Analyzing data using MapReduce

Mayank Gulaty
National College of Ireland
x15031705
August 5, 2016

***Abstract*—In this age of big data, everyone is focused on gaining insights with the data and new frameworks are being made to process this data in an efficient way as the data is not just on 1 computer. To overcome this problem MapReduce framework was built to process big data efficiently. This project applies MapReduce using 2 different techniques on a large movie dataset to see if there is a connection between the profile of the users and the ratings they post.**

Keywords*—*MapReduce, Hive, Recommendation System, Python

I. Introduction

The objective of this project is to apply MapReduce techniques to a big dataset for analysis and do some post visualization with the MapReduce result.

The project will start from giving information about the dataset followed by use cases of the project. Methodology and implementation of the project will be done afterwards after which visualization will be done of the acquired result.

A. Dataset Introduction

The dataset is acquired from Group Lens project. This project has a lot of datasets but for the purpose of this project 1M dataset will be used.[1]. The dataset has 3 files

Users This file contains the identities of the users, their gender, age and occupation and the zip code. The genders are depicted "M" for males and "F" for females. Age is depicted as a number which depicts the following.

- 1: "Under 18"
- 18: "18-24"
- 25: "25-34"
- 35: "35-44"
- 45: "45-49"
- 50: "50-55"
- 56: "56+"

Occupation is shown as the following:

- 0: "other" or not specified
- 1: "academic/educator"
- 2: "artist"
- 3: "clerical/admin"
- 4: "college/grad student"
- 5: "customer service"
- 6: "doctor/health care"
- 7: "executive/managerial"
- 8: "farmer"
- 9: "homemaker"
- 10: "K-12 student"
- 11: "lawyer"
- 12: "programmer"
- 13: "retired"
- 14: "sales/marketing"
- 15: "scientist"
- 16: "self-employed"
- 17: "technician/engineer"
- 18: "tradesman/craftsman"
- 19: "unemployed"
- 20: "writer"

Ratings This file contain the identities of the users posting the reviews and ranges between 1 and 6040. Each user has posted at least 20 ratings. Ratings are made on a star based model ranging between 1 and 5, 5 being the best. The values are discrete, means there is no value like 3.5.

Movies This file comprises of the id of the movie, its title and the genre of the movie. Titles also include the corresponding year of release. One movie can contain more than one genres and are separated by pipe. The genres are of 18 types as mentioned below

- Action
- Adventure
- Animation
- Children's
- Comedy
- Crime
- Documentary
- Drama
- Fantasy
- Film-Noir
- Horror
- Musical
- Mystery
- Romance
- Sci-Fi
- Thriller
- War
- Western

These all files are linked with movie ids and user ids.

O. Creanor is with the Department of School of Computing, National College of Ireland, Dublin, Ireland

B. Objectiveness of the analysis

This project has a lot of impact especially on the market research. It gives a relationship between the users profiles and movie ratings. Market researchers can make use of this information to target movies to a particular gender or age type. This project will also try to show that the role of audience is not neutral in the movie business rather it has a big impact on different types of movies. This project also tries to make a basic recommendation system by taking in consideration users ratings. All of the above mentioned things would be processed through a map reduce environment.

C. Use Cases

1) Calculate the average rating of a movie
2) How the different types of occupations have an effect on ratings?
3) How the age group can have an effect on ratings?
4) Males vs Females comparison w.r.t. ratings
5) Recommendation

D. Recommendation

Anyone using internet has one way or another come across a recommendation system be it on social media or e-commerce. E.g. when a person watches a video on YouTube, there is a section called You may also like. Buying a phone in amazon instantly gives us the deal for a mobile case and screen guard. Spotify curates a weekly playlist based on the listening history of the user. All these are the examples of the recommendation system. Recommendation system has many advantages as it helps users find the right information. Netflix movie recommendation is one of the best in the industry and they are so confident about it that it released a very large dataset in 2006 to the data science community and challenged to develop a recommendation system better than their own. [2] But after that, there were not much datasets like that when it was discovered by [3] that users identity could be revealed by reverse engineering skills.

E. What is MapReduce?

MapReduce is a programming framework developed by Google to process large datasets sitting on different computers. There are 2 key phases in it mapper and a reducer and it can be applied in many programming languages including the most popular ones Python and Java.[4] It works on key-value pairs as input and output which can be chosen by programmer [5]. For this, there are various design patterns as mentioned in [6]. There are projects being developed every now and then to make the map reduce job easier and dont require a design pattern. These convert the higher level query to MapReduce jobs. Hive [7] and pig [8] are the most popular ones that are used. Hive was developed by Facebook with a language similar to SQL called HiveQL. Most of the times using frameworks like these is beneficial, but there are times when it becomes very difficult to get the job done with these frameworks when a more complex task has to be done. And that is why MapReduce design patterns are important and native MapReduce code is still beneficial. This project will both use Hive and native MapReduce code as recommendation system task becomes very complex in pig or hive. Hive is used instead of pig because of the similar structure like SQL.

F. Related Work

The MovieLens dataset is very popular and the work has been done previously using the technologies like Spark and Pig. [9] showed how to make a recommender system using pig and [10] did the same with MLib. However, this project achieves to make the system with native MapReduce and the algorithm will be referenced as shown by [9].

II. Methodology

The first step is to load the data into MySQL by creating the appropriate tables. After that, using Sqoop [11], data is transferred to HDFS (Hadoop Distributed File System). After that , we created table in HIVE with HiveQL to store the data and apply MapReduce operations and store the results again in a table and get to the local file system using sqoop. The data then will be visualized and make it interactive if possible. Dynamic visualizations will be made in tableau and static ones will be made using ggplot package in R.

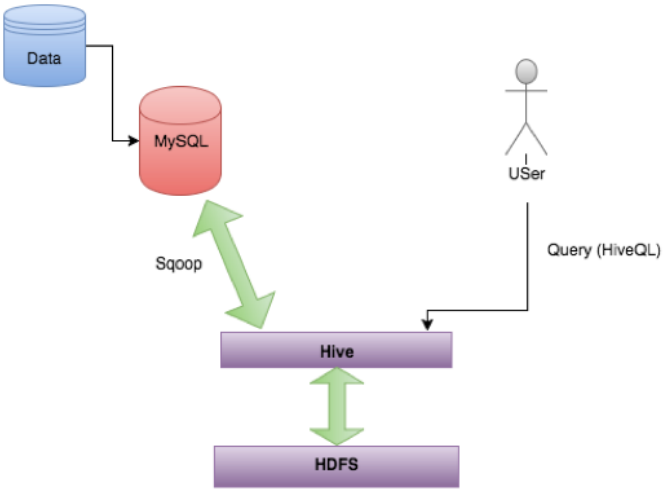

Fig. 1: Flow chart

Recommendation Algorithm

1) For every pair of movies A and B, extract the users who rated both A and B.
2) Make one vector for A and one for B.
3) Calculate the correlation between the two vectors and sort it in a descending order. So now, if a person watches a movie A, we have a movie B strongly correlated to A.

Firstly, the data will be loaded as above MapReduce 1 will run. The output generated by MapReduce 1 will be the input to MapReduce 2 and then finally the output will come as 3

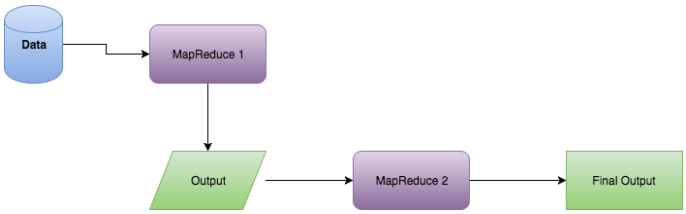

Fig. 2: Recommendation engine Flowchart

columns showing 2 movies and their similarity index which will be measured by the correlation.

For this mrjob package [12] will be used. Its a python package made to write multi step map reduce in just one program and as this task require 2 MapReduce operations, its easier to code using this package.

MapReduce 1
Mapper1 -
Input- userID,itemID,rating

Output
Key - userID
Values - (itemID, rating)

Reducer 1 - Make an array of itemID and rating the user has posted like in the form [itemID1,rating,itemID2,rating2].

Output - userID, [[itemID1,rating1,itemID2,rating2, itemIDn,ratingn]]

MapReduce 2
Mapper 2 - Make the pairs in the baove output as keys and values would be values.
Key - itemID1, itemID2 .. itemIDn
Value (rating1, rating2,..ratingn)

Reducer 2
key - Key - itemID1, itemID2 .. itemIDn
Value Correlation(value)

A. *Implementation*

Visualizations were implemented using Tableau and R. Tableau was used due to its simplistic model and easy to create visualizations. However more complex visualizations couldnt be made through Tableau, thats why R was used due to the power it gives. Shiny package was used to create the interactivity in the visualization. The visualizations are simple and self-explanatory for the end-user to understand.

1) Average Rating of the Movie
The average rating of the movies was calculated and shown with the help of an interactive bar chart where user can select the interval of ratings which he/she wants to see.

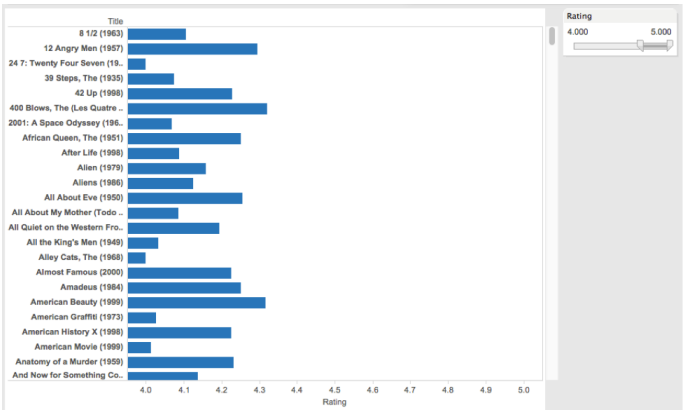

Fig. 3: Interactive Bar chart

2) How the different types of occupations have an effect on ratings?
We can see from the graph below that the managerial level people put most of the ratings and are behind 0 number code which is unspecified.

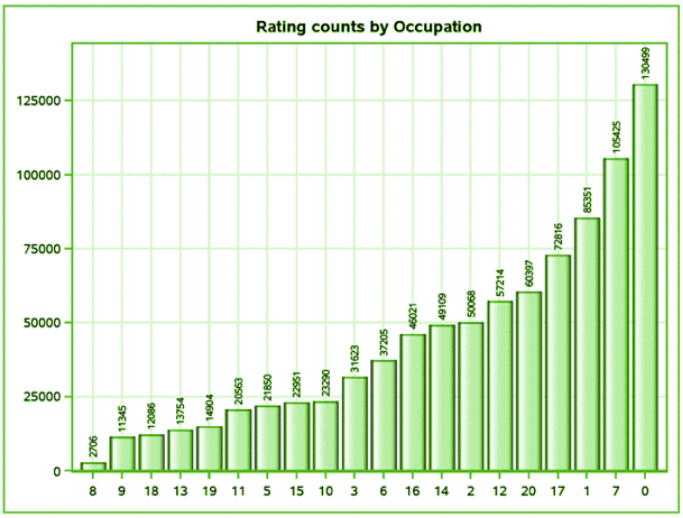

Fig. 4: Occupation vs Rating

3) How the age group can have an effect on rating?
We can see as the the age group between 25 and 34 are the most who post the ratings followed by the close competition between the age groups of 18-24 and 35-44.
4) Gender
The bar chart of gender vs ratings show that males posted almost triple the number of ratings than females.
5) Recommender
Here is a sample output of the recommender system made using the principles of MapReduce and was coded in python using mrjob package.
We can see a strong correlation between the empire

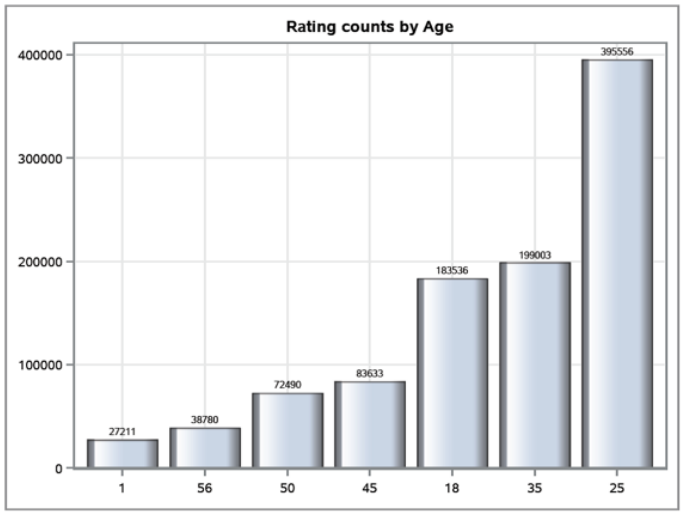

Fig. 5: Age vs Rating

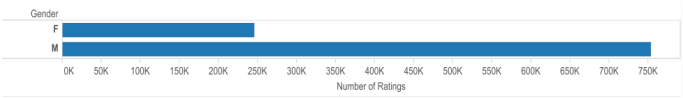

Fig. 6: Gender vs number of ratings

strikes back and the search for spock which was obvious. This also means that the recommender systems accuracy is moderately good just by seeing the users ratings.

III. Results and Conclusion

MapReduce techniques were applied to the specified dataset and some insights were revealed that could be useful for market research. Also, we saw the ease of writing MapReduce programs in hive as compared to native MapReduce but the latter is useful for writing more complex programs. A basic recommender system was made by grouping users ratings and using correlation technique.

IV. Future Work

The recommendation system can be integrated with RShiny package to make a web application and a nice graphical user interface so that users can select the movies according to their needs like by genre or by rating and the recommender system displays the name of the recommended movie. Due to time limitation, this was not achieved.

Star Trek III: The Search for Spock (1984)	Empire Strikes Back, The (1980)	0.371294
Star Trek IV: The Voyage Home (1986)	Star Trek VI: The Undiscovered Country (1991)	0.360103
Star Trek: The Wrath of Khan (1982)	Empire Strikes Back, The (1980)	0.35366
Stargate (1994)	Star Trek: Generations (1994)	0.347169
Star Trek VI: The Undiscovered Country (1991)	Empire Strikes Back, The (1980)	0.340193
Star Trek V: The Final Frontier (1989)	Stargate (1994)	0.315828

Fig. 7: Recommendation correlation output

References

[1] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 5, no. 4, p. 19, 2016.

[2] J. Bennett and S. Lanning, "The netflix prize," in *Proceedings of KDD cup and workshop*, vol. 2007, 2007, p. 35.

[3] D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl, "You are what you say: privacy risks of public mentions," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2006, pp. 565–572.

[4] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[5] T. White, *Hadoop: The definitive guide*. " O'Reilly Media, Inc.", 2012.

[6] D. Miner and A. Shook, *MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems*. " O'Reilly Media, Inc.", 2012.

[7] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, and R. Murthy, "Hive: a warehousing solution over a map-reduce framework," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1626–1629, 2009.

[8] C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins, "Pig latin: a not-so-foreign language for data processing," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 1099–1110.

[9] "Analysing MovieLens movie data using Pig A Step by Step tutorial," Mar. 2014. [Online]. Available: https://ashokharnal.wordpress.com/2014/03/25/analysing-movielens-movie-data-using-pig-a-step-by-step-tutorial/

[10] "Movie Recommendation with MLlib." [Online]. Available: https://databricks-training.s3.amazonaws.com/movie-recommendation-with-mllib.html

[11] "Sqoop -." [Online]. Available: http://sqoop.apache.org/

[12] "mrjob mrjob v0.5.3 documentation." [Online]. Available: https://pythonhosted.org/mrjob/