IMDB/Rotten Tomatoes Movies Ratings and Reviews

Caio Nogueira
up201806218@edu.fe.up.pt
University of Porto

Carlos Lousada
up201806302@edu.fe.up.pt
University of Porto

Tomás Mendes
up201806522@edu.fe.up.pt
University of Porto

ABSTRACT

In this article, we will explain how the data preparation process works for our information retrieval system. The goal here is to collect and analyze a data collection containing information about movies in both the IMDB and Rotten Tomatoes databases.

CCS CONCEPTS

• **Information systems** → *Information retrieval.*

KEYWORDS

Datasets, Data Preparation, Movies

1 INTRODUCTION

Cinema was not invented by one person. However, the first to present projected moving pictures were the Lumière brothers in December 1895 in Paris. They decided to use a device made by themselves, the Cinématographe, which was composed by a camera, a projector and a film printer. This way, movies evolved tremendously: from basic frames in black and white colours with no sound to an amazing explosion of colours with beautiful soundtracks and dialogues, passing through a golden age (during the 1930s and 1940s) and a rapidly improvement of the digital technology (3D, IMAX, CGI, ATMOS, Real|Laser, among other tools).

Bearing this in mind, this article describes the first development phase of a movie plataform that aims to break any geographical barrier and intends to connect fans of the seventh art from all over the world, using a friendly and intuitive search engine with several applicable filters.

To accomplish this, we need to take several steps. Firstly, it is crucial to collect datasets with relevant information. Then, the data should go through a preparation stage, i.e., a variety of refinement tasks, such as data scaling, normalising and cleaning, in order to be easier to handle it. Furthermore, the Conceptual Model demonstrates how the dataset is structured and, finally, by using and interpreting diverse graphics, we can have a better perception of the collected data.

2 DATA COLLECTION

The data used for the information retrieval system was collected from *kaggle* [5], which is a well-known data science website that allows users to publish and use free datasets in a collaborative environment.

In order to collect the necessary data, we downloaded two different datasets from *kaggle*: one of them containing information scrapped from the IMDB's database [1] and the other from the Rotten Tomatoes's database [2] .

The IMDB dataset contains information about over 85000 movies, dated between 1915 and 2020 (when both datasets were scrapped). The data is composed by 4 different *.csv* files, from which we selected the ratings and the movies' general information: title, actors, directors, release date, age suitability, etc. By doing this, we can retrieve information about the ratings given to the movies.

The Rotten Tomatoes dataset is less extensive when it comes to the number of movies: it contains information about roughly 17000 movies. Despite covering less movies, we can use this dataset to collect textual critics for each movie, which will be useful later on future milestones. In addition, we can also collect numeric classification for each movie, such as the *Tomatometer* score and *Audience Score* (rating given by the users of Rotten Tomatoes). For these reasons, we can say that the datasets complement each other.

Both datasets are owned by the same *Kaggle* user: Stefano Leone (a data Analyst at a financial company). The fact that the data was collected by the same developer using the same methods (Python with the Requests library) proved to be a huge advantage as the datasets have a very high usability score and the data types are consistent.

3 DATA PREPARATION

After having decided which datasets would be used for the information retrieval system, we needed to refine the data. This section consists in the different steps taken in order to prepare the data for the information retrieval phase.

For this task we used Pandas [3] library from Python, which allows reading, writing and manipulation of *csv* files.

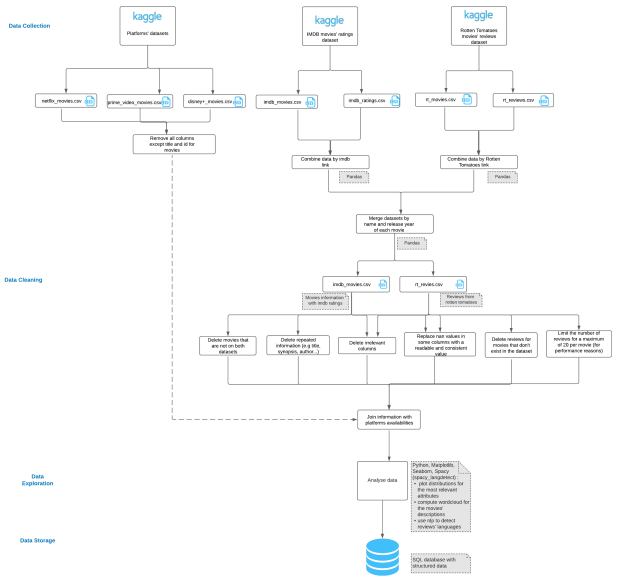

Figure 1: Pipeline design

3.1 Merging

First of all, since we selected two independent datasets we would need to merge them into one. For this reason, we could only use rows about movies that existed in both datasets. However, before calling the *merge* method, we dropped the common attributes(movie title, director, actors, genre, duration, etc).

The next step in this process was the proper merging of the datasets. We decided to join the files by title and release date.

The merging process resulted in a *csv* file containing aproximately 9000 movies and over 80 columns. In the reviews file, we would need to drop the rows that didn't refer to movies that existed in the merged dataset.

3.2 Refinement

After having merged the data, we started looking into the different columns. Since some of the columns had NaN (Not a Number) values, we decided to replace them for a more readable text ("Unknown" or "Not defined") depending on the respective attribute. At this point, we had over 80 columns, that translates into an excessive amount of data (per movie) that would not be used further on, so we disposed it.

The critics dataset, on the other hand contained a reasonable amount of columns (8), but an excessive amount of reviews per movie (about 500000 reviews in total). For performance reasons, we limited the maximum amount of critics for each movie to 20. In order to get the most relevant ones, we sorted the reviews by top critic and text length. This means that if a movie has, for instance, 50 reviews, where **1)** 15 of them are top critics and the rest of them are not, we will pick the 15 top critics and the 5 normal critics with higher word count; **2)** all of them are top critics, we will pick the 20 top critics with higher word count. After this operation, the reviews file was slightly less than 130000 rows.

3.3 Enrichment

In order to provide more information and, therefore more possibilities to query the dataset on future milestones, we used another dataset from *Kaggle*, with information about movies from three popular streaming services: *Netflix*, *Amazon Prime Video* and *Disney Plus*. The analysis of this new information resulted in three more columns added to the previous (movies) dataset. Each of the additional columns contain boolean values indicating whether the corresponding movie is available on the respective streaming service or not.

4 CONCEPTUAL MODEL

Having completed all the necessary data preparation tasks, we designed a domain conceptual model, which is displayed in Figure 1.

Movie is the main class, since it connects all other entities in the domain. This class contains relevant information about each movie. The classes *ContentRating*, *ProductionCompany*, *Writer*, *Director*, *Actor*, *Genre* and *Country* are association classes that provide additional information about the movies.

Furthermore, the classes *RTReview* and *IMDBRatings* provide information about the Rotten Tomatoes critics and IMDB ratings, respectively: *RTReviews* entries have as attributes not only the review itself but also some information about the critic who published it (name and reputation on the website). *IMDBRatings'* entries have the rating itself associated (1-10). This table will then be analyzed to compute the mean rating from *imdb*.

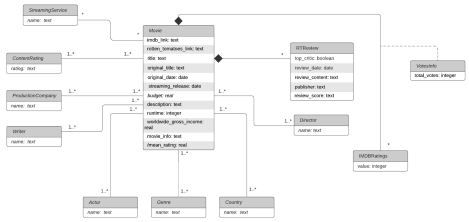

Figure 2: Conceptual Model

5 SEARCH TASKS

The data collected and prepared can be obtained by the information retrieval system in several ways. Some of the possible retrieval tasks are listed below:

- Search for a specific movie by name, Rotten Tomatoes link or IMDB link
- Search movies from a certain actor
- Search movies from a certain director
- Search the most popular/highest rated movies from a certain genre.
- Search for movies with a specific length interval
- Search reviews for a movie made by a verified Rotten Tomatoes user (top critic)
- Search the most popular movies from a certain genre.
- Search for movies from a specific year.
- Search for most popular movies available on a certain streaming service.

6 DATA CHARACTERIZATION

With the data prepared, we started to analyse its distribution and characterization. To complete this task, we used the following Python modules: *Matplotlib*, *Seaborn* and *WordCloud*.

One of our concerns was to have a wide variety of data. This means that we wanted our final dataset to contain a decent number of movies (and respective reviews and ratings) from a relevant range of years. This proved to be the case as seen in the following graph.

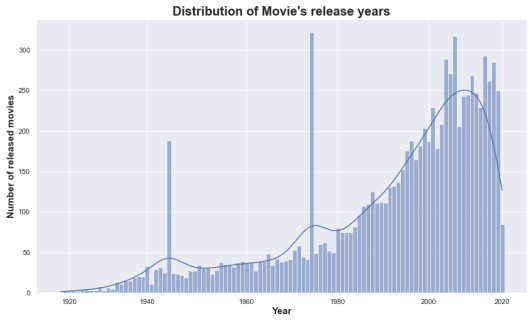

Figure 3: Number of movies per year

As expected, on average, our dataset contains more movies that were released since the 2000s. However, there's still a good representation of movies from the previous century.

Following this line of thought, we also strived for a wide representation of movie production countries, genres and content-ratings. This can be observed in the following 3 graphs.

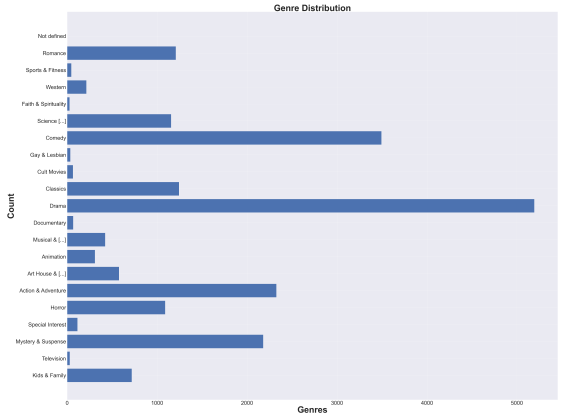

Figure 4: Genre distribution

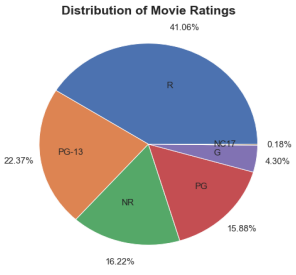

Figure 5: Content-rating distribution

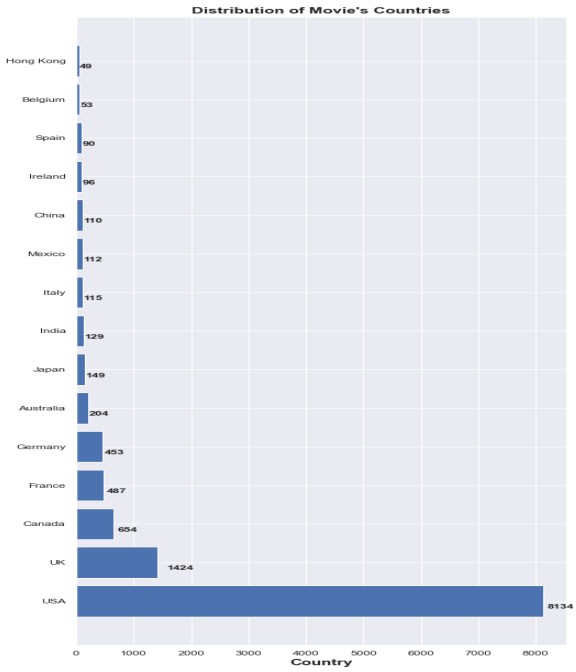

Figure 6: Production Country distribution

Analysing these graphs, we can see that there is a decent distribution of movies in all of the mentioned parameters. We found the results to be very accurate, as it shows Drama, Comedy and Action/Adventure as the most common genres and that R-Rated is the most popular content-rating.

Furthermore, regarding rating data, we felt the urge to verify if there's a wide range of ratings and differently rated movies. To do that, we can analyse the following graphs.

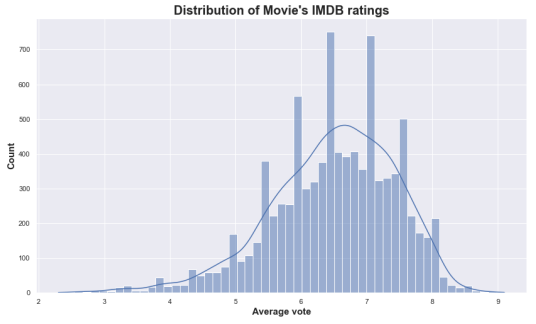

Figure 7: IMDB's rating distribution

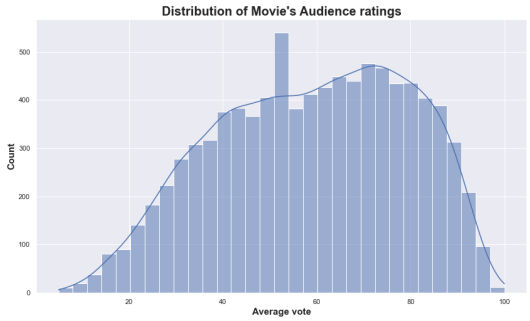

Figure 8: Rotten Tomatoes's Audience rating distribution

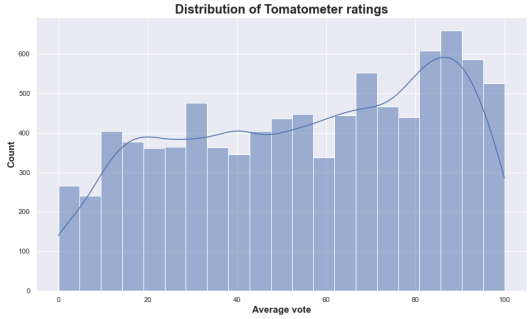

Figure 9: Tomatometer rating distribution

Once again, we can conclude that our dataset includes a wide variety of movies, rated in very different ways in several different ratings.

Finally, we decided to analyse a bar plot to check the evolution of ratings throughout the years, and a Word Cloud plot, to verify that there are several different plot points in the movies taken into account (for instance, family, love, friend, life, etc). Those plots can be seen, respectively, below.

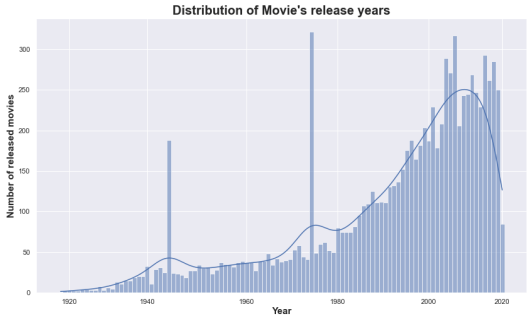

Figure 10: Average IMDB's ratings per year

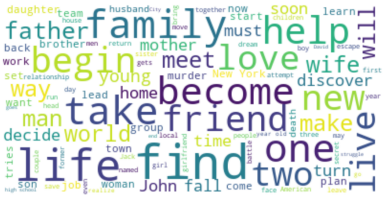

Figure 11: Movie description's Word Cloud

6.1 Text characterization

In our dataset, there are two main textual values for each movie: its description and the critics. Our goal at this point is to improve the understanding of both and, to achieve that objective, we computed the word cloud for the movies' descriptions, which is displayed on **Figure 10**. As to critics, we used *NLP (Natural Language Processing)* with *Spacy* [4] to detect the percentage of English reviews in the dataset. The natural language processing algorithm pointed out that approximately 99 percent of the reviews were English. The other reviews are written in Portuguese.

7 CONCLUSIONS

During this first milestone, our focus was to choose a dataset with "recent" data and manipulate it in order to better understand its information and make it ready for the information retrieval phase. This goal was accomplished successfully, despite having some problems with the merging and refinement process as well as some missing rows' values. The end result is a reasonably large dataset (over 9000 movies) with coherent data about movies, ratings and reviews.

In future milestones, the goal will be to create a search engine powerful enough to query the dataset and thus obtaining data from it.

REFERENCES

[1] Stefano Leone. *IMDB movies extensive dataset.* Sept. 2020. URL: https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset.
[2] Stefano Leone. *Rotten tomatoes movies and critic Reviews Dataset.* Nov. 2020. URL: https://www.kaggle.com/stefanoleone992/rotten-tomatoes-movies-and-critic-reviews-dataset?select=rotten_tomatoes_critic_reviews.csv.
[3] *Pandas.* URL: https://pandas.pydata.org/.
[4] *Spacy · industrial-strength natural language processing in python.* URL: https://spacy.io/.
[5] *Your machine learning and Data Science Community.* URL: https://www.kaggle.com/.