Seminar Report on

Big Data and Hadoop

Submitted by

Manoj Valeti
Roll No: B160091CS

Department of Computer Science and Engineering
National Institute of Technology Calicut
Calicut, Kerala, India - 673 601

September 12, 2019

Contents

List of Figures

List of Tables

Abstract

The term Big Data describes innovative techniques and technologies to capture, store, distribute, manage and analyze petabyte- or larger-sized datasets with high-velocity and different structures. It also refers to the data that can't be processed using traditional dataprocessing application software.

Hadoop is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance

This report provides a detailed analysis on BigData using Hadoop as a software example.

Chapter 1

Big Data

1.1 Introduction

In the era of information explosion enormous amounts of data have become available on hand to decision maker.Big Data[3] usually refers to data sets with sizes beyond the ability of commonly used software tools to capture, curate, manage, and process data within a tolerable elapsed time. In simpler words if we can't store the data and process it using a single computer then we can call that data as Big Data. There's a general misconception about Big Data that it only refers to large volumes of data, but actually Big Data accounts to 5 characteristics they are
i)Volume
ii)Velocity
iii)Variety
iv)Value
v)Veracity.

1.2 Characteristics of Big Data

The 5 characteristics that account for Big Data are usually referred to as 5V's of Big Data.

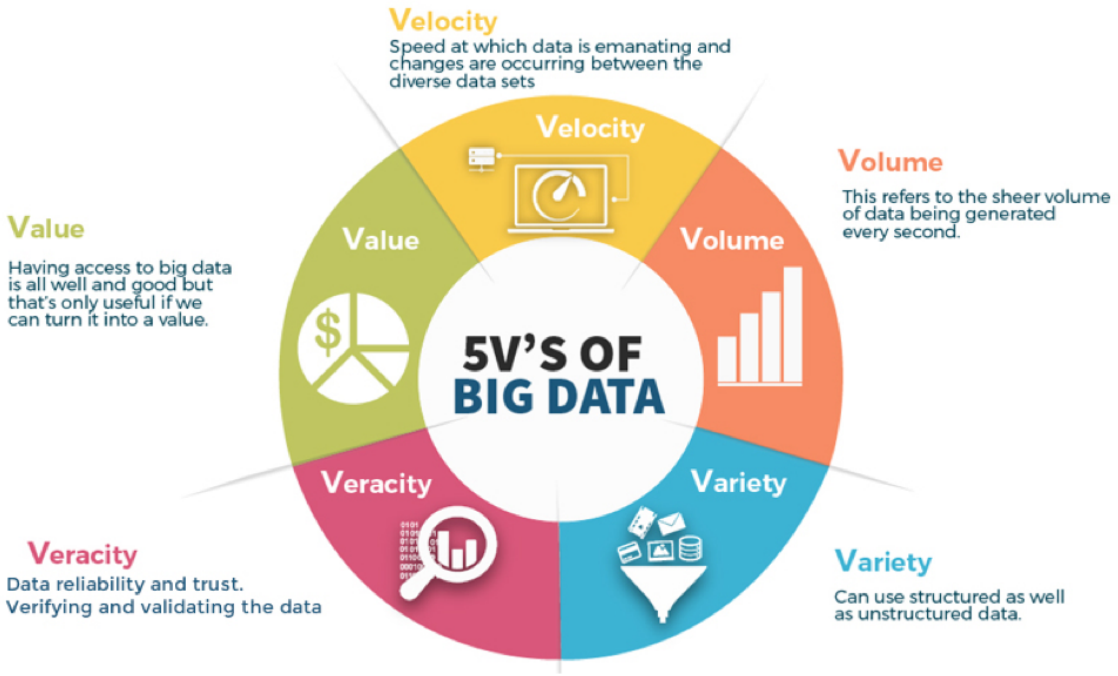

Figure 1.1: The 5V's of Big Data

1.2.1 Volume

Volume refers to the incredible amounts of data that are being generated every second from various sources like social media(Facebook, YouTube, Instagram, etc), cell phones, video calls, sensors, etc. All this generated data is in scales of petabytes which is almost impossible to process them using traditional techniques. In order to process this huge data Big Data provides a distributed architecture in which the data is split and stored at various locations which helps in easy processing of the data.

1.2.2 Velocity

Velocity refers to the speed at which vast amount of data is being generated everyday. Everyday number of mails, tweets, posts in social media is increasing rapidly. All this generated data should not only be analyzed but also the transmission rate and availability of data should be high to provide real-time access to that particular website. Big Data technology avoids usage of database for faster analysis of data.

1.2.3 Variety

Variety of data refers to types of data being generated. Data can be structured, semi-structured or unstructured. Structured data are easy to store and process. However, In today's world almost 80% of the generated data is unstructured and it is not easily stored in database using traditional methods. Big Data technology provides methods to store both structured and unstructured data simultaneously.

1.2.4 Value

Value refers to the worthiness of data being abstracted. It is of no use to store data which we are not going to use it anywhere. There should always be an effective cost benefit analysis which helps to know whether or not to abstract data.

1.2.5 Veracity

Veracity is the quality or trustworthiness of the data. A lot of data from social media that includes hashtags, images, emojis, abbreviations, typos, etc are less reliable so there is no use to store loads of these data. Big Data equips with the technology that can deal with this type of data.

1.3 Big Data Use Cases

- **Security Intelligence** - Helps many organisations to identify the internal and external information to help them prevent, detect and mitigate the attacks.
- **Recommendation engines** - Helps to provide recommendations based on the historical data.
- **Banking** - Helps to understand customers, manage sources, minimise risk and fraud.
- **Price Optimization** - Helps companies to know which price points have yielded the best overall results under the various historic market conditions.
- **Internet of Things** - The data collected by the sensors can be analyzed to achieve actionable insights.
- **Government** - Helps in managing utilities, running agencies, traffic congestion, crime prevention.

1.4 Problems with Big Data Processing

- **Heterogeneity and Incompleteness** - Most of the machine analysis algorithms expect homogeneous data. In contrast the data we get is mostly heterogeneous and incomplete so the first step in our data analysis should be careful structuring of our data and store all the identical pieces of data together to make the systems work efficiently.

- **Scale** - In olden days the challenge of increasing data was mitigated by increasing the number of computing resources but today data volume is scaling faster than computing resources and CPU speeds are static.
- **Privacy** - Managing privacy is both a technical and a sociological problem, which must be addressed jointly from both perspectives to realize the promise of big data.
- **Human Collaboration** - A Big Data analysis system must accept input from many users who are separated in space and time to support their collaboration.

Big Data has taken the world by storm. It is said that the next decade will be going to be dominated by Big-data wherein all the companies will be using the data available to them to learn about their companys ecosystem and improving fall-backs. All major universities and companies have started investing in building tools that would help them understand and create useful insights from the data that they have access to. One such tool that helps in analyzing and processing Big-data is **Hadoop**.

Chapter 2

Hadoop

2.1 What is Hadoop?

Hadoop is an open source distributed processing framework that manages data processing and storage for big data applications running in clustered systems. Hadoop was developed by Google's MapReduce that is a software framework in which application is broken down into various parts. Hadoop can handle various forms of structured and unstructured data giving users more flexibility for collecting, processing and analyzing the data. It is designed to scale up from a single server to thousand machines each offering local computation and storage.

2.2 Hadoop Architecture

Hadoop follows a Master Slave architecture for the transformation and analysis of large data sets using Hadoop MapReduce paradigm. The 4 most important components that play an important role in Hadoop architecture are

(i) Hadoop Common

(ii) Hadoop Distributed File System (HDFS)

(iii) Hadoop MapReduce

(iv) Yet Another Resource Negotiator (YARN)

2.2.1 Hadoop Common

Hadoop common package is considered as the base/core of the framework as it provides essential services and basic processes such as abstraction of the underlying operating system and it's file system. It also contains the necessary Java Archive (JAR) files and scripts required to start Hadoop. Hadoop Common package also provides source code and documentation, as well as contribution section that includes different projects from the Hadoop Community.

2.2.2 Hadoop Distributed File System (HDFS)

Hadoop comes with a distributed file system called HDFS that runs on commodity hardware. HDFS is a fault-tolerant storage system which is able to store huge amounts of information, scale up incrementally and survive the failure of significant parts of the storage infrastructure without losing data. The incoming data is broken down into pieces called blocks and each block is stored redundantly. HDFS stores the application data and file system meta data separately on dedicated servers. DataNode and NameNode are two crucial components of HDFS architecture. Application data is stored on DataNodes and file system meta data is stored on NameNodes. HDFS follows Master/Salve architecture in which NameNode is considered as Master and DataNode is considered as Slave.

- **What is HDFS NameNode?**
  NameNode stores meta-data i.e the number of data blocks, replicas and other details. It is present in memory in the master for faster retrieval of data. It manages file system namespace, regulates clients access to files and also executes file system operations such as naming, opening, closing files and directories. It is also responsible for taking care of replication factor of all the blocks

- **What is HDFS DataNode?**
  DataNode stores actual data in HDFS. It performs operations like read, write, block creation, deletion and replication. It manages the data storage of the system.

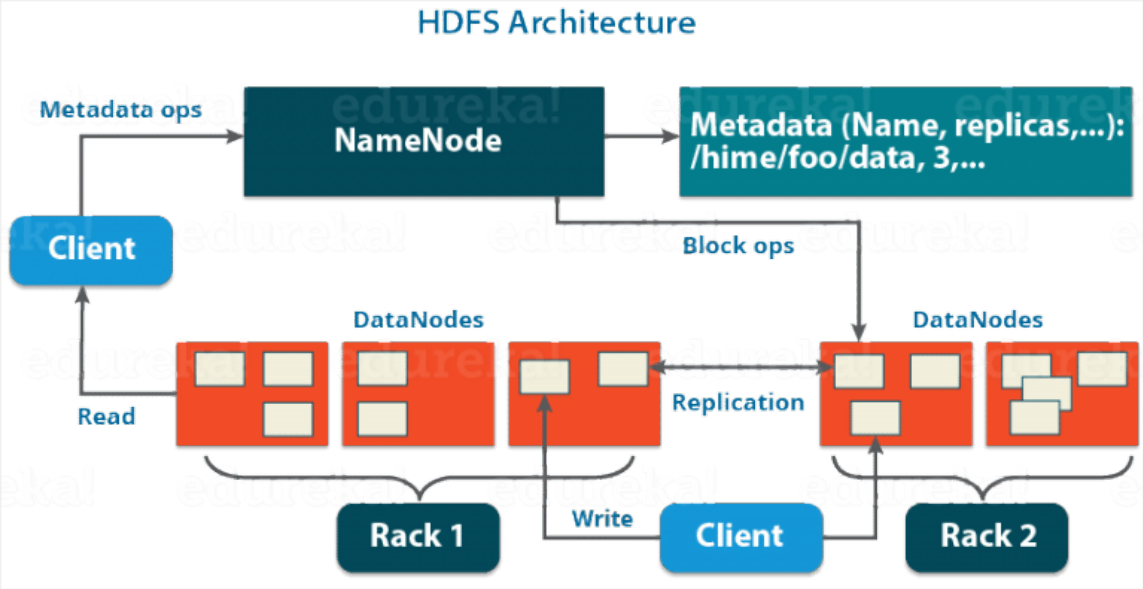

Figure 2.1: HDFS Architecture

2.2.3 Hadoop MapReduce

The primary objective of MapReduce is to split the data into independent chunks and process them parallelly. It is based on Java. There are two important tasks in MapReduce algorithms they are Map and Reduce. The function of Map is to convert a set of data into another in which individual elements are broken to form key-value pairs. Reduce takes the output of Map to generate a smaller set of key-value pairs. As the name suggests Reduce function is always performed after Map is finished.

MapReduce consists of 3 stages: Mapping stage, Shuffling and Sorting stage and Reduce stage. In Mapping stage a key-value pairs are generated for a given input. This output generated by mapping function is also know as intermediate output which is stored in local disk. It is not stored on HDFS as it is temporary data. Shuffling is the physical movement of data which is done over the network. Once all the mappers are finished and their output is shuffled on the reduced nodes then the intermediate output is combined and sorted, which is provided as input for reduce phase. Reduce function is run on all the input that is provided to reduce phase and output generated is the final output which is stored in HDFS.

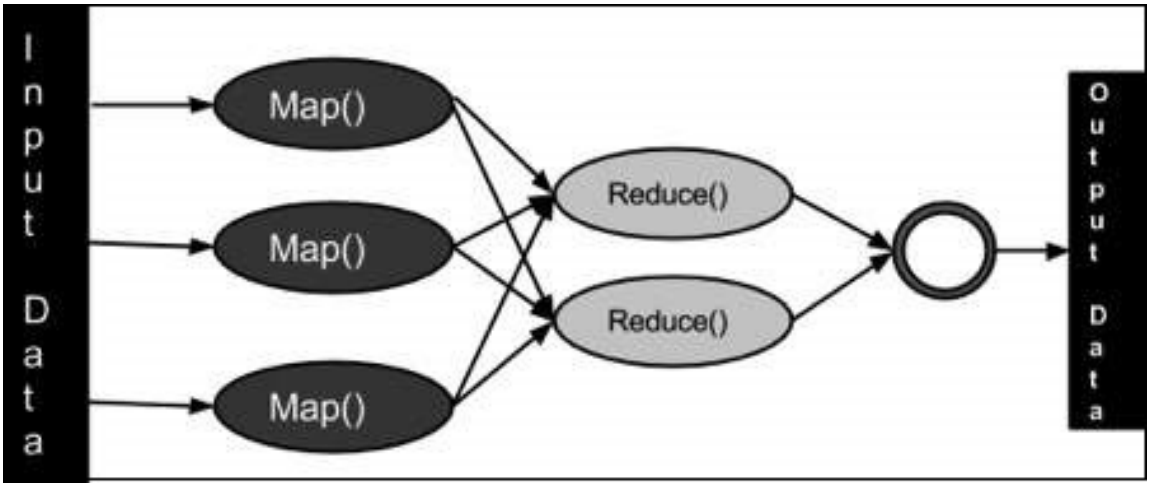

Figure 2.2: MapReduce algorithm

Table 2.1: YARN Vs MapReduce.

YARN	**MapReduce**
Supports variety of processing engines and applications	Only supports it's own batch processing ap
YARN is more isolated and scalable	Less scalable as compared to YARN.
By default the size of DataNode is 128MB	By default the size of DataNode is 64MB.
Dynamic allocation of resources	static allocation of resources
Separates its duties across multiple components	Combines most of it's work into a single co

2.2.4 Hadoop YARN

YARN (Yet Another Resource Negotiator) is the resource management layer of Hadoop. The basic principle of YARN is to separate resource management and job scheduling/monitoring function into separate daemons. YARN framework consists of master daemon called Resource-Manager, slave daemon called Node-Manager(one per application) and Application Master(one per application). The Resource-Manager arbitrates resources among all the competing applications in the system. The job of Node-Manager is to monitor the resource usage by the container and report the same to Resource-Manager. The Application-Master negotiates resources with Resource-Manager and works with Node-Manager to execute and monitor the job.

2.3 Who uses Hadoop?

- Financial companies use analytics to access risk, build investment models, and create trading algorithms. Hadoop has been used to help build and run those applications.
- Retailers uses it to help analyze structured and unstructured data to better understand and serve their customers.
- Provides scalable storage and processing power for machine-learning enthusiasts, for building better and more accurate models, as opposed to the expensive old datasets.
- Reduces the time for database migration and scheme redesign.

2.4 Pros and Cons of Hadoop

Pros of Hadoop are:

- Open source availability when compared to expensive software licenses.
- Automatic data replacement and re-balancing when data grows.
- It saves a lot of time in extraction of valuable data from the given different forms of data.
- It supports access to file-based external data.
- Support for automatic and incremental forward recovery of jobs with failed tasks
- It supports replication and machine failover.

Cons of Hadoop are:

- Hadoop is not suitable for large number of small files as the size of files will be less than the default block size. This overloads the Namenode and makes it difficult for Hadoop to function.

- Hadoop is easily vulnerable.
- Hadoop has a batch processing engine which is not efficient in stream processing. It cannot produce output in real-time with low latency.
- Hadoop has a processing overhead when dealing with large data read/write operations become expensive.

Chapter 3

Conclusion

The world is changing the way it is currently operating and Big Data is playing an important role in it. Hadoop is a framework that makes an engineers life easy while working on large sets of data. There are improvements possible on all the fronts.

References

[1] Harshawardhan S. Bhosale , Prof. Devendra P. Gadekar: A REVIEW PAPER ON BIG DATA AND HADOOP.

[2] Deepak Motwani, V.K. Chaubey, A. S. Saxena:Hadoop based Information Extract from Text Document

[3] Big Data, `https://en.wikipedia.org/wiki/Big_data#Definition`.

[4] Hadoop Tutorial, `https://www.javatpoint.com/hadoop-tutorial`