ISSN 2278-3091

Volume 8, No.1, January – February 2019

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse01812019.pdf

https://doi.org/10.30534/ijatcse/2019/01812019

A Review and Analysis of Big Data and MapReduce

Danish Ahamad[1], MD Mobin Akhtar[2], Shabi Alam Hameed[3]

[1] Collage of Science & Arts, Sajar City, Shaqra University, Saudi Arabia, danish.ahamad@gmail.com

[2] Preparatory Health Studies, Riyadh Elm University, Riyadh, Saudi Arabia, jmi.mobin@gmail.com

[3] Collage of Science & Humanities, Huraymla, Shaqra University, Saudi Arabia, shabiazam@gmail.com

ABSTRACT

The recent years consume the exemplary growth of data generation. This enormous amount of data has brought new kind of problem. The existing RDBMS systems are unable to process the Big Data, or they are not efficient in handling it. The significant problems appeared with the Big Data are storage and processing. Hadoop is brought in the solutions for storage and processing in the form of HDFS (Hadoop Distributed File System) and MapReduce respectively. The traditional systems not construct for keeping the Big Data, and also they can only process structured data. One of the industries, first to face the Big Data challenges is financial sector. In this work, an unstructured stocks data is processed using Hadoop MapReduce. Efficient processing of unstructured data is analyzed, and all the phases involved in implementation explicated.

Key words : Big data, Hadoop, HDFS, MapReduce

1. INTRODUCTION

We live in the data age. It is difficult to quantify the aggregate volume of data put away electronically, yet an IDC gauge put the extent of the "computerized universe" at 4.4 Zetta Bytes in 2013 and is gauging a ten times development by 2020 to 44 ZB. Around one thousand EB, one million PB, or one billion TB. That is more than one circle drive for each on the planet. So there's a great deal of data out there are presumably thinking about how it influences. The vast majority of the data is secured up in the biggest web properties (like web crawlers) or logical or money related establishments, would it say it is not? Does the coming of large data influence littler associations or people? The pattern is for each's data impression to develop, yet maybe more fundamentally, the measure of data produced by machines as a piece of the Internet of Things will be significantly more noteworthy than that produced by individuals. Machine logs, RFID peruses, sensor systems, vehicle GPS follows, local exchanges these add to the developing heap of data. The volume of data being made freely accessible builds each year, as well. Associations never again need to deal with their data; achievement, later on, will be directed to a vast degree by their capacity to extricate an incentive from other associations' data. For a long time, clients who need to store and investigate information would store the information in a database and process it utilizing SQL queries. The Web has changed the vast majority of the suspicions of this time. On the Web, the information is unstructured and substantial, and the databases can neither catch the information into an outline nor scale it to store and process it. Google was one of the principal associations to confront the issue, where they needed to download the entirety of the Internet and index it to help look questions. They constructed a framework for substantial scale information handling obtaining from the "map" and "reduce" capacities.

[10] What is big data? Basically refers to the big amount of data that cannot be stored and processed using the traditional approach the given specified period of time. The question arises in order to categories as big data. Faulty thinking while allude to the term big data. We generally use the phrase big data to ascribe to the data that is either Gigabyte, Exabyte, TB, PB or anything that is bigger than this size. This does not define the word big data comprehensively. The small amount of data can be described as big data build upon the context it is being used. Try to clarify it to you, for example, try to add a document 100 megabyte may not be capable to do email system would not support and attachment of this 100 megabytes of attachment which can be referred to as big data.

2. STRUCTURE FOR MAPREDUCE

Basically, it is very important programming model that encour-ages inclusive scale and appropriated making ready for big data on a machine. MapReduce characterizes the gauge as 2 capacities: map and reduce. The information becomes a group of value merg-es, and also yield may be a posting of esteem sets. The map col-lection takes Associate in the standard information set Associate in standard ends up in an accumulation of ordinary key/esteem sets (which is vacant). The decrease work gets a standard key and a summing up of ordinary qualities contrasted which key as its data and results set of definite key/esteem matches because the yield. Execution of a MapReduce program includes 2 stages. with-in the main stage, every data mix is given to delineate, and a ren-dezvous of information sets is delivered. a brief time later, within the second stage, the larger a part of the center of the road esteems that have the connected key area unit collected into a summing up, and every moderate key and its connected

commonplace esteem list area unit given to diminish limit. additional knowledge and representations area unit originated in.

The MapReduce program is executed in two strategies. Regularly, taken MapReduce is performed utilizing expert/slave structure [9]. the primary machine has met all needs for task of undertakings and handling the slave PCs. A schematic for the accomplishment of a MapReduce program is presented in Figure one. the data is placed away in shared capability sort of a circulated record activity and is isolated into lumps. Initial, a reproduction of the guide and diminish parts' code is distributed to any or all laborers. At that time, the professional doles out guide and diminish undertakings to directors. each specialist assigned a guide trip peruses the indistinguishable data split and passes the bulk of its sets to delineate and conveys the aftereffects of the guide position into moderate documents. Following the guide prepare is completed, the reducer specialists indicate middle of the road data and exchange the halfway combines to decrease work last the couples took when by reduce undertakings area unit routed to terminal yield documents.

3. LITERATURE REVIEW

Literature review described as the summary or re-organizing the related information from different sources to understand the research problem. In this review how data analysis done for big data through data mining techniques is studied.

[1] MapReduce is a programming model and a related execution for handling and producing comprehensive datasets. Clients determine a map work that procedures a key/esteem combine to create an arrangement of the middle of the road key/esteem sets and a reduce work that unions all moderate values related to a similar middle key. Numerous exact undertakings are expressible in this model, has appeared in the paper. Projects written in this useful style are naturally parallelized and executed on an expansive cluster of production machines. The run-time framework deals with the subtle elements of dividing the information data, booking the program's execution over an arrangement of machines, taking care of machine disappointments, and dealing with the required between machine correspondence. It permits software engineers with no involvement with parallel and distributed frameworks to effortlessly use the assets of a substantially distributed framework. Our usage of MapReduce keeps running on a substantial the cluster of ware machines and is exceptionally versatile: a typical MapReduce calculation forms numerous terabytes of data on a large number of machines. Developers find the framework simple to utilize: several MapReduce programs have been actualized and upwards of one thousand MapReduce employments are executed on Google's clusters consistently.

[2] In reality hone, programming frameworks frequently worked without building up any express forthright model. It can cause difficult issues that may ruin the relatively inescapable future development since best case scenario the main documentation about the product is as source code remarks. To address this issue, look into has been concentrating on the programmed induction of models by applying machine learning algorithms to execution logs. Be that as it may, the logs produced by a good programming framework might be substantial and the derivation calculation can surpass the handling limit of a single PC. This paper proposes a flexible, general way to deal with the deduction of conduct models that can deal with substantial execution logs using parallel and distributed algorithms actualized utilizing the MapReduce programming model and executed on a cluster of interconnected execution hubs. The approach comprises of two distributed stages that perform follow cutting and model blend. For each stage, a distributed calculation utilizing MapReduce created. With the parallel data handling limit of MapReduce, the issue of deducing conduct models from huge logs can proficiently unravel. The strategy is actualized over Hadoop. Analyses on Amazon clusters indicate productivity and adaptability of our approach.

[3] Google's MapReduce programming model serves for processing large data sets in a massively parallel manner. We convey the first hard report of the typical with its improvement as Google's domain-specific language Sawzall. we reverse-engineer the important papers on MapReduce and Sawzall in end, and we detention our results as an executable condition. We also classify and decide some doubts in the casual demonstration specified in the seminal papers. We practice entered handy encoding (specifically Haskell) as a device for project retrieval and executable condition.

[4] MapReduce is a mainstream framework for the data-serious distributed processing of bunch employments. To streamline blame resilience, numerous executions of MapReduce appear the total yield of each map and reduce undertaking before it can be devoured. In this paper, we propose an altered MapReduce engineering that enables data to pipelined between administrators. This broadens the MapReduce programming model past cluster handling and can reduce finish times and enhance framework use for clump occupations too. We introduce an adjusted form of the Hadoop MapReduce framework that backings online total, which enables clients to see "early returns" from a vocation as it is being processed. Our Hadoop Online Prototype (HOP) likewise underpins nonstop inquiries, which empower

MapReduce projects to be composed for applications for example, occasion observing and stream handling. Bounce holds the adaptation to non-critical failure properties of Hadoop and can run unmodified client characterized MapReduce programs.

[5] A fast development of data in late time, Industries what's more, scholarly world required a wise data examination apparatus that would be useful to fulfill the need to dissect a colossal measure of data. MapReduce framework is fundamentally intended to register data escalated applications to help viable basic leadership. Since its presentation, wonderful research endeavors have been put to make it more natural to the clients in this way used to bolster the execution of huge data escalated applications. Our overview paper underscores the cutting edge in enhancing the execution of different applications utilizing later MapReduce models [7] and how it is helpful to process expansive scale dataset. A relative investigation of given models compares to Apache Hadoop and Phoenix will be talked about fundamentally based on execution time and adaptation to internal failure. At last, an abnormal state the talk will do about the upgrade of the MapReduce calculation in a particular region, for example, Iterative calculation, consistent question handling, half-breed database and so on.

4. PROBLEM STATEMENT

The traditional database systems are not built to handle the kind of vast amounts of data we are experiencing in the recent years. It is also costly to increase the processing power on these systems. Also, the traditional systems can only process structured data. The significant part of the data generated in the last two years is unstructured data. The existing system works on a single server, which makes it difficult and costly to grow it vertically.

5. LIMITATIONS

The main two disadvantages are:

1. Cannot store and process Big Data [6]
2. Cannot process unstructured data

There is the limit to how much one can grow this system vertically. Most of these single servers are high end or custom made, hence not cost effective.

5. RECOMMENDED SYSTEM

The traditional RDBMS is not made to process unstructured data, and there is also a limit on the size of the data it can handle. The main problems that appeared with Big Data processing [8] are storage and processing. In this work, the Hadoop Framework used, which solve the problems of storage and processing. Hadoop stores and processes the data in a cluster which is a distributed network. It solves the problem of increasing the storage and processing power. It can be achieved by merely increasing the number of nodes in a cluster. The data storage, as well as the processing, distributed across the nodes in the cluster, bring down the processing time of Big Data drastically. This also avoids the need for high end or custom-made hardware which is very expensive.

6. CONCLUSION

This work elucidates how unstructured log data can efficiently process by using the MapReduce, programming model. Hadoop Framework is the solution for the two main problems of the Big Data processing. In this work, an unstructured stocks dataset is used to demonstrate the implementation of MapReduce jobs. Also, the phases involved exemplified. This work gives a better understanding of implementing MapReduce jobs on unstructured log data and encourages more research on coming up with more efficient ways to process unstructured log data.

REFERENCES

1. P. Zadrozny and R. Kodali, **“Big Data Analytics using Splunk, Berkeley”**, CA, USA: Apress, 2013. https://doi.org/10.1007/978-1-4302-5762-2
2. F. Ohlhorst, **“Big Data Analytics: Turning Big Data into Big Money”**, Hoboken, N.J, USA: Wiley, 2013.
3. J. Dean and S. Ghemawat, **"MapReduce: Simplified data processing on large clusters,"** Commun ACM, 51(1), pp. 107-113, 2008. https://doi.org/10.1145/1327452.1327492
4. F. Li, B. C. Ooi, M. T. Özsu and S. Wu, **"Distributed data management using MapReduce,"** ACM Computing Surveys, 46(3), pp. 1-42, 2014. https://doi.org/10.1145/2503009
5. C. Doulkeridis and K. Nørvåg, **"A survey of large-scale analytical query processing in MapReduce,"** The VLDB Journal, pp. 1-26, 2013.
6. S. Sakr, A. Liu and A. Fayoumi, **"The family of mapreduce and large-scale data processing systems,"** ACM Computing Surveys, 46(1), pp. 1-44, 2013. https://doi.org/10.1145/2522968.2522979
7. The emergence of **“big data”** technology and analytics Bernice Purcell –Holy Family University.
8. The Forrester Wave™: **Big Data Predictive Analytics Solutions**, Q1 2013 by Mike Gualtieri, January 3, 2013
9. Bhavani Buthukuri, Sivaram Rajeyyagari, “Investigation on Processing of Real-Time Streaming Big Data.” **International Journal of Engineering & Technology,** *7 (3.13) (2018) 79-83*
10. Iswharappa, Anuradha, “**A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology**” (ICCC2015)