BIG DATA ANALYTICS PROJECT REPORT

Apache http log analysis in Hadoop

BY:

Prafful Singh Jadon (I059)

Karan Merchant (I070)

Vedika Mittal (I072)

Saurabh Nair (I073)

Kunal Goyal (C011)

Astha Gupta (C012)

TABLE OF CONTENTS

1. PROJECT OVERVIEW

1.1. Brief Overview of the Project

Server logs are computer-generated log files that capture network and server operations data. They are useful for managing network operations, especially for security and regulatory compliance. Security breaches happen. And when they do, your server logs may be your best line of defense. Hadoop takes server-log analysis to the next level by speeding and improving security forensics and providing a low cost platform to show compliance.

There are various applications which have a huge amount of database in different format. All databases maintain log files that keep records of database changes in a system formation. This can include tracking various user events and activity. Apache Hadoop can be used for log processing at scale of a system. Log files have become a standard part of large applications and large scale industry, computer networks and distributed systems. Log files are often the only way to identify and locate an error in software as well as system, because log file analysis is not affected by any time-based issues known as probe effect or a system effect. This is opposite to analysis of a running program, when the analytical process can interfere with time critical or resource critical conditions within the analyzed program in a system. Log files are often very large and can have complex structure of database. Although the process of generating log files is quite simple and straightforward, log file analysis could be a tremendous task that requires enormous computational resources, long time and sophisticated procedures and time consuming. This often leads to a common situation, when log files are continuously generated and occupy valuable space on storage devices, but nobody uses them and utilizes enclosed information.

The overall goal of this project is to use HDFS, HIVE, PIG and SQOOP commands using hadoop map-reduce framework and analyze NASA's user log files. This generic log analyzer can analyze different kinds of log files such as- Email logs, Web logs, Firewall logs Server logs, or System produces and try to answer the pre-determined questions.

1.2. Learning Objective

- This program enables the participants to review the learnings of the Hadoop Technical Workshop.
- The primary objective of the project is to enhance the participant's skills in HDFS, Pig, Hive, Sqoop.
- Get a hands on experience of all the above mentioned platforms.
- To enhance problem solving and analytical skills

2. COMMANDS / CODE

2.1. HDFS Commands

```
su root

hdfs dfs -mkdir /myassign

hdfs dfs -mkdir /myassign/input

hdfs dfs -mkdir /myassign/output

hdfs dfs -put /home/cloudera/apache/apache_dataset.log /myassign/input

hdfs dfs -put /home/cloudera/apache/piggybank-apache.jar /myassign/input

hdfs dfs -cp /myassign/output/dataset.csv /myassign/output/apache_http.csv

hdfs dfs -rm -r /myassign/output/dataset.csv/_logs

hdfs dfs -rm -r /myassign/output/dataset.csv/_SUCCESS
```

2.2. PIG Commands

```
REGISTER /home/cloudera/apache/piggybank-apache.jar;

DEFINE ApacheCommonLogLoader org.apache.pig.piggybank.storage.apachelog.CommonLogLoader();

DEFINE LogLoader org.apache.pig.piggybank.storage.apachelog.CombinedLogLoader();

DEFINE DayExtractor org.apache.pig.piggybank.evaluation.util.apachelogparser.DateExtractor('yyyy-MM-dd');

LOGLINES = LOAD '/myassign/input/apache_dataset.log' USING ApacheCommonLogLoader AS (host, hclient, userid, logtime, method, pagerequest, protocol, serverstatus, sentbytes);

DUMP LOGLINES;

STORE LOGLINES INTO '/myassign/output/dataset.csv' USING PigStorage(',');
```

2.3. HIVE Commands

```
show databases;

create database myassign;

create table dataset(host STRING,

  identity STRING,

  user STRING,

  time STRING,

  request STRING,

  page STRING,

  protocol STRING,

  status STRING,

  size STRING)

ROW FORMAT DELIMITED

FIELDS terminated by ','

LINES terminated by '\n'

STORED AS TEXTFILE;

desc dataset;

load data inpath '/myassign/output/dataset.csv' OVERWRITE into table
dataset;
```

2.4. SQOOP Commands used to transfer files from HDFS to LINUX

```
sqoop export --connect jdbc:mysql://localhost/myassign --username
root -P --table dataset --export-dir /myassign/output/dataset.csv --
input-fields-terminated-by ',' --lines-terminated-by '\n'
```

3. RESULTS SECTION (AS HIVE TABLE)

3.1. MySQL command for each table and Result of each command

```
mysql -uroot

show databases;

create database myassign;

use myassign;

create table dataset(host
varchar(50),

  identity varchar(5),

  user varchar(20),

  time varchar(50),

  request varchar(10),

  page varchar(100),

  protocol varchar(50),

  status float,

  size float);
```

Q1. How many times each individual host has connected to our server? Store data sorted by highest count first.

```
select host, count(*) as cnt from dataset group by host order by cnt
DESC limit 10;
```

```
+------------------------------+-------+

| host                         | cnt   |

+------------------------------+-------+

| duke.usask.ca                | 38165 |

| sask.usask.ca                | 24477 |

| moondog.usask.ca             | 11344 |
| hist6629.usask.ca              |  8444 |
| skynet.usask.ca                |  7227 |
| broadway.sfn.saskatoon.sk.ca   |  6992 |
| cwis.usask.ca                  |  6937 |
| abyss.usask.ca                 |  4631 |
| herald.usask.ca                |  4480 |
| scooter.pa-x.dec.com           |  4238 |
+------------------------------+-------+
10 rows in set (1.13 sec)
```

Q2. How many times each individual page has been requested from our server? Store data sorted by highest count first.

```
select page, count(*) as pg from dataset group by page order by pg
DESC limit 10;
```

```
+--------------------------+--------+
| page                     | pg     |
+--------------------------+--------+
| /                        | 199998 |
| /images/logo.gif         | 141315 |
| /images/logo_32.gif      |  44744 |
| /cgi-bin/hytelnet        |  32557 |
| /images/letter_32.gif    |  23881 |
| /images/question_32.gif  |  23653 |
| /departments.html        |  16376 |
| /search/people_uofs.html |  15340 |
| /education/              |  14657 |
| /search.html             |  14592 |
+--------------------------+--------+
10 rows in set (1.27 sec)
```

Q3. How much data has been downloaded by each individual host that has connected to our server? Store data sorted by highest count first.

```
select host, SUM(size) as size from dataset group by host order by
size DESC limit 10;
```

```
+--------------------+-----------+
| host               | size      |
+--------------------+-----------+
| duke.usask.ca      | 198079339 |
| sask.usask.ca      |  90200609 |
| huey.usask.ca      |  63100478 |
| grapes.usask.ca    |  38568747 |
| moondog.usask.ca   |  35087536 |
| agora.carleton.ca  |  32643512 |
| skynet.usask.ca    |  31424480 |
| mac40199.usask.ca  |  31024528 |
| palona1.cns.hp.com |  30772140 |
| krause.usask.ca    |  30448911 |
+--------------------+-----------+
10 rows in set (1.42 sec)
```

Q4. How much data was sent out as each individual page was downloaded from our server? Store data sorted by highest count first.

```
select page, SUM(size) as size_page from dataset group by page order
by size page DESC limit 10;
```

```
+---------------------------------+-----------+
| page                            | size_page |
+---------------------------------+-----------+
| /                               | 552584493 |
| /education/edbldg.gif           | 327725744 |
| /uofs/ivany_movie.mov           | 234787776 |
| /cgi-bin/hytelnet?file=US000OTH | 215339380 |
| /Mosaic/Docs/whats-new.html     | 214741142 |
| /images/logo.gif                | 206661160 |
| /education/                     | 113365162 |
| /cusi/cusi.html                 |  97720436 |
| /departments.html               |  85954250 |
| /logs/access_log1               |  80563437 |
+---------------------------------+-----------+
10 rows in set (1.29 sec)
```

4. Results Section (As HDFS FILE)

4.1. Answers to all questions

Q1. Which host has connected the maximum number of times to our server? Give the host name & count of connections from that host.

```
select host, count(host) from dataset group by host having
count(host)= (select max(mycount) from (select host,
count(host)mycount from dataset group by host) as abc) into outfile
'/home/cloudera/apache/hdfs1.csv';


create table hdfs1(host varchar(50), host_count float);


load data local infile '/home/cloudera/apache/hdfs1.csv' into table
hdfs1 fields terminated by '\t' lines terminated by '\n';
```

ANSWER: duke.usask.ca,38165.0

Q2. Which page that has been requested the maximum number of times from our server? Give the page name & count of the times the page was requested.

```
select page, count(page) from dataset group by page having
count(page)= (select max(mycount) from (select page,
count(page)mycount from dataset group by page) as def) into outfile
'/home/cloudera/apache/hdfs2.csv';


create table hdfs2(page varchar(100), page_count float);

load data local infile '/home/cloudera/apache/hdfs2.csv' into table
hdfs2 fields terminated by '\t' lines terminated by '\n';
```

ANSWER: /,199998.0

Q3. How many unique hosts have connected to our server? Give counts.

```
select  count(DISTINCT host)  from dataset into outfile
'/home/cloudera/apache/hdfs3.csv';


create table hdfs3(Unique_hosts float);
```

```
load data local infile '/home/cloudera/apache/hdfs3.csv' into table
hdfs3 fields terminated by '\t' lines terminated by '\n';



sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs3 --m 1 --target-dir /myassign/output/sqoop/hdfs3
```

ANSWER: 78390.0

Q4. How many unique pages have been requested from our server? Give counts.

```
select  count(DISTINCT page)  from dataset into outfile
'/home/cloudera/apache/hdfs4.csv';

create table hdfs4(Unique_pages float);

load data local infile '/home/cloudera/apache/hdfs4.csv' into table
hdfs4 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs4 --m 1 --target-dir
/myassign/output/sqoop/hdfs4
```

ANSWER: 28971.0

Q5. Which host has caused maximum data transfer from our server? Give host name & the data transfer for the host.

```
select host, size from dataset where size = (select max(size) from
dataset where status = 200) group by host into outfile
'/home/cloudera/apache/hdfs5.csv';

create table hdfs5(host varchar(50), size float);

load data local infile '/home/cloudera/apache/hdfs5.csv' into table
hdfs5 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs5 --m 1 --target-dir
myassign/output/sqoop/hdfs5
```

ANSWER: agora.carleton.ca,3.01938E7

duke.usask.ca,3.01938E7

grapes.usask.ca,3.01938E7

krause.usask.ca,3.01938E7

mac40199.usask.ca,3.01938E7

palona1.cns.hp.com,3.01938E7

Q6. Which page has caused maximum data transfer from our server? Give page name & the data transfer for the page.

```
select page, size from dataset where size = (select max(size) from
dataset where status = 200) group by page into outfile
'/home/cloudera/apache/hdfs6.csv';

create table hdfs6(page varchar(50), size float);

load data local infile '/home/cloudera/apache/hdfs6.csv' into table
hdfs6 fields terminated by '\t' lines terminated by '\n';

  sqoop import --connect jdbc:mysql://localhost/myassign --username
              root -P --table hdfs6 --m 1 --target-dir
                    /myassign/output/sqoop/hdfs6
```

ANSWER of: /uofs/ivany_movie.mov,3.01938E7

Q7. Which page has maximum download size from our server? Give page name & the size for the page.

```
select page, size from dataset where size = (select max(size) from
dataset) group by page into outfile
'/home/cloudera/apache/hdfs7.csv';

create table hdfs7(page varchar(100), page_maxSize double);

load data local infile '/home/cloudera/apache/hdfs7.csv' into table
hdfs7 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs7 --m 1 --target-dir /myassign/output/sqoop/hdfs7
```

ANSWER: `/uofs/ivany_movie.mov,3.01938E7`

Q8. What is the download count of the page that has maximum download size from our server? Give page name & download count

```
select page, count(size) from dataset where size = (select max(size)
from dataset) group by page into outfile
'/home/cloudera/apache/hdfs8.csv';

create table hdfs8(page varchar(100), page_MaxSize_Count int);

load data local infile '/home/cloudera/apache/hdfs8.csv' into table
hdfs8 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs8 --m 1 --target-dir /myassign/output/sqoop/hdfs8
```

ANSWER: `/uofs/ivany_movie.mov,6`

Q9. Which page has minimum download size from our server? Give page name & the size for the page.

```
select page, size from dataset where size = (select min(size) from
dataset) group by page limit 10 into outfile
'/home/cloudera/apache/hdfs9.csv';

create table hdfs9(page varchar(100), size float);

load data local infile '/home/cloudera/apache/hdfs9.csv' into table
hdfs9 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username
root -P --table hdfs9 --m 1 --target-dir /myassign/output/sqoop/hdfs9
```

ANSWER:

```
"GET,-0.0
/ ,-0.0
/,0.0
/!philmic,-0.0
/!scottp/free.html,-0.0
/!scottp/iii.html,-0.0
/!turner/duesouth.html,-0.0
/",-0.0
/"averyc,-0.0
/"averyc/menu.html,-0.0
```

Q10. What is the download count of the page that minimum download size from our server? Give page name & the size for the page.

```
select page, size, count(size) from dataset where size = (select min(size) from dataset where status=200) into outfile '/home/cloudera/apache/hdfs10.csv';

create table hdfs10(page varchar(100), size float, page_MinSize_Count int);

load data local infile '/home/cloudera/apache/hdfs10.csv' into table hdfs10 fields terminated by '\t' lines terminated by '\n';

sqoop import --connect jdbc:mysql://localhost/myassign --username root -P --table hdfs10 --m 1 --target-dir /myassign/output/sqoop/hdfs10
```

ANSWER: /,0.0,215647

4.2. Additional Analytics

1. UNESTABLISHED CONNECTIONS DUE TO SERVER ERROR

```
select count(status) from dataset where status like '5%';
```

```
+---------------+
| count(status) |
+---------------+
|           560 |
+---------------+
1 row in set (1.07 sec)
```

2. UNESTABLISHED CONNECTIONS DUE TO CLIENT ERROR

```
select count(status) from dataset where status like '4%';
```

```
+---------------+
| count(status) |
+---------------+
|         21580 |
+---------------+
1 row in set (1.10 sec)
```

3. HOSTS HAVING CONNECTION ERROR FROM CLIENT SIDE & NUMBER OF TIMES CONNECTION IS NOT ESTABLISHED

```
select host, count(status) from dataset where status like '4%' group by
host order by count(status) DESC limit 10;
```

```
+----------------------+---------------+
| host                 | count(status) |
+----------------------+---------------+
| cwis.usask.ca        |          1716 |
| jcooke.usask.ca      |           601 |
| 128.95.226.85        |           398 |
| skynet.usask.ca      |           377 |
| duke.usask.ca        |           283 |
| scooter.pa-x.dec.com |           266 |
| crown.uunet.ca       |           183 |
| wheat.usask.ca       |           161 |
| sask.usask.ca        |           129 |
| mac40212.usask.ca    |           126 |
+----------------------+---------------+
10 rows in set (1.05 sec)
```

4.PAGES HAVING CONNECTION ERROR FROM SERVER SIDE & NUMBER OF TIMES CONNECTION IS NOT ESTABLISHED

```
select page, count(status) from dataset where status like '5%'group by
host order bv count(status) DESC limit 10;
```

```
+---------------------------------------------------------------------
----------------------------------+---------------+
|
page
                                  | count(status) |
+---------------------------------------------------------------------
----------------------------------+---------------+
|
/dcs/pts/record.html
                                  |            39 |
|                                                                 /cgi-
bin/lycos.pl?query=fogel&maxhits=15&minterms=1&minscore=0.01&terse=on
                             |            31 |
|                                                                 /cgi-
bin/phonebook.pl?fogel
                             |            14 |
|
/httpd_docs/info/star_trek/
                                 |             9 |
bin/phone.pl
                           |             9 |
|                                                     /registrar/cgi-
bin/schedule
                 |             8 |
|                                                               /cgi-
bin/digger?Value=peng&mode=nice&Server=University+of+Saskatchewan%09%
5Bduke.usask.ca+63%5D      |             8 |
|                                                               /cgi-
bin/phone.pl
                           |             8 |
|                                                               /cgi-
bin/cusi?query=tennyson&service=http%3A%2F%2Fwebcrawler.com%2Fcgi-
bin%2FWebQuery%3F_cusi-search |             7 |
|                                                               /cgi-
bin/cusi?service=http%3A%2F%2Fwww.pathfinder.com%2F@@ITYnPgAAAAAQBn0
%2Fcgi-bin%2Fsearch%3FSear |             7 |
+--------------------------------------------------------------------
----------------------------------+---------------+
10 rows in set (1.08 sec)
```

5.SUCCESSFULLY ESTABLISHED CONNECTIONS

```
select count(status) from dataset where status like '2%';
```

```
+---------------+
| count(status) |
+---------------+
|       1032464 |
+---------------+
1 row in set (1.65 sec)
```

6. REDIRECTIONS

```
select count(status) from dataset where status like '3%';
```

```
+---------------+
| count(status) |
+---------------+
|        191244 |
+---------------+
1 row in set (1.07 sec)
```

7. UNSUCCESSFUL CONNECTIONS

```
select count(status) from dataset where status like '4%' && '5%';
select count(status) from dataset where status like '4%' && '5%';
```

```
+---------------+
| count(status) |
+---------------+
|         21580 |
+---------------+
1 row in set, 1 warning (1.08 sec)
```

8. HOST, PAGE WHICH HAS THE MINIMUM DOWNLOAD SIZE OTHER THAN SERVER NOT RETURNING THE CONTENT TO THE CLIENT

```
select host, page, size from dataset where size = 1;
```

```
+--------------------------+----------------+------+
| host                     | page           | size |
+--------------------------+----------------+------+
| moondog.usask.ca         | /digger/       |    1 |
| welchlink.welch.jhu.edu  | /medicine/ces/ |    1 |
+--------------------------+----------------+------+
2 rows in set (0.56 sec)
```

4.3. Translating statistics to analytics

```
select host, count(*) as cnt from dataset group by host order by cnt
```

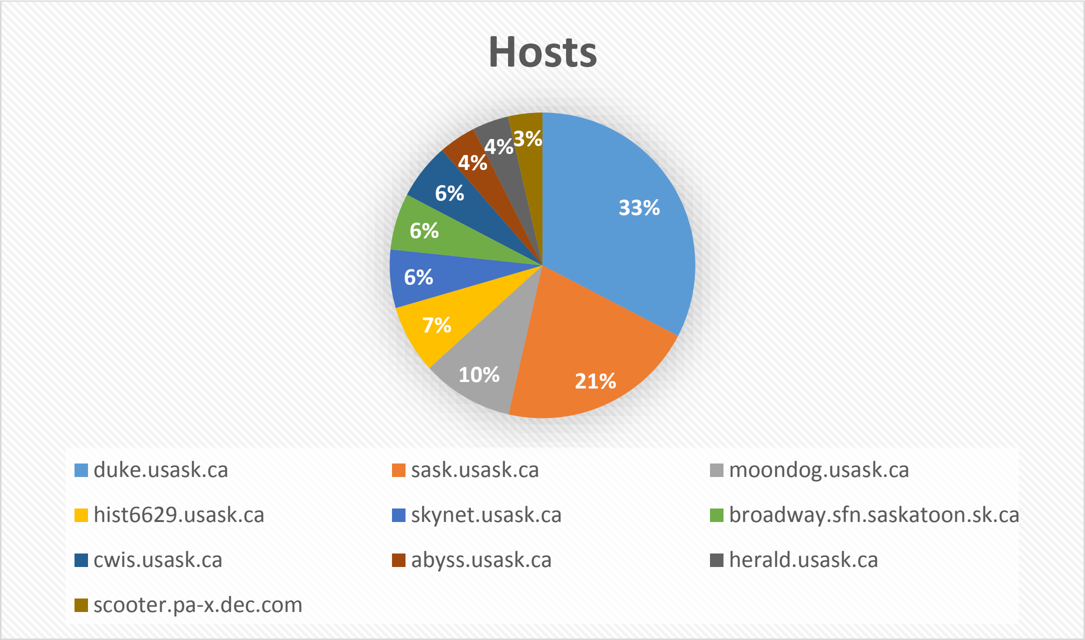

Fig.1 depicts the top 10 maximum number of times the hosts have accessed APACHE SERVER

The above pie-chart indicates the number of times each different host accesses the website.

Also from the chart is becomes clear that maximum number of times host from a particular geographical location to access the website is from Canada.

Host name Duke.usask.ca makes a maximum hit of 33% being the highest from among all the other host accessing the website. And second being sask.usask.ca with 21% times access to the website. Whereas the host scooter.pa-x.dec.com accesses the least times with just 3% being the lowest host to access the website.

```
select page, count(*) as pg from dataset group by page order by pg DESC
limit 10
```

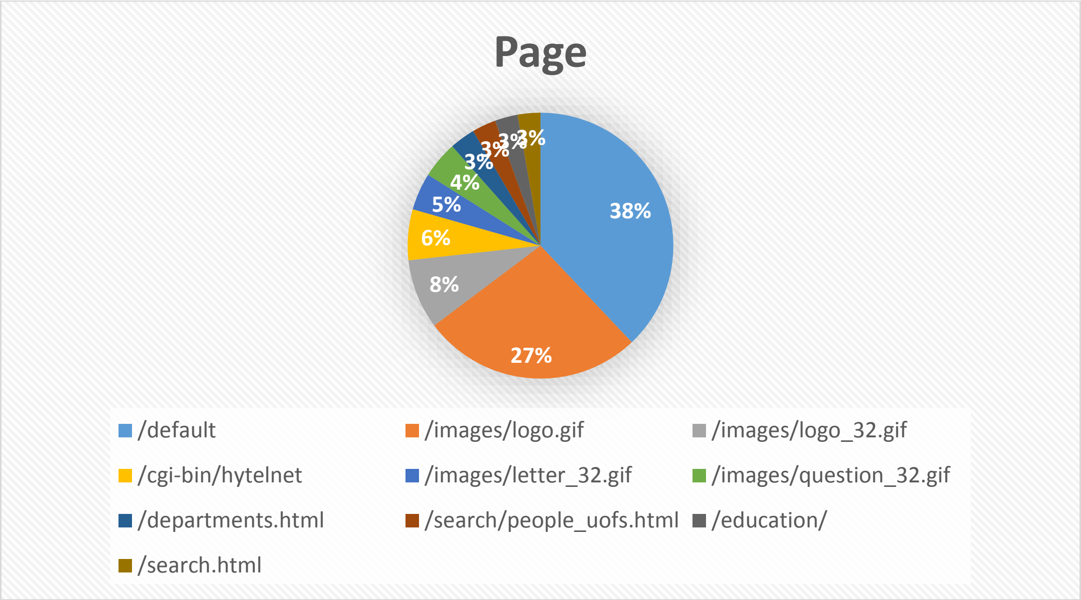

Fig.2 depicts the top 10 maximum number of times the page has been accessed by the hosts.

The above chart describes relation of the percentages of number of times a particular page of the website is visited. And as seen /default page being the initial page has he maximum number of hits as compared to any other page on the website with 38%. After /default which is a webpage the second highest percentage of a hits is on a media file on server rather than a webpage; the file being the Logo.

Minimum Number of times a page visited is the search page on the website with just 3% of hit ratio.

```
select host, SUM(size) as size from dataset group by host order by
```

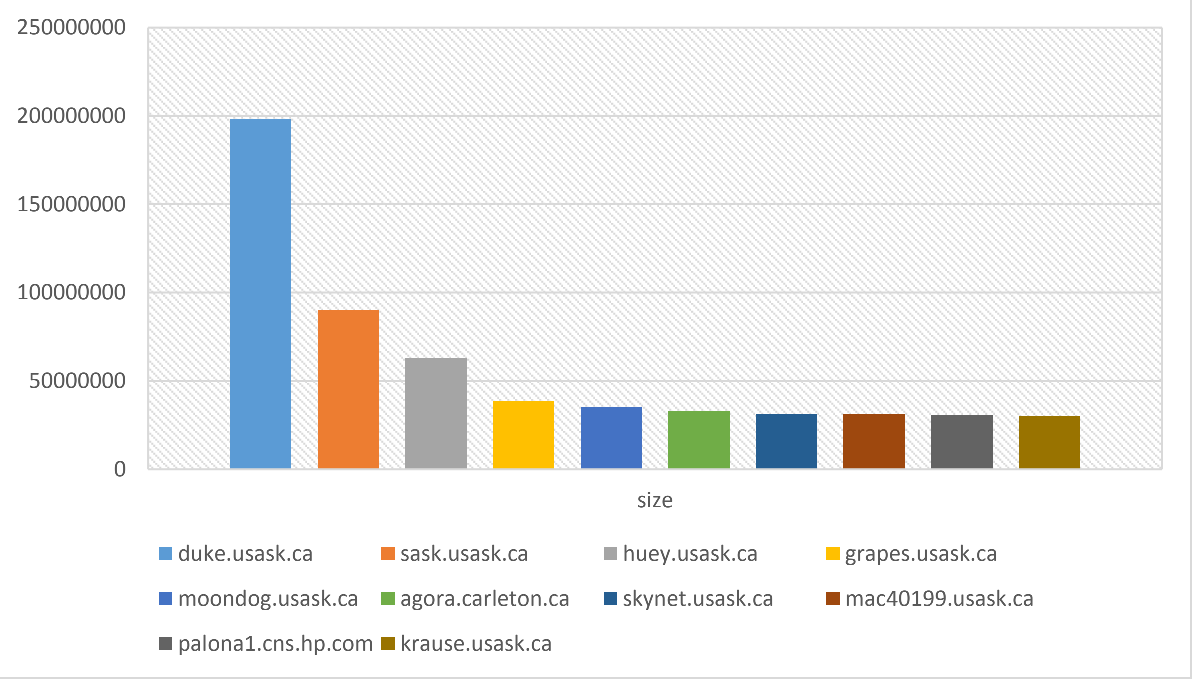

Fig.3 depicts the top 10 hosts that have downloaded the maximum data.

The bar chart above indicate the total number of data transfer by each host accessing the website.

This transfer of data can be incoming to and/or outgoing data from the website with respect to the individual host. Duke.usask.ca has the maximum number of transfer of the data to and from the website whereas as seen karause.usask.ca has the minimum number of data transfer among all the servers.

```
select page, SUM(size) as size_page from dataset group by page order
```

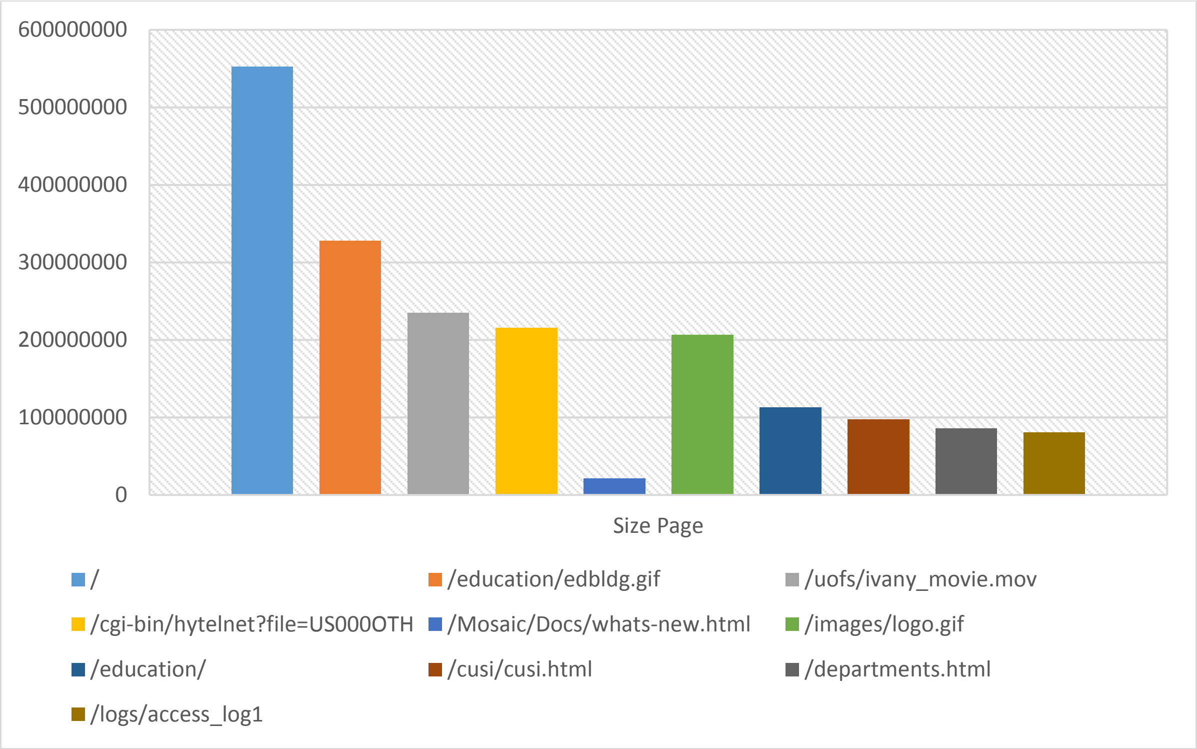

Fig.4 depicts the top 10 pages that has maximum download size

Looking at the above chart which graphically indicates total data transfer by each page and the lowest seen is the whats-new.html page and highest data transfer is by the /default page.

The above statistics to show that hosts which are connected to the APACHE server is mainly from Canada (.ca domain). People from Canada view their pages very often and download a lot. There is a page containing a video (.mov format) from which 6 of the distinct hosts have accessed and downloaded the movie. This movie has the maximum download size out of all the pages on the APACHE server.

There are a lot of unsuccessful connections due to the error from the client as well as server side. They have also been stated in the above statistics (Successful - 1032464; Unsuccessful - 21580; Redirections - 191244). Number of unsuccessful connections due to client side issues are 21580 while due to server side issues are 560. Since there are few server problems, there must be a patch resolved for the same. There is an additional analytic showing the page which has the maximum unsuccessful connection with the host due to server issues. They must resolve the pages (like /dcs/pts/record.html) as soon as possible.

5. SUMMARY

With the growing speed of data in the web, a framework is needed to process and store the data. The bigger in size the log files are, the more useful the information they can furnish. However, processing huge sized log data can be very challenging and it is definitively not an easy task. Therefore, the log files should be analyzed in Big Data environment using the Big Data application such as Hadoop. There were researches done to analyze various types of log file using Big Data solution. The outcome is very significant.

From the results it is concluded that processing a huge file in distributed fashion reduces the time and data transfer cost, without moving the data. The text files can be screwed in order to produce a statistical report for better understanding of users view. Also performing the same task for multiple files of large volume of data reduces the memory utilization, CPU load and other factors that results the process in easy head. The processing of log files went through ordinary sequential ways in order to perform preprocessing, session identification and user identification etc. The developed non Hadoop approach loads the log file dataset, to process each line one after another.

During the initial phase of parsing the Apache log file using PIG was a hurdle to cross. During this phase there were various resource limitations. As time passed and we got acquainted with our machines, things got smooth. We parsed the Apache log file successfully and stored the log file as Comma Separated Value (.csv file) on hadoop cluster. We created its replicate on our linux system so as to prevent data loss.

We created a prototype using the workset data log at first then proceeded with analyzing the dataset as a whole. As we were not great coders, we initially struggled to get the sql commands right and then we cracked them all. It was a tough as well as brain-teasing time to get the appropriate stats.

We analyzed them using our mental abilities, pie charts & histograms as seen in the report. We have also come up with certain additional analytics which we thought maybe useful which we have explained in the above section.

From the results it is observed that the framework developed in Hadoop has high performance when compared to the other approaches. Future work can be extended to number of nodes for prediction analysis to find the next page that will be accessed by the user in a particular website. The results of the related research works presented in this project report showed that by using Hadoop Framework the response and operation time of the analysis process have been improved.

This practical assignment has given us an opportunity to rack our brains and come out with a solution. This hands-on experience has given us such a joy when we accomplished this task. Thank you sir for all that you have taught us.