The term tremendous information actually considerations about data volumes, HACE theorem suggests that the key characteristics of the tremendous data are: A Huge with various and miscellaneous data sources:

The major characteristics of the big data is the large volume of knowledge represented by quite a lot of sources. This gigantic quantity of data comes from quite a lot of websites like Twitter, Myspace, Orkut and LinkedIn and so on. This is due to the fact one of a kind knowledge collectors prefer their possess representation or approach for data recording and the character of specific applications additional layout come in more than a few knowledge representations. Autonomous Sources with circulated &decentralized Control:-

Independent Sources with circulated & decentralized manipulate are a essential characteristic of enormous data applications. Complex and evolving associations:-In an early stage of knowledge centralized information systems, the focus is on discovering first-class feature values to represent every remark. This type of pattern feature representation inherently treats each man or woman as an impartial entity without given that their social connections, which is likely one of the important reasons of the human society.

VRESULT

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

It provides a group wise processing operator Translate that takes state as an explicit input to support incremental analysis. But it adopts a new programming model that is very different from MapReduce. In addition, several research studies, support incremental processing by task-level re-computation, but they require users to manipulate the states on their own. In contrast, i2MapReduce exploits a fine-grain pair level re-computation that are more advantages. Incremental processing for iterative application. Naiad. Big data proposes a timely dataflow paradigm that allows stateful computation and arbitrary nested iterations. To support incremental iterative computation, programmers have to completely rewrite their MapReduce programs for Naiad. In comparison, we extend the widely used MapReduce model for incremental iterative computation. Map- Reduce programs can be slightly changed to run on i2MapReduce for incremental processing. And data mining results compared to re-computation on both plain and iterative MapReduce.

VICONCLUSION

The paper proposed using map reduce and indexing techniques are performed by the HACE theorem. The HACE theorem used for the Hadoop based on information in complex data. This paper, using the HACE theorem for map reduce using parallel processing study, and the test results show that the using indexing technique that create common HDFS file system, and it has very good validity. Especially, the dataset size is bigger, the efficiency will be higher than compared to past evaluation. The HACE theorem characteristic of big data in 1) huge with heterogeneous and diverse data sources, 2) autonomous with distributed and decentralized control, and 3) complex and evolving in data and knowledge associations. The propose system overcome the previous challenging on big data in complex relation between sample data and collected data set to provide accurate data source to users with in time consume & possible factor.

REFERENCES

1. G. Malewicz, M. H. Austern, A. J. Bik, J. C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski, "Pregel: A system for large-scale graph processing," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2010, pp. 135–146.
2. J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S.-H. Bae, J. Qiu, and G. Fox, "Twister: A runtime for iterative mapreduce," in Proc. 19th ACM Symp. High Performance Distributed Comput., 2010, pp. 810–818.
3. M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica, "Resilient distributed datasets: A fault-tolerant abstraction for, in-memory cluster computing," in Proc. 9th USENIX Conf. Netw. Syst. Des. Implementation, 2012, p. 2.
4. R. Power and J. Li, "Piccolo: Building fast, distributed programs with partitioned tables," in Proc. 9th USENIX Conf. Oper. Syst. Des. Implementation, 2010, pp. 1–14.
5. S. Ewen, K. Tzoumas, M. Kaufmann, and V. Markl, "Spinning fast iterative data flows," in Proc. VLDB Endowment, 2012, vol. 5, no. 11, pp. 1268–1279.
6. S. R. Mihaylov, Z. G. Ives, and S. Guha, "Rex: Recursive, deltabased data-centric computation," in Proc. VLDB Endowment, 2012, vol. 5, no. 11, pp. 1280–1291.
7. Y. Bu, B. Howe, M. Balazinska, and M. D. Ernst, "Haloop: Efficient iterative data processing on large clusters," in Proc. VLDB Endowment, 2010, vol. 3, no. 1–2, pp. 285–296.
8. Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, and J. M. Hellerstein, "Distributed graphlab: A framework for machine learning and data mining in the cloud," in Proc. VLDB Endowment, 2012, vol. 5, no. 8, pp. 716–727.