ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

A Narrative Study on Big Data

Meenakshi Jaiswal, Rubal Jeet

Assistant Professor, Department of IT, Chandigarh Engineering College, Landran, India

Assistant Professor, Department of CSE, Chandigarh Engineering College, Landran, India

ABSTRACT: Big Data, the analysis of large quantities of data to gain new insight has become a ubiquitous phrase. In recent years, day by day the data is growing at a staggering rate. One of the efficient technologies that deal with the Big Data is Hadoop and Apache Spark, which will be discussed in this paper. This paper includes many libraries, bindings for different popular languages etc. Objectives were to compare Hadoop and Spark .As a result Spark continues to be heavily developed and maintained, next generation software for big data processing are being released. Apache Spark was able to analyze streamed tweets with very minor latency of few seconds. This proves that, despite being big general purpose, Interactive and flexible big data processing engine [3]. The process of analyzing big data using spark, the couple of improvement areas were identified as of importance should be persuaded as future work.

KEYWORD: Big Data, Apache Spark, Hadoop YARN, HDFS, Hadoop MapReduce.

I. INTRODUCTION

Big data analytics helps to discover pattern, unknown correlations, market trends, customer preferences and other useful information by the process of collecting, organizing and analyzing large sets of data ("big data") Not only will big data analytics help you to understand the information contained within the data, but it will also help identify the data that is most important to the business and future business decisions. [7] Techniques. A huge collection of both structured and unstructured data named “big data” cannot be handled by traditional database and software techniques. In most enterprise scenarios the volume of data is too big or it moves too fast or it exceeds current processing capacity. Despite these problems, big data has the potential to help companies improve operations and make faster, more intelligent decisions. This huge volume of variety data with high velocity is captured, formatted, manipulated, stored, and analyzed The primary goal of big data analytics is to help companies to gain useful insight to increase revenues, get or retain customers, and improve operations.

3 Vs of Big Data

1. Volume of data: Volume refers to amount of data. Volume of data stored in enterprise repositories have grown from gigabytes and petabytes to zettabytes.
2. Variety of data: Variety refers to different types of data and sources of data. Data variety exploded from structured and legacy data stored in enterprise repositories to unstructured, semi structured, audio, video, XML etc.
3. Velocity of data: Velocity refers to the speed of data processing. The real-time data can help researchers and businesses make valuable decisions that provide strategic competitive advantages and ROI if you are able to handle the velocity. [8]

II. SOLUTION TO BIG DATA PROBLEM: HADOOP

Apache Hadoop is an open-source software framework framework that allows for distributed processing of large data sets (big data) across computer clusters using simple programming models. Hadoop can scale from single computer systems up to thousands of commodity systems that offer local storage and compute power. Hadoop, in essence, is the ubiquitous 800-lb big data gorilla in the big data analytics space. Hadoop is composed of modules that work together to create the Hadoop framework.

The primary Hadoop framework modules are:

· Hadoop Distributed File System (HDFS)
· Hadoop YARN
· Hadoop MapReduce

A. HDFS- Hadoop Distributed File System- Hadoop Distributed File System [6] is an opensource file system that has been designed specifically to handle large files that traditional file system cannot handle. The large amount of data is split, replicated and scattered on multiple machines. The replication of data facilitates rapid computation and reliability. That is why HDFS can also be called as self-healing distributed file system meaning that, if a particular copy of the data gets corrupt or more specifically to say if the DataNode on which the data was residing fails then replicated copy can be used. This ensures that the on-going work continues without any disruption**. [6]** HDFS has master and slave architecture.

B. Hadoop YARN- To solve these limitations, the open source community proposed the next generation MapReduce called YARN (Yet Another Resource Negotiator) [6].Computer scientists and engineers are trying hard to eliminate these limitations and improve Hadoop. YARN eliminates scalability limitation of the first generation MapReduce paradigm. The earlier version of Hadoop did not have YARN but it was added in the Hadoop 2.0 version to increase the capabilities. A more general processing platform is provided by YARN-based architecture of Hadoop 2.0 that is not limited to MapReduce. YARN's basic idea is to split up the two major functionalities of the JobTracker, resource management and job scheduling into separate daemons. The idea is to have a global ResourceManager and perapplication ApplicationMaster. The ResourceManager arbitrates resources among all the applications in the system and it has two components: Scheduler and Applications Manager. The Scheduler is responsible for allocating the resources among running applications. The ApplicationsManager accepts job-submissions, negotiating the first container for executing the application and provides the service for restarting the ApplicationMaster container on failure. The resource management abilities that were present in MapReduce are also acquired by YARN which tones up MapReduce in processing the data more efficiently. With YARN multiple applications can run in Hadoop all sharing a common resource management**.[6]**

C. Hadoop MapReduce- MapReduce [5] [6] is an important technology which was proposed by Google. MapReduce is a simplified programming model and is a major component of Hadoop for parallel processing of vast amount of data. It relieves programmers from the burden of parallelization issues while allowing them to freely concentrate on application development. **[ijetae]** The original data will be given as input to the Map phase which performs processing as per the programming done by the programmers to generate intermediate results. Parallel Map tasks will run at a time. Firstly, the input data is split into fixed sized blocks on which parallel Map tasks are run. The output of the Map procedure is a collection of key/value pairs which is still an intermediate output. These pairs undergo a shuffling phase across reduce tasks. Only one key is accepted by each reduce task and based on this key the processing will be done. Finally the output will be in the form of key/value pairs.
Hadoop is one of the widely-adopted cluster computing frameworks for processing of the Big Data. Although Hadoop arguably has become the standard solution for managing Big Data, it is not free from limitations. MapReduce has reached scalability limit of 4000 nodes . Another limitation is Hadoop's inability to perform fine-grained resource sharing between multiple computation frameworks.

III. TECHNIQUES AND TECHNOLOGY: SPARK ON HADOOP

The Apache Spark was developed as "a fast and general engine for large-scale data processing." By comparison, and sticking with the analogy, if Hadoop's Big Data framework is the 800-lb gorilla, then Spark is the 130-lb big data cheetah. Spark is 100 times faster than Hadoop MapReduce in context of in-memory processing, ten times faster on disk. Spark can also perform batch processing, however, it really excels at streaming workloads, interactive queries, and machine-based learning.

Spark has its own page because, while it can run in Hadoop clusters through YARN (Yet another Resource Negotiator), it also has a standalone mode. The fact that it can run as a Hadoop module and as a standalone solution makes it tricky to directly compare and contrast. However, as time goes on, some big data scientists expect Spark to diverge and perhaps replace Hadoop, especially at instances where faster access to processed data is critical. Apache Spark is a distributed and highly scalable system, providing the ability to develop applications using languages like Java, Scala (the language used to write Spark itself), Python and R. It was originally developed at the University of California, Berkeley and donated to the Apache Software Foundation in 2013[4]

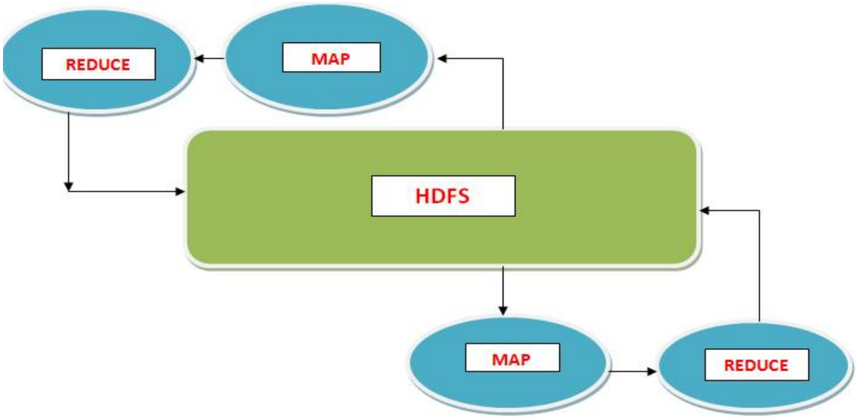

Fig 1 HADOOP

Spark is a cluster-computing framework, which means that it competes more with MapReduce than with the entire Hadoop ecosystem. For example, Spark doesn't have its own distributed filesystem, but can use HDFS.

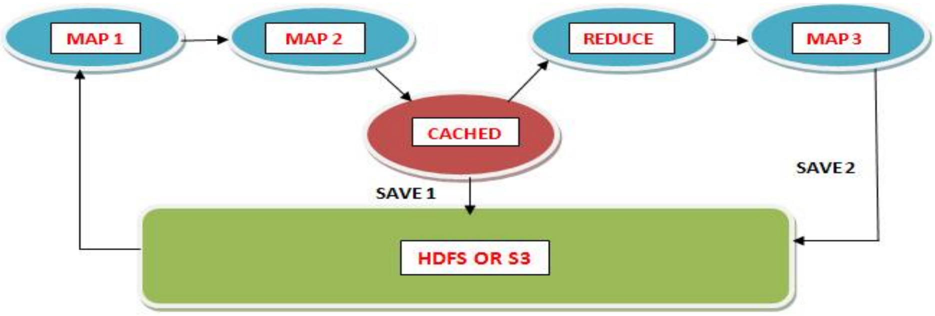

Fig 2 SPARK

Spark consists of four main interoperable components showed in figure 1. Spark Core is the foundation of the Spark project and contains features such as task scheduling, memory management and fault recovery. On top of it lies four modules, of which two are used [4] are:

• Spark Streaming leverages Spark Core capabilities to enable processing of live streams of data. It is used extensively in this project for fetching data from social networks.

• MLlib is a library containing machine learning algorithms that can be applied to compute statistical models from data.

When Not to Use Apache Spark:-

The following two scenarios can hinder the suitability of Apache spark:-

1) Low Tolerance to Latency requirements: If big data analyses are required to be performed on data streams and latency is the most crucial point rather anything else. In this case using Apache Storm may produce better results, but again reliability to be kept in mind.[3]

2) Shortage of Memory resources: Apache Spark is fasted general purpose engine due to the fact that it maintains all its current operations inside Memory. Hence requires access amount of memory, so in this case when available memory is very limited, Apache Hadoop Map Reduce may help better, considering huge performance gap.[3]

IV. CONCLUSION

The paper consists of applications and services developed around Hadoop and Apache Spark. A useful software for processing heavy data from various sources in real time, they are databases or online APIs. It's including many libraries, bindings for different popular languages etc. Objectives were to compare Hadoop and Spark .As a result Spark continues to be heavily solely focused on streaming, making it equivalent to the Spark Streaming module. Apache Spark was able to analyze streamed tweets with very minor latency of few seconds. This proves that, despite being big general purpose, Interactive and flexible big data processing engine [3]. The process of analyzing big data using spark, the couple of improvement areas were identified as of importance should be persuaded as future work.

REFERENCES

1. V Srinivas Jonnalagadda at al, " A Review Study of Apache Spark in Big Data Processing" International Journal of Computer Science Trends and Technology (I JCST) – Volume 4 Issue 3, May - Jun 2016.
2. Jorge L. Reyes-Ortiz et al "Big Data Analytics in the Cloud: Spark on Hadoop vs MPI/OpenMP on Beowulf" Procedia Computer Science Volume 53, 2015, Pages 121–130.
3. Abdul Ghaffar Shoro et al "Big Data Analysis: Ap Spark Perspective" Global Journal of Computer Science and Technology: C Software & Data Engineering. Volume 15 Issue 1 Version 1.0 Year 2015.
4. Benjamin Fovet "Machine Learning and DataMining with Apache Spark" 2015
5. J. Christy Jacksona et al "Survey on Programming Models and Environments for Cluster, Cloud, and Grid Computing that defends Big Data. ScienceDirect, 2nd International Symposium on Big Data and Cloud Computing (ISBCC'15).
6. Amogh Pramod Kulkarni et al "Survey on Hadoop and Introduction to YARN" International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 5, May 2014).
7. Dr. Siddaraju et al " Efficient Analysis of Big Data Using Map Reduce Framework" International Journal of Recent Development in Engineering and Technology Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 2, Issue 6, June 2014).
8. Harshawardhan S. Bhosale et al. " A Review Paper on Big Data and Hadoop" International Journal of Scientific and Research Publications, Volume 4, Issue 10, October 2014 1 ISSN 2250-3153.
9. Nada Elgendy and Ahmed Elragal "Big Data Analytics: A Literature Review Paper" Department of Business Informatics & Operations, German University in Cairo (GUC), Cairo, Egypt 2014.
10. Shilpa et al. "BIG Data and Methodology-A review" International Journal of Advanced Research in Computer Science and Software Engineering. Volume 3, Issue 10, October 2013.
11. Jens Dietrich "Efficient Big Data Processing in Hadoop MapReduce" Information Systems Group Saarland University, 2011.
12. Chen He, Derek Weitzel, David Swanson, Ying Lu "HOG: Distributed Hadoop MapReduce on the Grid" 2010.

DOI: 10.15680/IJIRCCE.2017. 0504033