CA4022 Analysis of MovieLens Dataset Using Hadoop's PIG and HIVE.

Seán Cummins
18404744
sean.cummins26@mail.dcu.ie
Github

October 2021

Abstract

Apache Hadoop is a software framework that allows for the distributed processing of large data across clusters of computers. Rather than relying on single hardware architectures to perform large amounts of computation at constant high-availability, Hadoop clusters divide the task amongst individual hardware setups called nodes. These nodes are often consumer-grade and in-expensive. Along with being relatively cheap, Hadoop clusters ensure that data operations are not compromised by nodes that become unavaiable by redistributing workload.

The Hadoop infrastructure deploys the MapReduce algorithm (Dean and Ghemawat, 2008), coined by Google, to divide or 'Map' the task into identical, independent parts amongst nodes in a cluster. The outputs from clusters are then combined and aggregated, or 'Reduced' where possible. Apache Hive and PIG are frameworks built on top of the Hadoop infrastructure designed to refine the process of writing custom MapReduce jobs. MapReduce scripts in native Java are long, complex, and difficult to to write. To this end, both Hive and PIG are capable of writing equivalent code with considerably less lines and complexity, increasing user efficiency

1 Introduction

The following text aims to foster the analysis of the MovieLens dataset which consists of: movies, users, ratings, and tag information. In the following subsection, we will briefly introduce two frameworks commonly used to write and execute custom MapReduce scripts, Apache Hive, and Apache Pig. These frameworks are also used to perform data pre-processing and wrangling. In Section 2, we will perform some analysis of the MovieLens data. In this particular dataset, we are provided with the tags that users create to describe movies they have viewed. This textual resource provides a wealth of information with respect to the opinions of the movie-watchers and researchers regularly apply Sentiment Analysis in this domain.

1.1 HIVE & PIG

As already discussed, writing MapReduce code in native Java is slow, requires expertise, is difficult to prototype, requires many lines of code for even trivial tasks. To solve these issues, frameworks such as PIG and Hive have been created.

Pig Latin is the language implemented on Pig, an open source software which runs on Hadoop. Users can write code in the Pig Latin scripting language, which is then converted by the Parser in to Logical Plan script which can be read by a Compiler and converted into MapReduce jobs (Fuad et al., 2014). Pig Latin is beneficial for it's ease of programming, optimization opportunities, and extensibility.

Hive is an SQL-like query language, more commonly known as HQL. Similarly to PIG, it can compile SQL-like queries in to MapReduce jobs. Hive has benefits similar to PIG in that it allows the user to work with structured data in a familiar manner. Hive can also be integrated with Business Intelligence (Du, 2015).

2 User-Generated Tags Analysis

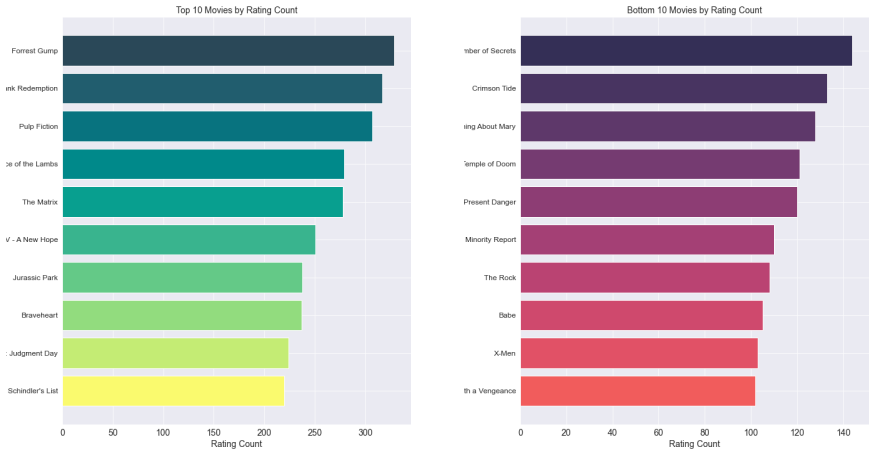
In the following section, we will be performing some exploratory data analysis, and trying to make inference from the tags attached to movies generated by users. We will firstly analyse the tags as a collection for the entire set, and compare this to the tags associated with both the best and worst rated movies. Following on from this, we will analyse the tags associated with particular genres and hypothesise whether or not these tags are good textual representations of the genre they describe.

2.1 Best and Worst Rated Movies

When a user decides to create a tag for a movie they have watched, they are providing useful information for developing an understanding of a populations

(a) (b) (c)

Figure 1: (a) Best Movies (b) All Movies (c) Worst Rated Movies



(a) (b)

Figure 2: (a) Best Rated Movies (b) Worst Rated Movies

generalized opinion about that movie. The same can be said for a movie rating. However, text intrinsically carries deeper meaning and more detailed information than numerical ratings. Figure 1 is made up of 3 Word Clouds representing the collection of all movies, the top rated movies, and the worst rated movies. It is interesting to note that the tags 'violence', and 'dark' appear as a common word only in the set of highly rated movies. It is also no surprise that the highly rated movies are often tagged with 'classic'. 'Space' is common amongst all sub figures including (b) All Movies. This could indicate that the dataset is disproportionately representing space movies with respect to other genres. Or possibly, there has been a considerable amount of space movies produced over time. A final hypothesis could be that space movies are difficult to make and consequently, producers take on a lot of risk when they create a space film. This would imply that poorly produced space movies would receive negative criticism. However, when a space movie is well done, it receives high praise, resulting in its strong presence in both the sub figures (a), and (c).

In Figure 2, you will find bar charts representative of the movies included in Figure 1 (a) and (b). These movies are the films with the best and worst average ratings according to the MovieLens dataset. As well as this, they also receive the highest amount of ratings overall. The list of movies in (a) consists of the exact movies you would expect to find here. So called 'cult-classics'. It's interesting to note that movies from the 'worst movies' receive a comparable amount of tags to the 'best movies.' This shows that tags and descriptions are not reserved solely for movies that people enjoy.

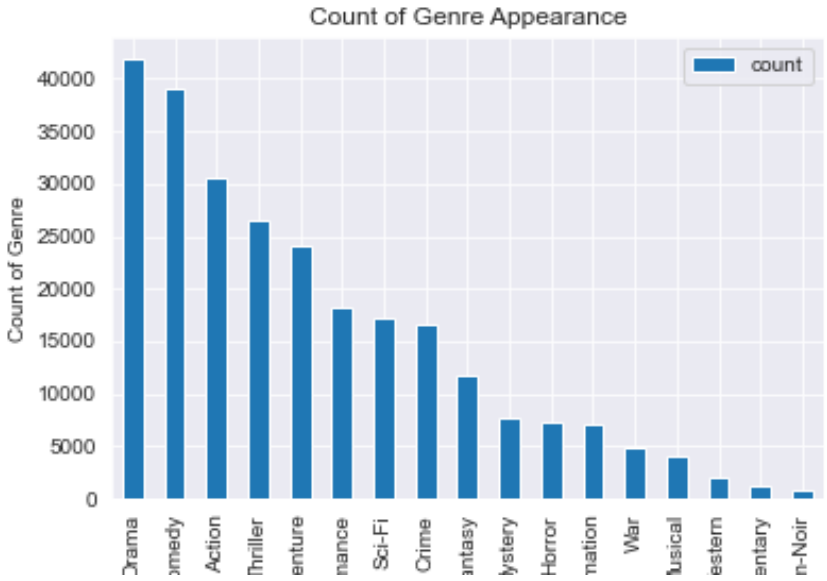


Figure 3: Genres ordered by Number of Appearances.

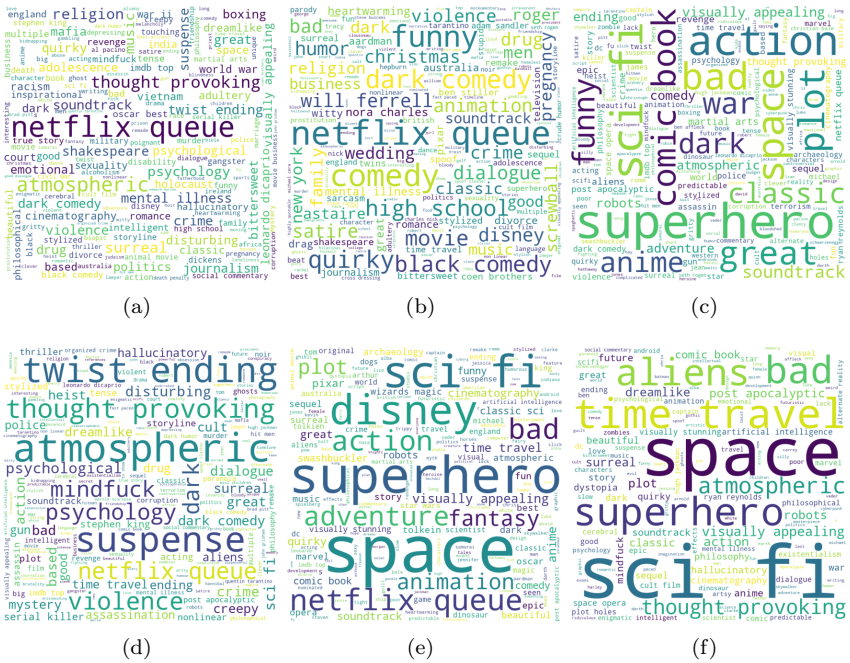
2.2 Movies Separated By Genre

Let's better understand if tags generated by users are reasonable estimators of the actual genre a movie is associated. To do this, let's visualise the abundance of movies associated with each particular genre. We will work with genres associated with many movies as it is likely they will have an equivalently large number of tags. This can be seen in Figure 3. We will select 6 of the most popular genres including; *Drama*, *Comedy*, *Action*, *Thriller*, *Adventure*, and *Sci-Fi*.

Figure 4 displays a word cloud for each of the genres mentioned above. We can clearly see that words we would expect to be associated with each particular genre do indeed appear in their word cloud. For example, in the Drama word cloud, we see tags such as 'atmospheric' and 'thought provoking'. The Comedy word cloud is very well described by its tags such as 'funny', and 'dark comedy'. We even see actor names such as 'Will Ferrell' appearing as notably large tokens. The same point can be made for the following 4 word clouds also.

3 Conclusion

We will now conclude this short review and analysis of Assignment one for CA4022. In this assignment, we were introduced to the Hadoop ecosystem, a powerful framework designed under the "Divide and Conquer" principles. The Hadoop architecture incorporates Google's MapReduce algorithm to divide big



(a) (b) (c)

(d) (e) (f)

Figure 4: (a) Drama (b) Comedy (c) Action (d) Thriller (e) Adventure (f) Sci-Fi

data tasks amongst clusters of machines rather than a single machine. This not only increases efficiency, but creates various safety mechanisms in the case of a system failure. Hadoop allows large difficult computations to be performed on a distributed system of machines that can be of consumer-grade. This eliminates the need for large, expensive, complex hardware.

We were also able to perform some light analysis on the MovieLens dataset. This analysis was shallow in the essence of keeping this review short. In our brief analysis we were able to prove that user generated tags for movies do indeed correlate with that movies genre. We've also shown that tags are good at representing viewers' overall opinions on movies.

There is room to build further on this. Understanding the emotions portrayed by the tags users write to describe movies is a common research task known as Sentiment Analysis. It would be interesting to better understand the sentiment behind movie reviews in the future.

Please find the link to my Github containing all the PIG and Hive cleaning and analysis scripts below the title of this document. Otherwise, access my Github here.

References

[1] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, jan 2008. ISSN 0001-0782. doi: 10.1145/1327452.1327492. URL https://doi.org/10.1145/1327452.1327492.

[2] Ammar Fuad, Alva Erwin, and Heru Purnomo Ipung. Processing performance on apache pig, apache hive and mysql cluster. In *Proceedings of International Conference on Information, Communication Technology and System (ICTS) 2014*, pages 297–302. IEEE, 2014.

[3] Dayong Du. *Apache Hive Essentials*. Packt Publishing Ltd, 2015.