Scientific Journal of Impact Factor (SJIF): 5.71

e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 5, Issue 03, March -2018

Performance Comparison of Hadoop Map Reduce and Apache Spark

Anju Parmar[1], Vikrant Bhardwaj[2], Divya Chauhan[3], K L Bansal[4]

[1, 2] *Department of Computer Science, HP University, Shimla*
[3] *JRF, Department of Computer Science, HP University, Shimla*
[4]*Professor, Department of Computer Science, HP University, Shimla*

Abstract -*With the advancement of the electronics and the communication technology information generation rate has gain tremendous growth. Huge –Huge Amount of the data has been generated per hour from various medium of the Internet of Thing (IoT), referred as big data, and is a trending term these days. Enormous Data has been the point of interest for Computer Science devotee around the globe, and has increased considerably more noticeable quality over the most recent couple of years. This paper discusses the comparison of Hadoop MapReduce and Apache Spark. Both Hadoop and Spark are framework for analyzing big data. Although both of these resources are based on the idea of Big Data, their performance varies significantly based on the application under consideration. In this paper two frameworks are being compared along with providing the performance comparison using word count algorithm. In this paper, various datasets has been analyzed over Hadoop MapReduce and Apache Spark environment for word count algorithm.*

Keywords- *Apache Spark, Big Data, Hadoop, HDFS, Map Reduce*

I. INTRODUCTION

Big data is immensely growing with the increase in users on various social networking sites or in industries. Big Data can be characterized in the form of structured, unstructured and semi-structured form. Traditional data processing techniques are unable to store and process this huge amount of data, and due to this, they face many challenges in processing Big Data. Nowadays web traffic, social media content, system data and machine-generated data are growing rapidly. Social networking websites generate new data every second and handling such data a data is one of the major challenges companies are facing. Data which is stored in data warehouses is causing disruption because it is in a raw format, proper analysis and processing is to be done in order to produce usable information out of it. New tools are being used to handle such structured and unstructured type of data in short time. Big data is a data which is difficult to store, process and manage. Big Data is demanding new techniques to analyze and process the data.

Hadoop, [1] a distributed processing framework addresses these demands. It is built across highly scalable clusters of commodity servers for processing, storing and managing data used in advanced applications. Hadoop has two main components-MapReduce and HDFS (Hadoop Distributed File System). HDFS is a file system used by Hadoop. Map Reduce is a programming model of Hadoop.

Apache Spark [2] is an open source big data processing framework with high speed, easy to use, and sophisticated analytics. Spark runs on top of existing Hadoop Distributed File System (HDFS) infrastructure to provide elevated and extra functionality.

II. LITERATURE REVIEW

C. Lakshmi, V. V. Nagendra Kumar [3] have surveyed various technologies to handle the big data and there architectures. In this paper they have discussed the challenges of Big Data (volume, variety, velocity, value, veracity) and various advantages and disadvantages of the technologies used for big data. This paper discusses an architecture using Hadoop HDFS distributed data storage, real-time NoSQL databases, and Map Reduce distributed data processing over a cluster of commodity servers.

. **Priya Dahiya et.al [4]** presented a **s**urvey on Big Data using Apache Hadoop and Spark. This paper shows the architecture and working of Hadoop and Spark. This paper also brings out the differences between Hadoop and Spark. The paper also discusses the challenges faced by Map Reduce during processing of large datasets and how Spark works on Hadoop YARN.

M. Zaharia et al. [5] presented architecture and utility of Apache Spark. The paper gives a brief overview of the Apache Spark programming model, which includes RDDs, parallel computing etc. It also introduces a few implementations in the environment.

Satish Gopalani, Rohan Arora [6] compared MapReduce and Spark. For the performance analysis; they have used a standard machine learning algorithm K-Means. They have taken datasets of with a single node and with two nodes and monitored the performance in terms of the time taken for clustering using K-Means algorithm. The results show that the performance of Spark is considerably high in terms of processing time.

Mantripatjit Kaur, Gurleen Kaur Dhaliwal [7] compared the Hadoop and Apache Spark framework. This paper also provides the performance comparison using word count algorithm. In this paper, various datasets has been analyzed over Hadoop Map Reduce and Apache Spark environment for word count algorithm. The system that comes out to be better is further used to analyze the research dataset of a university.

A.C.Priya Ranjani, **Dr. M.Sridhar [8]** discusses an overview of both Hadoop and Spark frameworks and efficiency of Spark over Hadoop. It was found that Spark is a very strong contender due its ability for in-memory computations, interactive querying and stream processing. Although Spark is reported to work up to 100 times faster than Hadoop in certain circumstances, it does not provide its own distributed storage system. So, it requires one provided by a third-party. For this reason many Big Data projects involve installing Spark on top of Hadoop, where Spark's advanced analytics applications can make use of data stored using the Hadoop Distributed File System (HDFS).

Akaash Vishal Hazarika et al. [9] presented the performance comparison of Hadoop and Spark Engine. In this paper, for performance comparison word count and logistic regression is used. The performance is compared based on execution time.

III. HADOOP

Apache Hadoop consists of a file system HDFS and a Map Reduce engine. Hadoop cluster consists of a single master node and many worker nodes. The master node provides instructions to the slave nodes and computations are performed on the slave nodes. Copies of the same data exist indifferent slave nodes ensuring a fault tolerant working [10].

- **HDFS Architecture [11]**

Apache Hadoop is a fast-growing big-data processing open source software platform. Hadoop is able to handle all kind of data like unstructured, structured, and audio. It runs OS/X, Linux, Solaris, and Windows. Hadoop is flexible, scalable and fault tolerant. It comprises of HDFS. Hadoop HDFS is distributed and scalable file system which is written in Java. HDFS has master/slave architecture. An HDFS cluster comprises of a single Name Node. It is able to handle the file system namespace. The name node is the equal to the address router for the big data application. Moreover, there are a many Data Nodes, typically one per node in the cluster, which manages storage connected to the nodes that they run on. Figure 1 explains [12] HDFS architecture.

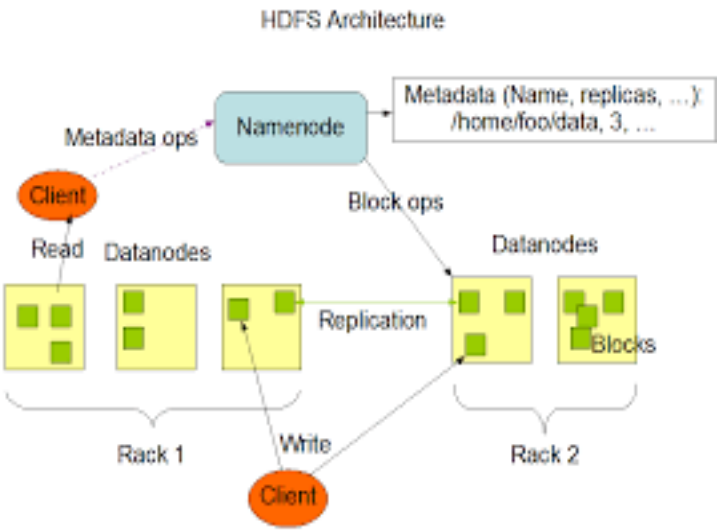

Figure 1 HDFS Architecture [12]

- **MapReduce**

MapReduce was developed by Google. It is massively scalable, parallel processing programming model and software framework for generating large datasets. The term MapReduce stands for two functions: Map and Reduce .Both of Map tasks and Reduce tasks work on key-value pairs. The man idea is to map the input data into key pairs and group together values with the same keys, then the reduce function merges together these values with the same keys [13]. Figure 2 explains [14] MapReduce architecture.

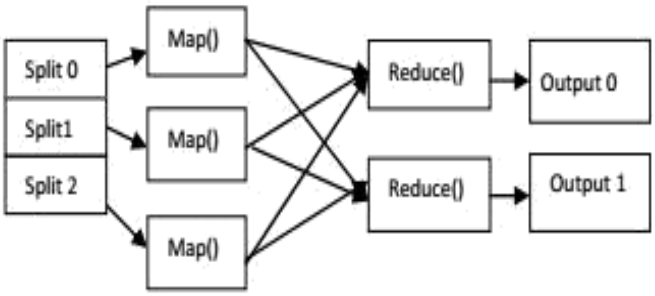

Figure 2 MapReduce Architecture [14]

IV. SPARK

Apache Spark [8] is a high performance framework for analyzing large datasets. It was initially developed by Matei Zaharia at UC Berkeley AMPLab in 2009, and open sourced in 2010 as an Apache project. Spark has several advantages compared to other Big Data and MapReduce technologies like Hadoop and Storm. Spark requires a cluster manager and a distributed storage system. For cluster management, Spark supports standalone (native Spark cluster), Hadoop YARN or Apache Mesos. For distributed storage, Spark can interface with a wide variety including HDFS, Cassandra, Amazon S3.

Spark is also designed to be used easily. It provides APIs in Java, Scala, Python and R shells. It can run on YARN and accessing data from HDFS. It supports iterative machine learning algorithms, batch applications, as well as interactive data analysis tools, while retaining the characteristics of MapReduce such as scalability and fault tolerance. To achieve these goals, Spark introduced RDDs, its fundamental data structure. An RDD is a read-only collection of objects partitioned across a set of machines that can be rebuilt if a partition is lost. It is an immutable distributed collection of objects. It runs faster than Hadoop by caching the data that is to be processed. It is quite flexible and can run on the existing Hadoop framework. Figure 3 explains the Spark Architecture [15].

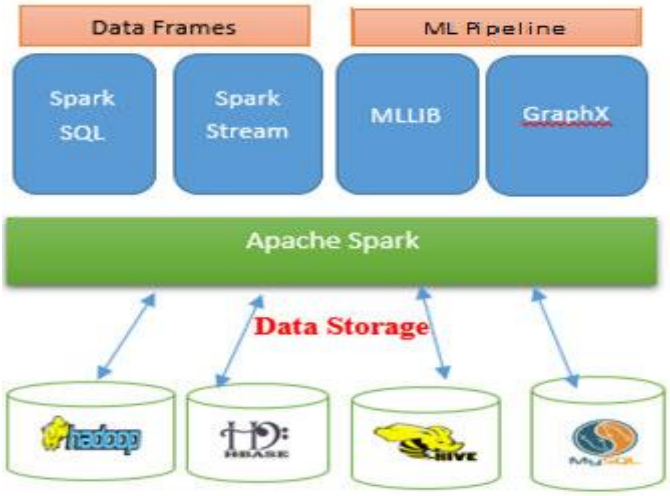

Figure 3 Spark Architecture[15]

V. DIFFERENCE BETWEEN HADOOP AND SPARK FRAMEWORKS

Table 1 shows the difference between Hadoop and Spark frameworks

Table 1 Difference between HADOOP and SPARK frameworks

Sr. No.	Parameters	Hadoop	Spark
1	**Distributed File System**	Own File System	Depends on HDFS
2	**Scalability**	Highly Horizontally Scalable	Horizontal
3	**Message Delivery guarantee**	Exactly-once	Exactly-once
4	**Streaming System**	Do not support streaming	Micro batching
5	**Data Processing Engine**	At the core map reduce is batch processing	At the core spark is batch processing engine
6	**Cost**	Less expensive	More Requirement of RAM increases cost
7	**Data Computation**	Disk-Based	In Memory
8	**Hardware Requirement**	commodity hardware	mid to high-level hardware
9	**Auto-Scaling**	Yes	Yes
10	**Languages Supported**	Primarily Java, but other languages like C, C++, Ruby, Groovy, Perl, Python also supported using Hadoop	Java, Scala, python and R

VI. EXPERIMENT AND PERFORMANCE EVALUATON

In this paper performance evaluation of Hadoop Map Reduce and Apache Spark done and compared. For Comparison Word count algorithm is used. In order to come to a conclusion about the practical comparison of Apache Spark and Map Reduce, we performed a comparative analysis using these frameworks on a datasets of different sizes using the word count algorithm with single node and performance is evaluated based on the execution time.
The system used for evaluation has the following configurations:

- Intel-core i5 processor with 4 GBs of RAM 64-bit architecture
- Linux (Ubuntu 16.04 LTS).
- 500 GB disk
- Hadoop 2.7.4
- Spark-2.2.0
- Eclipse-jee-luna-SR2

The tests were conducted for various datasets having different sizes .The Table 2 shows the execution time of both Hadoop Map Reduce and Apache Spark on various datasets of different sizes.

Table 2 Execution Time Comparison of Apache Spark and Hadoop MapReduce

Sr. No.	Dataset size	Execution Time(sec)	
		Spark	Hadoop
1	31 MB	6	21
2	124 MB	26	54
3	381 MB	43	103
4	761 MB	120	198
5	1.1 GB	180	312

The results related to time taken for execution has been noted for the different datasets for both Hadoop and Apache Spark. Fig. 3 shows the graphical representation of the results for the time taken for execution by both Hadoop and Apache Spark on the word count algorithm on different datasets.

Figure 4 Time taken for Execution by Hadoop and Spark on different datasets

It has been observed that Apache Spark gives better performance in terms of execution time than Apache Hadoop when compared by using word count algorithm on datasets of different sizes on single node.

VII. CONCLUSION

In this paper two programming model Map Reduce and Apache Spark are compared for analyzing their performance using word count algorithm on datasets of different sizes on single node. By implementing both frameworks on various datasets of different sizes using the word count algorithm, performance of Map Reduce and Apache Spark has been compared. In this paper we have found that Apache Spark gives better performance in terms of execution time as compared to Hadoop Map Reduce.

REFERENCES

[1] Jacob,J.P., Basu A, "Performance analysis of Hadoop mapreduce on eucalyptus private cloud" , International Journal of Computer Applications , Vol.17, 2013.

[2] Guanghui, X., Feng, X., Hongxu, M. , "Deploying and Researching Hadoop in Virtual Machines", Proceeding of the IEEE,International Conference on Automation and Logistics,Zhengzhou, China, 2012.

[3] C. Lakshmi*, V. V. Nagendra Kumar, "Survey Paper on Big Data", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 8, August 2016

[4] Priya Dahiya , Chaitra.B , Usha Kumari," Survey on Big Data using Apache Hadoop and Spark", International Journal of Computer Engineering In Research Trends, Volume 4, Issue 6, June-2017

[5] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," HotCloud, vol. 10, no. 10-10, p. 95, 2010.

[6] Satish Gopalani, Rohan Arora, "Comparing Apache Spark and Map Reduce with Performance Analysis using K-Means", International Journal of Computer Applications (0975 – 8887), Volume 113 – No. 1, March 2015

[7] Mantripatjit Kaur , Gurleen Kaur Dhaliwal ,"Performance Comparison of Map Reduce and Apache Spark on Hadoop for Big Data Analysis", International Journal of Computer Sciences and Engineering, Volume-3,Issue-11,2015

[8] A.C.Priya Ranjani , Dr. M.Sridhar **,** "Spark – An Efficient Framework for Large Scale Data Analytics", International Journal of Scientific & Engineering Research, Volume 7, Issue 2, February-2016

[9] Akaash Vishal Hazarika, G Jagadeesh Sai Raghu Ram, Eeti Jain, **"**Performance Comparision of Hadoop and Spark Engine", International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017)

[10] Ivanilton Polato, Reginaldo Ré, Alfredo Goldman,Fabio Kon, "A comprehensive view of Hadoop research—A systematic literature review", Journal of Network and Computer Applications, November 2014

[11] Zan Mo, Yanfei Li, "Research of Big Data Based on the Views of Technology and Application",American Journal of Industrial and Business Management, 2015, 5, 192-197Published Online April 2015 in SciRes.

[12] Raveena Pandya, VinayaSawant, Neha Mendjoge, Mitchell D'silva, "Big Data Vs Traditional Data",International Journal for Research in Applied Science & EngineeringTechnology (IJRASET), Volume 3 Issue X, October

[13] Jeffrey Dean and Sanjay Ghemawat. "Mapreduce: Simplified data processing on large clusters" OSDI'04: Sixth Symposium on Operating System Design and Implementation, December 2004

[14] Jimmy Lin and Chris Dyer, "Data- Intensive Text Processing with Map Reduce", pp. 18-38, Morgan and Claypool publishers.

[15] Jai Prakash Verma, Atul Patel," Comparison of MapReduce and Spark Programming Frameworks for Big Data Analytics on HDFS",IJCSE,, Volume 7 Number 2 March 2016

