ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

Youtube: Bigdata Analytics using Hadoop and Map Reduce

[1] P.Sushma, [2] Dr.S.Nagaprasad, [3] Dr.V.Ajantha Devi
[1] Research Scholar-JJT University, Rajasthan
[2] Faculty of Computer Science, Dept. Of Computer Science, S.R.R.Govt.Arts & Science College, Karimnagar, Telangana State
[3] Department of Computer Application, Guru Nanak College, Chennai, Tamil Nadu

Abstract: **- We live in a digitalized world today. An enormous amount of data is generated from every digital service we use. This enormous amount of generated data is called Big Data. According to Wikipedia, Big data is a word for data sets that are enormous in size or compound that traditional data supervision application software is pathetic to compact with them [5].Big data defies embrace receiving data, data storage, data analysis, search, sharing, transfer, visualization, querying, updating and information privacy. Google's video streaming services, YouTube, is one of the best examples of services which produces a huge quantity of data in a very short period. Data mining of such an enormous quantity of data is performed using Hadoop and MapReduce to measure performance. Hadoop is a system which delivers a consistent collective storage. The storage is provided by HDFS (Hadoop Distributed File System) and analysis by MapReduce. MapReduce is a programming model and an associated implementation for processing large data sets. This paper presents big data analysis on Youtube using Hadoop and MapReduce techniques.**

Keywords: **Big Data definition, Data mining, YouTube data analysis, Hadoop, HDFS, MapReduce, unstructured dataset analysis.**

I. INTRODUCTION

Analysis of structured dataset has demonstrated tremendous success. In a current whitepaper from Filene Exploration Organization, creator Philipp Kallerhoff states: Organizations, as fluctuated as Amazon, Google, Walmart, and Wells Fargo, are swinging too big data for part bits of knowledge that will enable them to serve clients and catch showcase share[6]. He included an essential for building up these (prescient) and different models is a very much kept up database with as much transactional detail as possible [6].. Financial companies and the finance departments of companies are already facing challenges in extracting the required information from the huge transactional data from the customers. However, the nature of such data is structural and easily manageable. Google's YouTube allows billions of people to connect, inform, and inspire others across the globe using originally created videos on a daily, every minute, basis. Thus, unsurprisingly, YouTube has a great impact on Internet traffic

Nowadays, yet itself is suffering from a severe problem of scalability. Storage, processing and efficient analysis of such enormous data over a short period of time is a very demanding task. The data generated from billions of YouTube videos is primarily unstructured. Quick, efficient and accurate analysis of this unstructured or semi-structured data remains a challenging task. According to statistics published by Google, YouTube has over a billion clients very nearly 33% surprisingly on the Internet, and every day those clients watch a billion hours of video, creating billions of perspectives [3]. YouTube has roughly 300h of video transferred each moment, and billions of the sentiments made each day [2].

YOUTUBE COMPANY STATISTICS	DATA
Total number of you Tube users	1,325,000,000
Hours of video uploaded every minute	300 Hours
Number of videos viewed everyday	4,950,000,000
Total number of hours of video watched every month	3.25 billion hours
Number of videos that have generated over 1 billion views	10,113
Average time spent on YouTube per mobile Session	40 Minutes

Figure1. YouTube statistics

The above image, Figure1 [3], provides us with important statistics and helps us infer that approximately 300K videos are uploaded to YouTube every day. YouTube collects a wide variety of traditional data points like the number of views, likes, votes, comments, and duration. The collection of the above-listed data points constitutes

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

a very interesting data set to analyze for obtaining implicit knowledge about users, videos, categories and community interests. Movie production houses release their movie promos and songs on YouTube. Company brands release their ads on YouTube for promotion. Budding artists present and promote their art on YouTube for publicity. These are just a few examples. The success rate of movies, songs, brand ads, and artists largely depends on the number of viewers, likes, and comments. Companies or artists can not only analyze their own performance but also analyze their competitors'. The Paper includes these modules:

Hadoop Common:
The common utilities that support the other Hadoop modules.

Hadoop Distributed File System (HDFS™):
A distributed file system that provides high-throughput access to application data.

HadoopYARN (Yet another Resource Negotiator):
A framework for job scheduling and cluster resource management.

Hadoop MapReduce:
A YARN-based system for parallel processing of large datasets. [8] .The main objective of this Paper is to help organizations or people in general, who use YouTube for marketing/promotion, understand how data mining and data analytics can prove them helpful by fetching meaningful results in terms of understanding their performance and changing trends among people. There are several Big Data analytics platforms available such as HIVE, HBASE, PIG to handle such volume of data. In this paper, we have chosen the MapReduce framework for analyzing our dataset. The Operating System chosen for this experiment is Ubuntu.
The procedure is very simple and broken down into 6 steps. The below Flow Diagram (Figure 2) helps illustrate the steps very effectively.

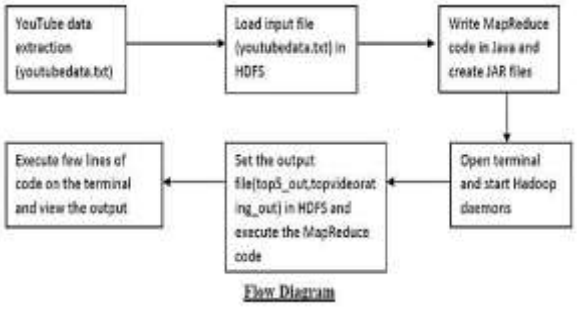

Figure 2-overview of methodology

A. APACHE HADOOP PLATFORM
Considering the magnitude of data produced by YouTube over a very short period of time, Hadoop is definitely the most preferred framework for data analysis.
The Apache Hadoop programming library is a system that takes into account the distributed processing of massive datasets crosswise over clusters of PCs utilizing basic programming models. It is intended to scale up from single servers to a large number of machines, each offering neighborhood calculation and capacity. As opposed to depending on equipment to convey high-accessibility, the library itself is intended to distinguish and handle disappointments at the application layer, so giving a straightforward administration over a cluster of PCs, every one of which might be inclined to failures[8].

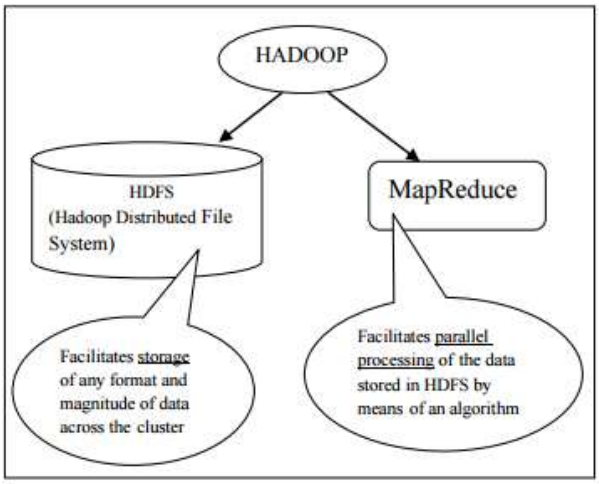

Figure 3-Apache Hadoop Ecosystem

From Figure 3, we understand that Hadoop stores any type of data across the cluster. A cluster is a group of interconnected systems which produce data, collectively called as nodes, which work together.

Hadoop Distributed File System (HDFS):

HDFS has two main classes:

1. Name Node: Contains metadata about the data stored

2. Data Node: Where actual data is stored

3. Secondary Name Node: Contains copy of Name Node DF – Data File.

This is best illustrated in Figure 4.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

Each data block in the Data Node is replicated by a factor of 3(default value) .i.e. there are 3 copies of each data block in the data node. This replication mechanism is provided to ensure that there is no loss of data in case any of the data nodes fail. The replication factor can be decided by the organization using Hadoop system as per their requirements for storing and processing their data.

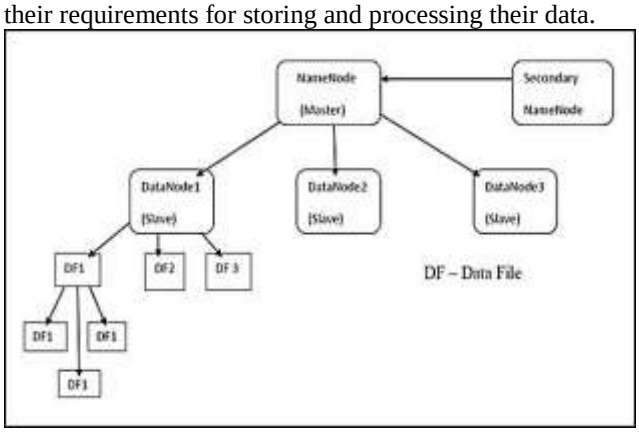

Figure 4 – Hadoop Cluster

B. DETERMINING WHICH SYSTEM IS BEST SUITED FOR THIS PAPER

While Hadoop provides the ability to store data on HDFS, there are many programming frameworks available that allow us to perform distributed and parallel processing and analyzing on large datasets in a distributed environment. The most popular ones are MapReduce, Pig, and Hive. Figure 5 [9] helped us analyze which application is better suited for this Paper.

Figure 5 – Comparison between MapReduce, Pig and Hive

Also, a well-developed MapReduce algorithm has a higher efficiency than Pig. Thus, MapReduce is the best choice for this Paper.

C. MAPREDUCE

The MapReduce programming is abridged in the accompanying statement [10]: The calculation takes an arrangement of information key/esteem combines and delivers an arrangement of yield key/value pairs. The client of the MapReduce library communicates the computation in two capacities: delineate diminish. Guide, composed by the client, takes an information combine and delivers an arrangement of the middle of the road key/value pairs. The MapReduce library bunches together all middle of the road esteems related with a similar medium key I and passes them to the lessen work. The lessen work, additionally composed by the client, acknowledges a moderate key I and an arrangement of qualities for that key. It blends these qualities to frame a potentially littler mechanism of conditions. Regularly only zero or one yield esteem is delivered per diminish conjuring. The middle esteems provided to the client's lessen work using an iterator. This enables us to deal with arrangements of qualities that are too vast to fit in memory.

Phases in MapReduce:

The main concept behind MapReduce job is splitting a large data set into independent smaller data sets, mapping those smaller data sets to form a collection of <key, value> pairs and reducing overall pairs having the same key for parallel processing. A key-value pair (KVP) is a set of two inter-connected data items: a key is a unique identifier for a particular data item in the dataset, and the value is either the count of the data that is identified or the position value of that data. Because this parallel processing mechanism follows the Divide and Process rule, it significantly improves the speed and reliability of the cluster, returning solutions more quickly and with greater reliability.

Every MapReduce job consists of the following two main parts:

I. The Mapper

II. The Reducer

I. Mapper Phase

The first phase of a MapReduce program is called mapping. A mapping algorithm is designed. The main objective of the mapping algorithm is to accept the large input dataset and divide it into smaller parts (sub-dataset). These sub data sets are distributed to different nodes by the Job Tracker. The nodes perform parallel processing (map task) on these sub-datasets and convert them into pairs as output. The value of 'Value' in each KVP is always set to 1. Each KVP output is then fed as input to the reducer phase.

II. Reducer Phase

The reducing phase aggregates values of KVP together. A reducer function receives the KVP input and iterates over each KVP. It then combines the KVP containing the same Key and increments the 'Value' by 1. It then combines these values together, returning a single output value which is the aggregate of same keys in the input dataset.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

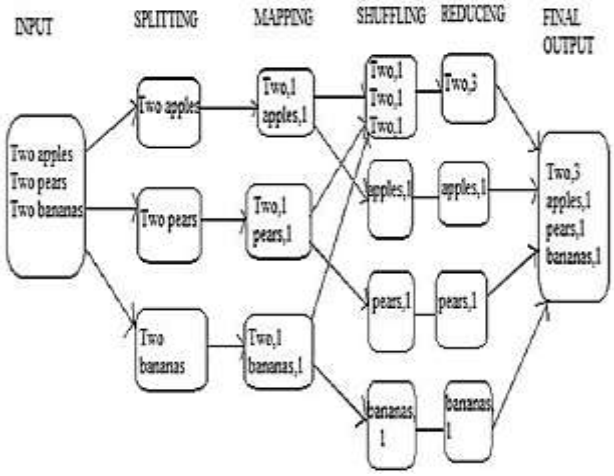

Figure 6 – Word Count Example illustrating MapReduce concept

The following diagram, Figure 7, gives an overview summary of the MapReduce concept.

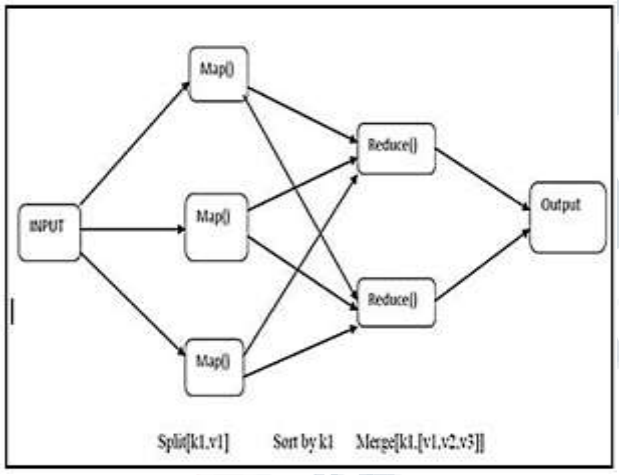

Figure 7 – MapReduce Overview concept

This unstructured dataset consists of data from approximately 3000 videos and contains 10 columns in total.

II. METHODOLOGY MAPREDUCE

Problem Statement 1: To determine top 5 video categories on YouTube

Mapper Algorithm:

We take a class by name Top5_categories. We then extend the Mapper class which has arguments and. We then declare an object ‘category‘which stores all the categories of YouTube. As explained before, in the pairs in MapReduce, the value of ‘v‘is always set to 1 for every key-value pair. In the next step, we declare a static variable ‘one‘and set it to the constant integer value 1 so that every ‘value‘in every pair automatically gets assigned to value 1. We override the Map method which will run for all pairs. We then declare a variable ‘line’ which will store all the lines in the input youtubedata.txt dataset. We then split the lines and store them in an array so that all the columns in a row are stored in this array. We do this to make the unstructured dataset structured. We then store the 4th column which contains the video category. Finally, we write the key and value, where the key is ‘category‘and value is ‘one‘. This will be the output of the map method.

Reducer Algorithm:

We first extend the Reducer class which has the same arguments as the Mapper class .i.e. and .Again, same as the Mapper code; we override the Reduce method which will run for all pairs. We then declare a variable sum which will sum all the values of the ‘v‘in the pairs containing the same ‘k‘(key) value. Finally, it writes the final pairs as the output where the value of ‘k‘is unique and ‘v‘is the value of sum obtained in the previous step. The two configuration classes (Map Output Key Class and Map Output Value Class) are included in the main class to clarify the Output key type and the output value type of the pairs of the Mapper which will be the inputs of the Reducer code.

Problem Statement 2: To find the top 5 video uploaders on YouTube:

The mapper and reducer algorithm for this problem statement is very similar to that of Problem statement1. Mapper Algorithm: In this mapper code, the pairs are associated as: key=uploader, and value=views where uploader is the username of the uploader and views is the number of views for the video. These pairs will be passed to the shuffle and sort phase and is then sent to the reducer phase where the total count (sum) of the values is performed. We take a class by name Top Uploader We then extend the Mapper class which has the same arguments as the Mapper class in Problem Statement 1.i.e. and . We then declare an object ‘uploader’ which will store the username of the uploader. Next we declare a variable ‘views’ which will store the video views. Then we override the map method so that it runs once for every line. Next we declare a variable ‘record’ which stores the lines. We then split the line and store them in an array. All the columns in a row are stored in this array. We then store the uploaders’ username. Finally, we write the key and value, where key is ‘uploader’ and value is ‘views’. This will be the output of the map method.

Reducer Algorithm: We first extend the Reducer class which has the same arguments as the Mapper class .i.e. and .Again, same as the Mapper code; we override the Reduce method which will run for all pairs. We then declare a variable 'total views' which will check all the values of the 'v' in the pairs containing the same 'k' (key) value. Finally, it writes the final pairs as the output where the value of 'k' is unique and 'v' is the highest value obtained in the previous step. The two configuration classes(Map Output Key Class and Map Output Value Class) are included in the main class to clarify the Output key type and the output value type of the pairs of the Mapper which will be the inputs of the Reducer code.

III. CONCLUSION

This Paper intends to hit on those key areas which companies and organizations use or can use to measure their product's/movie's success against their competitors. As seen from the methodology, the basic algorithm retrieves reports to better understanding and viewing statistics and trends for users' channel depending on the number of views and likes not only on their respective videos but also check if their competitors' are at the top. Another output result gives us insights on what categories of videos interest the public more. This can be done by analyzing the top video categories. This also helps budding You Tubers who upload YouTube videos to earn money. They can analyze the most popular video categories and upload videos accordingly to gain more views, more subscription and thus more money and popularity. As we have already seen above in depth, MapReduce is a very simple programming tool which makes use of basic programming languages like C, Python, and Java.

IV. FUTURE SCOPE

This Paper can be further advanced by designing a MapReduce algorithm to perform sentiment analysis on YouTube video comments. Also, an algorithm on comment analysis can help analyze and identify the numbers of trolls harassing authentic users and spam users.

REFERENCES

[1] Webster, John. "MapReduce: Simplified Data Processing on Large Clusters", "Search Storage",2004. Retrieved on 25 March 2013. https://static.googleusercontent.com/media/research.google.com/en//archive/mapreduce-osdi04.pdf.

[2]. Bibliography: Big Data Analytics: Methods and Applications by SaumyadiptaPyne, B.L.S. PrakasaRao, S.B. Rao.

[3]. YOUTUBE COMPANY STATISTICS. https://www.statisticbrain.com/youtube-statistics/.

[4]. Youtube.com @2017. YouTube for media. https://www.youtube.com/yt/about/press/

[5] Big data;Wikipedia https://en.wikipedia.org/wiki/Big_data

[6] Kallerhoff,Phillip. ―Big Data and Credit Unions: Machine Learning in Member Transactions https://filene.org/assets/pdfreports/301_Kallerhoff_Machine_Learning.pdf.

[7] Marr,Barnard.―Why only one of the 5 Vs of big data really matters http://www.ibmbigdatahub.com/blog/why-only-one-5-vs-big-data-really-matters.

[8] 2016. Information. "Chapter 1 - Big Data Overview". Big Data: Concepts, Methodologies, Tools, and Applications, Volume I. IGI Global. http://common.books24x7.com/toc.aspx?bookid=114 046

[9]. Apache Hadoophttp://hadoop.apache.org/

[10] How To Analyze Big Data With HadoopTechnologies ; 3pillarglobal.com. 2017. https://www.3pillarglobal.com/insights/analyze-big-data-hadoop-technologies

[11] J. Dean, S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, in:OSDI'04, 6th Symposium on Operating Systems Design and Implementation, Sponsored by USENIX, in cooperation with ACM SIGOPS, 2004, pp. 137– 150.