ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

HADOOP BASED APPLICATION USING MULTINODE CLUSTERS

Vanshika Bhati*, Meenakshi Sharma, Ajay Agarwal
* Engineering Student, Department of IT, Krishna Group of Institutions, Ghaziabad
Engineering Student, Department of IT, Krishna Group of Institutions, Ghaziabad
Professor, Department of IT, Krishna Group of Institutions, Ghaziabad

DOI: 10.5281/zenodo.800595

ABSTRACT

In the present era, data is considered as precious as gold for many organizations. Data management and storage is of utmost importance. In today's scenario, data is being generated in massive quantities every single day. Hence, the storage and processing of data using the conventional storing methods like RDBMS is not efficient and effective. So, new ways have been evolved to manage this massive amount of data, also termed as Big Data. This Big Data is a combination of both structured and unstructured data. Hadoop is an open source software that helps to store and process this Big Data. The Hadoop divides the data in blocks and stores them on different nodes and also does replication of these blocks for fault tolerance. The Hadoop Distribution File system (HDFS) and MapReduce are the two key components of Hadoop. MapReduce is used to process the data. In this paper 3-nodes cluster is proposed to store file and process the data for word-count application.

INTRODUCTION

Data sets are growing rapidly in part because they are increasingly gathered by cheap and numerous information-sensing mobile devices, aerial (remote sensing), software logs, cameras, microphones, radio-frequency identification (RFID) readers and wireless sensor networks. Data that is being generated is both structured and unstructured in nature. For a structured dataset, conventional techniques like relational database management system (RDBMS) can be used. But, all the data that is being generated is not structured and contain unstructured data like audio and video.

Consider scenario wherein 1GB of data that needs to be processed. The data is stored in a relational database on desktop computer and has no problem handling this load. If it starts growing very quickly, and that data grows to 10GB, then 100 GB, and there is a need to reach the limits of the current desktop computer. When the data grows to 10TB, and then 100TB, it is quickly approaching the limits of new computer also. Moreover, we may be asked to feed application with unstructured data coming from sources like Facebook, Twitter, RFID readers, sensors, and so on. The management wants to derive information from both the relational data and the unstructured data and wants this information as soon as possible. Hence, the problem lies in the amount of data which is to be processed.

In Section 2 existing and proposed systems are defined. Literature review is done in Section 3. Section 4 contains Hadoop framework. An example is illustrated in Section 5. Section 6 has concluding remarks while future scope is mentioned in Section 7.

SECTION 2

In this Section existing and proposed problem is defined:

EXISTING SYSTEM

When one types in a document, Word automatically counts the number of pages and words in ones document and displays them on the status bar at the bottom of the workspace. While writing code in java or in any other language, we can also create a program which does the same.
But in both these applications, there is a limit on the number of words which has to be processed.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

PROPOSED SYSTEM

To overcome the limitation of existing system we proposed a similar problem that provides in addition of total words, frequency of each word in a given input file. This can be obtained with the help of Map Reduce feature of Hadoop supporting BIG DATA (large files) and can be made to support input file of any type. In this work, code is designed in order to support .doc and .pdf files.

LITERATURE REVIEW

S. Vikram Phaneendra & E. Madhusudhan Reddy *et.al*. Illustrated that in olden days the data was less and easily handled by RDBMS but recently it is difficult to handle huge data through RDBMS tools, which is preferred as "big data". In this they told that big data differs from other data in 5 dimensions such as volume, velocity, variety, value and complexity [1].

Kiran kumara Reddi & Dnvsl Indira *et.al*. Enhanced us with the knowledge that Big Data is combination of structured , semi-structured ,unstructured homogenous and heterogeneous data [2].

Jimmy Lin *et.al*. used Hadoop which is currently the large –scale data analysis " hammer" of choice, but there exists classes of algorithms that aren't " nails" in the sense that they are not particularly amenable to the MapReduce programming model [3].

Wei Fan & Albert Bifet *et.al*. Introduced Big Data Mining as the capability of extracting Useful information from these large datasets or streams of data that due to its Volume, variability and velocity it was not possible before to do it [4].

Albert Bifet *et.al*. Stated that streaming data analysis in real time is becoming the fastest and most efficient way to obtain useful knowledge, allowing organizations to react quickly when problem appear or detect to improve performance. [5].

Bernice Purcell *et.al*. Started that Big Data is comprised of large data sets that can't be handle by traditional systems. Big data includes structured data, semi-structured and unstructured data. The data storage technique used for big data includes multiple clustered network attached storage (NAS) and object based storage. The advent of Big Data has posed opportunities as well challenges to business [6].

Sameer Agarwal *et.al*. Presents a BlinkDB, a approximate query engine for running interactive SQL queries on large volume of data which is massively parallel. [7].

Yingyi Bu *et.al*. Used a new technique called as HaLoop which is modified version of Hadoop MapReduce Framework, as Map Reduce lacks built-in-support for iterative programs HaLoop allows iterative applications to be assembled from existing Hadoop programs without modification, and significantly improves their efficiency by providing iteration caching mechanisms and a loop-aware scheduler to exploit these caches [8].

Shadi Ibrahim et.al. In this paper, author develop a novel algorithm named LEEN for locality aware and fairness-aware key partitioning in MapReduce. LEEN embraces an asynchronous map and reduce scheme [9].

Kenn Slagter et.al. He proposes an improved partitioning mechanism for optimizing massive data analysis using MapReduce for evenly distribution of workload [10].

Ahmed Eldawy et.al. presents the first full-fledged MapReduce framework with native support for spatial data a Spatial Hadoop [11].

Jeffrey Dean et.al. This allows programmers without any experience with parallel and distributed systems to easily utilize the resources of a large distributed system. Author proposes Simplified Data Processing on Large Clusters [12].

Chris Jermaine et.al. Proposes a Online Aggregation for Large-Scale Computing. [13].

Tyson Condie et.al. propose a modified MapReduce architecture in which intermediate data is pipelined between operators, while preserving the programming interfaces and fault tolerance models of other MapReduce frameworks [14].

HADOOP FRAMEWORK

Hadoop is open source software used to process the Big Data. It is very popular used by organizations/researchers to analyze the Big Data. Hadoop is influenced by Google's architecture, Google File System and MapReduce. Hadoop processes the large data sets in a distributed computing environment. An Apache Hadoop ecosystem consists of the Hadoop Kernel, MapReduce, HDFS and other components like Apache Hive, Base and Zookeeper.

A. Hadoop consists of two main components: 1) Storage: The Hadoop Distributed File System (HDFS): It is a distributed file system which provides fault tolerance and designed to run on commodity hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS can store data across thousands of servers. HDFS has master/slave architecture. Files added to HDFS are split into fixed-size blocks. Block size is configurable, but defaults to 64 megabytes.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

2) *Processing: MapReduce* : It is a programming model introduced by Google in 2004 for easily writing applications which processes large amount of data in parallel on large clusters of hardware in fault tolerant manner. This operates on huge data set, splits the problem and data sets and run it in parallel. Two functions in MapReduce are as following:

a) Map – The Map function always runs first typically used to filter, transform, or parse the data. The output from Map becomes the input to Reduce.

b) **Reduce** – The Reduce function is optional normally used to summarize data from the Map function.

Working of Hadoop

Stage 1

A user/application can submit a job to the Hadoop (a hadoop job client) for required process by specifying the following items:

- The job configuration by setting different parameters specific to the job.
- The location of the input and output files in the distributed file system.
- The java classes in the form of jar file containing the implementation of map and reduce functions.

Stage 2

The Hadoop job client then submits the job (jar/executable etc) and configuration to the JobTracker (running on Master node) which then assumes the responsibility of distributing the software/configuration to the TaskTrackers (running on slaves nodes), scheduling tasks and monitoring them, providing status and diagnostic information to the job-client.

Stage 3

The TaskTrackers on different nodes execute the task as per MapReduce implementation and output of the reduce function is stored into the output files on the file system.

In this work, we have created 3-nodes cluster in Hadoop. One is called Hadoop master and other two are its slave nodes. Given file is divided into two halves. Each half is used as input for two slave nodes where map and reduce operations are applied to obtain the combined result from the slave nodes by master node. The output is frequency of each word used in given input file.

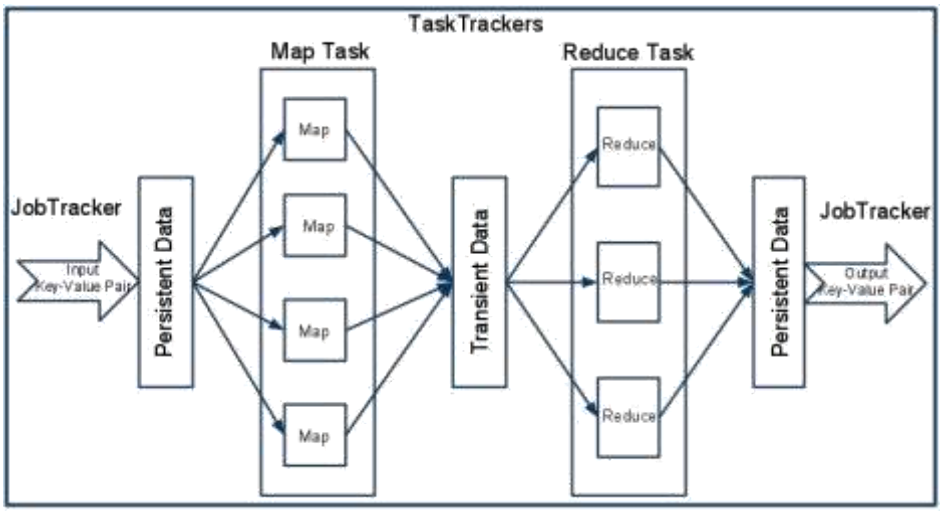

Figure 3.1: Working of Multi Node Cluster

In a multi node cluster, the data is divided into various nodes. It is aimed to implement a multi node cluster terminology by the use of MapReduce technique in Big Data. Firstly a single node cluster is setup, followed by setting several similar single nodes to configure a multi node cluster. Further a file is provided as an input to the cluster. The data gets divided (and may be replicated) among the different nodes and is processed by applying Map and reduce operations. A word count application runs on the data of the file to determine the number of times a word appears in the file and the data gets sorted alphabetically.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

Figure 3.1 illustrates working of Multi Node cluster. There are 2 important types of nodes and it's functions in Hadoop cluster – NameNode and DataNode:

Namenode

- NameNode is also known as the Master
- NameNode does not store the actual data or the dataset. The data itself is actually stored in the DataNodes.
- NameNode knows the list of the blocks and its location. With this information NameNode knows how to construct the file from blocks.
- NameNode is a single point of failure in Hadoop cluster.

DataNode

- DataNode is also known as the Slave
- DataNode is responsible for storing the actual data.
- NameNode and DataNode are in constant communication
- When a DataNode starts up it announce itself to the NameNode along with the list of blocks it is responsible for.
- When a DataNode is down, it does not affect the availability of data or the cluster. NameNode will arrange for replication for the blocks managed by the DataNode that is not available.
- DataNode is usually configured with a lot of hard disk space. Because the actual data is stored in the DataNode.
- JobTracker and TaskTracker are 2 essential processes involved in MapReduce execution.

Job Tracker

- JobTracker process runs on a separate node and not usually on a DataNode.
- JobTracker receives the requests for MapReduce execution from the client.
- JobTracker talks to the NameNode to determine the location of the data.
- JobTracker process is critical to the Hadoop cluster in terms of MapReduce execution.

TaskTracker

- TaskTracker runs on DataNode. Mostly on all DataNodes.
- Mapper and Reducer tasks are executed on DataNodes administered by TaskTrackers.

ILLUSTRATION

Word count is an example by which the number of words and their occurrences can be calculated in a given file. The word count operation takes place in two stages a mapper phase and a reducer phase. In mapper phase first the text is tokenized into words then it is formed as a key value pair where the key being the word itself and value '1' for each word. For example we have following two input files:

Input File 1:
Hello Girl Bye Girl

Input File 2:
Hello Hadoop Goodbye Hadoop
The execution has three steps:

Step 1:

For the given sample input file 1 the first map emits:

< Hello, 1>
< Girl, 1>
< Bye, 1>
< Girl, 1>

For the given sample input file 2 the second map emits:

< Hello, 1>
< Hadoop, 1>
< Goodbye, 1>
< Hadoop, 1>

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

Step 2:

The output of the first map file 1 after shuffling:

< Bye, 1>
< Hello, 1>
< Girl, 2>`

The output of the second map file 2 after shuffling:

< Goodbye, 1>
< Hadoop, 2>
< Hello, 1>`

Step 3:

The Reducer implementation, via the reduce method just sums up the values, which are the occurrence counts for each key (i.e. words in this example).

Thus the output of the job is:

Bye 1
Goodbye 1
Hadoop 2
Hello 2
Girl 2

This is how the MapReduce word count program executes and outputs the number of occurrences of a word in any given input file in Hadoop.
Hadoop installation procedure is given Appendix 1.

CONCLUSION

Hadoop possesses a distributed file structure and practically proves the concepts of distributed and parallel computing. Therefore the results of the computation are faster. Thus it provides a means to attain high efficiency. The application of Hadoop to various practical problems can provide a robust alternative way to find solutions to them. Also the results are highly optimized and complex to be understood by a person new to the field of Hadoop. The solutions provided are fast, short and can be utilized to predict various patterns in the data. Thus Hadoop can be used to encounter problems which are faced by the conventional data-mining issues and file sizes.

By practically implementing the Hadoop framework, *we see that the input file which is fed into the Hadoop master gets divided on the two. Hadoop after processing the file (which can be a word or pdf file) automatically combines the separate files into one output file. The client gets the desired result i.e. number of occurrences of words in the input file. By implementing such an application on the data, it is possible to predict patterns from it.* These predictions can be useful to an organization for decision-making.

FUTURE SCOPE

The world is becoming data driven. Every decision is now taken on data from stock markets to machines around. The data generated is growing exponentially; with this data great solutions can be developed around the data. For example Facebook is a company which entirely depends on the data generated by its users and this data could be audio, video, text and still images. The company's business model is dependent on this data. The more the users, more the data and more accurate ads company can target and more conversion of these ads. This was just a story of Facebook there are many companies, many apps and many sources of data. Considering all the above the dependency on data also increases for a better decision making and many problem statements can be designed from the data and also answer can be found from the same data. Almost every business problem or need involves the use and maintenance of data. Data is virtually the lifeblood of business so it will always be important. After all, data – and big data – are just point-in-time recordings of business or operational events. Big data analytics is going to be main stream with increased adoption among every industry and forma virtuous cycle with more people wanting access to even bigger data. However, often the requirements for big data analysis are really not well understood by the developers and business owners, thus creating an undesirable product.

For organizations to not waste precious time, money and manpower over these issues there is a need to develop expertise and process of creating small scale prototypes quickly and test them to demonstrate its correctness, matching with business goals.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

REFERENCES

[1] S.Vikram Phaneendra & E.Madhusudhan Reddy "Big Data- solutions for RDBMS problems- A survey" In 12th IEEE/IFIP Network Operations & Management Symposium (NOMS 2010) (Osaka, Japan, Apr 19{23 2013).
[2] Kiran kumara Reddi & Dnvsl Indira "Different Technique to Transfer Big Data : survey" IEEE Transactions on 52(8) (Aug.2013) 2348 { 2355}
[3] Jimmy Lin "MapReduce Is Good Enough?" The control project. IEEE Computer 32 (2013).
[4] Umasri.M.L, Shyamalagowri.D ,Suresh Kumar.S "Mining Big Data:- Current status and forecast to the future" Volume 4, Issue 1, January 2014 ISSN: 2277 128X
[5] Albert Bifet "Mining Big Data In Real Time" Informatica 37 (2013) 15–20 DEC 2012
[6] Bernice Purcell "The emergence of "big data" technology and analytics" Journal of Technology Research 2013.
[7] Sameer Agarwal†, Barzan MozafariX, Aurojit Panda†, Henry Milner†, Samuel MaddenX, Ion Stoica "BlinkDB: Queries with Bounded Errors and Bounded Response Times on Very Large Data" Copyright © 2013ì ACM 978-1-4503-1994 2/13/04
[8] Yingyi Bu _ Bill Howe _ Magdalena Balazinska _ Michael D. Ernst "The HaLoop Approach to Large-Scale Iterative Data Analysis" VLDB 2010 paper "HaLoop: Efficient Iterative Data Processing on Large Clusters.
[9] Shadi Ibrahim⋆ _ Hai Jin _ Lu Lu "Handling Partitioning Skew in MapReduce using LEEN" ACM 51 (2008) 107–113
[10] Kenn Slagter • Ching-Hsien Hsu "An improved partitioning mechanism for optimizing massive data analysis using MapReduce" Published online: 11 April 2013
[11] Ahmed Eldawy, Mohamed F. Mokbel "A Demonstration of SpatialHadoop:An Efficient MapReduce Framework for Spatial Data" Proceedings of the VLDB Endowment, Vol. 6, No. 12 Copyright 2013 VLDB Endowment 21508097/13/10.
[12] Jeffrey Dean and Sanjay Ghemawat "MapReduce: Simplified Data Processing on Large Clusters" OSDI 2010
[13] Niketan Pansare1, Vinayak Borkar2, Chris Jermaine1, Tyson Condie "Online Aggregation for Large MapReduce Jobs" August 29September 3, 2011, Seattle, WA Copyright 2011 VLDB Endowment, ACM
[14] Tyson Condie, Neil Conway, Peter Alvaro, Joseph M. Hellerstein "Online Aggregation and Continuous Query support in MapReduce" SIGMOD'10, June 6–11, 2010, Indianapolis, Indiana, USA. Copyright 2010 ACM 978-1-4503-0032-2/10/06.

CITE AN ARTICLE

Bhati, V., Sharma, M., & Agarwal, A. (2017). HADOOP BASED APPLICATION USING MULTINODE CLUSTERS. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, 6*(5), 706-711. doi:10.5281/zenodo.800595