YouTube Analyzer

CptS 415, Big Data
Instructor : Yinghui Wu

Insun Lee

Kim Ngyuen

Chao Zheng

YouTube Analyzer

Abstract

Social network such as Facebook, Twitter and YouTube has been popular entertainment for people. Especially, YouTube shares different contents of videos including movie, music, entertainment and many more. By the website Fortuneload, 300 hours of videos are uploaded on a day. This reflects the fact that number of videos are increasing drastically every day. As it increases, there will be challenges of extracting right information out of them. These analysis could benefit YouTube website to improve their services. It could also help the world to see the trend over some time. In this project, this team will analysis dataset from YouTube. Part of our project description has asked to use specific algorithms, PageRank. Also, to find some useful information about statistics of dataset, search of top k item, user pattern, and influence analysis. During the process team has found out extracting top k item, range queries, and statistics was simple to find out. The challenging part was to deal with user pattern and influence analysis. Part of the reason why there was a struggle is because of dataset. Based on the dataset extracted, it was not detailed enough to create the sophisticated recommendation system. This applied on influence analysis as well. Overall, in this project, different methods of analysis was made using efficient algorithms such as PageRank.

I. Introduction

CptS 415 has taught different approach on handling big dataset. As the main goal of this project is to reflect what was taught on the class, this team decided to go with YouTube Analyzer project. Using the dataset extracted from YouTube, this group will analysis the dataset to extract important informations. This project could be challenging because of the dataset. Currently, using the link provided by the project is simple. It contains basic information about each video, but to perform more detailed analysis, this team requires more detailed dataset. For instance, if this team make analysis about music videos, what genre of music is it playing could be a important information. Such information could not be provided by the video itself. Title of the video could help this team to find genre but title of the music is not always right. Other than challenges with extracting right dataset, handling big dataset is not always easy. With a lot of information, details or crucial analysis could not be made.

With these challenges, this team has explored related works to get some direction of the project. One work is 'Statistics and Social Network of YouTube Videos'. This work made analysis of youtube video data. Such are category distribution on videos, uploading trend, distribution of video trend, and relationship between different categories provided by dataset. This paper definitely helped this team where to start. Unlike 'Statistics and Social Network of YouTube Videos', where it focused on analyzing video, this project will explore more detailed categories of videos to create recommendation system.

Another paper is 'Understanding User Behavior in Large-Scale Video-on-Demand Systems'. Unlike the first paper, this work has explored the user's pattern on watching videos over time using poisson distribution. By analyzing user's pattern, they have discovered changes of interest over time. As new videos were uploaded, changes of user interest moved based on existing videos. With this paper, this team hope to achieve detailed analysis over time. Part of making analysis is to observe the changes over time, and this project gave an idea of making different analysis one specific factor.

Goal of this project is to implement a Youtube data analyzer which could give useful information. This project will be supported by MapReduce, SQL and/or graph algorithms. This analyzer should be able to report network aggregation, return the top k search result, and perform influence analysis.

II. Problem Statement

This team acquired dataset from a link, http://netsg.cs.sfu.ca/youtubedata/. Both input and output are in text format. Input has been extracted from YouTube API with 10 categories of each video.

Video ID	DjdA-5oKYFQ
uploader	tv3636
updated_date	690
category	Comedy
length	151
views	24280
rate	3.92
rating	73
comments	71
related_videos_IDs	LKh7zAJ4nwo NxTDlnOuybo c-8VuICzXtU, y5kwKp6y8t4, oHknurFKx64, LpCCsPergb4, qF2v6mW9J2k, vcVYexixsmA, LZi2ryWsShY iO8qKGBL9vY, sV0E-cRjVSQ, OPmYbP0F4Zw zdnuTg9K5XU, UnfbKKvUG9Q, t7Y8yKNGEo8, Yph6sRK0vog wDQUKSLEEvE, GQbGA_QSXDU, qR8WRLrO2aQ, goQu8PksNn0

This chart shows a sample data extraction of a video. Video ID represents a unique 11-digit string. Age shows the date between creation of YouTube and uploaded date. Category is chosen by uploader. For related IDs, it contains 20 other videos which are related with the video. Other categories are described by its name.

As the analysis are made, this team has noticed the limitation of dataset. Many of the videos in this dataset is outdated. Due to this, there are missing videos either removed by the publisher or erased by different issues. Once the video got removed, it was hard to find contents and even name of the video. Another problem was that it is hard to create recommendation system nor detailed analysis using the information above. Members have noticed that knowing contents of the video. Knowing about if this music video is rock music or not made a huge difference in making analysis.

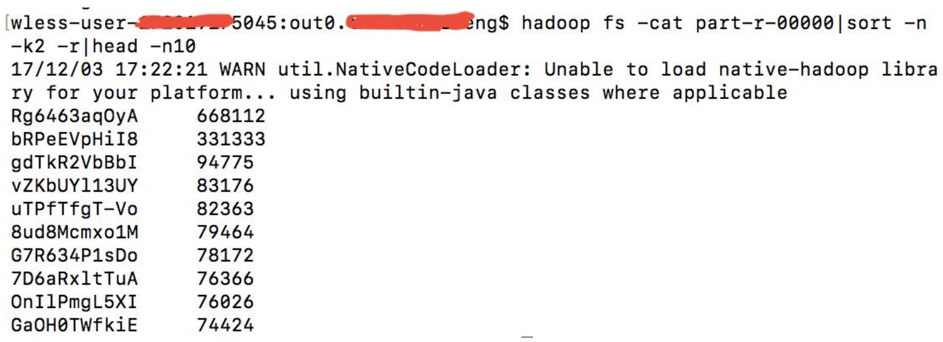
III. Solution

In this project, this team has used a software called Apache Hadoop for MapReduce algorithm. Some of its advantage is that it can handle large dataset with exponential growth rate, and support different algorithms. This includes locally weighted linear regression, naive bayes, k-means, and many more. With all this advantages, Hadoop can handle large dataset by using parallel processing. This means, it will be very efficient when it comes to processing big data. MapReduce algorithm could be explained in map and reduce. First, map converts data set into another set, where it breaks down into tuples with key and value pairs. Next, reduce part will take the output of map as input and combine data tuples into smaller set of tuples. Specifically, map will tokenize input, or sort the dataset. Then, reduce will search and reduce then return the output. Another algorithm this project used is PageRank. It shows the probability distribution to display the most relevant data among the dataset. Equation could be represented as
PageRank(A) = (1-d) + d(PageRank (T_1)/ C(T_1) + … + PageRank (T_n) / C(T_n)), where d is a damping factor between 0 to 1, T_1 means related item to A, C is the number of outbound link to T_1 . To simplify, PageRank of given page = Initial PageRank (related with given page) + (total ranking power / number of outbound $link_A$) + …. .
MapReduce algorithm is tricky to calculate its correctness and time complexity. Both can vary depending on what dataset is being evaluated and how the programmer is using the algorithm. For complexity, it could be computed using this equation provided by stackoverflow. O(n log n * s * (1/p)), where n is the number of items, s is the number of nodes, and p is the ping time between nodes. For p, this team assumes that it is equal for all nodes since data is being processed in the same network. So, time complexity will be O (n log n). For correctness, it is hard to be measured. By the insideBigData website, it is stated that developer of Hadoop had hard time configuring correctness of MapReduce algorithm. It is simply because it will vary between cases and this is common issue between data scientists. Instead, it recommends using ScaleOut Software which could help out increasing the correctness.
PageRank algorithm's correctness could be evaluated using Markov Chain. Using this, PageRank scores could be calculated and from here, scores will form the principal eigenvector of the transition matrix. Based on Markov chain coverage, and PageRank score, it is found that principal vector is invariant under the transition matrix. Thereby, Perron-Frobenius theorem will prove that PageRank's eigenvector exists. Time complexity of PageRank is O(n+m) because it will go through every vector in the data, then touch whatever nodes related to the already touched node.

IV. Experimental Study

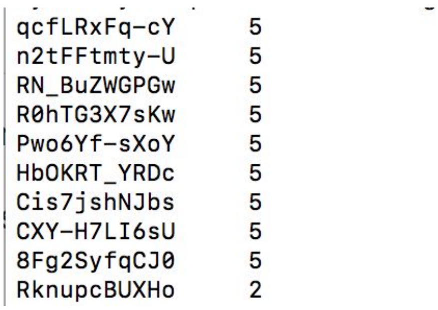
Using the information described above, this team produced some results. Based on some categories like number of views, category, and rate, top k items has been produced.

Top 10 viewed videos result:



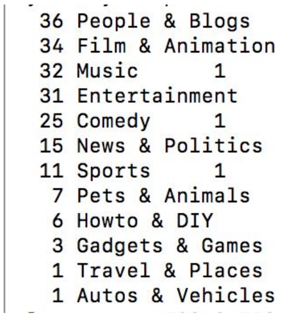
```
wless-user-[redacted]5045:out0.[redacted]eng$ hadoop fs -cat part-r-00000|sort -n
 -k2 -r|head -n10
17/12/03 17:22:21 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
Rg6463aqOyA     668112
bRPeEVpHiI8     331333
gdTkR2VbBbI     94775
vZKbUYl13UY     83176
uTPfTfgT-Vo     82363
8ud8Mcmxo1M     79464
G7R634P1sDo     78172
7D6aRxltTuA     76366
OnIlPmgL5XI     76026
GaOH0TWfkiE     74424
```

Top 10 Rated Videos



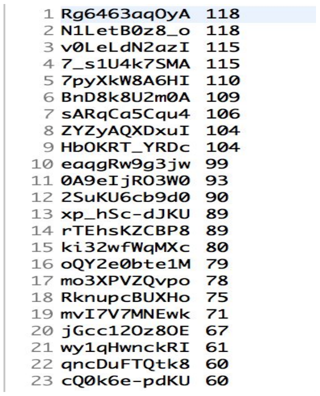
```
qcfLRxFq-cY     5
n2tFFtmty-U     5
RN_BuZWGPGw     5
R0hTG3X7sKw     5
Pwo6Yf-sXoY     5
HbOKRT_YRDc     5
Cis7jshNJbs     5
CXY-H7LI6sU     5
8Fg2SyfqCJ0     5
RknupcBUXHo     2
```

Top 10 categories



```
 36 People & Blogs
 34 Film & Animation
 32 Music      1
 31 Entertainment
 25 Comedy     1
 15 News & Politics
 11 Sports     1
  7 Pets & Animals
  6 Howto & DIY
  3 Gadgets & Games
  1 Travel & Places
  1 Autos & Vehicles
```

Search of videos with ragne [60,120]:



```
Rg6463aqOyA  118
N1LetB0z8_o  118
v0LeLdN2azI  115
7_s1U4k7SMA  115
7pyXkW8A6HI  110
BnD8k8U2m0A  109
sARqCa5Cqu4  106
ZYZyAQXDxuI  104
HbOKRT_YRDc  104
eaqgRw9g3jw  99
0A9eIjRO3W0  93
2SuKU6cb9d0  90
xp_hSc-dJKU  89
rTEhsKZCBP8  89
ki32wfWqMXc  80
oQY2e0bte1M  79
mo3XPVZQvpo  78
RknupcBUXHo  75
mvI7V7MNEwk  71
jGcc12Oz8OE  67
wy1qHwnckRI  61
qncDuFTQtk8  60
cQ0k6e-pdKU  60
```

For influence analysis, this team has used PageRank algorithm to find top k most influence videos in YouTube network.

Top 10 category

Top	Category	Total
1	Comedy	654
2	Entertainment	575
3	Music	500
4	Film & Animation	371
5	People & Blogs	318
6	News & Politics	257
7	Sports	163
8	Gadgets & Games	101
9	Howto & DIY	62
10	Pets & Animals	52

Top 10 rated

Top		
1	rpIAHWcCJVY	5
2	k7_i-K5uKSo	5
3	bHnAETS5ssE	5
4	ahrbm2G0N7g	5
5	O7oHhyIdPmc	5
6	NHf2igxB8oo	5
7	J3XM6cl_44I	5
8	F0L-tYSBw-Y	5
9	3zWi2Ig91_Q	5
10	2VHU9CBNTaA	5

Top 10 View

Top		
1	4wGR4-SeuJ0	3085087
2	B8H29jU8Wrs	2806422
3	h_CayCjo3XA	2792082
4	hut3VRL5XRE	2684989
5	ut5fFyTkKv4	2401671
6	JMvMzQ4Vu-8	2359461
7	MC8Zvl-8ziA	2317777
8	59ZX5qdIEB0	1814798
9	o_uln6CurFk	1767646
10	cvHeVdlZQPs	1707028

Total video of length [60, 120] = 557
Total Video between range[60,120] seconds in category X

Category	Total Video
Comedy	125
Entertainment	118
Music	51
Film & Animation	75
People & Blogs	50
News & Politics	49
Sports	33
Gadgets & Games	15
Howto & DIY	12
Pets & Animals	11

Page Rank

Top	Line#	PageRank Score
1	1090	0.15230242746613892
2	1497	0.15168760879025242
3	811	0.151531893028099
4	1009	0.15147045107345972
5	1549	0.1514101936466493
6	1507	0.15136395778938208
7	881	0.15134156041956306
8	623	0.15130953057948002
9	1514	0.15122525021758051
10	999	0.15121963744246494

Top	Video ID	PageRank Score
1	rkvEuAtErwQ	0.15230242746613892
2	skn2nNHH8co	0.15168760879025242
3	wvPOOXdlO8U	0.151531893028099
4	5c9OBqgmjzE	0.15147045107345972
5	CCz1kmfqL7g	0.1514101936466493
6	U4bk0UH1poQ	0.15136395778938208
7	uAIIMI0BUQ8	0.15134156041956306
8	OIkUJnXLU74	0.15130953057948002
9	nMl6B9m1rDw	0.15122525021758051
10	RG2kMfInkFA	0.15121963744246494

Conversion between Video ID to actual name of the video has failed since the video does not exist anymore.

For the user recommendation system, instead of using the given dataset, extracted new dataset from YouTube in the same format. Additionally, initial category to be evaluated was chosen before extracting the dataset. In here, movie recommendation system will be provided. At first, data cleaning process was performed to extract videos with movie category. Then, top 10 videos were chosen based on their rating. This top 10 videos will be big nodes in graph below. After that, using related videos of top 10 is placed as small red dot around the big node. Distance between each nodes are determined based on the rating. As rating is good, the distance gets smaller, while bad rating will get longer distance with dot line.

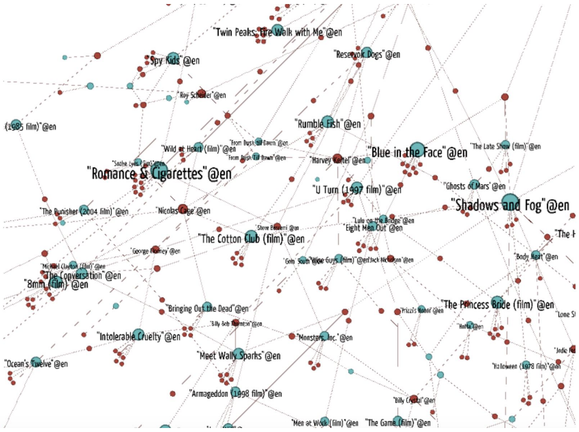


Figure 1: lined graph between movies

In Figure 1, this team has noticed the limitation of this dataset. This recommendation system will return the names of node around the chosen movie. However, depending on the category of movie, such as comedy, romance and many more, user may not necessarily like the recommendation. To improve this, another dataset has been extracted. With 10 categories from the original, genre of the movie has been extracted. With many of the genres, based on top k genre, this group only evaluated those movies with comedy, romance, Sci-Fi and action videos.

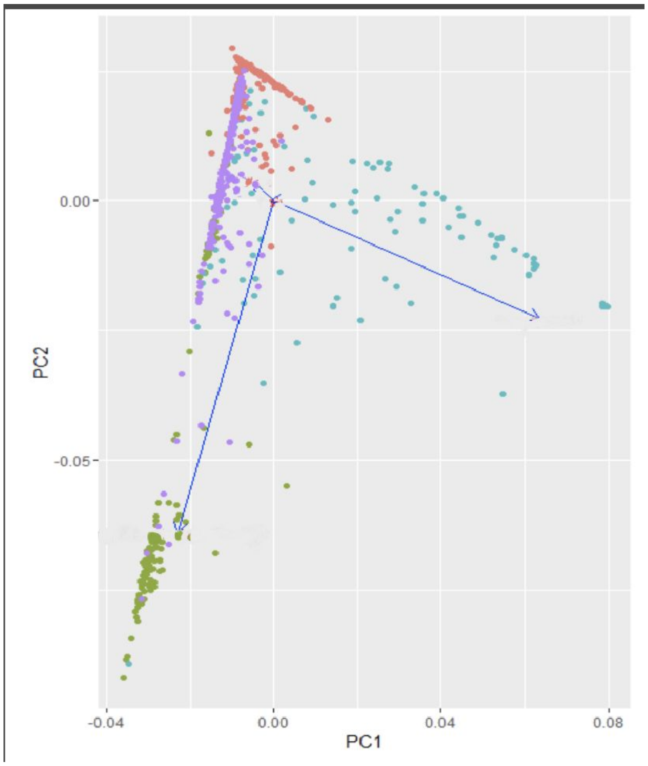


Figure 2: distribution of movies on genre

If you refer Figure 2, comedy is marked as red, purple as romance, blue as Sci-Fi, and green as action movie. As graph shows, there are many movies with Romance and Comedy. Although each genre is different, many movies are combined with at least 2 different genres. Using the graph above, now recommendation system can provide recommendation based on genre of movie that user likes.

V. Conclusion & Future Work

Although this team has produced results, failed to make analysis due to outdated dataset. As the data gets larger, there could be many different ways to make analysis. Not only based on user's activity but also based on rates, views and many more. As a future work, both top k item and search should be run with a new extracted dataset. As the converter from video ID to video name is working, using a new dataset, this group should be able to evaluate the dataset in better way.

Reference

Cheng, Xu. “Statistics and Social Network of YouTube Videos.” 2008, www.cs.sfu.ca/~jcliu/Papers/YouTube-IWQoS2008.pdf.

“Youtube Statistics - 2017.” *Digital Marketing Education*, 16 Dec. 2017, fortunelords.com/youtube-statistics/.

Yu, Hongliang. “Understanding User Behavior in Large-Scale Video-on-Demand Systems*.” UCSB, www.cs.ucsb.edu/~ravenben/publications/pdf/vod-eurosys06.pdf.

“What Is the Computational Complexity of the MapReduce Overhead.” *Hadoop - What Is the Computational Complexity of the MapReduce Overhead - Stack Overflow*, stackoverflow.com/questions/3371110/what-is-the-computational-complexity-of-the-mapreduce-overhead.

“Hadoop 101: Simplifying MapReduce Development.” *InsideBIGDATA*, 29 Sept. 2015, insidebigdata.com/2015/05/13/hadoop-101-simplifying-mapreduce-development/.