Techniques Management Big data Apache Hadoop and Apache Spark and which is better In structuring and processing data

Ahmed M Altriki
Department of Information System
IT College, University of Benghazi
Benghazi, Libya
Ahmed.Altriki@uob.edu.ly

Osama Alarafee
Department of Information System
IT College, University of Benghazi
Benghazi, Libya
Osama.Alarafee@uob.edu.ly

Abstract:
Newly considered both Big data Apache Hadoop and Apache Spark It is an important techniques in managing huge databases and thus this article provides an overview of Hadoop MapReduce. Apache Spark is used in big data increase the number of users of social networking sites of various types or industries. Big Data can be classified 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 and demands of the end user. This paper also shows a comparison of certain criteria to determine which is better accordingly.

Keywords: Big Data; Hadoop; Map Reduce; Spark.

1. Introduction:

Recently, Big data is gaining popularity with the increase in the number of users of social networking sites of various types. Big Data can be classified 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 and demands of the end user.

Nowadays web traffic, social media content, system data and machine-generated data are growing rapidly. We should know the five V's of Big Data 1, 2 are velocity, volume, veracity, variability, and variety. Velocity depends on the speed how fast data is growing and processing. Volume determines the size of data whether it can be called as big data or not. Veracity determines the quality of captured and processed data. Variability depends on the inconsistency in data, if data is more inconsistent various techniques are used to manage this. Variety is the type i.e. structured, unstructured and semi-structured and nature of data. Apache Hadoop is a software framework which is open-source and used to store, manage and process data sets using MapReduce programming model. [1]

Also, Apache Hadoop has two parts; the Hadoop Distributed File System (HDFS) and Map Reduce. HDFS is used for storing data in distributed environment. MapReduce is one of the key approaches to meet the demands of computing massive datasets as well as it processes information in the nodes by executing parallelly in the system. MapReduce is popular for its simplicity, scalability, and fault-tolerance. The MapReduce challenges

are put into three categories: online processing, security and privacy, and data storage. Apache Spark is a cluster computing framework which is open-source and used to program clusters with implicit parallelism and fault-tolerance. It is used for performing fast and real-time analysis at a lightning fast speed which cannot be done by Hadoop.

2. Literature Review

First, we show the paper proposed by Vibhavari Chavan and Rajesh. N. Pursue [1] describes what is Big data and what are five V's, i.e., velocity, volume, veracity, variability, variety. In this paper, the author has given the working of Hadoop HDFS and MapReduce working. To analyze the huge amount of data author has considered Apache Hadoop as a working model. Data is stored in HDFS and to process a large amount of data it has used the concept of key-value pairs.

Ankush Verma [2] proposed a strong paper in which gives the difference between Hadoop and Spark. Also, the paper shows the working of the two, and a comparative study is done on them and in good detail. Spark has overcome the limitations which are present in the traditional system and Better performance is an important factor for maintaining a massive amount of data.

The working and architecture of Spark when it is used along with Hadoop YARN is shown by the authors Wei Huang, Lingkui Meng, Dongying Zhang, and Wen Zhang [3]. YARN works in a heterogeneous environment for Apache storm and Tez. Also, analyzing and processing large data is quite a tedious job. The traditional systems are not capable enough to MapReduce. Spark uses (DAG) to divide operators into many stages of tasks and (RDD) for fault tolerance. Also spoken by Priya Dahiya 1, Chaitra.B 2, Usha Kumari [4] regarding their search which focus on the architecture and working of Apache Hadoop and Apache Spark and the challenges faced by MapReduce.

Finally, Katarina Grolinger, Michael Hayes [5] proposed a paper which describes briefly about the challenges faced by MapReduce while processing this huge amount of data.

3. Hadoop and Spark Architecture

3.1 Hadoop Architecture

Hadoop Map Reduce: It is an open-source framework for writing applications. It also processes structured and unstructured data that are stored in HDFS. Hadoop MapReduce is designed in a way to process a large volume of data on a cluster of commodity hardware.

Map Reduce can process data in batch mode**.** Hadoop stores and compute data by creating clusters using many computers. It can be designed for a single server or thousands of machines to compute and store significant data. It uses MapReduce algorithm to run the application, where the data is processed in parallel with others. In MapReduce (MR) one node acts as a master node and other nodes as slave nodes where master node handles the slave nodes that mean it follows master-slave architecture. Hadoop consists of HDFS and MR. **It consists of two layers (HDFS Layer** and **MapReduce layer)**

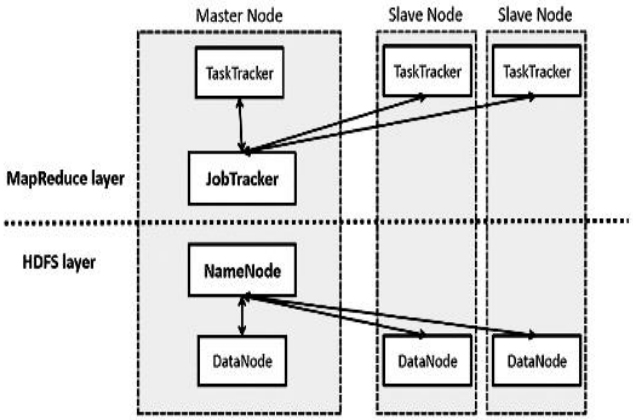

Fig 1. Architecture of Hadoop

3.1.1 HDFS Layer: since the data is huge and single node is not capable of storing it, HDFS is used as an alternative. HDFS is based on Google file system (GFS) to store data in multiple nodes by splitting the huge data into smaller parts. The HDFS is mainly designed for storing datasets and to use them for user applications. HDFS has Data Nodes which act as a master and Name Node which act as a worker 3.

These nodes are used for performing functions like reading, write, create and delete. Name Node is used for requesting the access permission. If the request is granted, Name Node converts filename into block IDs and multiple Data Nodes stores the block and return the list to the client as shown in figure 1. Name node has information about all Data Nodes i.e., data stored by them, which nodes are working actively and which are not i.e. passive nodes, about the free space and job tracker is working efficiently or not. [2]

3.1.2 Map Reduce layer: a large amount of data needs to be properly processed in a distributed environment, and for processing, this data Map Reduce is used. The advantage that MapReduce has is, it makes the workload for handling large data by processing it parallelly on clusters of computers. This makes the system more reliable and fault-tolerant. As shown in fig.1 MapReduce layer has a Job Tracker which is used for assigning tasks to the Task Tracker 3. Task Tracker of slave node gets the tasks to be completed from the Master node Job Tracker.[5]

3.2 Spark Architecture

It is an open-source big data framework. It provides a faster and more general-purpose data processing engine. Spark is basically designed for fast computation. It also covers a wide range of workloads — for example, batch, interactive, iterative, and streaming**.** Spark ran on top of Hadoop and used for streaming of data which is in real-time. Spark supports machine learning, SQL queries, graph data processing and streaming data for analysis of big data [4]. Since traditional MapReduce failed to work properly for real-time data, Spark is used as an alternative. Virtual Machine (JVM). Spark consists of cluster manager, driver program (spark context), executor or worker and HDFS as shown in fig 2. In spark, a Driver program is considered as the main program.

Spark Context is for the coordination of the applications which run on clusters as a set of processes. Processes used for applications are assigned uniquely i.e. they all have their processes and due to these tasks run in multiple threads, and they must have connectivity to worker nodes. These worker nodes run computations and store the data. Programming is written in java or python language which is sent to the executor, and it runs the tasks8. The two main key concepts used in Apache Spark are Resilient Distributed Datasets (RDD) and Directed Acyclic Graph (DAG)[3].

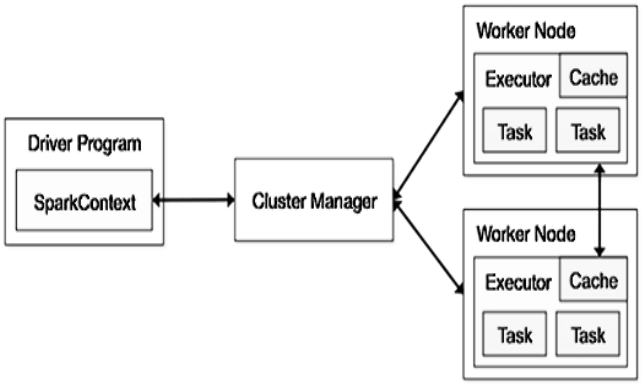

Fig 2. Architecture of Spark

3.2.1 Resilient Distributed Dataset (RDD): it works as a collection of elements which are operated in parallel. In the distributed file system Spark runs on Hadoop cluster, and RDD is created from files in the format of text or sequence files. RDD is used for reading the objects in the collection and when some partition is lost, it can be rebuilt because RDDs are distributed across a set of machines. The two operations supported by RDD: Transformations and Actions.

3.2.2 Directed Acyclic Graph

The data flow is cyclic in Directed Acyclic Graph (DAG) engine. To perform on cluster, a DAG of task stages is created by each Spark job. In map and reduce stage DAG is created 4. It just takes one single stage to complete simple jobs and multiple stages to complete complex jobs in one single run. Thus, jobs completed faster than MapReduce [3].

4. Discussion

After studying considering the previous studies in this research remains There is an increasing curiosity among big data professionals to choose the best framework between Apache Spark and Hadoop, often mistaking them to be the same. However, with considerable similarities, Hadoop and Spark are not directly comparable products. Hadoop and Spark are both Big Data frameworks but they do not necessarily perform exactly the same task. Neither are they exclusive of each other.[7]

Also, several big data projects require installing Spark on top of Hadoop so that the advanced analytics applications of Spark can work on the data stored by Hadoop Distributed File System (HDFS). However, Hadoop and Spark can also work without the other. In addition to HDFS, Hadoop also comes with a processing component called MapReduce to get the data-processing done. On a similar note, Spark can also be integrated into any cloud-based data platform, besides HDFS, whose data can be used for its analytics function.[6]

We have also tried to establish criteria for a clearer and more comprehensive comparison and to shed some more light on the Spark vs Hadoop debate, let's take a look at each of them separately.

Standard	Hadoop Map reduce	Apache Spark
Data Processing	Slow as MapReduce operates in various sequential steps	Its real-time data processing capability makes Spark a top choice for big data analytics
Real-Time Analysis	MapReduce fails when it comes to real-time data processing, as it was designed to perform batch processing on voluminous amounts of data	It can process real-time data
Ease of Use	easy to use	Spark easier to use than Hadoop
Graph Processing	Slow in graphic processing	Fast in graphic processing
Fault Tolerance	Hadoop achieves fault tolerance through replication (have good fault tolerance ability more tolerant than Spark)	(have good fault tolerance ability)Spark uses RDD and various data storage models for fault tolerance by minimizing network I/O
Security	Hadoop MapReduce has better security features than Spark	Spark's security is currently in its infancy, offering only authentication support through shared secret (password authentication)
Cost	Low cost	High cost
Compatibility	yes	yes

Terms of Performance	Fast performance	is 100 times speedier than Hadoop in performance
Resilience	is naturally resilient to system faults or failures as data are written to disk after every operation	has built-in resiliency by virtue of the fact that it arranges data in Resilient Distributed Datasets
fault tolerance ability	yes	yes
Open Source	yes	yes
Expandable	yes	yes

5. Conclusion

Due to the importance of data storage and processing on a daily basis with the large number of users, here is the importance of the most important to develop a storage mechanism to illustrate these data and explain the treatment with large size. In this research clarification of previous studies of a number of research and then was studied all points of difference and similarity in the comparison of Apache Spark vs Hadoop Map reduce Clarify which is best in handling and storing data in terms of several criteria (security, performance, flexibility, cost, processing speed, etc.)

As a future work, we seek to expand our study of more papers in this field and apply one of the two methods to a case study and to obtain results that contribute to explain the differences in more details.

6. References

[1] Chavan, V . and Phursule, R.N., 2014. Survey paper on big data. *Int. J. Comput. Sci. Inf. Technol*, *5*(6), pp.7932-7939

[2] Verma, A., Mansuri, A.H. and Jain, N., 2016, March. Big data management processing with Hadoop MapReduce and spark technology: A comparison. In *Colossal Data Analysis and Networking (CDAN), Symposium on* (pp. 1-4). IEEE

[3] Huang, W., Meng, L., Zhang, D. and Zhang, W., 2017. In-memory parallel processing of massive remotely sensed data using an apache spark on hadoop yarn model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(1), pp.3-19

[4] Marone, R.M., Camara, F. and Ndiaye, S., 2017, December. LSIS: Large scale instance selection algorithm for big data. In *Computer and Communications (ICCC), 2017 3rd IEEE International Conference on* (pp. 2353-2356). IEEE.

[5] Grolinger, K., Hayes, M., Higashino, W.A., L'Heureux, A., Allison, D.S. and Capretz, M.A., 2014, June. Challenges for mapreduce in big data. In Services (SERVICES), 2014 IEEE World Congress on (pp. 182-189). IEEE

[6] Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J. and Ghodsi, A., 2016. Apache spark: a unified engine for big data processing. *Communications of the ACM*, *59*(11), pp.56-65.

[7] Reyes-Ortiz, J.L., Oneto, L. and Anguita, D., 2015, January. Big data analytics in the cloud: Spark on hadoop vs mpi/openmp on beowulf. In *INNS Conference on Big Data* (Vol. 8, p. 121).