Research Paper on Big Data and Hadoop-Map Reduce Real Time Scheduling

N. Deshai¹, S. Venkataramana², Dr. G. P. Saradhi Varma³

Information Technology^{1, 2, 3} S.R.K.R, JNTUK, India^{1, 2, 3}

ABSTRACT

The Big Data environment came in all computing life where the bulk volume of data collects and store at escalating rates. Due to widely use of social networking sites, transactional data, sensor networks, mobile devices, machines generated data, software logs, videos and many more producing petabytes to terabytes of massive data. The large data sets are referred to volume (size), variability (complexity), velocity (growth of data), and veracity(vary data greatly) to manage, store, process and analyzed very difficult. The traditional data management system is not convenient for storing, processing and analyzing such large data. The working of Big data can be done efficient manner using Apache Hadoop for distributed and parallel processing of large data sets across clusters of commodity hardware on one server to hundreds and thousands of machines along with more fault tolerance, scalable, economic and efficient. MapReduce is a popular parallel programming model used to solve a wide range of Big Data applications. This paper aims at reviewing the role of big data, Hadoop-Map Reduce and a comprehensive survey of the various job scheduling algorithms has been performed. In this paper, a new optimal task selection scheme is introduced in to assist the scheduler when multiple local tasks are available to a node.

Keywords: Big data, Hadoop, Map Reduce, Scheduling algorithm.

I.INTRODUCTION

We are living in the computerized information world. The information surge is originating from various wellsprings of databases in reality with a high exponential development in reality databases. Information is past with the capacity measure and past the preparing power and in light of the fact that the span of worldwide advanced information is developing and developing in petabytes to zeta-bytes are alluded as Big Data. Huge information has increasingly volume and more mind boggling, semi-organized information, unstructured information, heterogeneous information, and information is twofold consistently moreover. Since consistently the Big Data has been produced in Terabytes of information where the huge volume of information is persistently gathered at exponential rates. Facebook share about 10 billion of pictures which takes one petabyte of information stockpiling. The New York Stock Exchange produces exchange information in Terabytes consistently. Web based keeping money exchange, diverse sensor systems, Airlines record, Global Position System, RFID perusers, PC logs, Closed Circuit cameras, Internet of Things information can be produced on the web in different structures like writings, pictures, sound,

recordings and online networking posts these are made exponential development in information. Huge information and its investigation innovation are the focuses of present day science and business regions. For instance, as indicated by IDC gauges that 16.1 ZettaByte of information produced in 2016 and the worldwide information circle will develop to 163 Zetta Byte in 2025 (that is a trillion gigabytes).IBM demonstrates that 2.5 exa-bytes information is made ordinary which is exceptionally hard to break down. Indeed, even typical information preparing strategies and devices not deal with proficiently. Apache Hadoop Open Source, HDFS and MapReduce Schedulers assume imperative parts to store and process Big Data ongoing calculations.

II BIG DATA

We are living in the computerized world information. Enormous Data is as an accumulation of expansive dataset that can't be handled utilizing conventional figuring systems .Big Data isn't simply an information rather it has turned into a total subject which include different structure, instruments and strategies. The enormous information created from the vast organizations like Facebook, Yahoo, Google, YouTube, Amazon, Microsoft and Twitter, eBay and so forth. Web and its applications have turned into a fundamental piece of the present human way of life. Huge information is the term for informational collections so expansive and confounded that it winds up hard to process utilizing customary information and deficient to manage them. To process these a lot of the dataset in an economical and effective way, parallel, conveyed, versatile, solid way and some arrangement of programming devices and structures are utilized. Breaking down Big Data is a testing undertaking as it includes extensive appropriated record frameworks which ought to be blame tolerant, adaptable and versatile. Huge information challenges incorporate information stockpiling, information examination, information seek, information sharing, information exchange, catching information, perception, questioning, and refreshing and data protection. Enormous information it is portrayed by the 9 Vs. Volume, Velocity, Variety, Variability, Validity, Vulnerability, Volatility, Visualization, and Value.



Figure 1: Challenges in Big Data

2. I Volume

It alludes to tremendous measures of information created by an alternate association inconsistently. The world current measure of information, for the most part, the best-known normal for huge information is volume. The volume (estimate) of the information is spoken to in terabytes, petabytes, and zettabytes. In YouTube, consistently 300 hours of recordings are transferred. 1.1 trillion photographs were taken in 2016, and that number is 9 percent in 2017. The quantity of photographs put away is likewise

anticipated that would develop from 3.9 trillion of every 2016 to 4.7 trillion out of 2017.

2.2 Variety

Alludes to sorts of information now we can use, in certainty 80% of world's information is unstructured (content, picture, sound, video and so forth) with enormous information innovation, we can dissect and unite information of various kinds, for example, messages, online networking discussion, photographs, sensor information, video or voice accounts, log records, click information, machine, and sensor information.

2.3 Velocity

Velocity refers to the speed at which data are being generated, produced, created, or refreshed and moves around the world with high speed. Sometimes 1 minutes is too late, so big data is time sensitive. Facebook data warehouse stores upwards of 300 peta-bytes of data, but the velocity at which new data is created should be taken into account. Facebook claims 600 terabytes of incoming data per day. Google processes on more than 40,000 search queries every second, it translates to more than 3.5 billion searches per day.

2.4 Variability

Variability refers to the inconsistent speed at which big data are loaded into the database. These need to be found by anomaly and outlier detection methods in order for any meaningful analytics to occur. Because a multitude of data dimensions resulting from multiple disparate data types and sources.

2.5 Veracity

Veracity refers to noise, biases and abnormality when we are dealing with high volume, velocity and variety of data, all of the data are not going 100% correct, there will be dirty data. Veracity is confidence, trust, quality, provenance and reliability of the data source.

2.6 Vulnerability

Big data bring new security concerns. Security is also about building brand reputation and trust. Strong security practices, including the use of advanced analytics capabilities to manage privacy and security challenges, can set businesses apart from the competition and create comfort and confidence with the public.

2.7 Volatility Old data need to be before it is considered irrelevant, historic, or not useful any longer How long does data need to be kept for Before big data, organizations tended to store data indefinitely

2.8 Visualization

Traditional graphs when trying to plot a billion data points, we need different ways of representing data such as data clustering, tree maps, sunbursts, parallel coordinates, circular network diagrams and cone trees.

Combine this with the multitude of variables resulting from big data's variety and velocity and the complex relationships between them, and you can see that developing a meaningful visualization is not easy.

2.9 Value

The potential value of Big data is huge. Value is the main source for big data because it is important for businesses, IT infrastructure system to store large amount of values in the database.

III HADOOP

Hadoop is an open source Java-based programming framework also it is the most popular and powerful big data tool in the data world. Hadoop was designed especially for the analysis of large data sets to build scalable, distributed applications. Apache Hadoop is an open-source software framework can be used currently for distributed processing and parallel processing of large data set in a distributed computing environment using the HDFS and MapReduce programming model. Open source, Apache Hadoop was created by Doug Cutting and Mike Cafarella in 2005. Hadoop was originally motivated by a paper published by Google for their approach to lever their data. Hadoop has been successfully used by many companies like Google, Yahoo, Facebook, IBM , Twitter, AOL, YouTube Cloudera, Intel, and New York Times for running their applications on clusters with Distributed, Parallel, Fault Tolerance, Reliability, High Availability, Scalability, Data Locality, and the Cost Effective manner and process extremely large data sets on commodity hardware.

Apache Hadoop, which was sponsored by Yahoo, it is to provide open-source distributed computing framework that supports the data storing and processing of large data sets in a distributed computing environment and it is very much appropriate for high volume of data to handle thousands of tera-bytes and peta-bytes of data. With the help of Hadoop there is no limit for storing and computation process.Hadoop framework allows developers to write parallel processing programs that focus on their computation problem, rather than parallelization issues. Hadoop plays an important role for storing and retrieving huge set of data. If any system loses its data, Hadoop can be used to retrieve it. Apache Hadoop open source splits the files into a number of blocks and distributes them across multiple nodes in a cluster and allows to system continue work in case of a node failure. Hadoop continues to operate the cluster without losing data or interrupting work, by shifting work to the remaining machines in the cluster. Hadoop transfers packaged code into multiple nodes to process the data in parallel manner. This frame takes advantage of data locality, where nodes manipulate and process the data they have easily access them. This allows the Dataset to be processed fast, efficient, flexible and reliable. Hadoop open source can able to provide much needed robustness, scalability and reliable storage on inexpensive of a distributed system and it can hold a large amount of data and provide access to this data to many clients distributed across a network. Apache Hadoop allows the clients to configure the job, submit it, control its execution, and query the state. Each job consists of independent tasks, and all the tasks need to have a system slot running. With Hadoop all scheduling and allocation decisions are made on a task and node slot level for both the map and reduce phases. It provides an abstracted environment for running large scale data intensive applications in a scalable and fault tolerant manner.



Figure1: Hadoop Map Reduce Architecture

Hadoop Characteristics in Big Data:

Robust: It can handle failures of hardware of data is

stored in multiple data nodes.

Scalable: it can able to increase cluster size by add more and more nodes.

Simple: It focuses on code rather than data and is can

write parallel wise so it's simple.

Portable: Because Structured (In table format), Semi structure- Not in well organized format (XML), Unstructured. It is no format (Text, Image, Videos).

Cost Effective: Hadoop uses commodity hardware to store the data so it is inexpensive and economic.

Fault Tolerance: the tasks are automatically redirected to another node if a node fails it is fault tolerance automatically stored multiple copies of all the data. If one node fails, same data is available on some other nodes is based on replication factor.

Hadoop includes HDFS and Hadoop MapReduce.

Hadoop Distributed File System (HDFS):

It is a distributed file system that stores large amount of data with high throughput access to data on clusters.

Hadoop MapReduce:

It is a software framework for distributed processing of large data on clusters. This framework allows developers to write parallel processing programs that focus on their computation problem, rather than parallelization issues.

4. Hadoop Distributed File System (HDFS)

The Hadoop Distributed File System (HDFS) is a distributed, scalable, portable file system designed to run on commodity hardware for Hadoop framework. HDFS is a highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS was built to support high throughput, streaming reads and writes of extremely large files. HDFS consists of a single Master NameNode and number of Slave DataNodes in a cluster. Hadoop HDFS is a selfhealing became more popular amongst all the Big Data tools as it is open source with good flexibility, high scalability, less total cost of ownership and allows data stores of any form like text, images, audio, and videos, without the need to have data types or schemas defined. HDFS is the world's most reliable storage system. The goals and assumptions of HDFS include Hardware failure, Streaming data access, storing large data sets, simple coherence mode. Gracefully deal with component failures of machines and disks. HDFS is able to store huge amounts of information, scale up incrementally and survive the failure of significant parts of the storage infrastructure without losing data. HDFS stores large files (range of tera-bytes to peta-bytes) across a number of machines. HDFS can be support of tens of millions of files in a single instance. It can provide high aggregate data bandwidth and scale to hundreds of nodes in a single cluster. Support the functionality and scale requirements of MapReduce processing. HDFS is designed more for batch processing rather than interactive use by users. HDFS provides high throughput access to different application's data and is suitable for applications that have large data sets. HDFS stores large files (gigabytes to terabytes) spread across multiple machines which achieves reliability by replicating the data across multiple hosts.



Figure 2: Hadoop Architecture

It achieves the reliability by replicating the data across a number of hosts. With the default replication factor value is 3,so data is stored on three nodes: two on the same rack, and one on a different rack. Therefore, detection of faults and quickly, automatically recovery from them is a core architectural intention of HDFS.HDFS was originally built as infrastructure for the Apache Nutch web search engine project. HDFS is a master/slave architecture which comprises of NameNode, Secondary and DataNode .

4.1 NameNode

Name node is the master, hence it can control large numbers of DataNodes. The Name node spreads the data to the data nodes. It stores the each file metadata it includes name, location of each block, block size and file permission

4.2 DataNodes

Data node is the slave where actual data is stored sent by Name Node. DataNodes are stored and retrieve blocks when they are requested by the client or name node. They perform operations such as block creation, deletion and replication as stated by the node name.

4.3 Secondary NameNode

When name node down, secondary node will be online, but this node only have the read access to the fsimage and editlog files and don't have the write access to them. All the metadata is stored into fsimage and editlog files continuously by primary node is Name Node. All the secondary node operations will be stored to temp folder. When name node back to online this temp folder will be copied to name node and the namenode will update the fsimage and editlog files.

V MAPREDUCE

Hadoop MapReduce is a production system for effortlessly composing applications that procedure immense volumes of information in parallel by partitioning the work into an arrangement of free assignments in expansive gatherings (a great many hubs) of fundamental equipment in a solid blame tolerant and practical way. MapReduce is a brilliant model for conveyed figuring, exhibited by Google in 2004. Hadoop MapReduce is the most endorsed PC system that utilizations versatile and adaptable ways to deal with disseminated figuring to process vast informational indexes. Extraordinary compared to other known strategies is MapReduce to change over crude information into helpful data. One of the fixings in MapReduce's mystery source is the thought of information areas as it can perform counts on a similar machine where the prepared information is put away. The MapReduce motor uses JobTracker and TaskTracker to oversee, screen and execute numerous employments. The Hadoop MapReduce work partitions the info informational collection into particular sections that the guide's exercises procedure in a totally parallel manner. A MapReduce work comprises of four unique stages, executed all together: sending customer employments, performing map undertakings, revamping and arranging and decreasing assignment execution. Hadoop MapReduce is inalienably mindful of HDFS and can be utilized as a NameNode when arranging exercises to settle on the best mapping assignments. It maintains a strategic distance from critical system overhead amid handling, it isn't important to duplicate information over the system to get to it and kill one of the primary bottlenecks when preparing a lot of information



Figure 3: MapReduce Job Execution Flow

On general system trying to make process multiple files concurrently, then the centralized system creates too much of a bottleneck, map reduce can remove the bottleneck problem. The framework sorts the outputs of the maps, which are then input to the reduce tasks. Typically both the input and the output of the job are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks. MapReduce is that it allows non-expert users to easily run analytical tasks over big data. Apache Hadoop MapReduce gives full control to users on how input datasets are processed. Users programming code in their queries using Java rather than SQL. After MapReduce program is written it can easily be extrapolated to work over a cluster which has hundreds or even thousands of nodes. In this framework, computation is sent to where the data resides.However, Hadoop MapReduce jobs are far behind parallel databases in their query processing efficiently. Hadoop MapReduce jobs, achieve decent and good performance.Hadoop MapReduce easily to use for a

A larger number of developers in background no databases are required only a basic knowledge in Java is required. MapReduce framework is a processing pillar, processing layer and center processing component on HadoopEcosystem as it gives the rationale of processing. The biggest strength of the MapReduce framework is scalability because able to process peta-bytes of data on a single cluster. HDFS has great Flexibility hence it can be developed in any language like Java, c++, python, etc. On a daily basis the micro-blogging site Twitter receives nearly 500 million tweets, i.e., 3000 tweets per second. MapReduce is a programming model intended for processing substantial volumes of data in parallel by partitioning the work into an arrangement of free undertakings. Another ideal errand determination plot is acquainted in with help the scheduler when different neighborhood undertakings are accessible to a hub. There are two noteworthy daemons in Hadoop MapReduce: the job tracker and the task tracker.

JobTracker: Hadoop MapReduce Job tracker is a daemon which it keeps running on a name node. Hadoop MapReduce has one and just is work tracker for name hub, yet Hadoop MapReduce has numerous errand trackers

for every datum hub. Hadoop MapReduce work tracker which relegates the undertakings to the distinctive arrangement of errand tracker. Which Hadoop MapReduce is the single purpose of disappointment. Since if the activity tracker goes down all the running employments are stopped. Which it gets a pulse from undertaking tracker in light of which Job tracker chooses whether the relegated errand is finished or not.

Following the strategy of Hadoop MapReduce JobTracker:

1. Hadoop HDFS JobTracker gets the administration from the customer.

2. JobTracker reacts to the Mater Node (NameNode) for the area of the information is to discover.

3. JobTracker can be executed errands in light of best TaskTracker hubs found by JobTracker.

4. The JobTracker puts the work to the chose TaskTracker hubs.

5. Screen each TaskTracker hubs in MapReduce. In the event that pulse signs won't submit then the work is appointed to an alternate TaskTracker.

6. In the event that the work is complete, refreshes the status and submit back to the general status of the activity back to the customer by the JobTracker. The smidgen disappointment in JobTracker if outline benefit is going down, at that point every running activity is ended.

TaskTracker:

Errand tracker is additionally a daemon that keeps running on datanodes. Assignment Trackers deal with the execution of individual errands on the slave hub. At the point when a customer presents an occupation, the activity tracker will introduce the activity and partition the work among various undertaking trackers to perform MapReduce assignments. While playing out this activity, the assignment tracker will be all the while speaking with work tracker by sending a pulse. On the off chance that the activity tracker does not get a pulse from errand tracker inside the predefined time, at that point it will expect that undertaking tracker has slammed and relegated that assignment to another assignment tracker in the group.

TaskTracker process:

1. The JobTracker presents the work to the TaskTracker hubs.

2. TaskTracker runs the errands and reports the status of an assignment to JobTracker.

3. It has a component of following the requests of the activity tracker and refreshing the activity tracker with its encouraging status intermittently.

4. TaskTracker will be in consistent correspondence with the JobTracker.

5. TaskTracker disappointment isn't viewed as deadly. At the point when a TaskTracker winds up inert, JobTracker will relegate the assignment to another hub.



Figure 4: MapReduce Architecture

6 SCHEDULING IN HADOOP

In a Hadoop open source, plan the quantity of undertakings in a more effective way since adding up to the consummation of each activity execution transforms into great speed and give an upgraded execution condition. In Apache Hadoop, the Task booking process is a basic part which controls the assignments and allotment of every asset in a productive way. Hadoop MapReduce planning strategies can be sorted into two kinds in view of runtime conduct

A versatile calculation called dynamic:

These techniques are utilized the past, present and future estimations of a parameter to make booking decisions.Example reasonable scheduler, limit scheduler.

Non-versatile calculations called static:

These techniques don't contemplate the progressions occurring in a situation and timetables work assignments according to predefined approach arrange illustration FIFO.

The neighborhood information handling takes lesser time when contrasted with moving the information to the system so as well. Enhance the execution of occupations a large portion of the calculations work to enhance the information area to meet the client desires directing calculations utilize expectation strategies in light of the estimation of information to be handled and the basic equipment. For future work, we ought to build up the calculations which can plan the number employments proficiently on a heterogeneous group. It is specifically identified with the framework assets usage and general execution of the stage. The fundamental aim is to appoint errands to a hub, all the time the nearby guide undertakings are favored on-neighborhood outline regardless of the assignment has a place with which kind of occupation, and a territory marker can be utilized to check hubs and to guarantee every hub a reasonable opportunity to snatch it neighborhood errands.

6.1 FIFO scheduling:

Apache Hadoop open-source utilizes default scheduler is First In First Out. In the event that common bunch gives incredible potential to offering substantial assets to numerous clients, the issue of sharing assets reasonably way between clients we require a superior schedule. In FIFO scheduler absence of execution regarding reaction time of little employment and reasonable sharing among clients. Apache Hadoop's worked in scheduler runs occupations like FIFO way. FIFO scheduler has the single line for all employment and the bunch most established occupations picked first from the leader of the Queue, however, no understanding of the need for the extent of the employment. Not suited for generation testing and improvement. Its preemptive administration implies that the scheduler can suspend errands in pools running over the limit in the request to give the openings to the pool running under limit Priority criteria are likewise doled out to different pools. Employment Tracker pulls most established occupations

first from the activity line. It isn't considered the need or the measure of the activity. Here undertakings are booked in an interleaved way in view of need. After a vocation can be apportioned into singular undertakings, they are stacked into the line and occupations are doled out to free openings subsequently they end up accessible on TaskTracker hubs. Creation occupations are expected to complete in an opportune way while permitting clients who have specially appointed questions to get comes about back arranged by seconds. In the event that a vocation possesses assets for a more drawn out timeframe, at that point, the consequent occupations sitting tight for the execution may endure with a more drawn out holding up period. It is fundamentally created for a solitary sort of employment and exhibits debasement in execution when various occupations are required to be executed by numerous clients. In any case, this schedule won't be a decent decision for a few employees. The disservice of FIFO booking is poor reaction times for short occupations contrasted with expansive jobs.FIFO scheduler is thought to be basic in execution and cost impact scheduler. Indeed, even FIFO scheduler is basic, however, it experiences various constraints.

6.2 Fair scheduler

The fair scheduler purpose is to provide Quality of service (QoS) and fast response times to respond to each job. Fair scheduler manages access to its Hadoop cluster and subsequently released to the Hadoop community. The Fair scheduler follows steps Multiple Jobs are grouped and combined into pools. Jobs are assigned to pools with guaranteed minimum resources. Its capacity is split between jobs. Jobs are not categorized into a default pool. Pools can be specified that the minimum number of map slots, reduce slots, and maximum limit on the number of running jobs. Fair Scheduler can support preemption, if a pool has not received its fair share for a certain period of time, and then the scheduler can kill tasks in pools running over capacity in order to give the slots to the pool running under capacity. Main advantages of Fair schedulers are giving equal priority to short jobs and long jobs with short jobs completing faster. All jobs get an average, an equal share of resource over time. The main objective is to do an equal distribute of computing resource among the jobs in the system. Free resources can be allocated to other pool's excess pool capacity is shared between jobs. Preemption supports fairness among fools priority supports importance within a pool.

6.3 Capacity scheduler

The reasonable scheduler intention is to give Quality of administration (QoS) and quick reaction times to react to each activity. Reasonable scheduler oversees access to its Hadoop bunch and in this manner discharged to the Hadoop people group. The Fair scheduler takes after advances Multiple Jobs are gathered and joined into pools. Occupations are alloted to pools with ensured least assets. Its ability is part between occupations. Employments are not ordered into a default pool. Pools can be indicated that the base number of guide spaces, lessen openings, and greatest cutoff on the quantity of running employments. Reasonable Scheduler can bolster seizure if a pool has not gotten what's coming to its for a specific timeframe, and after that the scheduler can kill assignments in pools running over limit so as to give the openings to the pool running under limit. Fundamental points of interest of Fair schedulers are giving equivalent need to short occupations and long employments with short employments finishing

speedier. All employments get a normal, an equivalent offer of asset after some time. The primary target is to complete an equivalent appropriate of figuring asset among the occupations in the framework. Free assets can be apportioned to other pool's abundance pool limit is shared between occupations. Seizure underpins decency among fools need bolsters significance inside a pool.

7. Conclusions

The coming of Big Data has acted openings like well difficulties to computerized world. The Review center around Challenges of Big Data, Hadoop, MapReduce and Schedulers the present condition of workmanship scheduler, for example, FIFO, Fair and limit is experiencing no less than one of the accompanying issues: discontinuities, over the designation and scarification of execution over reasonable portion. Despite the fact that arrangements, for example, Multi asset pressing scheduler are being created to drop down the issue of Fragmentation, over distribution and enhance the execution yet at the same time they are experiencing the issue of reasonableness.

REFERENCES

[1] Bakshi, K.,(2012)," Considerations for big data: Architecture and approach"

[2] Mukherjee, A.; Datta, J.; Jorapur, R.; Singhvi, R.; Haloi, S.; Akram, W., (18-22 Dec., 2012), "Shared disk big data analytics with Apache Hadoop"

[3] Harshawardhan S. Bhosale1, Prof. Devendra Gadekar, JSPM's Imperial College of Engineering & Research, Wagholi, Pune, a review on Big Data

[4] Aditya B. Patel, Manashvi Birla, Ushma Nair,(6-8 Dec. 2012),"Addressing Big Data Problem Using Hadoop and Map Reduce"

[5] Shilpa, Manjeet Kaur, LPU, Phagwara, India, a review on Big Data and Methodology

[6] Yu Li; Wenming Qiu; Awada, U. ; Keqiu Li,,(Dec 2012)," Big Data Processing in Cloud Computing Environments"

[7] Garlasu, D.; Sandulescu, V; Halcu, I. ; Neculoiu, G. ;,(17-19 Jan. 2013),"A Big Data implementation based on Grid Computing", Grid Computing

[8] Sagiroglu, S.; Sinanc, D. ,(20-24 May 2013),"Big Data: A Review"

[9] Grosso, P. ; de Laat, C. ; Membrey, P.,(20-24 May 2013)," Addressing big data issues in Scientific Data Infrastructure"

[10] Kogge, P.M., (20-24 May, 2013), "Big data, deep data, and the effect of system architectures on performance"