JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-11 E-ISSN: 2347-2693

Performance Comparison of Map Reduce and Apache Spark on Hadoop for Big Data Analysis

Mantripatjit Kaur *1 and Gurleen Kaur Dhaliwal2

1*,2 *CSE Department, BBSBEC, Fatehgarh Sahib, India*

www.ijcseonline.org

Received: Oct/22/2015 Revised: Nov /04/2015 Accepted: Nov/17/2015 Published: Nov/30/2015

***Abstract*—**With the unremitting advancement of internet and IT, tremendous growth of data has been observed. Data creation occurring at very fast pace, referred as big data, is a trending term these days. Big Data has been the topic of fascination for Computer Science fanatic around the world, and has gained even more prominence in the last few years. This paper scrutinizes the comparison of Hadoop Map Reduce and the newly introduced Apache Spark – both of which 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 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.

***Keywords*—** Big Data, Hadoop, HDFS, Map Reduce, Apache Spark.

I. INTRODUCTION

We live in data age. It is not easy to measure the total amount of data stored electronically. Amount of data generated every day is expanding in enormous manner. Big data is a popular term used to describe the data which is very large in size. Government, companies many organizations try to acquire and store data about their citizens and customers to know them better and predict the customer behavior [5].Social networking websites generate new data every second and handling such 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-Map Reduce and HDFS (Hadoop Distributed File System). HDFS is a file system of Hadoop. Map Reduce is a programming model of Hadoop.

Apache Spark [14] 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. RELATED TECHNOLOGIES

A. Hadoop

Hadoop, which is a framework that supports the processing of large sets of data in a distributed computing environment. It is a part of Apache project. Hadoop cluster uses a Master/Slave Architecture.

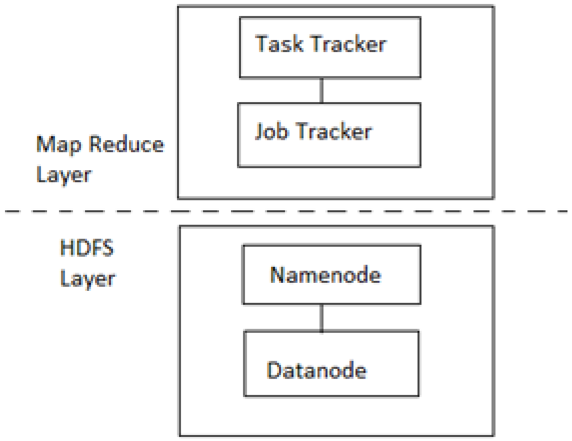

Figure-1 High Level Architecture of Hadoop

Hadoop is a well-known implementation of the Map Reduce model. Hadoop enables applications to work with thousands of nodes and terabytes of data, without bothering the user with too much detail on the assignment and distribution of data and calculation. Hadoop has become a well known and excellent platform in the area of Big Data for data processing. It produces an authentic storage and high performance in the area of Big Data. The hadoop system has two main components: Map Reduce [10] and Hadoop Distributed File System (HDFS) [8]. The Figure-1 above represents the two main components of Hadoop.

B. Map Reduce

Map reduce is a programming model that is used by Hadoop framework to process the data [3]. Map-Reduce basically uses the java programming with Hadoop. Map Reduce model breaks the big data into small portions called chunks and performs operations on those chunks of data. The Map Reduce programming model simplifies the complexity of running parallel data processing functions across various nodes in a cluster, by allowing a programmer with no specific knowledge of parallel programming to create Map Reduce functions running in parallel on the cluster. Map Reduce automatically handles the collecting of results across the multiple nodes and returns a single result or a set of results. More importantly, the Map Reduce runtime system offers fault tolerance that is completely transparent to programmers. Figure-2 below represents the high level architecture of Map reduce.

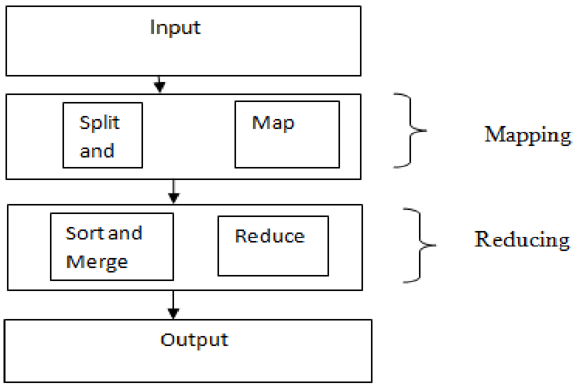

Figure-2 Map Reduce Architecture

C. Hadoop Distributed File System (HDFS)

HDFS is a distributed file system that runs on top of the local file systems of the cluster nodes and can store abundantly large files suitable for streaming data access. HDFS is exceedingly fault tolerant and can scale up from a single node to thousands of nodes, each offering local computation and storage. HDFS has of two types of nodes, namely, a name node called master and several data nodes called slaves.

D. Apache Spark

Spark is an open source computing framework specialized in data analytics, It is build on the top of Hadoop HDFS. The spark [13] programming model is inspired by the parallel abstraction of Map Reduce. Keeping favorable properties of Map Reduce, such as parallelization, fault-tolerance, data distribution and load balancing, Spark adds support for iterative classes of algorithms, interactive applications and algorithms containing common parallel primitive methods like join and match.

Spark loads the necessary data in to the cluster memory based on user's application and apply calculation directly in memory making it faster than the traditional Hadoop-HDFS approach. In opposition to Hadoop that requires loading the necessary data from the HDFS in every iteration, Spark keeps the data in memory in between iterations. Because of this mechanism, Spark is very suitable for algorithms that iterates on the data. Figure-3 below presents the model of Apache Spark which shows the reason why Spark is gaining its popularity.

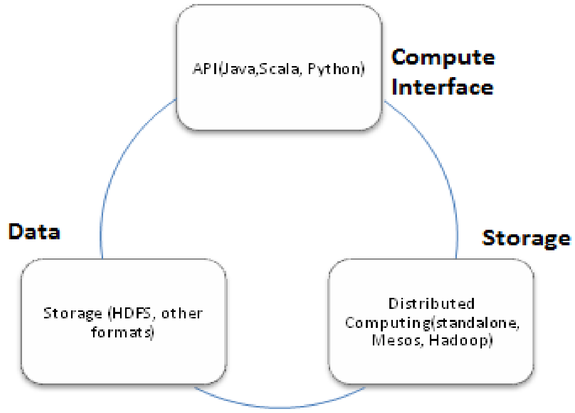
API(Java,Scala, Python)
Compute Interface
Data
Storage
Storage (HDFS, other formats)
Distributed Computing(standalone, Mesos, Hadoop)

Figure-3 Apache Spark Model

III. PERFORMANCE EVALUATION

A. Methodology

In this paper a systematic evaluation of Hadoop Map Reduce is done and its performance is compared with another Big data framework, Apache Spark. For this purpose, we evaluated the word count algorithm on various datasets of different sizes [15]. Considering word count example for text documents, experiments were performed on both map reduce and spark. All the experiments are implemented in Java, where the application runs on single node cluster, and results of analysis are shown in the form of bar charts. From the result analysis, the system that gives better performance, is used to analyze the research data of a university.

B. Experimental Setup

Tests were conducted on a Hadoop and Apache Spark. Hadoop is built on a single node having Intel-core i5 processor with 8 GBs of RAM 64-bit architecture. The operating system is Linux (Ubuntu 14.04). To benchmark the performance, the stable release of Hadoop and Spark namely Hadoop-2.5.2 and Spark 1.5.1 were chosen. All the experiments were performed in Java using Eclipse 3.8 IDE for both map reduce and spark. Daemons for hadoop running on the machine include Name Node, Secondary Name Node, Job Tracker, Data Node, Task Tracker, where as daemons for spark includes Master and Workers

C. Results

1). As mentioned earlier, the purpose of this study was to evaluate the Map Reduce-HDFS performance with Spark-HDFS performance under the same setup for word count algorithm. The tests were conducted for various datasets having different sizes ranging from 1.5 MB (approx) to 322 MB (approx). The datasets are stored on HDFS. Map Reduce job and Apache Spark job are run on these datasets one by one to get the desired output. The analyzed result shows count of each keyword in a file and time to get the result. The Table-1 shows the execution time of both Map Reduce and Apache Spark on various datasets.

Dataset Size	**Execution Time (in sec)**	
	Apache Spark	***Hadoop Map Reduce***
1.5 MB	7.203	30.549
3.8 MB	7.961	31.422
5.7 MB	8.106	31.940
13.5 MB	9.413	32.075
38.1 MB	12.841	32.783
61.3 MB	14.694	39.281
81.5 MB	15.881	50.295
321.5 MB	58.266	140.021

Table 1. Execution Time Comparison of Apache Spark and Map Reduce

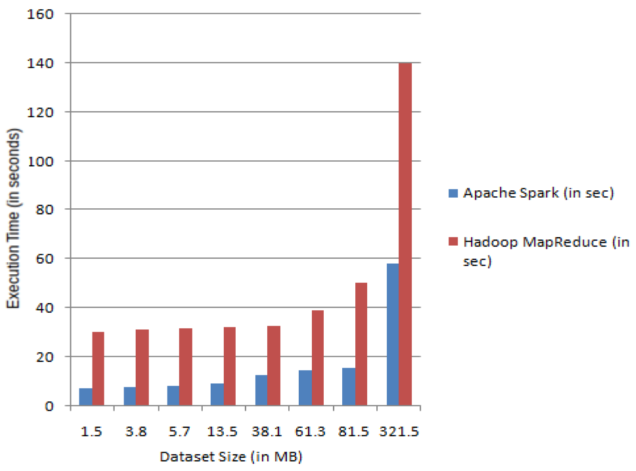

Figure- 4 Performance Comparison of Spark And Map Reduce

The Figure-4 above represents the performance comparison of both the models in graphical manner. The figure clearly displays that Apache Spark is far better than Map Reduce.

2). From the result analysis of first experiment, Apache spark comes out to be the better choice for word count application. On the basis of these results word count algorithm has been performed on a research dataset to find the top ten research areas in the field of Politics and Public Administration. The research study demands an appropriate technique of data analysis with the word count algorithm. Data collection is done mainly from secondary sources. The published data set has been collected from website of SavitriBai Phule Pune University [12]. The dataset contains records from 1953 to 2011. Research dataset is stored on local file system. Text cleaning has been done by removing corrupted, erroneous, misleading and empty fields. It has been then copied from local file system to HDFS. The Word Count module is developed in the Java programming language and then the .JAR file is uploaded to single node storage. The data is then processed using spark for word count algorithm in order to find top ten interest areas of research. Top ten keywords are selected form dataset which are having maximum occurrences. After processing the data, the output is stored on HDFS and copied back to local file system. The Word Count Algorithm has been successfully applied to the datasets. The results are shown in the figure below. Figure 5 shows the top ten research data trends in a particular domain in graphical format. Figure show that in the PPA(Politics and Public Administration) domain, the majority of focused research areas are Politics, Administration, Public Health, Government Policy. Results are possible using Apache Spark Word Count algorithm.

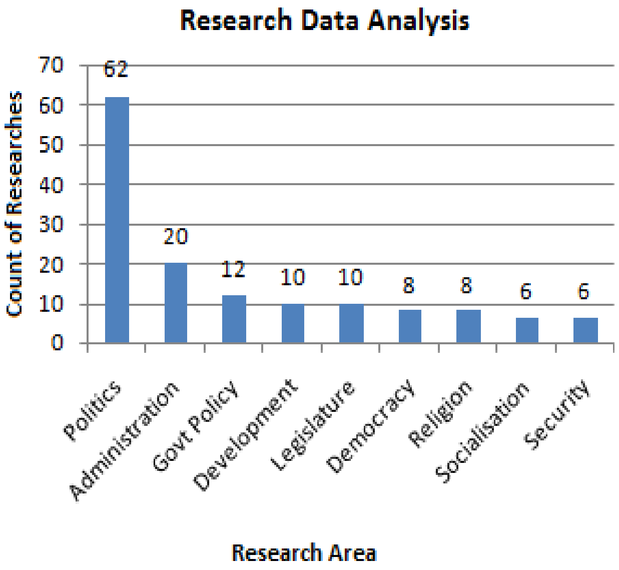

Figure-5 Trend of Research Studies in a particular domain

IV. CONCULUSION

In this paper two programming model Map Reduce and Apache Spark has been presented for analyzing their performance. Hadoop Map Reduce and Apache Spark both can cope with every type of data- structured, unstructured or semi-structured. By implementing both frameworks on various datasets of different sizes, performance of Map Reduce and Apache Spark has been compared. Apache Spark gives far better performance in terms of execution time as compared to Map Reduce. Hence Apache Spark is used to analyze the research dataset of a university in a particular domain. Outcome of the study can be useful for the researchers in their study.

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] Ezhilvathani, A., Raja, K.,“Implementation of Parallel Apriori Algorithm on Hadoop Cluster”, IJCSMC, Vol. 2, **2013** pp**.513 – 516.**

[4] Zaharia,M., Chowdhury, M., Franklin J, Shenker, S., Stoica, I., " Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing". Technical Report UCB/EECS-2011-82, EECS Department, UC Berkeley, **2011.**

[5] Peng, W.,, Yan, Q., Hua, Y. “Analysis and Study on the Performance of Query based on NoSQL Database”, Computer modelling & new technologies , **2014**, pp.**153-159** **.**

[6] Wang, L., Tao, J., Ranjan, R., Marten, H., Streit, A., Chen, J., Chen, D.,. “G-Hadoop: MapReduce across distributed data centers for data-intensive computing” , Parallel and Distributed Processing Symposium Workshops and Phd Forum ,IEEE 26th International , **2012,** pp.**2004-2011.**

[7] Rao,B.T., Sridevi N.V.,Reddy V.K., Reddy L.S.S.“Performance Issues of Heterogeneous Hadoop Clusters in Cloud Computing”, Global Journal of Computer Science and Technology ,**2011**,Vol.11, Issue 8.

[8] Pradeepa, A., Thanamani, A.S. “ Hadoop file system and fundamental concept of mapreduce interior and closure rough set approximations**”,** International Journal of Advanced Research in Computer and Communication Engineering ,Vol. 2, Issue 10, **2013**.

[9] Lee, C., Hseieha, K., Hsieha, S., Hsia, H.“ A Dynamic Data Placement Strategy for Hadoop in Heterogeneous Environments," Big Data Research, Vol. 1, **2014**, pp.**14–22**.

[10] “Hadoop in Action” by Chuck Lam.

[11] White, Tom, 2011.“Hadoop the definitive guide” O’ Reilly media, Inc., CA.

[12] SBPU University Research Dataset: http://www.unipune.ac.in/dept/mental_moral_and_social_science/politics_and_public_administration/ppa_webfiles/pdf/new11/Link_Archives_PhDThesisList2011.pdf

[13] Apache Spark, http://spark.apache.org/

[14] Amp Lab web page : https:// amplab.cs.berkeley.edu/projects/spark- lightning-fast-cluster-computing

[15] http://www.gutenberg.org/ebooks/2600