

YOUTUBE DATA ANALYSIS USING HADOOP

A report submitted in fulfillment of the requirements for
the award of the degree of

Bachelor of Technology

in

Department of Computer Science and Engineering

by

CH. SUJAN (CS14B1009)

K.P.N.S.ANIRUDH (CS14B1016)



**DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY PUDUCHERRY
KARAIKAL – 609609
MAY-2018**

BONAFIDE CERTIFICATE

This is to certify that the project work entitled “**Youtube data analysis using Hadoop**” is a bonafide record of the work done by

CH.SUJAN (CS14B1009)
K.P.N.S.ANIRUDH (CS14B1016)

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** of the **NATIONAL INSTITUTE OF TECHNOLOGY PUDUCHERRY** during the year 2017 - 2018.

Dr. B.Surendiran
Assistant Professor
Project Guide

Dr. B.Surendiran
Head of the Department

Project viva-voce held on _____

Internal Examiner

External Examiner

ABSTRACT

There is a large amount of data being collected in the world today. Although analysis of structured data involving textual and semi-textual elements has seen tremendous success in the past few years, analysis of large scale unstructured data in the form of video format still remains a challenging area of research. With the amount of video data being collected today, via surveillance cameras, digital devices and other recording devices present everywhere, this large scale video data requires in depth analysis, and may provide vital insights. However, analysis of large scale unstructured video data is faced with a number of issues. YouTube, a Google company, has over a billion users and generates billions of views. Since YouTube data is getting created in a very huge amount and with an equally great speed, there is a huge demand to store, process and carefully study this large amount of data to make it usable . This project attempts to discuss about how youtube data will be processed and analysed using Apache's Hadoop platform.

ACKNOWLEDGEMENT

We would like to show our kind regards towards our respected Director , **Dr.K.Sankaranarayanasamy** for permitting us to undertake this project work.

We would like to thank our project guide and Head of the Department, Assistant Professor **Dr.B.Surendiran**, for his constant motivation and guidance during the project. We want to genuinely convey our thanks to all the faculties of our Computer Science and Engineering department for their motivation in various reviews throughout the course of the project phase-I. We would like to thank our project Coordinator **Dr.Narendran Rajagopalan** for his consistent encouragement.

We would like to thank the project review members for their valuable suggestion throughout the period of project.

We are at the dearth of words to express gratitude to our wonderful parents for their unconditional support both financially and emotionally. I thank our parents for inculcating the dedication and discipline to do whatever we undertake well.

We have been fortunate to have friends who cherish us despite our eccentricities. By their remarks comments or compliments and unavoidable questions, we were able to make our project reviews better each time. Thank you all for making it possible for us to reach the final stage of our endeavor.

We would also like to thank all our sources, mentioned in the references, and our friends who helped us by providing mental and logistical support. Last but not the least we would like to thank our parents and God Almighty.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	3
	ACKNOWLEDGEMENT	4
	TABLE OF CONTENTS	5
	LIST OF FIGURES	8
1	INTRODUCTION	
	1.1 Bigdata	9
	1.2 Hadoop	10
	1.3 Objective	10
	1.4 Motivation	11
	1.5 Issues	11
	1.6 Organization of Thesis	12
2	LITERATURE REVIEW	13

3	SOFTWARE REQUIREMENTS	
	3.1 VMware Workstation	15
	3.2 Cloudera Os	16
	3.3 Eclipse IDE	17
	3.4 Visual Studio	17
4	DATA SET	
	4.1 Data set 1	18
	4.2 Data set 2	18
5	SYSTEM MODELING	
	5.1 System Modeling	20
	5.2 Advantages	20
	5.3 Algorithm	21
	5.4 Stages	22
	5.5 Example	22
6	IMPLEMENTATION AND RESULTS-1	
	6.1 Implementation	23
	6.2 Problem Statement 1	23
	6.2.1 Mapper Code	23
	6.2.2 Reducer Code	24
	6.2.3 Output	25
	6.3 Problem Statement 2	25
	6.3.1 Configuration code	25
	6.3.2 Output	29
	6.4 Procedure	30
	6.5 Analysis	30
7	IMPLEMENTATION AND RESULTS-2	
	7.1 Implementation	32

	7.2 Generate API Key To Fetch YouTube Data	32
	7.3 Creating a .Net(C#) Console Application to Use API	36
	7.4 Result	38
8	CONCLUSION	39
	REFERENCES	42

LIST OF FIGURES

Figure .No	Title	Page No
3.1.1	VMware	16
3.2.1	Cloudera OS	17
4.1	Sample Dataset	18
5.3.1	MapReduce	21
6.2.3.1	Output 1	25
6.5.1	Analysis 1	30
6.5.2	Analysis 2	31
7.2.1	Project creation	31
7.2.2	Naming Project	33
7.2.3	Enabling API	33
7.2.4	Adding Credentials	34
7.2.5	Providing Name For Client ID	35
7.2.6	J-son File(key)	36
7.3.1	Adding DLL's	37
7.3.2	Required Packages	37
7.3.3	Adding API Key	38
7.3.4	Including Channel Id	38
7.3.5	Creating Header File	38
7.4.1	Result	40

CHAPTER 1

INTRODUCTION

1.1 Bigdata :

Big Data is "a collection of data sets so large and complex that it becomes difficult to process using the available database management tools. The challenges include how to capture, curate, store, search, share, analyze and visualize Big Data". In today's environment, we have access to more types of data. These data sources include online transactions, social networking activities, mobile device services, internet gaming etc. Big Data is a collection of data sets that are large and complex in nature. They constitute both structured and unstructured data that grow large so fast that they are not manageable by traditional relational database systems or conventional statistical tools. Big Data is defined as any kind of data source that has at least three shared characteristics:

- Extremely large Volumes of data
- Extremely high Velocity of data
- Extremely wide Variety of data

Data sources are ever expanding. Data from Facebook, Twitter, YouTube, Google etc., are to grow 50X in the next 10-13 years. Over 2.5 exabytes of data is generated every day. Some of the sources of huge volume of data are:

1. A typical large stock exchange captures more than 1 TB of data every day.
2. There are over 5 billion mobile phones in the world which are producing enormous amount of data on daily basis.
3. YouTube users upload more than 48 hours of video every minute.
4. Large social networks such as Twitter and Facebook capture more than 10 TB of data daily.
5. There are more than 30 million networked sensors in the world which further produces TBs of data every day.

Structured and semi-structured formats have some limitations with respect to handling large quantities of data. Hence, in order to manage the data in the Big Data world, new emerging approaches are required, including document, graph, columnar, and geospatial database architectures. Collectively, these are referred to as NoSQL, or not only SQL, databases. In essence the data architectures need to be mapped to the types of transactions.

1.2 Hadoop :

As organizations are getting flooded with massive amount of raw data, the challenge here is that traditional tools are poorly equipped to deal with the scale and complexity of such kind of data. That's where Hadoop comes in. Hadoop is well suited to 14 meet many Big Data challenges, especially with high volumes of data and data with a variety of structures. At its core, Hadoop is a framework for storing data on large clusters of commodity hardware — everyday computer hardware that is affordable and easily available — and running applications against that data. A cluster is a group of interconnected computers (known as nodes) that can work together on the same problem. Using networks of affordable compute resources to acquire business insight is the key value proposition of Hadoop. Hadoop consists of two main components

1. A distributed processing framework named MapReduce (which is now supported by a component called YARN(Yet Another Resource Negotiator)).
2. A distributed file system known as the Hadoop Distributed File System, or HDFS.

In Hadoop you can do any kind any kind of aggregation of data whether it is one month old data or one-year-old data. Hadoop provides a mechanism called MapReduce model to do distributed processing of large data which internally takes care of data even if one machine goes down.

1.3 Objective:

The main objective of this project is to demonstrate by using Hadoop concepts, how data generated from YouTube can be mined and utilized to make targeted and informed decisions.

1.4 Motivation:

Analysis of structured data has seen tremendous success in the past. However, analysis of large scale unstructured data in the form of video format remains a challenging area. YouTube, a Google company, has over a billion users and generates billions of views. Since YouTube data is getting created in a very huge amount and with an equally great speed, there is a huge demand to store, process and carefully study this large amount of data to make it usable .

1.5 Issues:

RDBMS finds it challenging to handle such huge data volumes. To address this, RDBMS added more central processing units (or CPUs) or more memory to the database management system to scale up vertically.

The majority of the data comes in a semi-structured or unstructured format from social media, audio, video, texts, and emails. However, the second problem related to unstructured data is outside the purview of RDBMS because relational databases just can't categorize unstructured data. They're designed and structured to accommodate structured data such as weblog sensor and financial data.

Also, "big data" is generated at a very high velocity. RDBMS lacks in high velocity because it's designed for steady data retention rather than rapid growth. Even if RDBMS is used to handle and store "big data," it will turn out to be very expensive. As a result, the inability of relational databases to handle "big data" led to the emergence of new technologies.

To overcome these problems , we use map reduce algorithm which is a eco system of Hadoop software.

1.6 Organization of Thesis:

Introduction about Big Data and Hadoop, Our Objective ,Motivation and organization of the thesis are presented in this **Chapter 1**.

Chapter 2 of this document summarizes a literature review .

Chapter 3 presents system requirements specifications.

Chapter 4 Significance of datasets used in project.

Chapter 5 will give idea about Algorithm used in the project.

Chapter 6 presents implementation and results.

Chapter 7 summarizes the project work and concludes along with the future direction of work.

CHAPTER-2

LITERATURE REVIEW

This briefly discuss the application of Hadoop Big Data in different fields of technology.

A) Mukherjee, A.; Datta, J.; Jorapur, R.; Singhvi, R.; Haloi, S.; Akram, W. (18-22 Dec. 2012) “Shared Disk Big Data Analytics With Apache Hadoop”

This paper discusses the necessity of Big Data and Big Data techniques which is required to process huge amount of data and to discover insights. Hadoop is a open source platform used for implementing Mapreducer Model. The performance of VERITAS Storage Foundation Cluster File System (SF CFS) is compared with Hadoop distributed file system (HDFS) for shared data Big Data analytics. Analytics with clustered file system is best suited for this proposed model.

B) “Towards MapReduce Performance Optimization: A Look Into The Optimization Techniques In Apache Hadoop For Big Data Analytics” Kudakwashe Zvarevashe1, Dr. A Vinaya Babu

Traditional database management system can’t handle huge distributed, structured and unstructured data. Big Data plays a role in solving the issues of handling huge, complicated and dynamic data. Hadoop and NoSQL databases supported to eradicate these problems. Various technologies associated to MapReduce discussed in this paper. Difference research problems related to the improvement of the MapReduce problem discussed.

C) Jimmy Lin and Chris Dyer: “Data-Intensive Text Processing with MapReduce “

This paper discusses the need for computing the data in the clouds, the algorithms and the softwares used, the use of Hadoop frame work for handling big data. It also discuss the basis of map reduce algorithm its working principle and implementation details with the additional programming languages used for handling bigdata like PIG,HIVE etc, and discuss the limitations of map reduce.

D) <https://hortonworks.com/apache/hdfs/> : An Overview of HDFS File System of Hadoop.

This blog discuss the overview of file system implemented in Hadoop for processing big data, it's architecture and working principle. It also gives a brief idea about how it is implemented in the real time using a random applications.

E) <https://csusdspace.calstate.edu/bitstream/handle/10211.3/182685/Youtube> : Accessing of data of YouTube from Google API's.

This blog discuss how to get client id's for accessing API's using which we can extract datasets directly from youtube.

CHAPTER 3

SOFTWARE REQUIREMENTS

3.1 VMware (virtual machine) :

VMware Workstation is a hosted hypervisor that runs on x64 versions of Windows and Linux operating systems. It enables users to set up virtual machines (VMs) on a single physical machine, and use them simultaneously along with the actual machine. Each virtual machine can execute its own operating system, including versions of Microsoft Windows, Linux, BSD, and MS-DOS. VMware Workstation is developed and sold by VMware, Inc., a division of Dell Technologies. There is a free-of-charge version, VMware Workstation Player, for non-commercial use. An operating systems license is needed to use proprietary ones such as Windows. Ready-made Linux VMs set up for different purposes are available from several sources.

VMware Workstation supports bridging existing host network adapters and sharing physical disk drives and USB devices with a virtual machine. It can simulate disk drives; an ISO image file can be mounted as a virtual optical disc drive, and virtual hard disk drives are implemented as .vmdk files.

VMware Workstation Pro can save the state of a virtual machine (a "snapshot") at any instant. These snapshots can later be restored, effectively returning the virtual machine to the saved state,^[5] as it was and free from any post-snapshot damage to the VM.

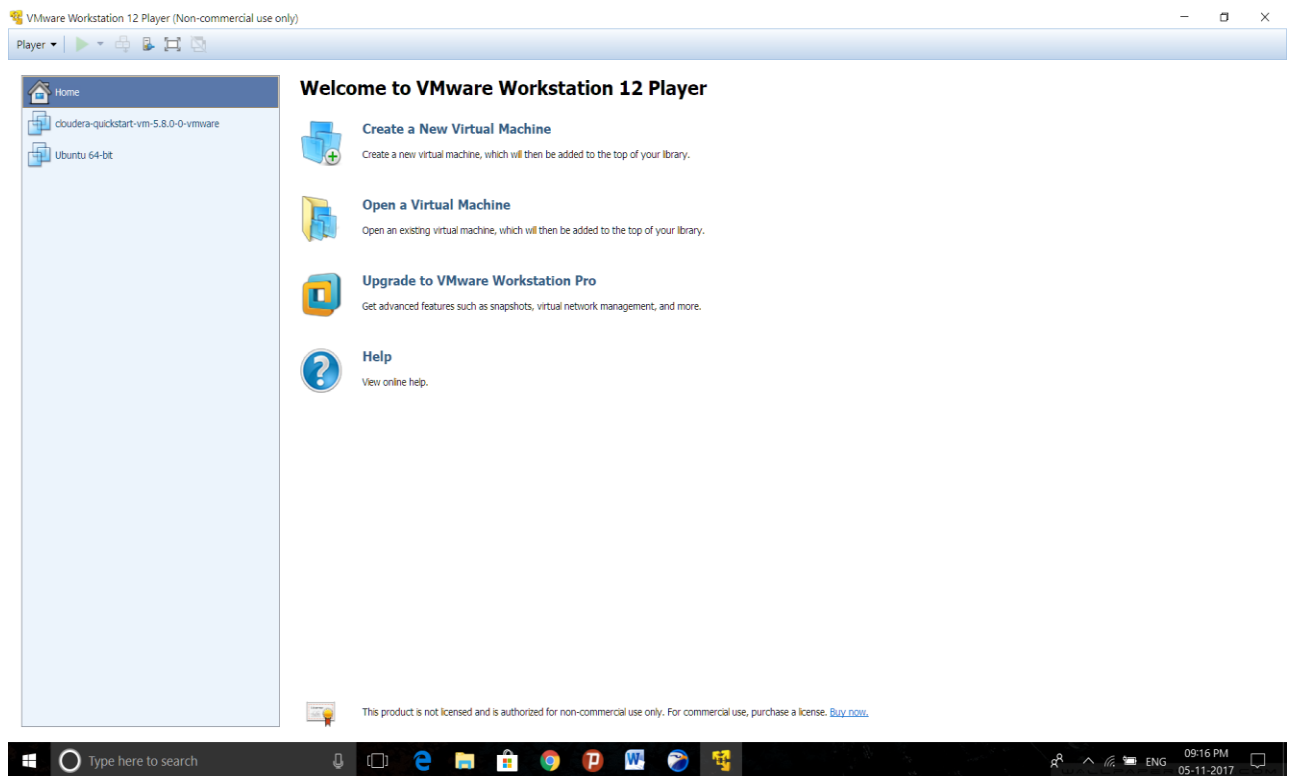


Figure 3.1.1-VMware

3.2 Cloudera os :

Cloudera Express includes CDH and a version of Cloudera Manager lacking enterprise features. CDH contains the main, core elements of Hadoop that provide reliable, scalable distributed data processing of large data sets (chiefly MapReduce and HDFS), as well as other enterprise-oriented components that provide security, high availability, and integration with hardware and other software.

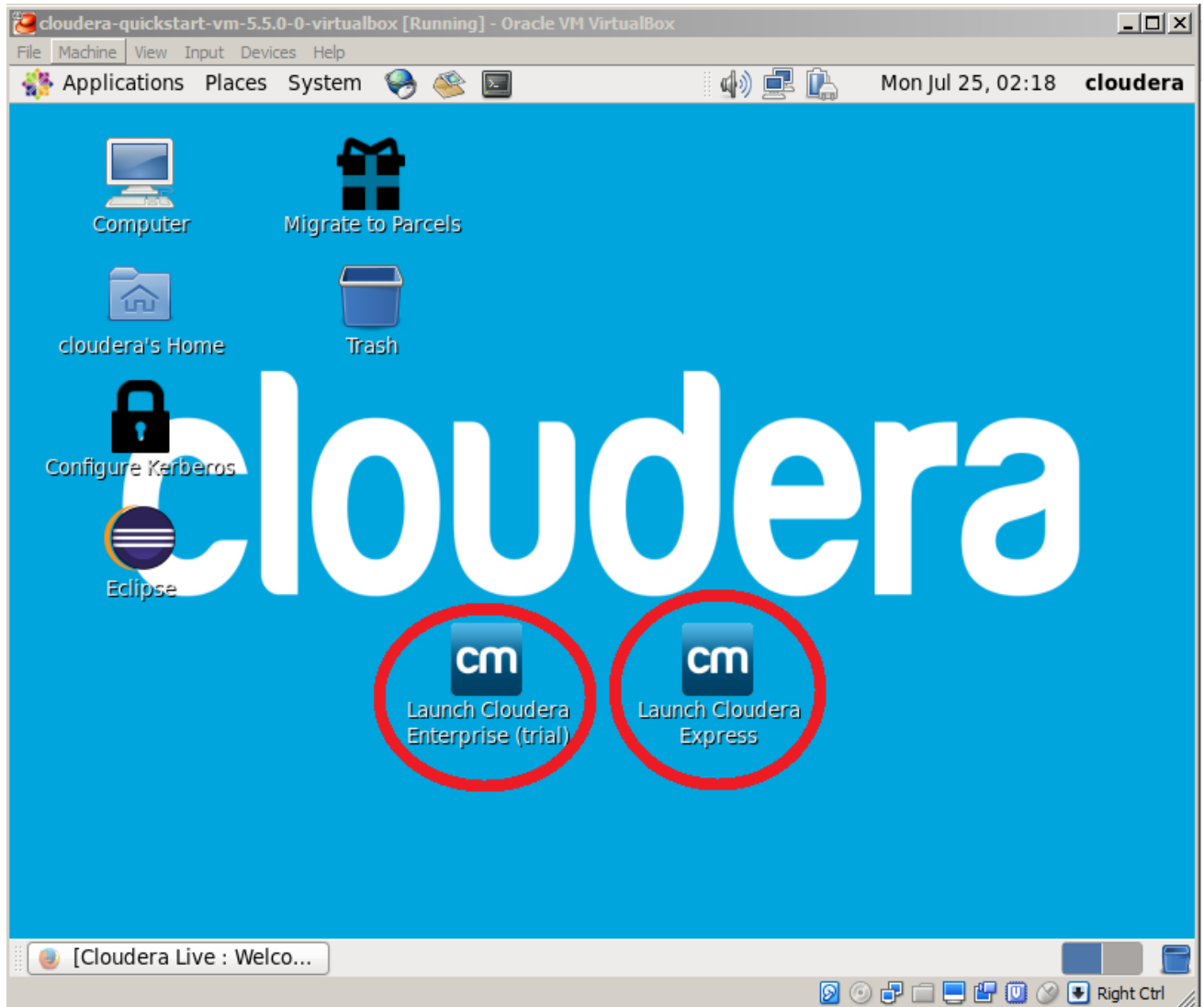


Figure 3.2.1-cloudera os

3.3 Eclipse IDE :

It will be used to write map reduce code and produce jar files which are used for data processing. The code will be in java .

3.4 Visual studio :

It will be used to write C# code to extract the required datasets from API's.

CHAPTER 4

DATA SET

Data set is in csv format (comma separated value).

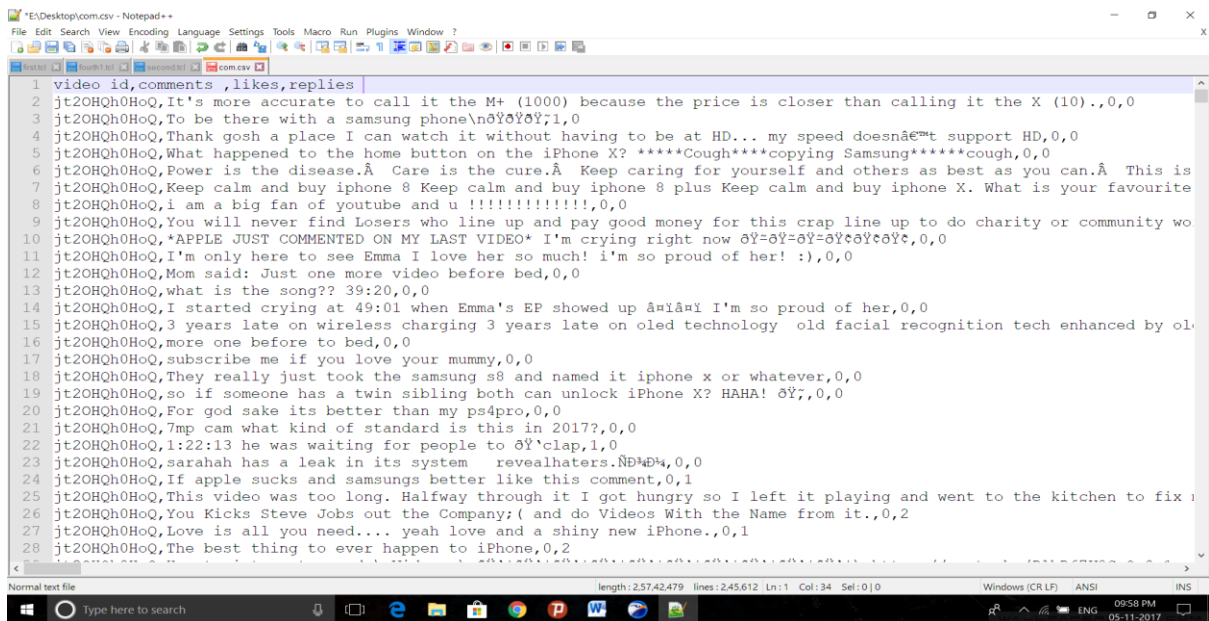
4.1 Data set 1: The dataset we use for this analysis contain nearly 240000 lines with each line representing following attributes. We use this dataset to find total number of comments for each distinct video so that we can find the video with more number of comments.

It contains four columns (video id, comments, like , replies).

Video Id : A unique id for videos to distinguish them separately.

Comments : Certain statements given by users regarding video.

Like & Replies : Each comment will have likes and replies.



```
1 video id,comments ,likes,replies
2 jt2OHQh0HoQ,It's more accurate to call it the M+ (1000) because the price is closer than calling it the X (10)..,0,0
3 jt2OHQh0HoQ,To be there with a samsung phone\nðŸðŸðŸ;1,0
4 jt2OHQh0HoQ,Thank gosh a place I can watch it without having to be at HD... my speed doesnâ€™t support HD,0,0
5 jt2OHQh0HoQ,What happened to the home button on the iPhone X? *****Cough*****copying Samsung*****cough,0,0
6 jt2OHQh0HoQ,Power is the disease.Â Care is the cure.Â Keep caring for yourself and others as best as you can.Â This is
7 jt2OHQh0HoQ,Keep calm and buy iphone 8 Keep calm and buy iphone 8 plus Keep calm and buy iphone X. What is your favourite
8 jt2OHQh0HoQ,i am a big fan of youtube and u !!!!!!!!!!!!!,0,0
9 jt2OHQh0HoQ,You will never find Losers who line up and pay good money for this crap line up to do charity or community wo
10 jt2OHQh0HoQ,*APPLE JUST COMMENTED ON MY LAST VIDEO* I'm crying right now ðŸ=ðŸ=ðŸ=ðŸcðŸsðŸc,0,0
11 jt2OHQh0HoQ,I'm only here to see Emma I love her so much! i'm so proud of her! :),0,0
12 jt2OHQh0HoQ,Mom said: Just one more video before bed,0,0
13 jt2OHQh0HoQ,what is the song?? 39:20,0,0
14 jt2OHQh0HoQ,I started crying at 49:01 when Emma's EP showed up â€œIâ€™m so proud of her,0,0
15 jt2OHQh0HoQ,3 years late on wireless charging 3 years late on oled technology old facial recognition tech enhanced by old
16 jt2OHQh0HoQ,more one before to bed,0,0
17 jt2OHQh0HoQ,subscribe me if you love your mummy,0,0
18 jt2OHQh0HoQ,They really just took the samsung s8 and named it iphone x or whatever,0,0
19 jt2OHQh0HoQ,so if someone has a twin sibling both can unlock iPhone X? HAHA! ðŸ;,0,0
20 jt2OHQh0HoQ,For god sake its better than my ps4pro,0,0
21 jt2OHQh0HoQ,7mp cam what kind of standard is this in 2017?,0,0
22 jt2OHQh0HoQ,1:22:13 he was waiting for people to ðŸ'clap,1,0
23 jt2OHQh0HoQ,sarahah has a leak in its system revealhaters.ND4D%,0,0
24 jt2OHQh0HoQ,If apple sucks and samsungs better like this comment,0,1
25 jt2OHQh0HoQ,This video was too long. Halfway through it I got hungry so I left it playing and went to the kitchen to fix
26 jt2OHQh0HoQ,You Kicks Steve Jobs out the Company:( and do Videos With the Name from it.,0,2
27 jt2OHQh0HoQ,Love is all you need.... yeah love and a shiny new iPhone.,0,1
28 jt2OHQh0HoQ,The best thing to ever happen to iPhone,0,2
```

Figure 4.1-Sample Datset

4.2 Dataset 2: This dataset contains following attributes using which we will be finding categories with number of videos uploaded.

Column No	Attribute	Description
1	Video ID	A unique identification value
2	Channel Name	Name of channel
3	Upload Details	Time and date when video is uploaded
4	Category	Eg : Entertainment ,education
5	Length	Duration of the video
6	No of views	Indicates how many times it is played.
7	Rating	User ratings
8	No. of ratings	Total number of ratings for each video
9	No. of comments	Total number of comments for each video
10	Related Video Ids	Similar videos as of video id's

Sample dataset:

QuRYeRnAuXM	EvilSquirrelPictures	1135	Pets & Animals	252	1075	4.96	46	86
gFa1YMEJFag	nRcovJn9xHg	3TYqkBJ9YRk	rSJ8QZWBegU	0TZqX5MbXMA				
UEvVksP91kg	ZTopArY7Nbg	0RvGi2Rne8	HT_QIOJbDpg	YZev1imoxX8	8qQrrfUTmh0			
zQ83d_D2MGs	u6_DQQjLsAw	73Wz9CQFDtE						
3TYqkBJ9YRk	hggh22	1135	Comedy	169	228	5	5	3
gFa1YMEJFag	UEvVksP91kg	rSJ8QZWBegU	nRcovJn9xHg	sVkuOk4jmCo				
ZTopArY7Nbg	HT_QIOJbDpg	0RvGi2Rne8	ShhClb6J-NA	g9e1alirMhc				
YZev1imoxX8	I4yKEK9o8gA	zQ83d_D2MGs	1GKaVzNDbul	yuZhwV24PmM				
DomumdGQsG8	hiSmlmXp-aU	pFUYi7dp1WU	2l6vwAIAqNU					
rSJ8QZWBegU	TimeGem	1135	Entertainment	95	356	4.31	13	1
QuRYeRnAuXM	gFa1YMEJFag	UEvVksP91kg	3TYqkBJ9YRk	nRcovJn9xHg				
sVkuOk4jmCo	ZTopArY7Nbg	gBcu22Vv1nY	HT_QIOJbDpg	0RvGi2Rne8	ShhClb6J-NA			
g9e1alirMhc	YZev1imoxX8	I4yKEK9o8gA	zQ83d_D2MGs	1GKaVzNDbul				
yuZhwV24PmM	DomumdGQsG8	hiSmlmXp-aU	pFUYi7dp1WU					
nRcovJn9xHg	wooochacha	1135	Entertainment	118	1115	2.23	57	73
QuRYeRnAuXM	gFa1YMEJFag	UEvVksP91kg	3TYqkBJ9YRk	ZTopArY7Nbg				
gBcu22Vv1nY	HT_QIOJbDpg	0RvGi2Rne8	ShhClb6J-NA	g9e1alirMhc				
YZev1imoxX8	I4yKEK9o8gA	zQ83d_D2MGs	1GKaVzNDbul	yuZhwV24PmM				
DomumdGQsG8	hiSmlmXp-aU	pFUYi7dp1WU	2l6vwAIAqNU	WWqed9u6rr4				
UEvVksP91kg	johnx113	1135	Entertainment	83	281	2.67	9	16
gFa1YMEJFag	QuRYeRnAuXM	3TYqkBJ9YRk	0TZqX5MbXMA	rSJ8QZWBegU				
nRcovJn9xHg	sVkuOk4jmCo	ZTopArY7Nbg	HT_QIOJbDpg	0RvGi2Rne8	g9e1alirMhc			
YZev1imoxX8	I4yKEK9o8gA	zQ83d_D2MGs	1GKaVzNDbul	yuZhwV24PmM				
DomumdGQsG8	hiSmlmXp-aU	pFUYi7dp1WU	2l6vwAIAqNU					

CHAPTER 5

SYSTEM MODELING

5.1 System Modeling : In this section , we will discuss about algorithm ,block diagrams, the stages of map reduce, example for mapreduce. We will also discuss about the main advantages of using Hadoop and MapReduce.

5.2 ADVANTAGES:

Fast - Hadoop uses a storage method known as distributed file system, which basically implements a mapping system to locate data in cluster. The tools used for data processing, such as MapReduce programming, are also generally located in the very same servers, which allows for faster processing of data.

Parallel processing - One of the primary aspects of the working of MapReduce programming is that it divides tasks in a manner that allows their execution in parallel. Parallel processing allows multiple processors to take on these divided tasks, such that they run entire programs in less time.

Scalability - This is largely because of its ability to store as well as distribute large data sets across plenty of servers. These servers can be inexpensive and can operate in parallel. And with each addition of servers one adds more processing power.

Flexibility -Business organizations can make use of Hadoop MapReduce programming to have access to various new sources of data and also operate on different types of data, whether they are structured or unstructured. This allows them to generate value from all of the data that can be accessed by them.

Simple model of programming-Among the various advantages that Hadoop MapReduce offers, one of the most important ones is that it is based on a simple programming model. This basically allows programmers to develop MapReduce programs that can handle tasks with more ease and efficiency.

The programs for MapReduce can be written using Java, which is a language that isn't very hard to pickup and is also used widespread. Thus, it is easy for people to learn and write programs that meets their data processing needs sufficiently.

5.3 Algorithm:

MapReduce is a set of Java classes run on YARN with the purpose of processing massive amounts of data and reducing this data into output files. HDFS works with MapReduce to divide the data in parallel fashion on local or parallel machines. Parallel structure requires that the data is immutable and cannot be updated. It begins with the input files where the data is initially stored typically residing in HDFS. These input files are then split up into input format which selects the files, defines the input splits, breaks the file into tasks and provides a place for record reader objects. The input format defines the list of tasks that makes up the map phase. The tasks are then assigned to the nodes in the system based on where the input files chunks are physically resident. The input split describes the unit of work that comprises a single map task in a MapReduce program. The record reader loads the data and converts it into key value pairs that can be read by the Mapper. The Mapper performs the first phase of the MapReduce program. Given a key and a value the mappers export key and value pairs and send these values to the reducers. The process of moving mapped outputs to the reducers is known as shuffling. Partitions are the inputs to reduce tasks, the partitioner determines which key and value pair will be stored and reduced. The set of intermediate keys are automatically stored before they are sent to the reduce function. A reducer instance is created for each reduced task to create an output format. The output format governs the way objects are written, the output format provided by Hadoop writes the files to HDFS.

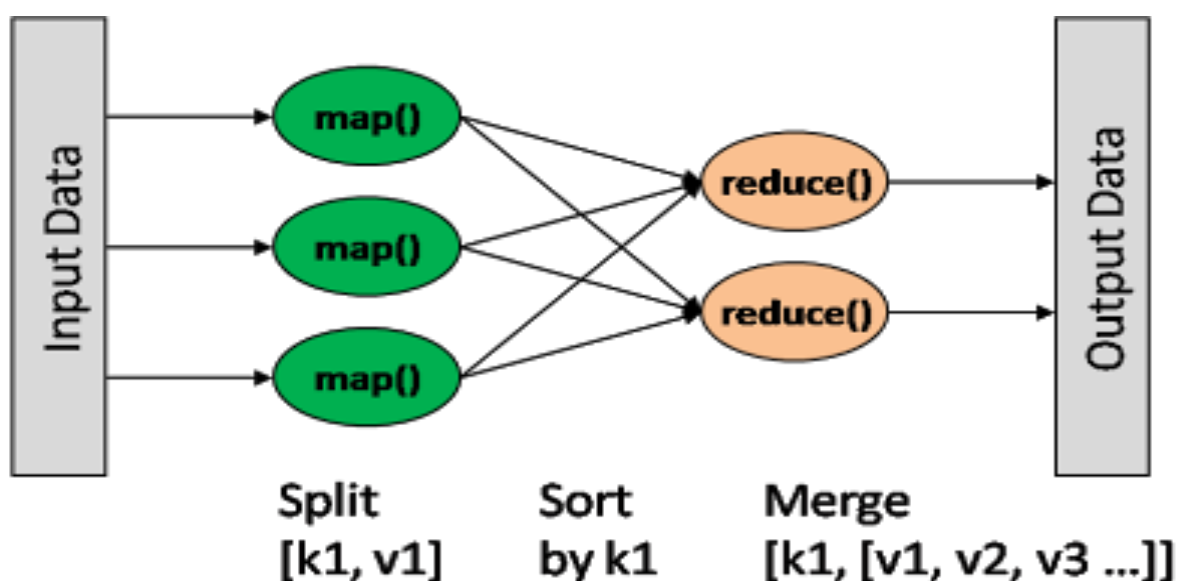


Figure 5.3.1-Map Reduce

5.4 Stages :

5.4.1 Prepare the Map() input – the "MapReduce system" designates Map processors, assigns the input key value $K1$ that each processor would work on, and provides that processor with all the input data associated with that key value.

5.4.2 Run the user-provided Map() code – Map() is run exactly once for each $K1$ key value, generating output organized by key values $K2$.

5.4.3 "Shuffle" the Map output to the Reduce processors – the MapReduce system designates Reduce processors, assigns the $K2$ key value each processor should work on, and provides that processor with all the Map-generated data associated with that key value.

5.4.4 Run the user-provided Reduce() code – Reduce() is run exactly once for each $K2$ key value produced by the Map step.

5.4.5 Produce the final output – the MapReduce system collects all the Reduce output, and sorts it by $K2$ to produce the final outcome.

5.5 Example :

Assume that a text file contains “hello , how are you ?, are you there?”. This file will be the input of program.

In mapper function input will be divided as follows:

(hello,1),(are,1),(how,1),(you,1),(are,1),(you,1),(there,1)

In shuffle and sort stage the above pairs will be arranged like this :

(are,1), (are,1), (hello,1), (how,1),(there,1),(you,1),(you,1)

In reduce stage the counter for the same words will be incremented . The key value pairs will look like this:

(hello,1),(are,2),(how,1),(you,2),(there,1).

CHAPTER 6

IMPLEMENTATION AND RESULTS

6.1 Implementation : In this section ,we will discuss about the programming code and the results .

6.2 Problem statement 1: In this problem , we extract the file which contains video ids and corresponding comments count.

6.2.1 Mapper code:

```
public class comments2 extends Configured implements Tool{

    public static class MapClass extends Mapper<LongWritable,Text, Text,IntWritable>

    {

        public void map(LongWritable key,Text value,Context context)

        {

            try

            {

                String[] str=value.toString().split(",");

                String vid=str[0];

                context.write(new Text(vid),new IntWritable(1));

            }

            catch(Exception e)

            {

                System.out.println(e.getMessage());

            }

        }

    }

}
```

6.2.2 Reducer code:

```
public static class ReduceClass extends Reducer<Text,IntWritable,Text,IntWritable>

{

    private IntWritable result = new IntWritable();

    public void reduce(Text key,Iterable <IntWritable> values,Context
context)throws IOException,InterruptedException

    {

        int sum =0;

        for(IntWritable val : values)

        {

            sum+=val.get();

        }

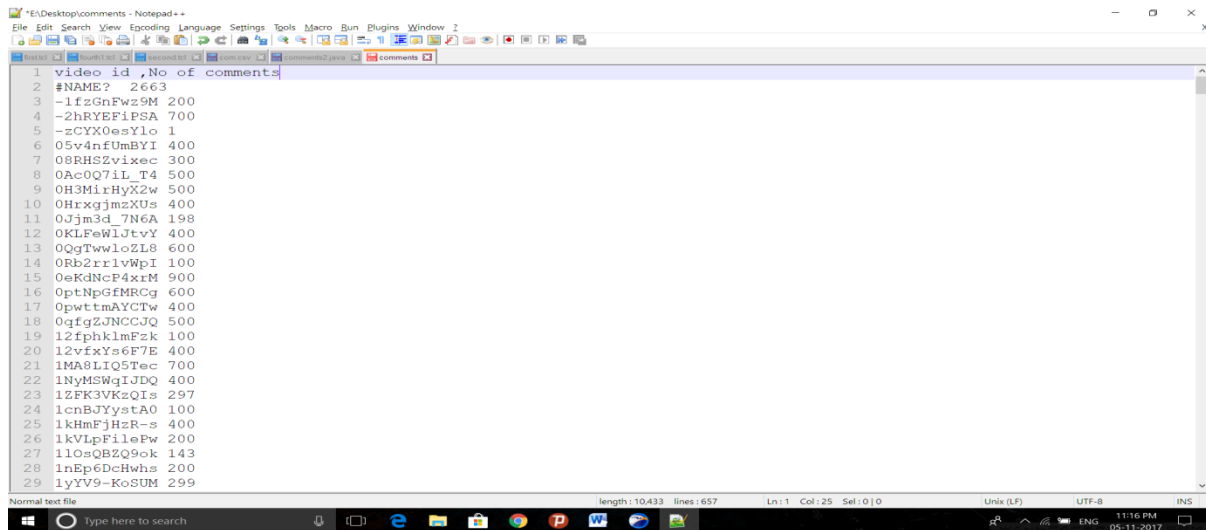
        result.set(sum);

        context.write(key,new IntWritable(sum));

    }

}
```


6.2.3 Output :



The screenshot shows a Notepad++ window with a file named 'comments'. The text inside is a list of 29 lines, each containing a video ID and its corresponding number of comments. The video IDs are alphanumeric strings, and the comment counts are integers. The window's status bar at the bottom indicates the file is a 'Normal text file', has a length of 10,433, and contains 657 lines.

video id	No of comments
#NAME?	2663
-1fzGnFwz9M	200
-2hRYEFiPSA	700
-zCYX0esYlo	1
05v4nfUmBYI	400
08RHSZvixec	300
0Ac0Q7iL_T4	500
0H3M1rHyX2w	500
0HrxgjmzXUs	400
0Jjm3d_7N6A	198
0KLFwWlJtvY	400
0QgTwWloZL8	600
0Rb2rrlvWpI	100
0eKdNcP4xrM	900
0ptNpGFMRcg	600
0pwttmAYCTw	400
0qfgZJNCCJQ	500
12fphklmFzk	100
12vfxYs6F7E	400
1MA8LIQ5Tec	700
1NyMSWqIJDQ	400
1ZFK3VKzQIs	297
1cnBJYystA0	100
1kHmFjHzR-s	400
1kVLpFilePw	200
1lOsQBZQ9ok	143
1nEp6DcHwhs	200
1yYV9-KoSUM	299

Figure 6.2.3.1-Output 1

The above output file contains video ID's with respective No. of comments.

6.3 Problem statement 2: In this statement, we will find category with most no. of videos uploaded.

CONFIGURATION CODE:

```
package project;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
public class project extends Configured implements Tool {
    @Override
    public int run(String[] args) throws Exception
    {
        if(args.length!=2)
        {
            System.out.printf("Usage : project<input dir><output dir>\n");
            return -1;
        }
    }
}
```

```

Job job = new Job(getConf());
job.setJarByClass(projectMapper.class);
job.setJobName("project");
org.apache.hadoop.mapreduce.lib.input.FileInputFormat.setInputPaths(job,
new Path(args[0]));
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.setOutputPath(jo
b,new Path(args[1]));
job.setMapperClass(projectMapper.class);
job.setMapOutputKeyClass(Text.class);
job.setMapOutputValueClass(IntWritable.class);
job.setNumReduceTasks(0);
boolean success=job.waitForCompletion(true);
if(success)
{
long Education=job.getCounters().findCounter("project"
,"Education").getValue(); 13

```

```

long FilmAnimation=job.getCounters().findCounter("project"
,"FilmAnimation").getValue();
long AutosVehicles=job.getCounters().findCounter("project"
,"AutosVehicles").getValue();
long Music=job.getCounters().findCounter("project" ,"Music").getValue();
long PetsAnimals=job.getCounters().findCounter("project"
,"PetsAnimals").getValue();
long Sports=job.getCounters().findCounter("project" ,"Sports").getValue();
long TravelEvents=job.getCounters().findCounter("project"
,"TravelEvents").getValue();
long Gaming=job.getCounters().findCounter("project" ,"Gaming").getValue();
long PeopleBlogs=job.getCounters().findCounter("project"
,"PeopleBlogs").getValue();
long Comedy=job.getCounters().findCounter("project" ,"Comedy").getValue();
long Entertainment=job.getCounters().findCounter("project"
,"Entertainment").getValue();
long NewsPolitics=job.getCounters().findCounter("project"
,"NewsPolitics").getValue();
long HowtoStyle=job.getCounters().findCounter("project"
,"HowtoStyle").getValue();
long ScienceTechnology=job.getCounters().findCounter("project"
,"ScienceTechnology").getValue();
long NonprofitsActivism=job.getCounters().findCounter("project"
,"NonprofitsActivism").getValue();
long UNA=job.getCounters().findCounter("project" ,"UNA").getValue();
long []
set={Education,FilmAnimation,AutosVehicles,Music,PetsAnimals,Sports,TravelE
vents,Gaming,PeopleBlogs,Comedy,Entertainment,NewsPolitics,HowtoStyle,Sc
ienceTechnology,NonprofitsActivism,UNA};
String []
setvalue={"Education","FilmAnimation","AutosVehicles","Music","PetsAnimals
","Sports","TravelEvents","Gaming","PeopleBlogs","Comedy","Entertainment",
"NewsPolitics","HowtoStyle","ScienceTechnology","NonprofitsActivism","UNA"
};

```

```

System.out.println("Education =\t"+Education);
System.out.println("FilmAnimation =\t"+FilmAnimation);
System.out.println("AutosVehicles =\t"+AutosVehicles);
System.out.println("Music =\t"+Music);
System.out.println("PetsAnimals =\t"+PetsAnimals);
System.out.println("Sports =\t"+Sports);
System.out.println("TravelEvents =\t"+TravelEvents);
System.out.println("Gaming =\t"+Gaming);
System.out.println("PeopleBlogs =\t"+PeopleBlogs);
System.out.println("Comedy =\t"+Comedy);
System.out.println("Entertainment =\t"+Entertainment);
System.out.println("NewsPolitics =\t"+NewsPolitics);
System.out.println("HowtoStyle =\t"+HowtoStyle);
System.out.println("ScienceTechnology =\t"+ScienceTechnology);
System.out.println("NonprofitsActivism =\t"+NonprofitsActivism);
System.out.println("others =\t"+UNA);
for(int i=0;i<16;i++)
{
    for (int j=0;j<15;j++)
    {
        if(set[j]<set[j+1])
        {
            long temp=set[j];
            set[j]=set[j+1];
            set[j+1]=temp;
            String s=setvalue[j];
            setvalue[j]=setvalue[j+1];
            setvalue[j+1]=s;
        }
    }
}
System.out.println("top 5 categories with maximum number of videos
uploaded\n");
for(int k=0;k<5;k++)
{
    System.out.println(setvalue[k] +"="+ set[k]+"\n");
} 15

```

```

return 0;
}
else
{
return 1;
}
}
public static void main(String[] args) throws Exception
{
int exitcode=ToolRunner.run(new org.apache.hadoop.conf.Configuration(),
new project(),args);
System.exit(exitcode);
}
}

```

6.3.1 Output :

Category	Number of videos uploaded
Education	65
FilmAnimation	260
AutosVehicles	77
Music	862
PetsAnimals	95
Sports	251
TravelEvents	112
Gaming	0
PeopleBlogs	398
Comedy	414
Entertainment	908
NewsPolitics	333
HowtoStyle	137
ScienceTechnology	80
NonprofitsActivism	42
others	32

6.4 Procedure:

The Mapper code and Reducer code will be converted into jar file and jar file will be used for executing the program.

- Command for executing .jar file to get result

```
hadoop jar project12.jar 'inputpath in hdfs' 'outputpath in hdfs'
```

6.5 Analysis:

6.5.1: The below graph shows the video id's and the respective comment count of each video in a barchart.

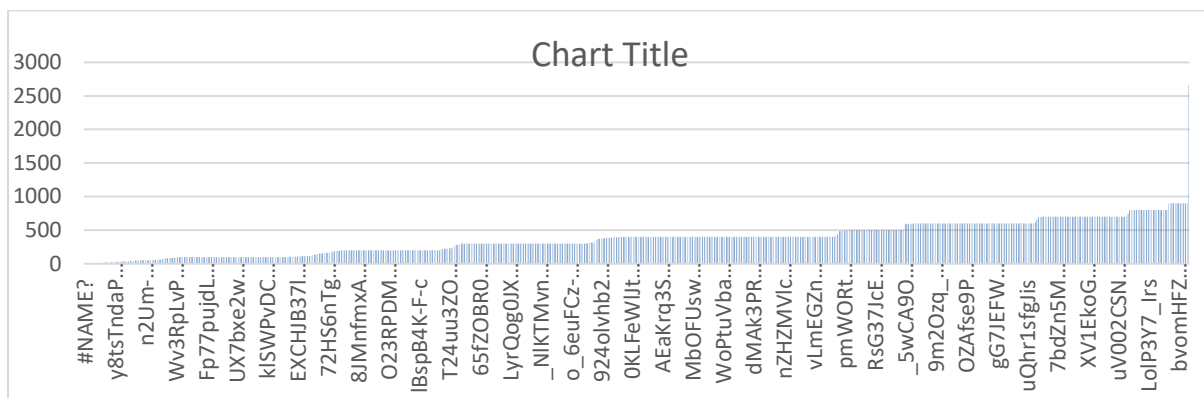


Figure 6.5.1- Analysis 1

6.2.2: The below graph shows the total number of videos in specific category or stream.

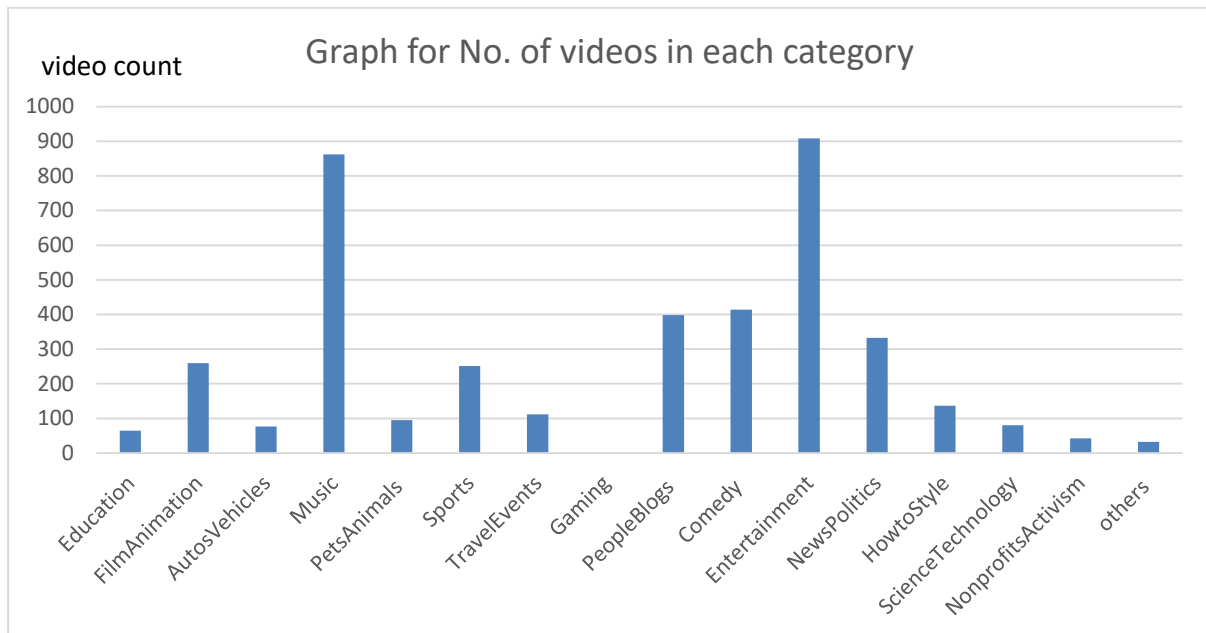


Figure 6.5.2-Analysis 2

CHAPTER 7

IMPLEMENTATION AND RESULTS -2

7.1 Implementation : In this section ,we will discuss about how to access data directly from youtube.

7.2 Generate API Key to Fetch YouTube Data : To communicate with YouTube API an Application program interface Key is required, Google Developer allows you to create a unique key to connect to YouTube.

Step 1 : Log into <https://developers.Google.com/> with existing credentials.

Step 2 : To create the unique API key for retrieving data, a new project needs to be created from the Google provided developer's console.

Step 3 : Go to <https://console.developers.Google.com/project>

Step 4 : Click create project.

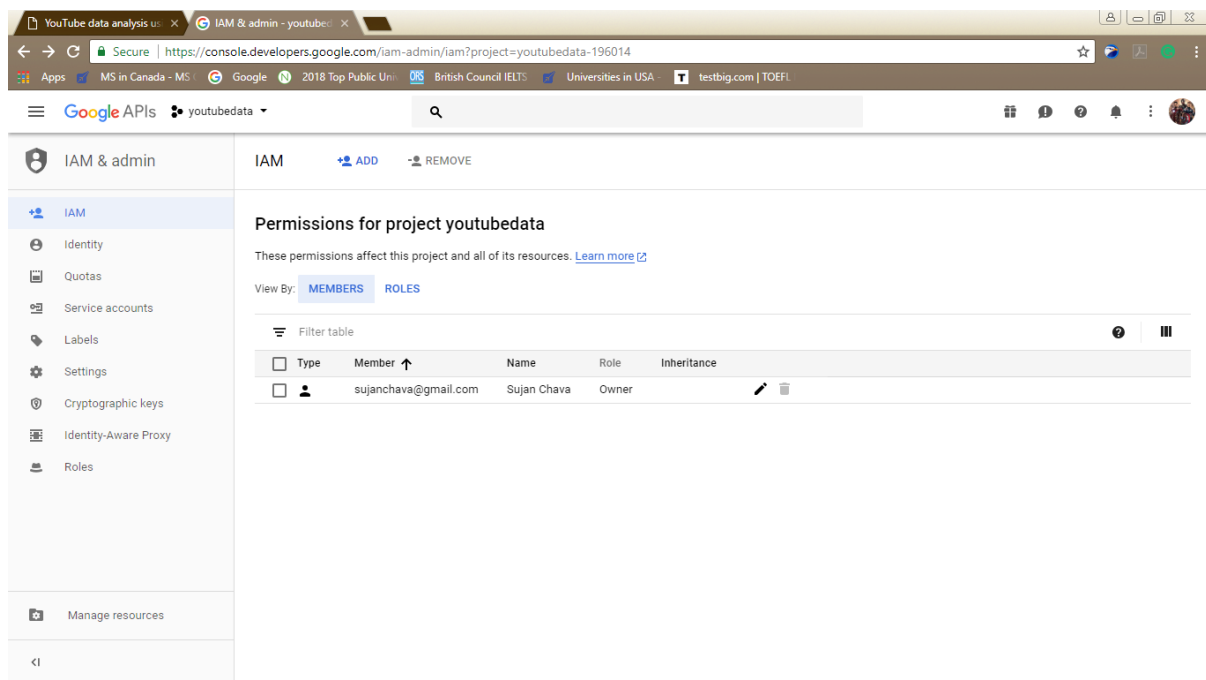


Figure 7.2.1- Project creation

Step 5 A new project needs to be created. Provide a name for the project.

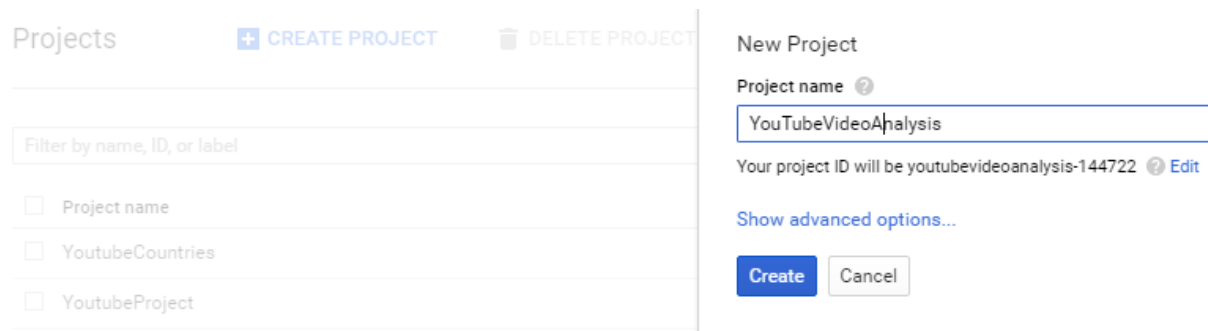


Figure 7.2.2- Naming project

Step 6 To create a new API key Google provides the YouTube Data API that is available under the developer tools.

Library > YouTube APIs > YouTube Data API

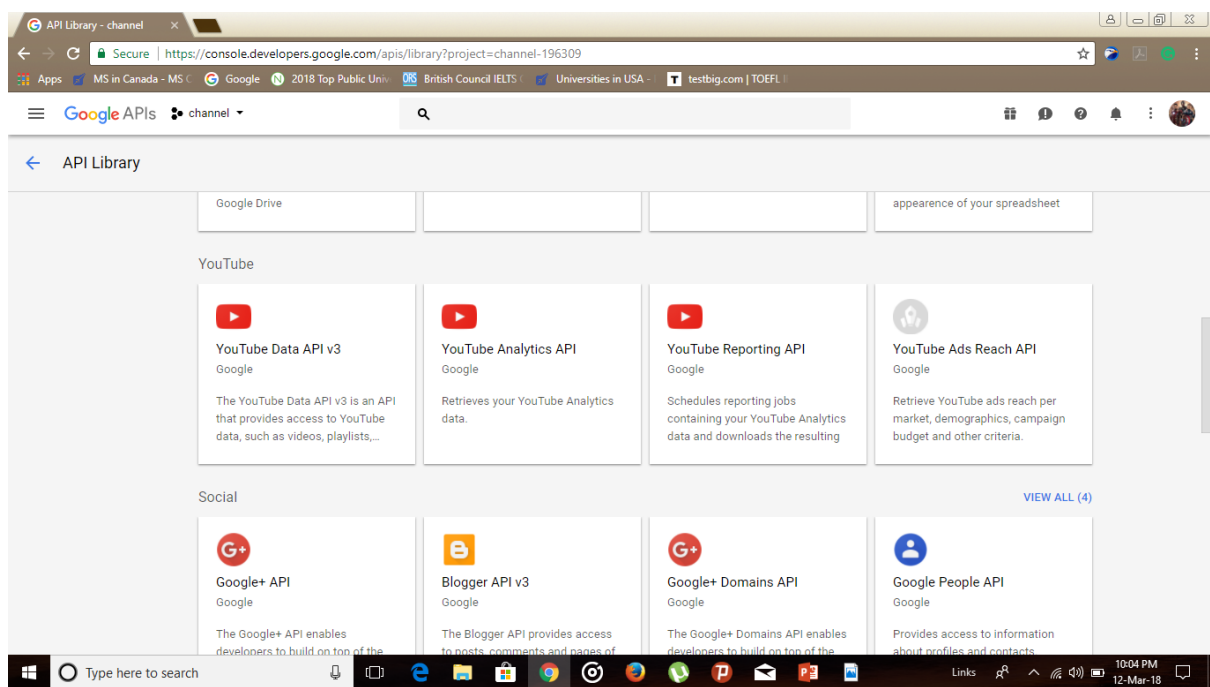


Figure 7.2.3- Enabling API

Step 7 To utilize the YouTube Data API, it needs to be enabled under the logged in credentials. Click “Enable” under the YouTube Data API v3.

Dashboard > YouTube Data API v3: Enable

Step 8 Once the YouTube data API is enabled, create credentials in order to utilize the API.

To create credentials

Dashboard > Go to Credentials Button

Step 9 Add credentials to the project. YouTube provides three options for creating an API Key.

- ☐ API key
- ☐ client ID
- ☐ service account

The project utilizes the “client ID” option.

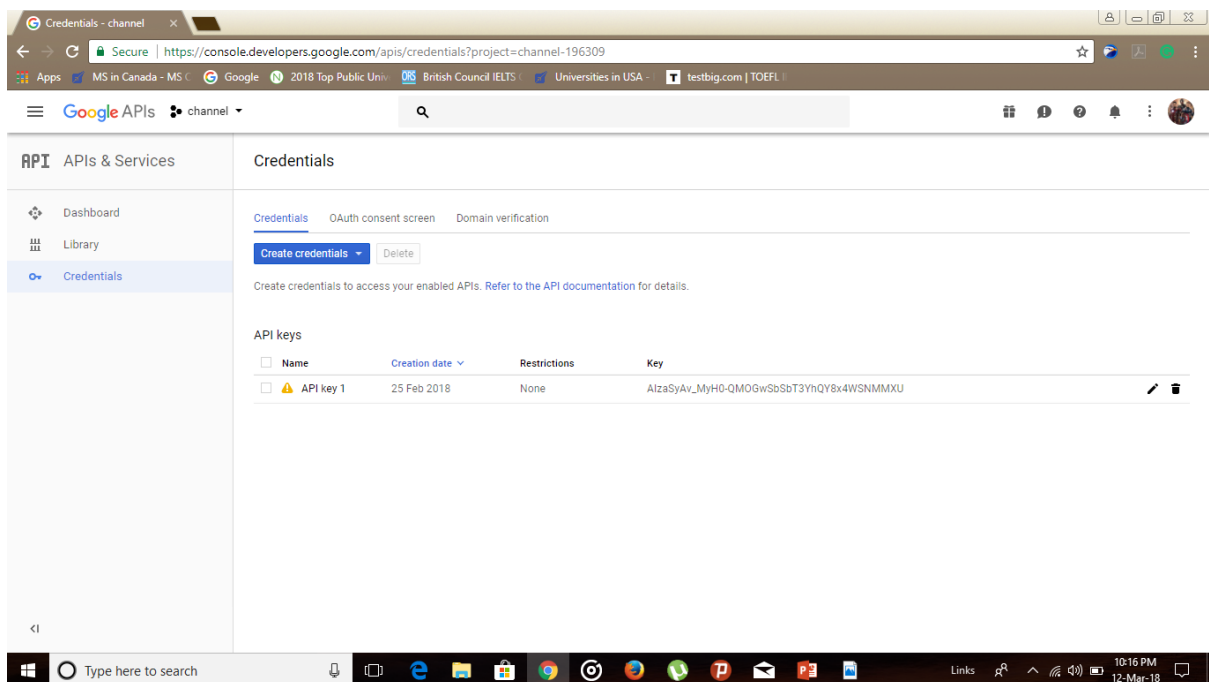


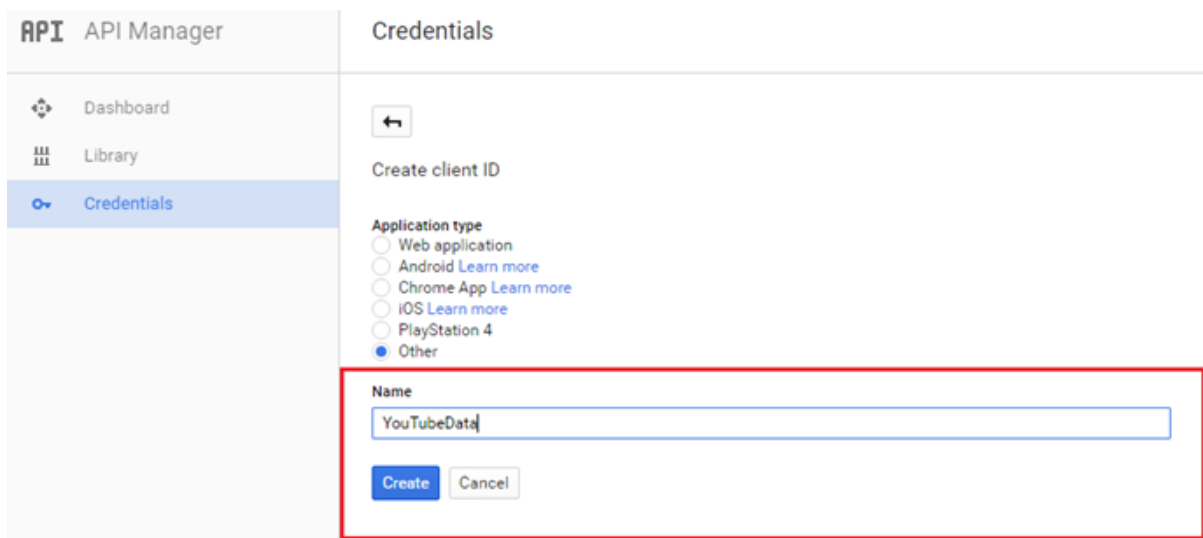
Figure 7.2.4- Adding credentials

Step 10 Create Client ID: To create a JSON file to fetch data, we need to select the application that will be using the data. YouTube Data API offers the following options

- ☐ Web application
- ☐ Android
- ☐ Chrome App
- ☐ iOS
- ☐ PlayStation 4
- ☐ Other

The project utilizes the “**Other**” option.

Step 11 Provide a name for the Client ID



The screenshot shows the Google API Manager interface. On the left is a sidebar with 'API Manager' at the top and a menu containing 'Dashboard', 'Library', and 'Credentials' (which is selected). The main area is titled 'Credentials' and contains a 'Create client ID' button. Below this, under 'Application type', several radio buttons are listed: 'Web application', 'Android [Learn more](#)', 'Chrome App [Learn more](#)', 'iOS [Learn more](#)', 'PlayStation 4', and 'Other' (which is selected). A red rectangular box highlights the 'Name' input field, which contains the text 'YouTubeData'. Below the input field are 'Create' and 'Cancel' buttons.

Figure 7.2.5- Providing name for client ID

Step 12 YouTube creates the Client ID for the project to utilize and provides the API Key.

Step 13 Once the OAuth 2.0 client IDs are created for the project the “**client_secret JSON**” file needs to be downloaded to be added to the project.

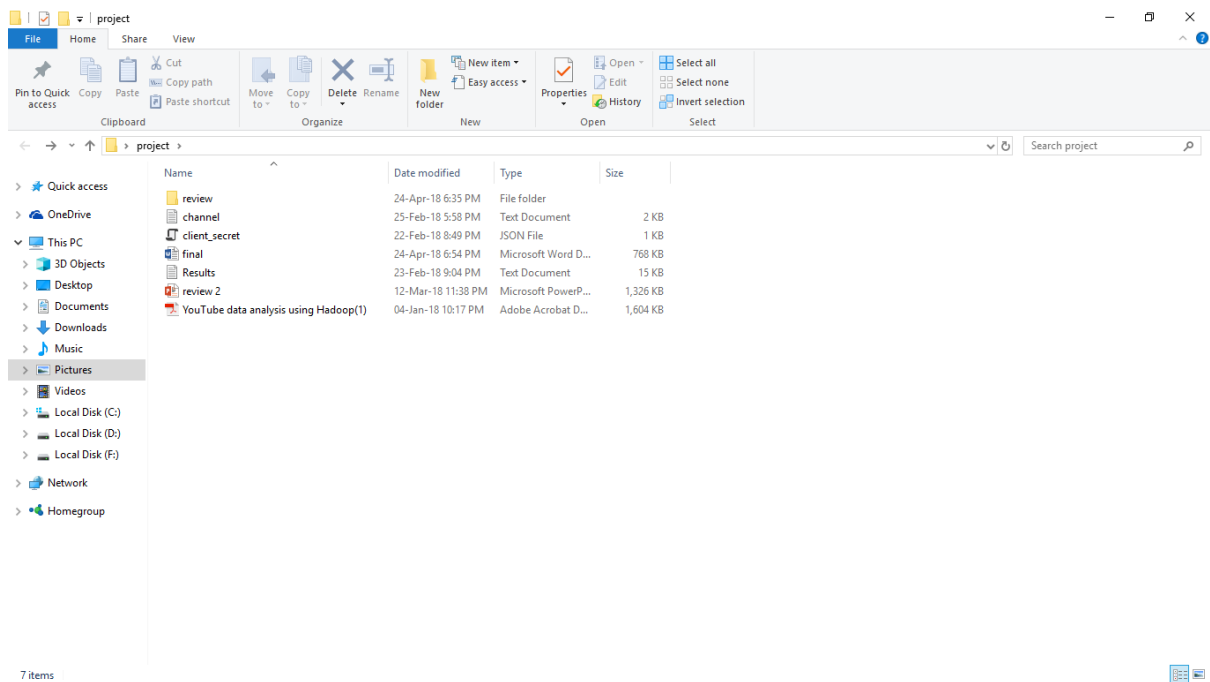


Figure 7.2.6- Project creation

7.3 Creating a .Net(C#) Console Application to Use the YouTube API

Step 1 Create a new c# console project.

Step 2 Next step is to add the required DLLs to the .NET project. The DLLs can be downloaded from Google’s NuGet Packages.

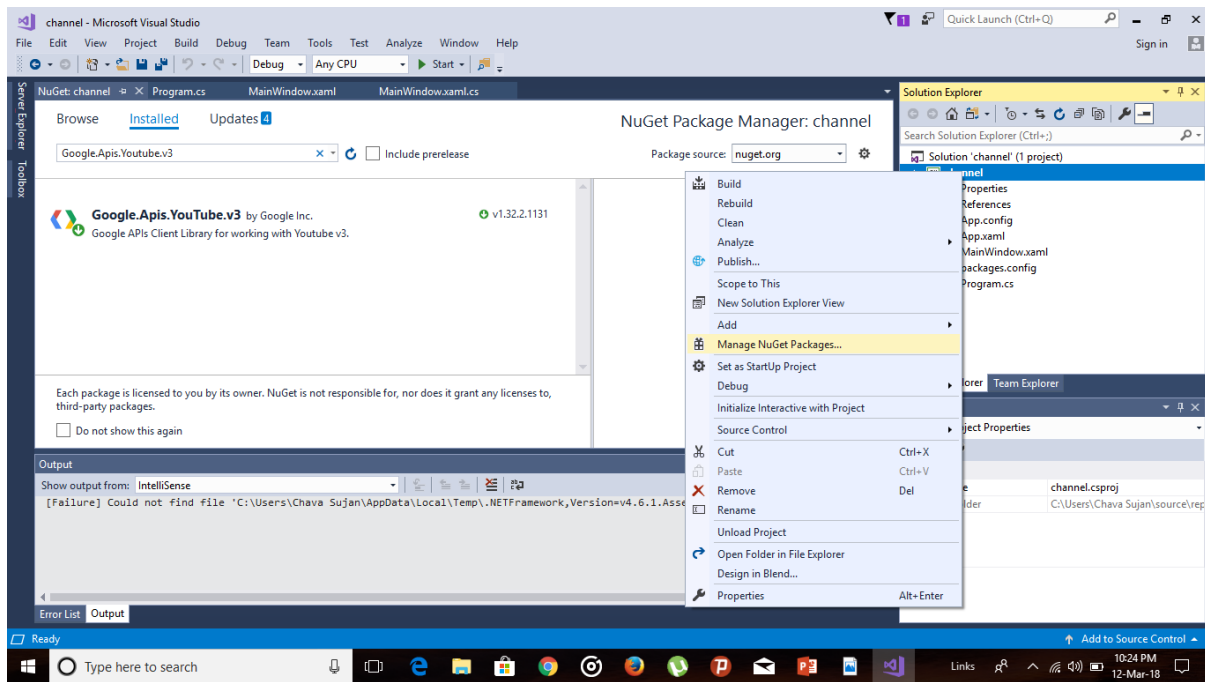


Figure 7.3.1-Adding DLL

Open NuGet Console and search for “Google.Apis.YouTube.v3”.

Step 3 Once all packages are installed, create a new class- **YouTubeInterfaceClass** and add the reference that will be used to fetch data.

```
using System;
using Google.Apis.YouTube.v3;
using Google.Apis.Services;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using Google.Apis.Auth.OAuth2;
using Google.Apis.Util.Store;
using Google.Apis.YouTube.v3.Data;
using System.IO;
using System.Net.Mime;
using System.Text;
using System.Xml;
```

Figure 7.3.2- Required Packages

Step 4 As mentioned above, Google's YouTube service needs to be authorized. We need to add the downloaded authentication JSON file to the project. To utilize the JSON file an Authorize function needs to be created.

```
namespace YoutubeVideos
{
    class Program
    {
        static void Main(string[] args)
        {
            YouTubeService yt = new YouTubeService(new BaseClientService.Initializer() { ApiKey = "AIzaSyAv_MyH0-QMOGwSbSbT
```

Figure 7.3.3- Adding API key

Step 5 Include the channel id from which we have to extract the videos.

```
var searchListRequest = yt.Search.List("contentDetails");
searchListRequest.ChannelId = "UCJjC1hn78yZqTf0vdTC6wAQ";
//var searchListResult = searchListRequest.Execute();
```

Figure 7.3.4-Including Channel ID

Step 6 Create the header for the export header file .

```
StringBuilder sbuilBuilder = new StringBuilder();
//Create header for export text file
//----- Comment/Remove this part if the header is not needed -----
sbuilBuilder.Append("ID" + "\t" + "Video Title " + "\n");
sbuilBuilder.Append(Environment.NewLine);
```

Figure 7.3.5-Creating header file

Step 7 Using YouTube services loop through the countries to get a list of records based on the search criteria.

```
var playlistItemsListRequest = yt.PlaylistItems.List("snippet");
playlistItemsListRequest.PlaylistId = "PLEbnTDJUr_I_f_BnzJkkN_J0Tl3iXtL8vq";
playlistItemsListRequest.MaxResults = 50;
var playlistItemsListResult = playlistItemsListRequest.Execute();
```

Step 8 Based on the search criteria list item. Loop through each video id to obtain additional information. The following modules were used to fetch the video information

- o Snippet
- o ContentDetails
- o Localizations
- o Statistics
- o Status

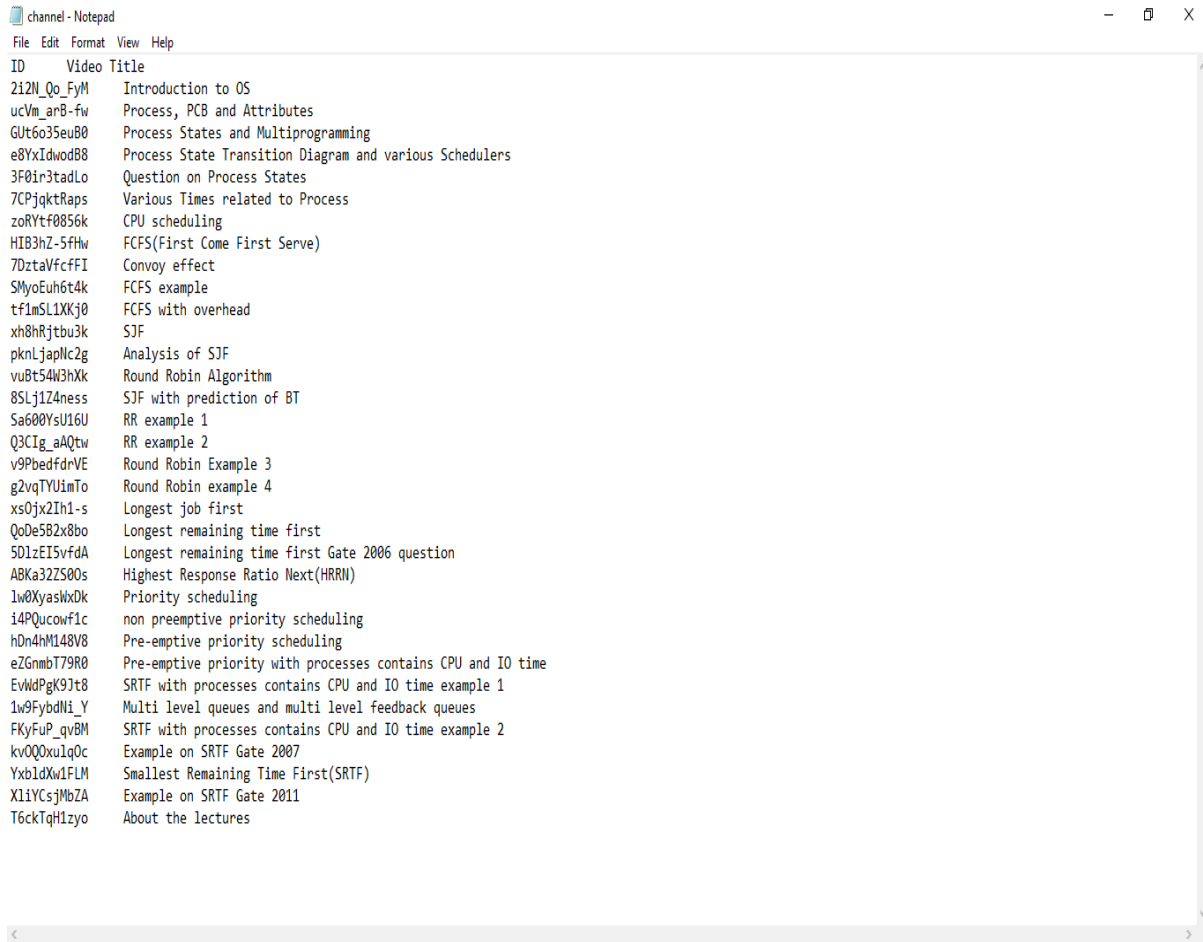
```
foreach (var ob in playlistItemsListResult.Items)
{
    sbuilBuilder.Append(ob.Snippet.ResourceId.VideoId
        + "\t" + ob.Snippet.Title + "\n"
    );
    sbuilBuilder.Append(Environment.NewLine);
}
```

Step 9 Populate the export file with the information about the video.

Step 10 Export the data into results.txt file

```
string appDirectory = AppDomain.CurrentDomain.BaseDirectory.Replace(@"C:\Users\Chava Sujan\source\repos\channel\channel\bin\Debug\", "");
File.WriteAllText(appDirectory + "channel.txt", sbuilBuilder.ToString());
}
```

7.4 Result:



ID	Video Title
212N_Qo_FyM	Introduction to OS
ucVm_arB-fw	Process, PCB and Attributes
GUt6o35euB0	Process States and Multiprogramming
e8YxIdwodB8	Process State Transition Diagram and various Schedulers
3F0ir3tadLo	Question on Process States
7CPjqktRaps	Various Times related to Process
zoRYtf0856k	CPU scheduling
HI83hZ-5fhw	FCFS(First Come First Serve)
7DztaVfcfFI	Convoy effect
SMyoEuh6t4k	FCFS example
tf1mSL1XKj0	FCFS with overhead
xh8hRjtbu3k	SJF
pknLjapNc2g	Analysis of SJF
vuBt54W3hXk	Round Robin Algorithm
8SLj1Z4ness	SJF with prediction of BT
Sa600YsU16U	RR example 1
Q3CIg_aAQtw	RR example 2
v9PbedfdrVE	Round Robin Example 3
g2vqTYUimTo	Round Robin example 4
xs0jx2Ih1-s	Longest job first
QoDe5B2x8bo	Longest remaining time first
5D1zeI5vfdA	Longest remaining time first Gate 2006 question
ABKa32ZS00s	Highest Response Ratio Next(HRRN)
lw0XyashxDk	Priority scheduling
i4PQucowf1c	non preemptive priority scheduling
hDn4hM148V8	Pre-emptive priority scheduling
eZGmbT79R0	Pre-emptive priority with processes contains CPU and IO time
EvWdPgK9Jt8	SRTF with processes contains CPU and IO time example 1
1w9FybdNi_Y	Multi level queues and multi level feedback queues
FKyFuP_qvBM	SRTF with processes contains CPU and IO time example 2
kv0Q0xulq0c	Example on SRTF Gate 2007
YxbldXw1FLM	Smallest Remaining Time First(SRTF)
X1iYCsJMbZA	Example on SRTF Gate 2011
T6ckTqH1zyo	About the lectures

Figure 7.4.1-Result

CHAPTER 8

CONCLUSION

8.1 Summary :

There is a need for a reliable and efficient mechanism for storage and processing of large scale video data today. With the amount of video data being collected every day, analysis of this data might provide valuable insights, which may aid organizations and governments in many ways. An efficient video processing analyzing system will not only help in analyzing existing video archives, but also move towards scaling existing video processing algorithms to frameworks such as Hadoop. This project has discussed the need for performing efficient big video data analytics in today's world. Moreover, it has also discussed the issues and challenges involved in performing analysis of the large scale video data using Hadoop, and finally proposed solutions to the same.

REFERENCES

- [1] SINTEF. "Big Data, for better or worse: 90% of world's data generated over last two years." ScienceDaily. ScienceDaily, 22 May 2013.
<www.sciencedaily.com/releases/2013/05/130522085217.htm>.
- [2] https://en.wikipedia.org/wiki/Unstructured_data
- [3] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust RealTime Unusual Event Detection Using Multiple Fixed-Location Monitors," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 555-560, Mar. 2008.
- [4] <http://blog.pivotal.io/data-science-pivotal/features/large-scale-video-analytics-on-hadoop>
- [5] Yamamoto, Muneto, and Kunihiro Kaneko. "Parallel image database processing with MapReduce and performance evaluation in pseudo distributed mode." International Journal of Electronic Commerce Studies 3.2 (2013): 211-228.
- [6] http://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html
- [7] <http://blog.pivotal.io/data-science-pivotal/products/using-hadoop-mapreduce-for-distributed-video-transcoding>
- [8] Xuggler, <http://www.xuggle.com/xuggler/>.
- [9] FFmpeg, <https://www.ffmpeg.org/>.
- [10] Sweeney, Chris, et al. "HIPI: a Hadoop image processing interface for image-based mapreduce tasks." Chris. University of Virginia (2011).
- [11] Hadoop tutorial :- <https://developer.yahoo.com/hadoop/tutorial/module4.html>.