IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 11, November 2016

Survey on Efficient Resource Utilization using Hadoop Cluster for Big Data Processing

Poonam Mahajan[1], Manish Patel[1], Amol Agarwal[1], Nikhil Raut[1], Devendra Gadekar[2]

Bachelors in Engineering, Computer Engineering, JSPM's Imperial College of Engineering and Research, Pune, India[1]

Professor, Computer Engineering, JSPM's Imperial College of Engineering and Research, Pune, India[2]

Abstract: Hadoop is a framework which is used to store and process large amount of data. Hadoop cluster is designed to analyze and store a huge amount of data. As day by day amount of data stored and processed is increasing rapidly, so we need such an optimal scheduling algorithm to meet the requirement. Job Scheduling is an important parameter to achieve high performance in Hadoop cluster. Hadoop scheduler is pluggable module used for resource allocation. Majorly used schedulers are FIFO, Fair and Capacity scheduler. In this paper we compare and classify parameter such as Average Response Time, Average waiting time and Fairness of various scheduling algorithm in different environment.

Keywords: Hadoop, MapReduce, Scheduling.

I. INTRODUCTION

Hadoop is a framework that allows distributed processing of large datasets using simple programming model. Two core components in Hadoop are Hadoop Distributed File System (HDFS) and MapReduce. HDFS is Hadoop file system is used to store huge data in distributed manner [1]. Parallel processing of these huge amounts of data is necessary for fast processing of data sets, MapReduce is the programming paradigms that enables to process huge amount of data in parallel. MapReduce is responsible for performing two tasks mapping task and reducing task. Hadoop frameworks make it possible to distribute all kinds of computation tasks to multiple computers instead of performing the whole task in a single computer. Hadoop allows the user to configure the job, submit it, control its execution, and query the state. The Hadoop scheduling model is a Master/Slave (Master/Worker) cluster structure. The master node (JobTracker) coordinates the worker machines (TaskTracker). JobTracker is a process which manages jobs, and TaskTracker is a process which manages tasks on the corresponding nodes.

The scheduler resides in the Jobrtacker and allocates TaskTracker resources to running tasks. [2] There are three important scheduling issues in MapReduce such as locality, synchronization and fairness. There are many algorithms to solve this issue with different techniques and approaches. Some of them get focus to improvement data locality and some of them implements to provide Synchronization processing. also, many of them have been designed to minimizing the total completion time. This paper provides an overview of four different scheduling algorithms for MapReduce namely Capacity scheduler, FIFO, Job aware scheduler, Fair scheduler. The advantages and disadvantages of these scheduling algorithms are also described.

II. REVIEW OF RELATED LITERATURE

A. FIFO Scheduler

In Hadoop cluster, FIFO is default scheduler in MapReduce [3]. FIFO means first come first serve. The objective of FIFO scheduler is to schedule jobs based on their priorities in first-come first-serve order. FIFO maintain one queue, as per the jobs are coming they store at queue in FIFO manner.

First job will at head of the queue and as per the new jobs are coming they are store at tail side of that queue[3].At the time of job execution, if one job is executed then next job will take from the head side of that queue.

So, all jobs are executed FIFO manner only. Advantages are simple one, easy to implement and perform well in case of short jobs and disadvantages are starvation and pre-emption will be there, less use of resources.

B. Fair Scheduler

The aim of the Fair scheduling algorithm is to do equivalent distribution of total resources among the jobs in the system. Fair scheduling is a process of assigning resources to jobs such that all the jobs get, on average, an equal share of resources over time [4]. If only one job is submitted then it will allocate complete cluster.

When numerous job are submit then free slot is owed and as previous jobs are entire slots are owed for new coming jobs if no new jobs are in queue then it will assign slots to current task for fast completion so that each job nearly get same amount of CPU time. It let small jobs complete within reasonable time while non-starving long jobs [5]. This scheduler in fact organizes jobs by resources pool, and share resources fairly between these pools. The fair scheduler can limit the number of concurrent running task as per pool and user and it can also limit the number of

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 11, November 2016

parallel running task each pool. The traditional algorithms have high data transfer rate and execution time of jobs. Fair scheduling algorithm overcomes the limitation of FIFO such as waiting time of jobs in queue, working in small as well as large cluster and less complex. The main disadvantage of the Fair scheduling algorithm is it assigns resources to job without considering weight of job for each node [6].

C. Capacity Scheduler

It is a default one scheduler in MapReduce 2 and YARN setup [7].In capacity scheduler queues are divided on the basis of users or group of users called organization and then resources mainly divided to serve this request for this organisation.

Multiple queues are specific for organization [7]. e.g.: Suppose job enters in organization a queue so, it will be picked up as no job is running; this will take up as many resources as much are available.

This will effectively utilise a cluster .When job in organization B queue appear task of the first job would be killed to free up the slots for the new job. Advantages are utilization of resources and job efficiency get improves and disadvantage are user requests to know all information of system to make queue set and select group for this job from queue.

D. Job Aware Scheduler

Various scheduling algorithms are being used in MapReduce Job scheduling, but many of them fails to be efficient in heterogeneous cluster [8]. Heterogeneous environment involves resources of various Configuration and composition. Job Aware scheduler overcomes limitations such as limited resource utilization of the cluster and limited applicability towards heterogeneous cluster [9].

Job aware scheduler mostly focuses on three important parameters, firstly reducing the execution time of submitted job, secondly fulfilling earliest deadline first criteria and lastly efficient management of workload of the job. The above parameter results in reducing the average waiting time.

On evaluation, the results show that it reduces the average waiting time of a job by 79% when evaluated using best case and 23% reduction in average case via scheduling the jobs based on job execution time [9].

III.RESEARCH METHODOLOGY

We are proposing a Novel scheduling algorithm which overcome the drawback of traditional scheduling algorithm such as waiting time of jobs in queue, reduce execution time, proper utilization of available resources so that each job can get the resources according to their requirement.

In Fair Scheduling algorithm when multiple jobs are submitted each job in queue get equal share of resources according to the maximum number jobs in each pool [4]. But all the jobs do not require all the assign resources so the part of resources which is not used is in ideal state, so it not proper utilization of resources.

So improving over Fair scheduling algorithm we can assign resources according to weight of the jobs in queue, in improved algorithm resources will shared equity.

Our Proposed algorithm will maintain a tree of jobs, jobs will be structured according to their weight; Lighter weight jobs will be on left most nodes and heavier weight will be on right most nodes. Weight of job will be calculated according to the number of blocks of data it will work on and according to weight of job resources will be allocated, similarly as like Linux process scheduler. When job complete its task it will be removed from tree.

Numbers of job executed at a time will be calculated on the basis of resources present so that all the jobs should be completed as fast as possible with less waiting time and less execution time, if all the jobs come at one time then it will take more time for execution. If we propose this algorithm then it may reduce waiting time of jobs in queue and every job will get proper resources for execution. Hence which will help to increase utilization of resources and decrease average waiting time of jobs in queue.

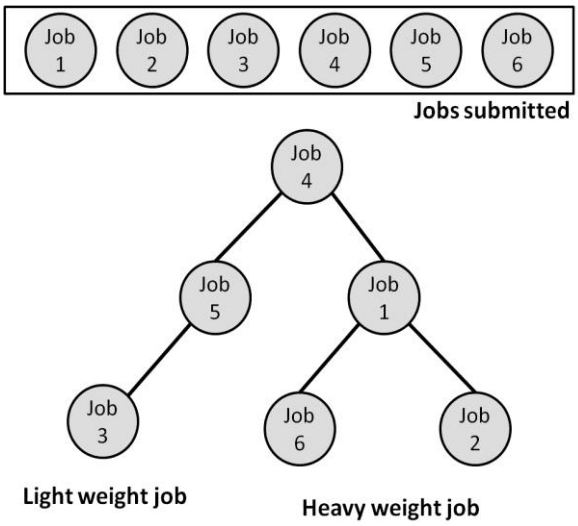

Fig. 1 Tree of Weighted jobs

Fig. 1 shows us the logically maintained tree by the scheduler which will help scheduler to assign resources.

IV.SYSTEM ARCHITECTURE

In the Fig. 2 is the basic architecture of our proposed algorithm which schedules the jobs according to their weight. The weight of the jobs is calculated using the information given by the Namenode to the scheduler, and then scheduler informs the JobTracker to assign the tasks.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 11, November 2016

A. **Basic architecture**

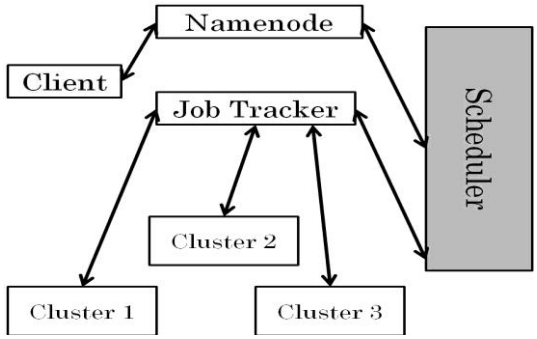

Fig. 2 System Architecture

Fig. 2 shows us the basic architecture of the how scheduler will communicate with Hadoop cluster and schedule the job.

V. DISCUSSIONS

Hadoop store and process huge amount of data and day by day the data is increasing rapidly and processing such huge amount of data has become big problem. There is requirement of such a scheduling algorithm which help job on data to get executed as fast as possible with less waiting time. There are numerous scheduling algorithms which have helped Hadoop to improve its performance in different factors such as Data locality rate, Synchronization overhead, Fairness, Average response time, Job execution time and Average waiting time.

Each of the traditional algorithm have improved it performance in some of its factors. Similarly our proposed algorithm will help to improve its performance. By implementing our proposed algorithm it may help to improve performance factors like in Fairness, Job execution time, Average waiting time, etc.

VI. CONCLUSION

Using this proposed scheduling algorithm we may able to make more and proper utilization of resources of Hadoop cluster. This will help to decrease waiting of jobs in queue and will also reduce the execution time of job. A comprehensive work for increasing the resource utilization, decrease waiting time of job and to decrease execution time of job is the primary purpose of our proposed algorithm. This will be an important module which will help Hadoop to work efficiently and reliably.

ACKNOWLEDGEMENT

We thank our colleagues who provided insight an expertise that greatly assisted the research.

We thank our guide **Prof. Devendra P. Gadekar for** inspiring us to do this project, guidance, support and his comments in the manuscript.

REFERENCES

[1] Hadoop home page. http://hadoop.apache.org/.
[2] "A Review of Adaptive Approaches to MapReduce Scheduling in Heterogeneous Environments" , 978-1-4799-3080-7/14/$31.00_c 2014 IEEE
[3] Yingjie Guo, Linzhi Wu, Wei Yu, Bin Wu, Xiaotian Wang," The Improved Job Scheduling Algorithm of Hadoop Platform", University of Chinese Academy of Sciences 100408, China
[4] Hadoop, "Hadoop fair scheduler" https://hadoop.apache.org/docs/r1.2.1/fair_scheduler.html
[5] B.P Andrews and A. Binu, " Survey on Job Schedulers in Hadoop Cluster ", IOSR Journal of Computer Engineering, Vol.15, NO. 1, Sep.- Oct. 2013, pp. 46-50.
[6] A Comprehensive View of Hadoop MapReduce Scheduling Algorithms, "Fair Scheduling" http://www.ijcncs.org/published/volume2/issue9/p5_2-9.pdf
[7] R.Thangaselvi, S.Ananthbabu, R.Aruna." An efficient Mapreduce scheduling algorithm in hadoop", International Journal of Engineering Research & Science (IJOER), December- 2015
[8] C.He, Y.Lu, and D. Swanson, "Matchmaking: A new mapreduce scheduling technique," IEEE 3rd International Conference on Cloud Computing Technology and Science, December 2011.
[9] Supriya Pati and Mayuri A. Mehta , " Job Aware Scheduling in Hadoop for Heterogeneous Cluster " , 2015 IEEE International Advance Computing Conference (IACC)