IT-478 MILESTONE 4

Movie Ratings Analysis

To

Dr. Bryan Hosack

Madhuri Choudhary
Mayank Shrivansh
Mayank Agrawal
Yesh Polishetty

Table of Contents

Goal:

To Implement Big Data Architecture in an organization which makes educational movies, to create increase revenue by analyzing the relevant data sets and to generate meaningful data from different type of data sets that will help client to make better decision.

Business Problem:

The organization is not able to generate enough revenue by making educational movies and they want to increase their financial status without changing their main goal. They want to

What we have done:

Implementation:

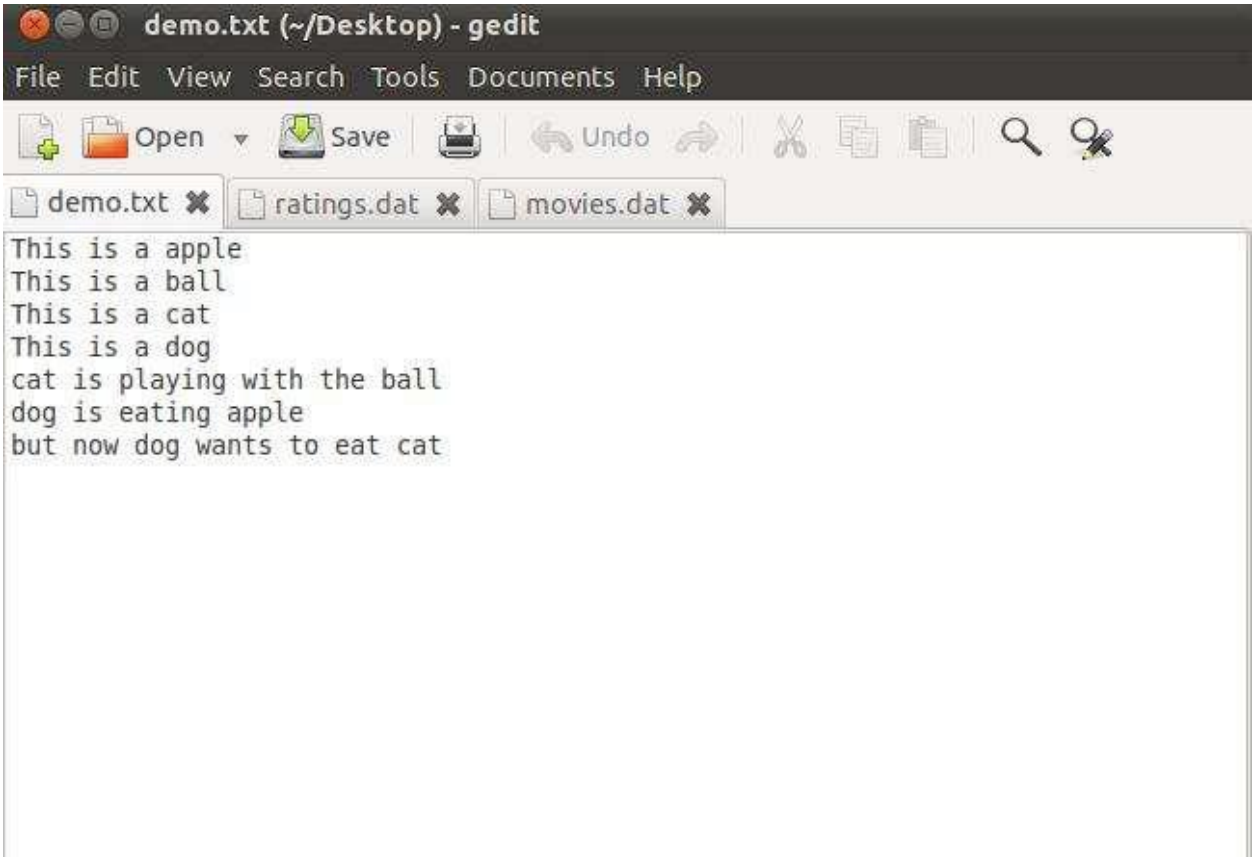
Step 1: Installation of Cloudera CDH3 virtual platform with VM Player.

Step 2: Then we have done demo for our understanding of the system.

Here is the files that we inserted in to the HDFS:

```
cloudera@cloudera-vm:/usr/lib/hadoop$ hadoop dfs -ls
Found 4 items
-rw-r--r--   1 cloudera cloudera       171308 2014-10-27 13:17 /user/cloudera/adb
-rw-r--r--   1 cloudera cloudera       134368 2014-10-27 09:17 /user/cloudera/adb1
-rw-r--r--   1 cloudera cloudera     24594131 2014-10-27 09:18 /user/cloudera/adb2
drwxr-xr-x   - cloudera supergroup          0 2014-10-27 13:38 /user/cloudera/wordcount
```

Sample Text File demo.txt contains the set of statements as below:



Then we load this file into HDFS:

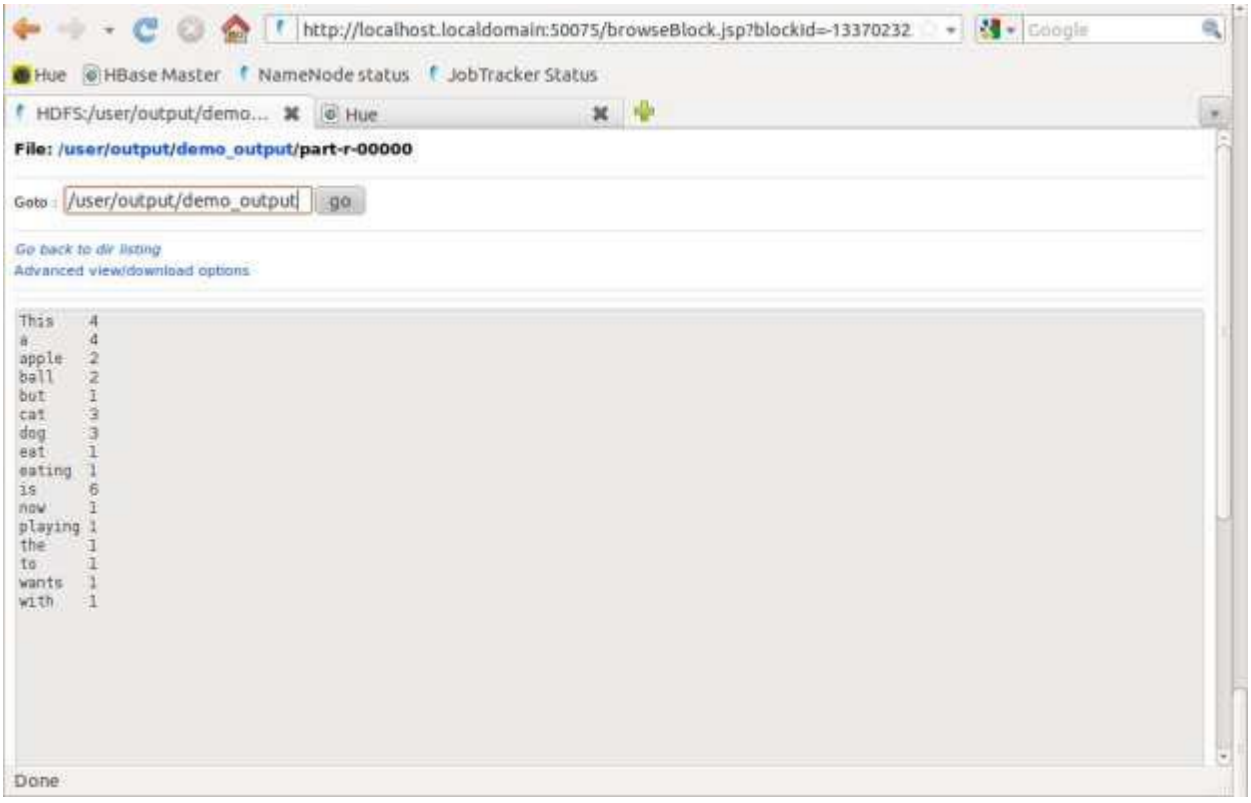
```
cloudera@cloudera-vm:~$ hadoop dfs -copyFromLocal /home/cloudera/Desktop/demo.tx
t /usr/training/demo.txt
```

After that we apply Map task on the file to get the word count of a particular word. The source of Map program is
http://www.cloudera.com/content/cloudera/en/documentation/HadoopTutorial/CDH4/Hadoop-Tutorial/ht_wordcount1_source.html

The detailed execution of Map task is shown below:

```
cloudera@cloudera-vm:/usr/lib/hadoop$ hadoop jar hadoop-examples.jar wordcount /
usr/training/demo.txt /user/output/demo_output
14/10/27 15:02:24 INFO input.FileInputFormat: Total input paths to process : 1
14/10/27 15:02:25 INFO mapred.JobClient: Running job: job_201410210013_0001
14/10/27 15:02:26 INFO mapred.JobClient:  map 0% reduce 0%
14/10/27 15:02:36 INFO mapred.JobClient:  map 100% reduce 0%
14/10/27 15:02:45 INFO mapred.JobClient:  map 100% reduce 100%
14/10/27 15:02:46 INFO mapred.JobClient: Job complete: job_201410210013_0001
14/10/27 15:02:46 INFO mapred.JobClient: Counters: 22
14/10/27 15:02:46 INFO mapred.JobClient:   Job Counters
14/10/27 15:02:46 INFO mapred.JobClient:     Launched reduce tasks=1
14/10/27 15:02:46 INFO mapred.JobClient:     SLOTS_MILLIS_MAPS=7287
14/10/27 15:02:46 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
14/10/27 15:02:46 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
14/10/27 15:02:46 INFO mapred.JobClient:     Launched map tasks=1
14/10/27 15:02:46 INFO mapred.JobClient:     Data-local map tasks=1
14/10/27 15:02:46 INFO mapred.JobClient:     SLOTS_MILLIS_REDUCES=9570
14/10/27 15:02:46 INFO mapred.JobClient:   FileSystemCounters
14/10/27 15:02:46 INFO mapred.JobClient:     FILE_BYTES_READ=176
14/10/27 15:62:46 INFO mapred.JobClient:     HDFS_BYTES_READ=240
14/10/27 15:02:46 INFO mapred.JobClient:     FILE_BYTES_WRITTEN=106676
14/10/27 15:02:46 INFO mapred.JobClient:     HDFS_BYTES_WRITTEN=106
14/10/27 15:02:46 INFO mapred.JobClient:   Map-Reduce Framework
14/10/27 15:02:46 INFO mapred.JobClient:     Reduce input groups=16
14/10/27 15:02:46 INFO mapred.JobClient:     Combine output records=16
14/10/27 15:02:46 INFO mapred.JobClient:     Map input records=7
14/10/27 15:02:46 INFO mapred.JobClient:     Reduce shuffle bytes=176
14/10/27 15:02:46 INFO mapred.JobClient:     Reduce output records=16
14/10/27 15:02:46 INFO mapred.JobClient:     Spilled Records=32
14/10/27 15:02:46 INFO mapred.JobClient:     Map output bytes=269
14/10/27 15:02:46 INFO mapred.JobClient:     Combine input records=33
14/10/27 15:02:46 INFO mapred.JobClient:     Map output records=33
14/10/27 15:02:46 INFO mapred.JobClient:     SPLIT_RAW_BYTES=103
14/10/27 15:02:46 INFO mapred.JobClient:     Reduce input records=16
```

The result of the Map task is as follows:

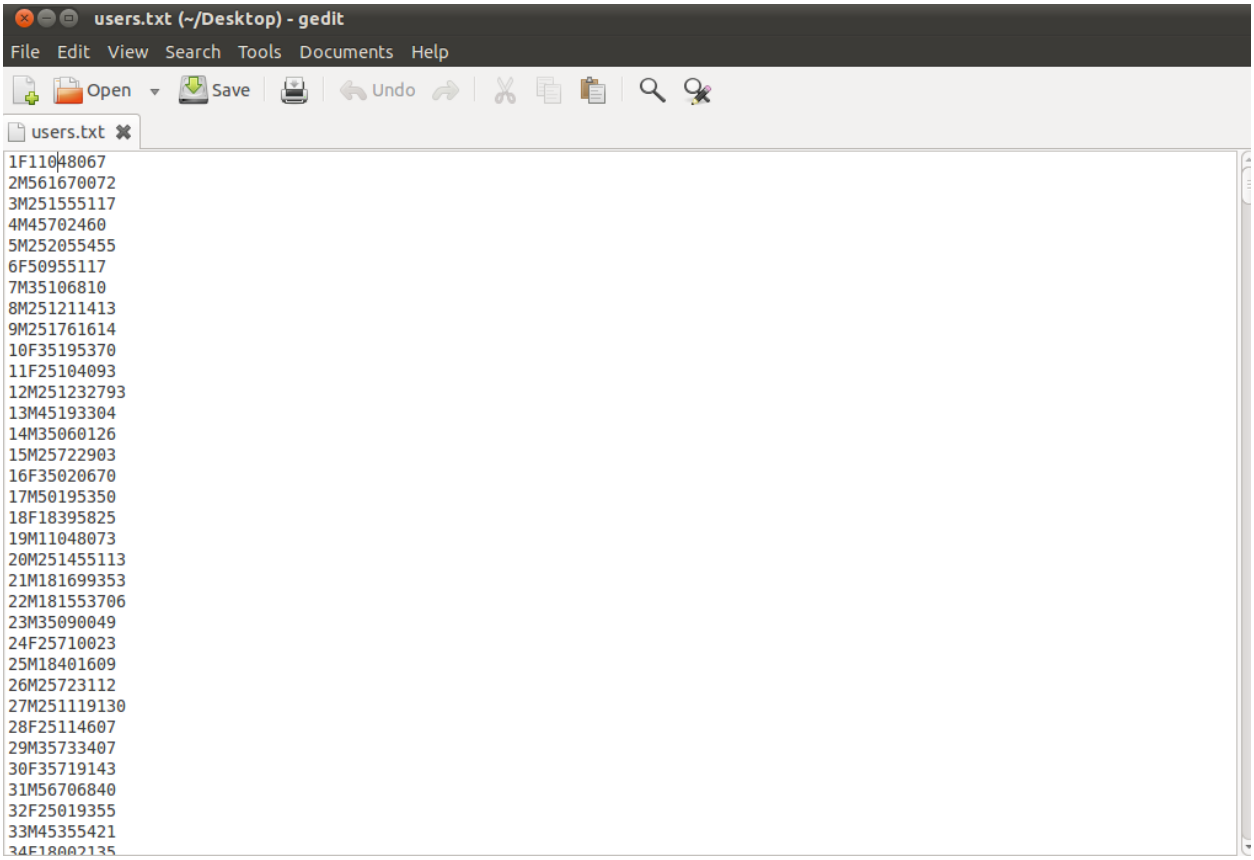


Step 3: Now we started using the system according to our requirements.

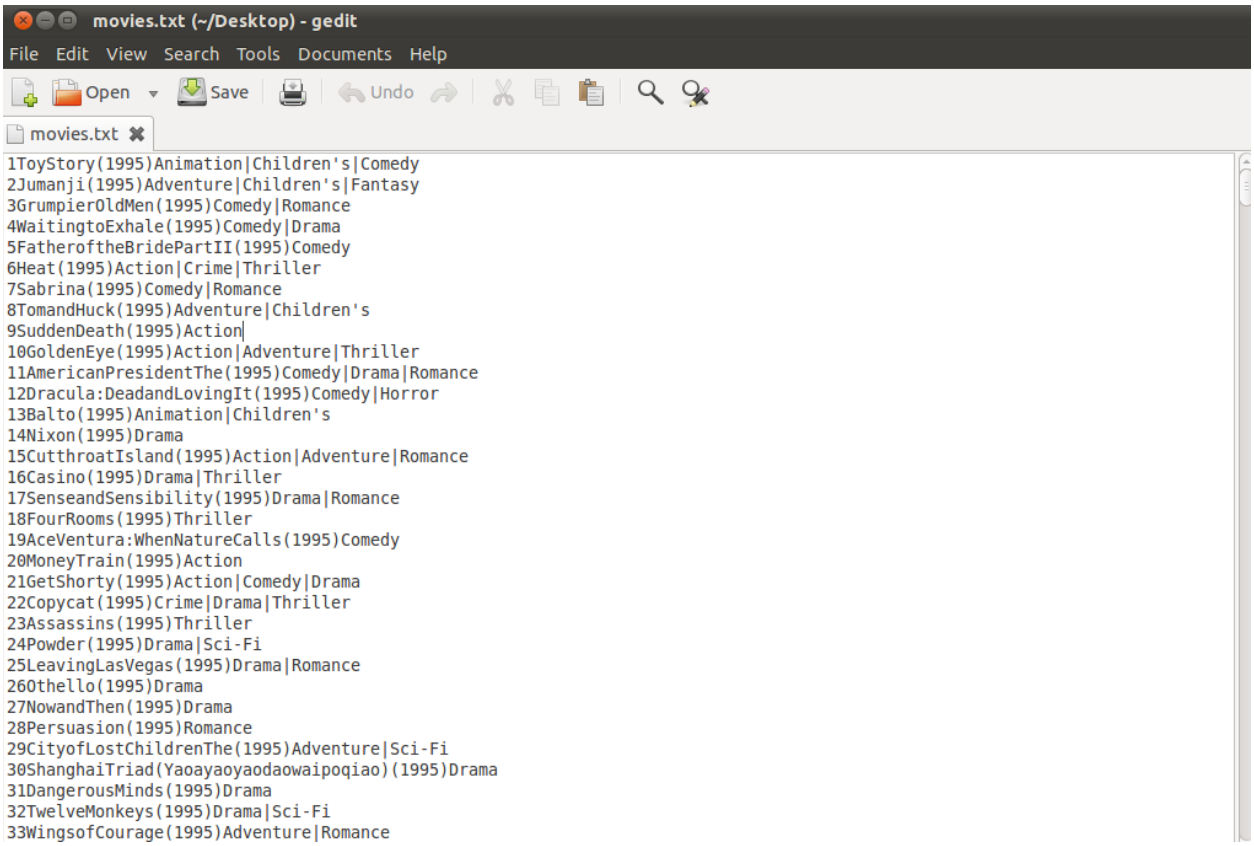
At first, we transfer our datasets from our local machine to Cloudera's VM through FileZilla FTP client.

And then we stored this unstructured data files in to the HDFS.

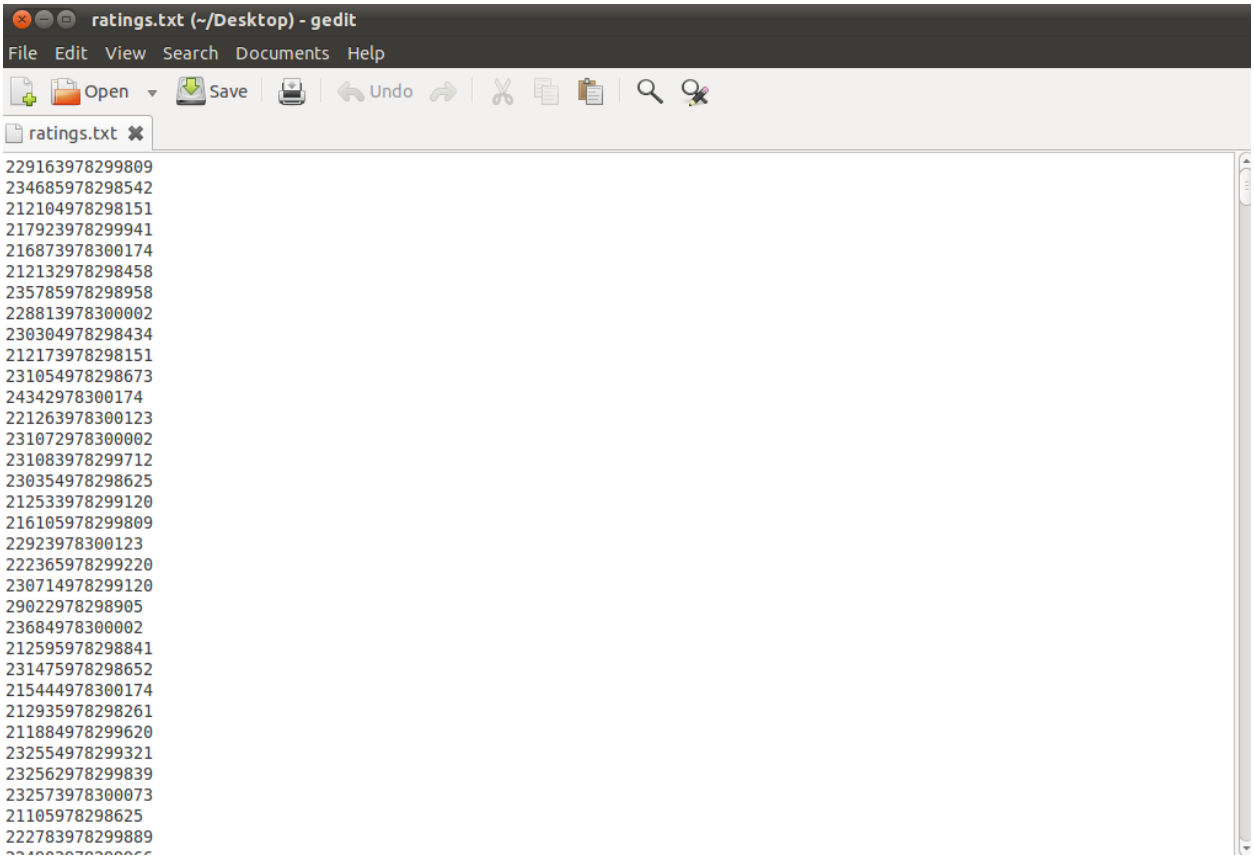
1. User.txt



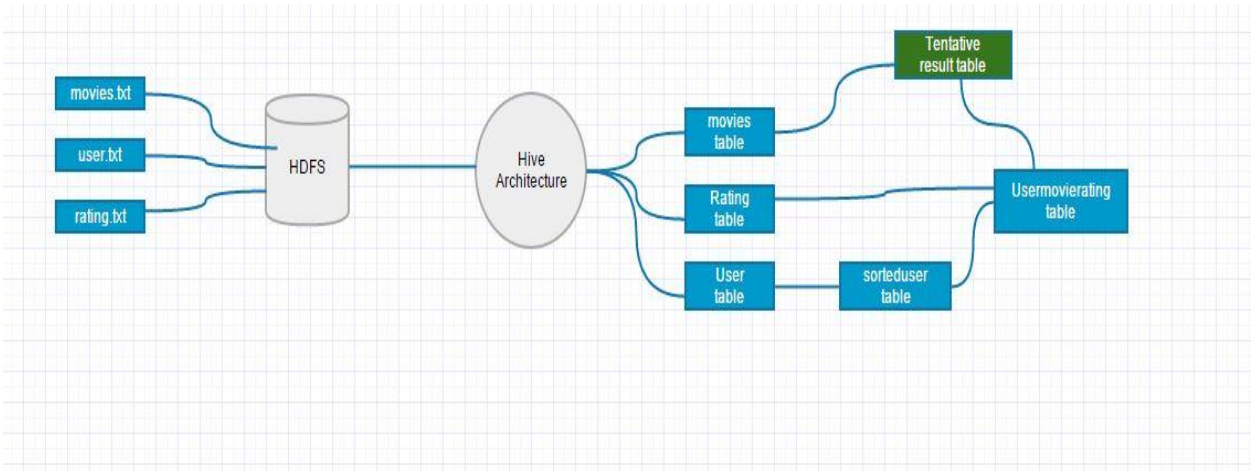
2. Movies.txt


movies.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
movies.txt
1ToyStory(1995)Animation|Children's|Comedy
2Jumanji(1995)Adventure|Children's|Fantasy
3GrumpierOldMen(1995)Comedy|Romance
4WaitingtoExhale(1995)Comedy|Drama
5FatheroftheBridePartII(1995)Comedy
6Heat(1995)Action|Crime|Thriller
7Sabrina(1995)Comedy|Romance
8TomandHuck(1995)Adventure|Children's
9SuddenDeath(1995)Action
10GoldenEye(1995)Action|Adventure|Thriller
11AmericanPresidentThe(1995)Comedy|Drama|Romance
12Dracula:DeadandLovingIt(1995)Comedy|Horror
13Balto(1995)Animation|Children's
14Nixon(1995)Drama
15CutthroatIsland(1995)Action|Adventure|Romance
16Casino(1995)Drama|Thriller
17SenseandSensibility(1995)Drama|Romance
18FourRooms(1995)Thriller
19AceVentura:WhenNatureCalls(1995)Comedy
20MoneyTrain(1995)Action
21GetShorty(1995)Action|Comedy|Drama
22Copycat(1995)Crime|Drama|Thriller
23Assassins(1995)Thriller
24Powder(1995)Drama|Sci-Fi
25LeavingLasVegas(1995)Drama|Romance
26Othello(1995)Drama
27NowandThen(1995)Drama
28Persuasion(1995)Romance
29CityofLostChildrenThe(1995)Adventure|Sci-Fi
30ShanghaiTriad(Yaoayaoyaodaowaipoqiao)(1995)Drama
31DangerousMinds(1995)Drama
32TwelveMonkeys(1995)Drama|Sci-Fi
33WingsofCourage(1995)Adventure|Romance

3. Rating.txt

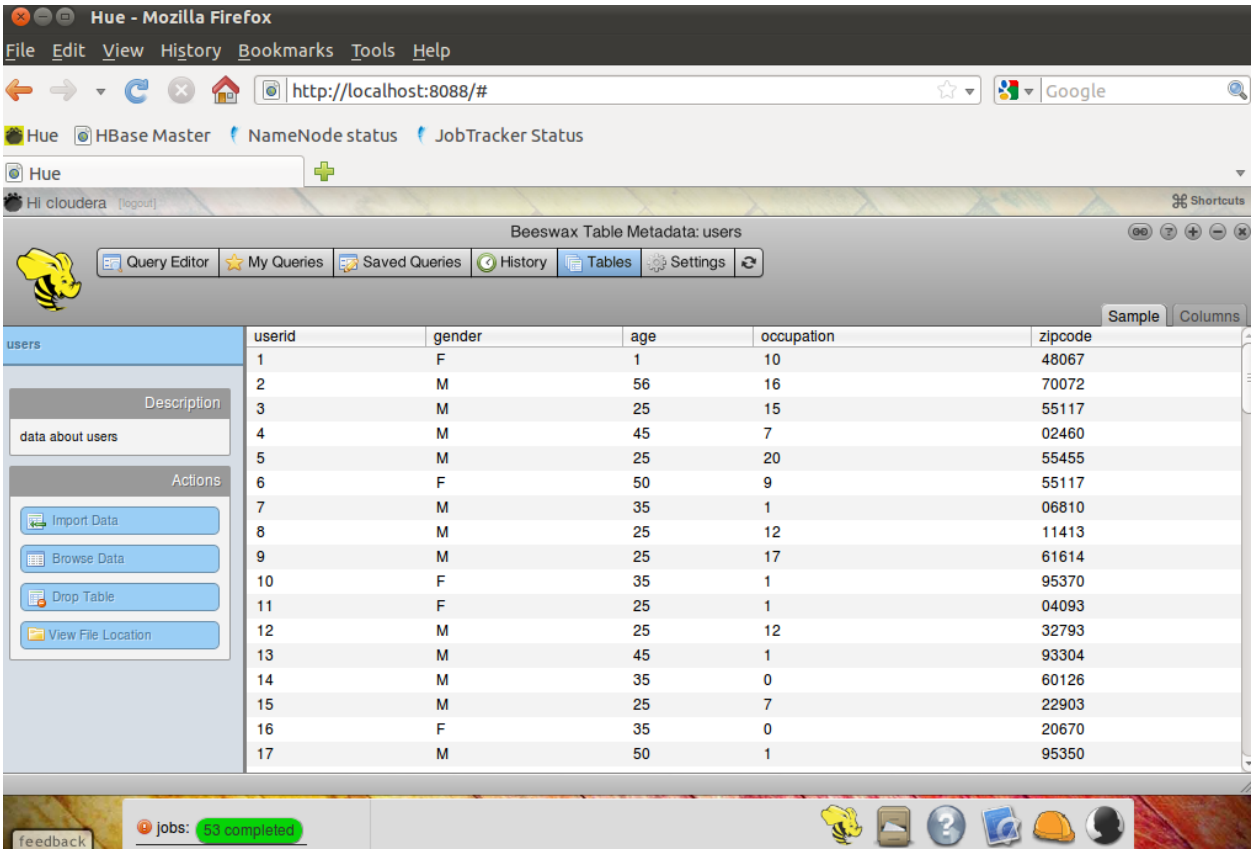


Here is the Data flow of the Overall Project:

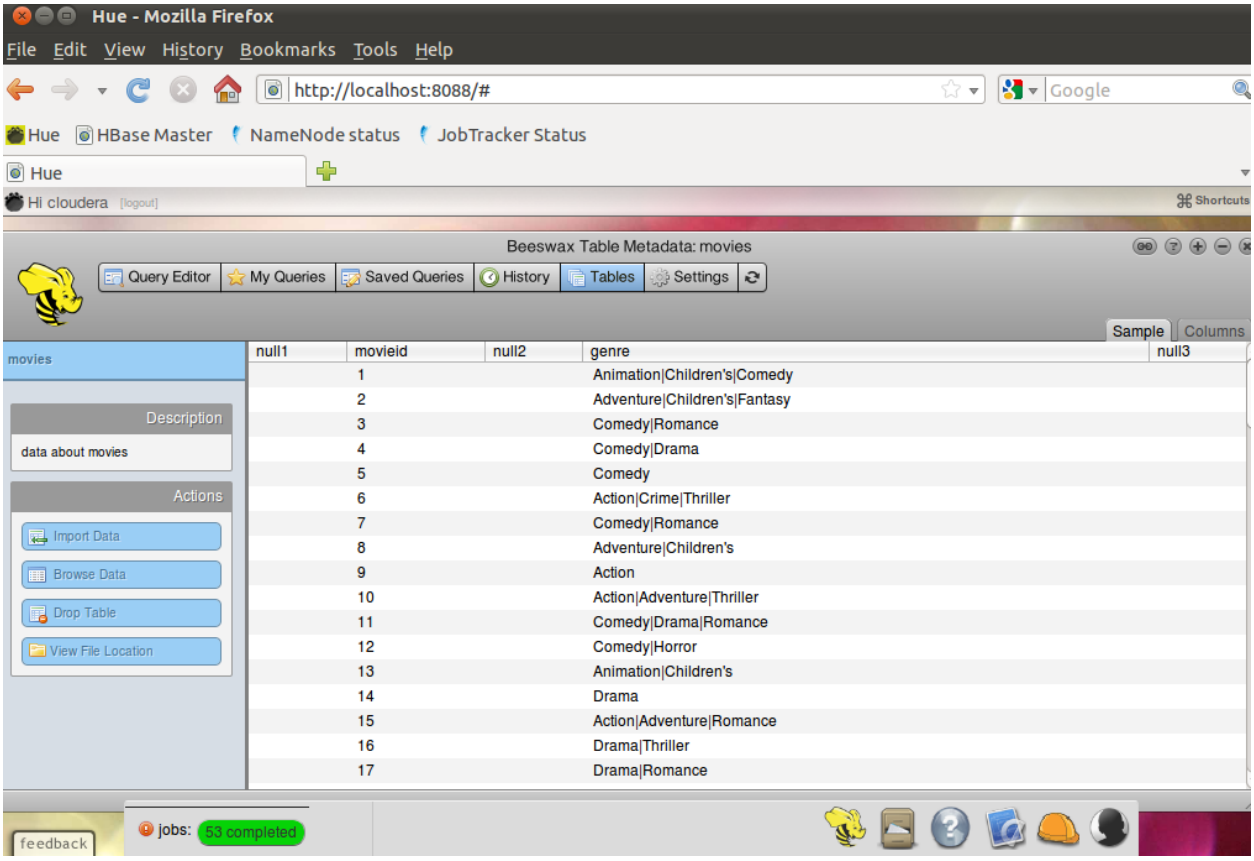


According to the data flow we proceed and convert the unstructured datasets in to a structured format like tables.

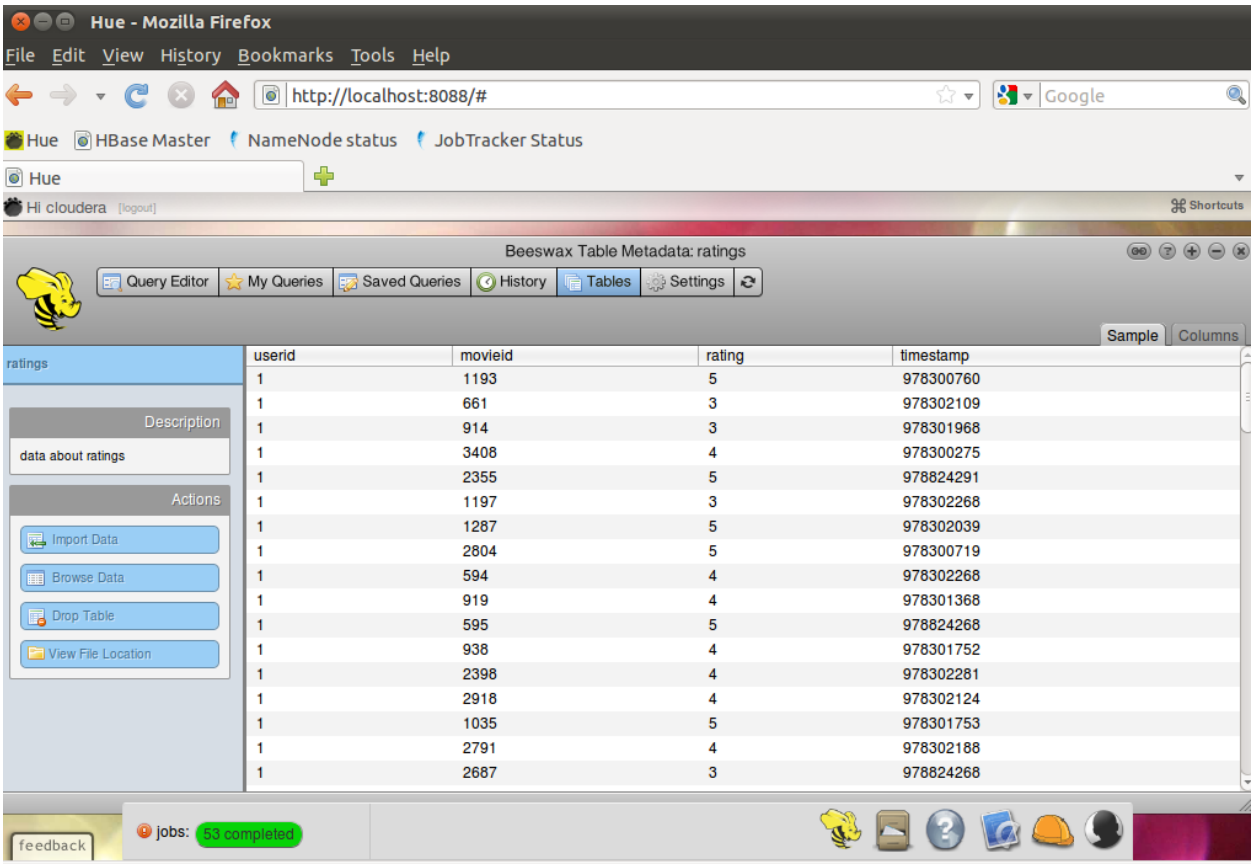
1. User Table.



2. Movies Table.



3. Ratings Table.



After that we apply some queries to the tables and came with the certain sets of Tables, by using certain delimiters and Serializable and Deserializable interfaces.

Some queries that we used are:-

- To sort the user with age group 18 (18 to 24yrs) and generate **sorteduser** tables.

  Select * from **usertable** where age =’18’;

- To create a new table called **usermovierating** from **sorteduser** and **rating** table**.**

  Select userid, gender, age, movieid, rating from **sortedusers** join **rating** on (users.userid=movies,userid) where ratings.rating=’5’;

- To generate the **tentativeresult** table from **movies** and **usermovierating** table.

  Select movieid, userid, gender, age, rating, genre from **movies** join **usermovierating** on (movies.movieid=usermovierating.movieid) where movies.movieid=usermovierating.movieid;

- To calculate the number of user under the age group 18 (18 to 24 yrs). i.e. **1098**

  Select distinct userid, gender, age, rating, genre from tentativeresult;

- To calculate the number of females from (18 to 24 yrs). i.e. **298**

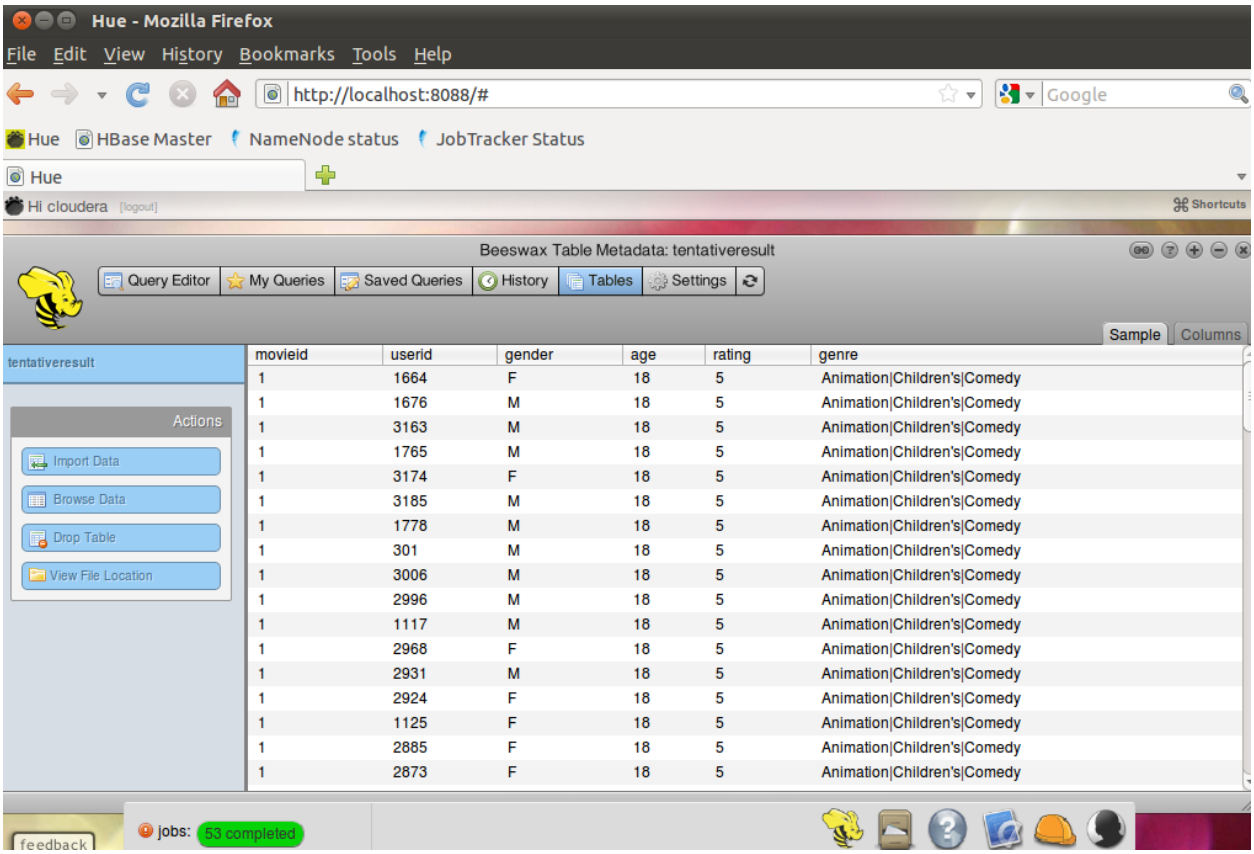
  Select * from tentativeresult where gender='F';

- To calculate the number of males from (18 to 24 yrs) i.e. **800**

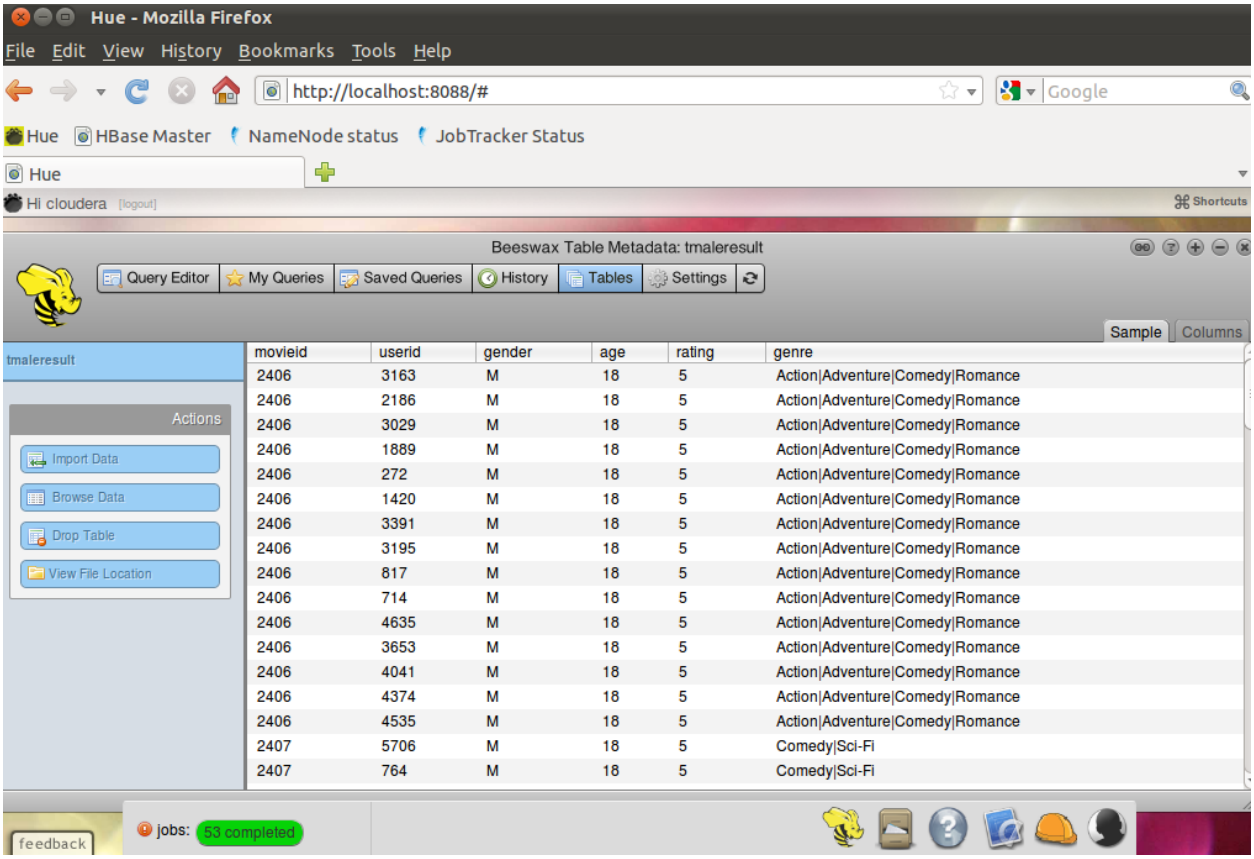
  Select * from tentativeresult where gender='M';

Some of the Screenshots of our result tables are:

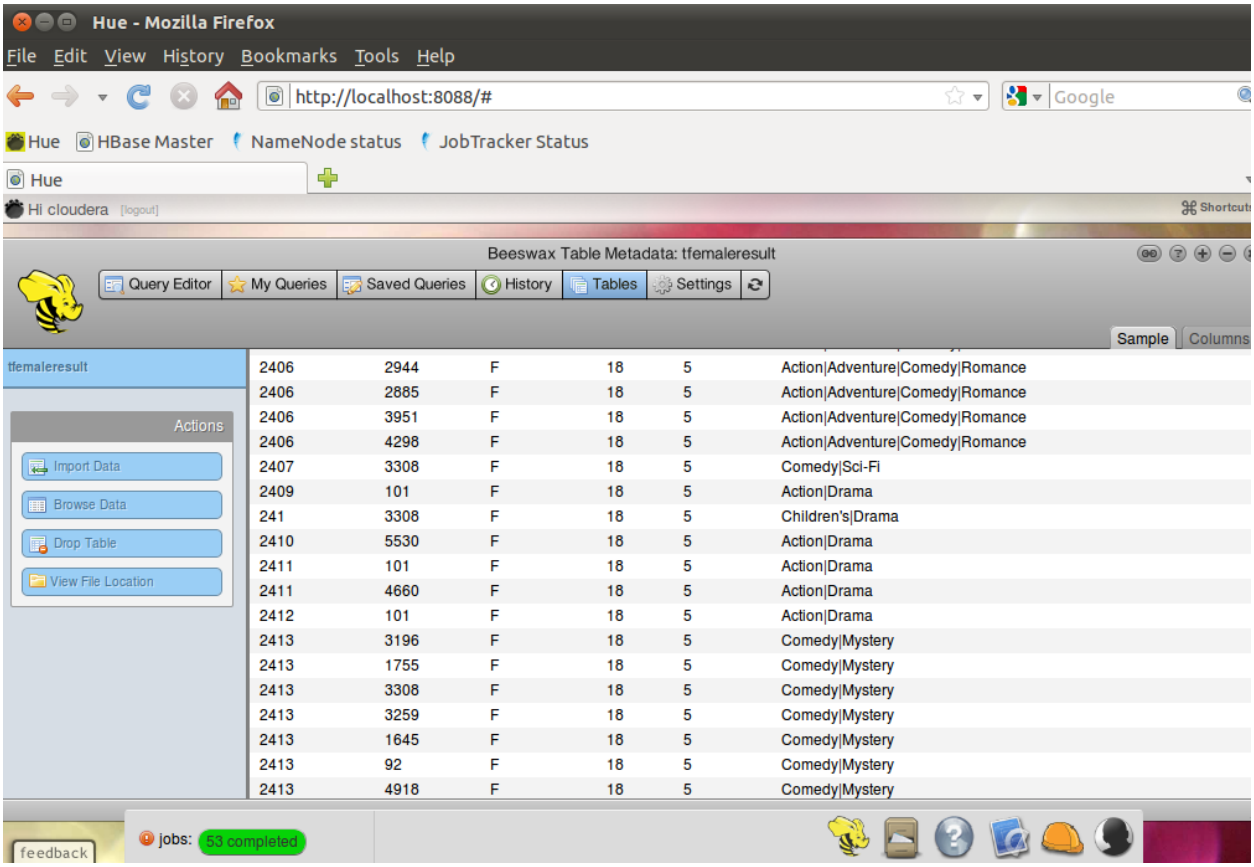
1. Tentative Result.



2. Male Table.



3. Female Table.



The above result tables are derived from the unstructured datasets that we have at the starting and with the help of this result tables the analyst can easily analyze the data according to the requirements which would be very hard to do with the unorganized data sets that we have earlier.

Some of the results that we have found are as follows:

- There are 1098 users in the datasets that are in the age range 18-24yrs.
- Among them 800 were males and most of the time they would like to watch Action/Adventure movies.
- And there are 298 females and most of the time they would like to watch Comedy/Drama type movies.

Team Project Planning:

Our job was to make meaningful analysis report from the available dataset that will help the organization to make better decision in their movie making choice. For that, our plan was to first identify the datasets based on the requirements of the organization. From those datasets we performed the required testing and proceed with the planned application. Firstly, data was inserted

in the HDFS and then we wrote map and reduce programs for it. Then we planned to apply multiple queries to the result data using map and reduce programs and to filter that data and then to analyze the filtered data and generate the final report for the end client.

We faced many problems in implementing this plan, like we had problems while creating map-reduce programs then as team we decided to switch over a tool called Beeswax Hive.

Hive can be defined as an infrastructure that runs on the top of Hadoop and it is used for the analysis of the structured as well as unstructured data. It works with HiveQL. HiveQL is a query based language similar to SQL with extra features like multi-table insert and create table as select. And Beeswax Hive is a tool which is similar to oracle developer, which is a UI to interact with Hive infrastructure (Data Warehouse).

Hive converts this HiveQL queries to the map reduce program and run accordingly. By using this tool we came up with certain sets of tables which are more organized and easy to understand.

After analyzing the sets of table we came up with the final report which contains the statistics related to movie viewers combined with movies reviews which will help the movie making company to take better decision in their movie making choice. We came up with the statistics related to the people, who are in the age group between 18years to 24years, like their favorite genre of movie based on the movie ratings and reviews. Now the movie making company may decide what movie genre they can opt to make the movie for the young population. We also categorized those tables and data according to the gender (Male and Female).

For the project planning and implementation as a team we collaborated very well and coordinated to work on this project. We faced some issues with time management; we took more time to do the tasks than we decided. But at the end we all did well according to our planning and completed the project on time. When we had issues while creating map-reduce programs, we decided as a team to switch over Beeswax Hive. During this project we also learnt to work in a team and skills to make our points or thoughts in a group.

Lesson Learnt:

- We have learned to work and collaborate in the team.
- Learned to present our ideas in a team.
- Learned the concepts of technologies relevant to the data technologies like Big Data, Hadoop, and the tools that works on top of Hadoop like Hive.
- Learned the project development cycle and the processes in each phase of the project.
- Learned time management and problem solving skills.

Conclusion:

Previously with low volumes of data, intuitive decisions would suffice for an organization. As the data size has grown human ability to make intuitive decisions has been completely reduced. This project is designed to retrieve the necessary information from the available data sets, analyze the information and provide to client in the form of reports.

Overall process and decisions are based on the quantitative methods which are cyclic process like -

- Problem solving definition and identification
- Design and build Hadoop framework
- Data sets management
- Analysis to produce models
- Execution and testing
- Creating final reports for decision making