Perception of literature and cinematography according to NLP

I. Problem statement

According to Global English Editing, India, Thailand, and China are the countries with the best reading habits in 2018, spending on reading between 8:00 and 10:42 hours per week. The U.S., with 5:42 hours, is in the 22nd position. 37% of American adults with a high school degree or less and 7% of college graduates have reported not reading a book in any format in the past year.

In the age of Netflix and instantaneous entertainment, it seems to be people who are still reading, but it is hard to keep the people interested in reading classic, historical and universal literature. How we can improve this reading behavior? Past year, Netflix developed around 50 literary projects (turning novels into series). Cinematographic adaptations bring books to people and could inspire them to read the books later.

This project tries to find relations between movies and books through genres ables to give us priceless information. Genres are determined by key-words of overviews, descriptions, using Part of Speech filters and NLP preprocessing pipelines to normalized unstructured data. If we are able to connect books and movies, we could recommend books depending on the people's preferences or give us a guide about how to predict the success of the metamorphosis from novels to cinematographic adaptations and discovering what genres would achieve more acceptance by the audience.

The inspiration for this project is educational. We are trying to understand the impressions that books and movies generate in population to find innovative ways to teach them and bring books to them. Also, we'll use all the available information on our database to predict trends related to how to decide what books could be interesting purposes for cinematographic adaptations. The educational sector, companies associated with bookselling and interested in turning books into audio-visual projects, could be interested in these results.

II. Description of the dataset

1. Data Acquisition

Meta-Data and Book Reviews Texts from Goodreads Datasets were used in this project. Meta-Data is a complete book graph of 2.36M books: titles, ratings, publication date, number of pages, similar books, authors, description, language, etc and Book Review Texts correspond to 1.38M scraped records with detailed review text: users, reviews, sentences, ratings and if the review contains a spoiler or not.

For movies, we choose Movies Dataset, with 45 thousand movies listed in the Full MovieLens Dataset and 26 million of ratings from 270 thousand of users for all 45.000 movies.

2. Data Wrangling in Movies

In this section, we import two files of **the movies Dataset: *movies_metadata*** and ***keywords***.

After importing the metadata, useless columns are dropped off the dataFrame as *budget*, *home page*, *poster path*, *revenues* and information about *videos*. Instead, we put attention in *movie id*, *original languages*, *title, overview, release date*, *genres*, *runtime*, *spoken languages*, *vote average*, *vote count* and *popularity*.

Some regular expression techniques are applied to get the *movie id* and *genres* because all of them are included as part of dictionaries of information. We note that one particular movie has two or more *genres*, then we pivot them, adding 18 columns to the dataFrame with the respective genre names: *Action, Adventure, Animation, Comedy, Crime, Documentary, Drama, Family, Fantasy, Foreign, History, Horror, Thriller, Music, Mystery, Romance, War, Western.*

Secondly, the *keywords* associated with the *movie id* are imported. Using regular expressions we extract a list of words for every movie and finally, both dataFrames are merged and saved in Data interim folder. An illustrative pipeline of the process is displayed in Fig. 1.

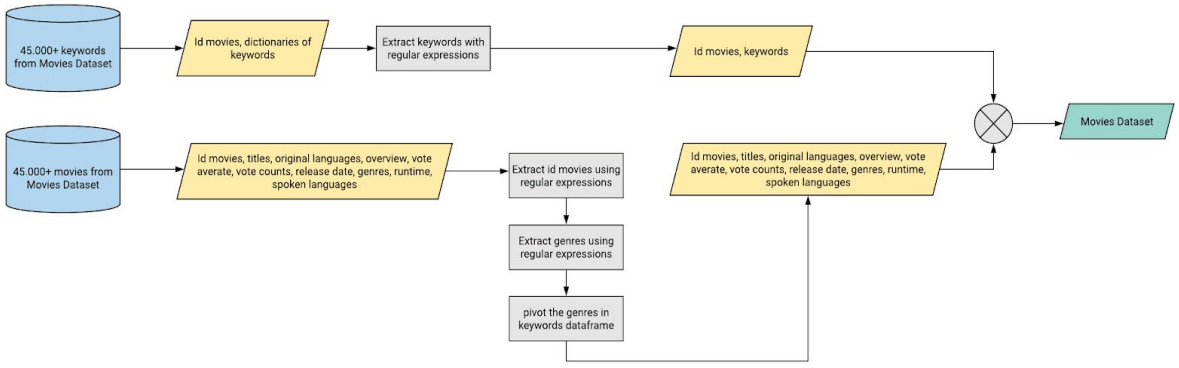

Fig. 1: Pipeline Movies Wrangling

3. Data Wrangling in Books

Book data consists of a dataset with general information about books, useful to discover languages, number of pages, publication years and an average rating of a wide variety of books. Additionally, we applied a pre-processing on the reader reviews dataset to get some NLP metrics thought sentiment analysis.

About the general information database, we focus on *titles, authors, average rating*, *book id*, *description of books*, *language code*, *number of pages, publication year*, *rating counts* and *similar books*. The wrangling process, step by step, is the following:

1. Are we including dictionaries in this analysis? We must consider that the number of pages of dictionaries is larger than common books and for that reason dictionaries are filtered using regular expressions. We found almost 500 documents belong to this category.
2. If we are curious to work with the *average rating* of books, we need to consider how many r*ating counts* have everyone and delete **outliers**. For instance, the *average rating* of books with just one *rating count* can't be compared with others with one hundred counts. To deal with these differences, we calculate the **z score** of every *rating count* and filter books with z higher than 3 (standard threshold).
3. We add absolute ratings to deal with ratings in linear and discrete versions.
4. Delete missing data in the publication year

Then, an NLP preprocessing is applied to read reviews before calculating the sentiment patterns. The techniques used are expanding contractions, subtraction of special characters, tokenization, and lemmatization of the words. We define a *pre-processing* function as a pipeline of the methods mentioned and a *sentiment parameters Pattern* function that calculate the polarity and subjectivity pattern of every review. In this case, we measure polarity using TextBlob and AFINN lexicon. The new metrics, patterns according to AFINN, TextBlob and normalized text are included in reviews of movies dataset. The pipeline of the process is displayed in Fig. 2.

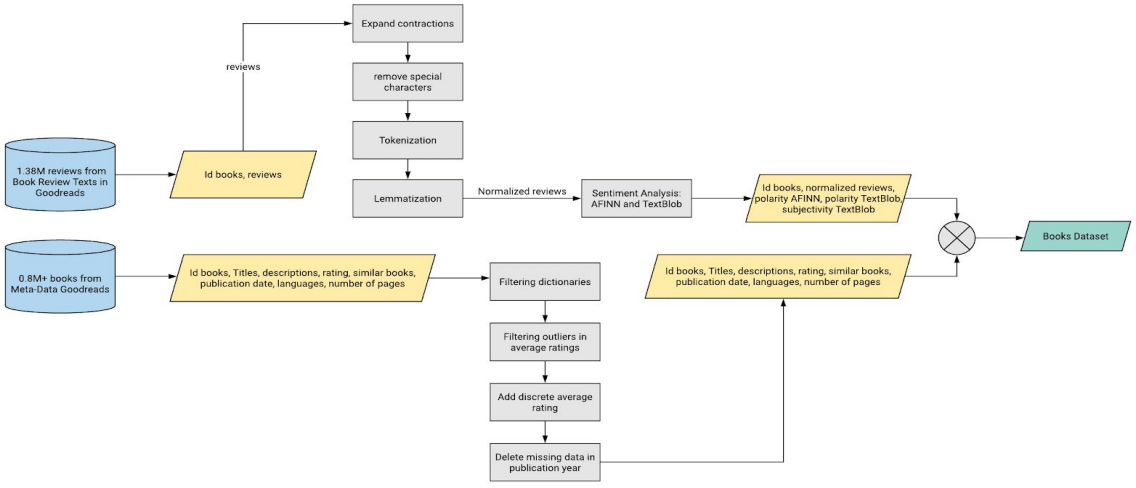

Fig. 2: Pipeline Books Wrangling

III. Initial findings from the exploratory analysis

1. Exploratory Data Analysis for Movies

The following analysis was applied to the movie dataset previously mentioned, which contains more than 40 thousand films. We are curious about release dates, languages, average scores, genres and time series of movies.

A. What are the most popular languages in movies?

As we can observe in Fig. 3, English is the most recurrent language in movies, followed by French, Italian, Japanese, Germain, Spanish and Russian. A little more down are Hindi, Korean and Chinese.

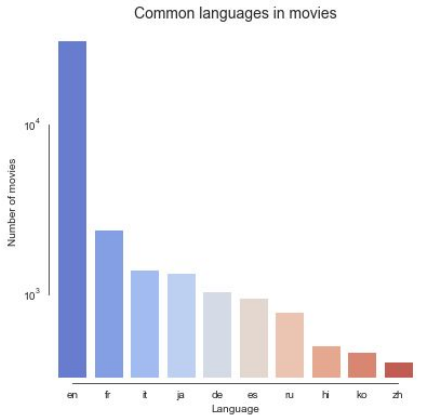

Fig. 3: Original languages in movies

B. How loved or hated are long movies?

For Hitchcock, *"the length of a film should be directly related to the endurance of the human bladder",* but certainly, some directors would disagree with him. In this section, we analyze the relation between vote rating (from 0 to 10 scores) and runtime of films. We only consider the movies with more than 4 vote counts and a runtime superior than 15 minutes, which represents three superior quartiles of data.

The mean of the runtime in movies is around 100 minutes and there are some movies that exceed the three hours.

We appreciate that the range of runtime is spreading as the vote rating increases. In other words, the movies little liked to have a runtime range shorter than the more liked movies. On the other hand, we get high scores in short and large movies. Then, there is not a clear trend to love or hate movies with a determine length, but films with excessive runtimes seem to have a good reception (vote rating lower than 4 only belongs to films with a maximum runtime of 200 minutes, meanwhile almost all the films with duration superior at 300 minutes have scores upper than 5 scores). If we measure the **z score** of the runtime to every movie, we get two classes: outliers or longer movies and regular movies, as is shown in Fig. 4a. Regular movies have a mean of 93 minutes and a maximum value of 204 minutes, meanwhile, when we treat all data as one, the mean is 100 minutes and the maximum value is 1256 minutes. Details are displayed in boxplots in Fig. 4b, where the percentiles 25 and 75 to regular movies are 86 and 107 minutes and for longest movies, 230 and 350 minutes.

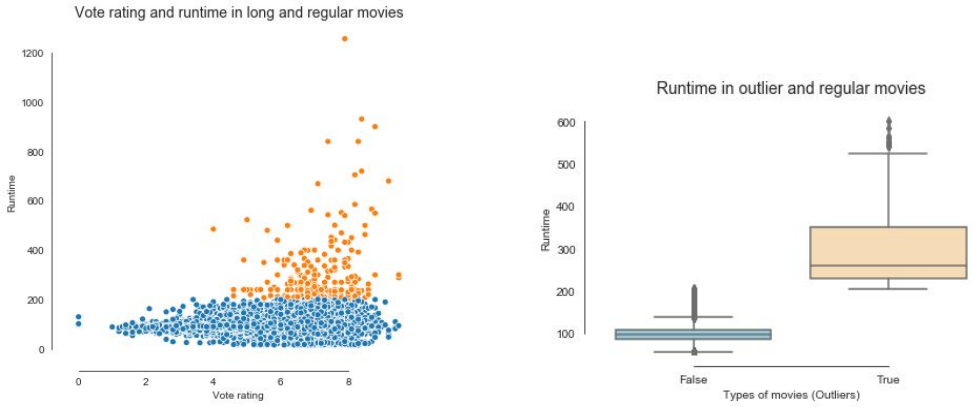

Fig. 4: a) Scatter plot runtime (left) and b) vote rating in movies and boxplot (right)

But what kind of movies exceed the 400 minutes? The list below is showing us those films, with the vote average included.

	title	release_date	runtime	vote_average
20312	Empire	1964-08-02	485.0	4.0
8469	War and Peace	1966-03-14	422.0	7.5
33132	Seventeen Moments in Spring	1973-01-01	840.0	7.4
30113	I, Claudius	1976-09-20	669.0	7.1
14216	Hitler: A Film from Germany	1977-07-07	442.0	7.6
23399	Centennial	1978-10-01	1256.0	7.9
13531	Berlin Alexanderplatz	1980-08-28	931.0	8.4
6605	Shoah	1985-11-01	566.0	8.7
25076	North and South, Book I	1985-11-03	561.0	6.9
37319	Shaka Zulu	1986-11-24	523.0	5.0
18194	The Civil War	1990-09-23	680.0	9.2
9839	Satantango	1994-02-08	450.0	8.1
8942	From the Earth to the Moon	1998-04-05	720.0	8.4
9027	The 10th Kingdom	2000-02-25	417.0	7.5
41614	Band of Brothers	2001-09-09	705.0	8.2
12834	Into the West	2005-06-10	552.0	7.8
32847	The Master and Margarita	2005-12-19	500.0	6.2
37866	Planet Earth	2006-12-10	550.0	8.8
37689	War and Peace	2007-10-19	480.0	5.6
12708	John Adams	2008-03-16	501.0	7.6
21966	Generation Kill	2008-07-13	470.0	7.8
32109	Little Dorrit	2008-10-26	452.0	7.5
37867	Life	2009-12-14	500.0	8.5
26654	The Pacific	2010-03-15	540.0	7.9
36455	Long Way Down	2010-11-30	543.0	7.4
26838	The Story of Film: An Odyssey	2011-09-03	900.0	8.8

Fig. 5: More extended films order by release date

This list has only five movies, with clear historical contents and all of them before 1995: Empire, topping the list, is a black-and-white silent film of more than eight hours of slow-motion footage of an unchanging view of the Empire State Building. War and Peace, based on Leo Tolstoy's 1869 novel is about the Napoleonic era. Hitler: A Film from Germany is a 1977 Franco-British-German experimental film; Shoah, a french movie about the Holocaust and Satantango, a Hungarian film based on the novel of the same name, about authoritarianism during the Hungarian People's Republic.

The rest of the names on the list are miniseries based -almost all of them- on novels or history. Seventeen Moments of Spring is a 1973 Soviet twelve-part television series; Berlin Alexanderplatz is a 14-part West German television miniseries, Heimat is a series of films about life in Germany from the 1840s to 2000. Shaka Zulu is a 1986 South African television series based on the story of the king of the Zulu, Shaka. The Civil War is a 1990 American television documentary miniseries about the American Civil War.

Perhaps they are not films of one part, we include them in the analysis due to their historical value and because of on the data there are more series with diverse lengths included as movies.

C. Are movies getting longer?

In Figure 6 we display the relationship between the length and release date of the movies in the most common languages, with runtimes upper than 15 minutes. As we know, English is the most popular language and almost all the runtime outliers are in English, but we can find a pair of Germain (Berlin Alexanderplatz, 1980; Hitler: A Film from Germany, 1977), French (Napoleon, 1927; The French Revolution, 1989), and Japanese (The Guyver: Bio-Booster Armor, 1989) outliers films.

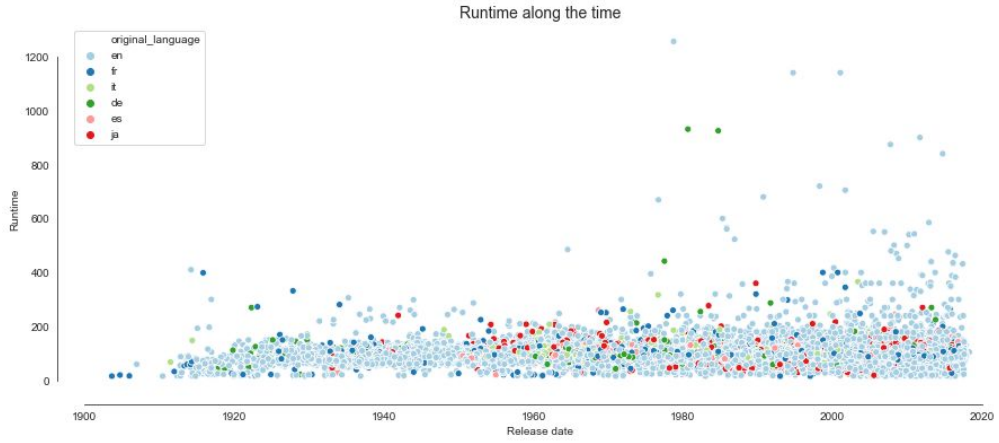

Fig. 6: Scatterplot of the length of movies by language and release date

What if we group the movies per year and determine the mean of the length as a time series? Let's have a look at the line chart below, where we can see a clear upward trend during the 20th century, but later, from the beginning of the present century until now, the length of films keeps around 100 minutes. We are using rolling means in the windows of 5 years here.

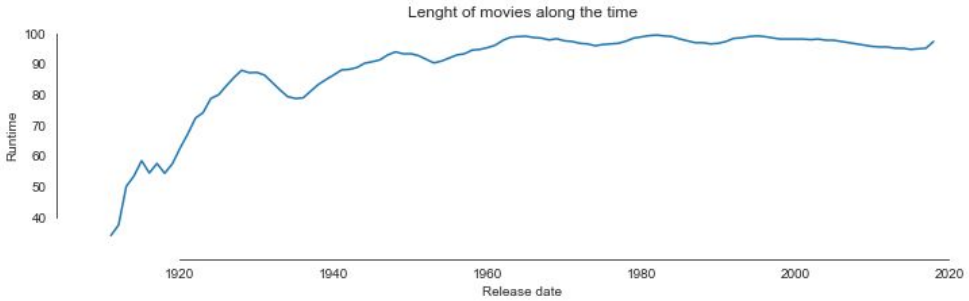

Fig. 7: Runtime of movies during the XX and XXI centuries

D. Historical release of movies

If you look at the following graph (see Fig. 8), you will notice an expected exponential growth of movies released every year from the beginnings of movies until now. Most interesting is to study the curves of movies released in different countries during the same period. Fig. 9a displays the most popular languages in movies in the logarithmic scale, to include English movies in the same plot and Fig. 9b is the same analysis in linear scale excluding the English movies.

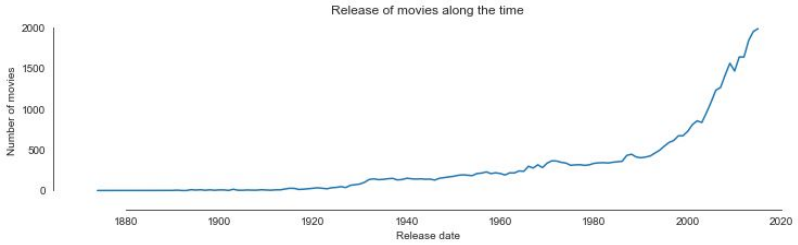

Fig. 8: Release of movies during the XX and XXI centuries

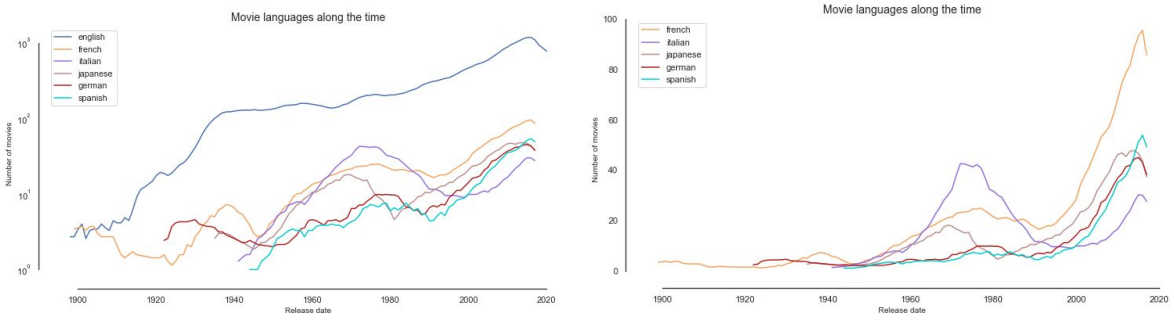

Fig. 9: Release of movies by language in a) Logarithmic scale and b) Linear scale

If this sample is representative of the development of films in the world, we could relate these behaviors with political and economic contexts from the past century. English and French movies have early development, but the first one experiment a dramatic increase before 1940. World Wars and repercussions of the Great Depression could have impacted the growth of the french seventh art during the first half of the past century. Italian movies have an explosion in the second half of the centuries, probably due to the same reasons. In the case of Germain movies, they played a fundamental role in the politic campaigns after WWI and we can observe a slight evolution of movies between 1920-1930 and a clear decay during the second war and after that. Finally, all the languages except English show a decrease in the movie evolution during the early 1990s recession.

In Fig. 10 we split English movies according to the production country. The curves of Britain's U.S have a bigger and younger growth than Canadian and Australian movies.

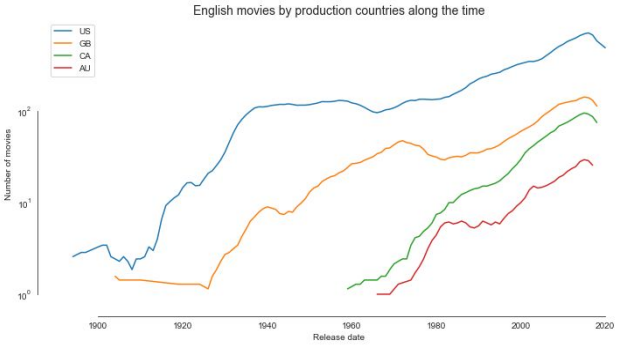

Fig. 10: Release of English movies by production country

E. How many languages can we find in the same movie?

Some movies have a list of spoken languages because they develop their stories in different countries or cultures. Can we measure the frequency and how many languages appear in every movie?

One or two languages are the most typical case, according to Fig. 11, that displays the histogram of languages in every movie using a logarithmic scale. We found less than ten movies including more than 9 languages and one film with 19 languages, called *Vision of Europe,* from 2004 and correspond to an anthology film that contains 25 short films about 25 directors from Europe.

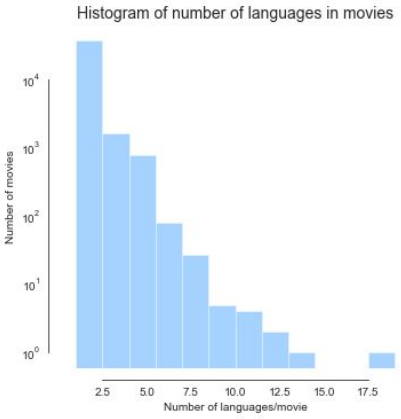

Fig. 11: Spoken languages in movies

F. Distribution of movies by language

In this section, statistical tests using a significance level $\alpha = 0.05$ are applied to determine if distributions of movies split by language have the same population mean. The one-way ANOVA test is used because we try two or more groups of different sizes. We build distributions using the most popular languages mentioned above. The histogram and respective KDE of English movies are displayed in Fig12a, where we observe that the mean of release does not exceed 250 movies every year. Deviation reaches 308 movies and the curve is leptokurtic (higher peak and profusion of outliers) and highly positively skewed. The same analysis of French and Italian films (Fig12b) reveals curves leptokurtic and highly skewed, but in this case, means are smaller (around 20 movies in French and 16 in Italian films) and the maximum value of release in French movies duplicate the biggest release in Italian productions.

The first test applied **rejects the null hypothesis about English, French and German distributions have the same population mean**, with a $p << 0.05$ and as we discuss in the historical time series section, the release of English movies has had a different and more speed up evolution. But, certainly, there are relationships between the other distributions analyzed. In the second test, the null hypothesis is **French and German distributions have**

the same population mean and in this scenario, we **fail to reject the null hypothesis**, with $p = 0.22$. Finally, the third test is related to the other popular languages (Japanese, German and Spanish films). The same ANOVA test **fails to reject the null hypothesis** with $p = 0.059$. Fig. 13 displays the KDE of these languages.

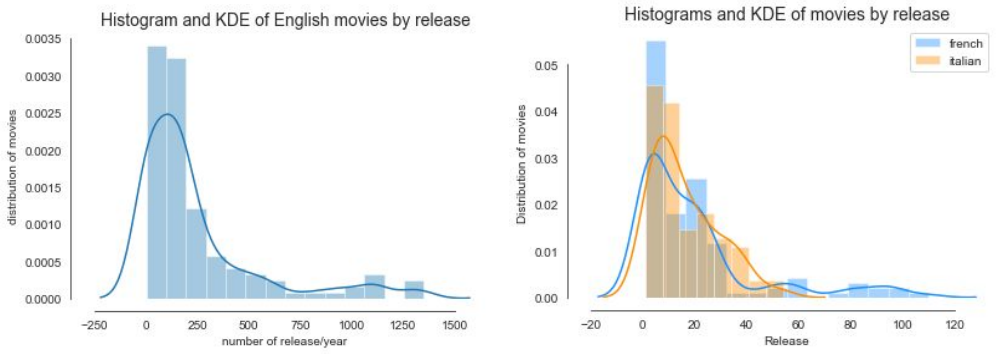

Fig. 12: Histograms and KDE of movies by release/year in a) English and b) French and German languages

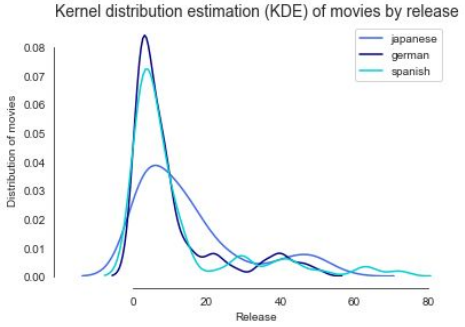

Fig. 13: KDE of Japanese, German and Spanish movies by release/year

G. Recurrent genres in movies

Every movie on the dataset belongs at least two genres of the following categories. The graph in Fig.14 shows how recurrent is every genre and as we could expect, *Drama* and *Comedy* are the most popular, followed by *Action*, *Horror-Thriller*, and *Romance*. The following table lists some of the movies of every genre order by popularity.

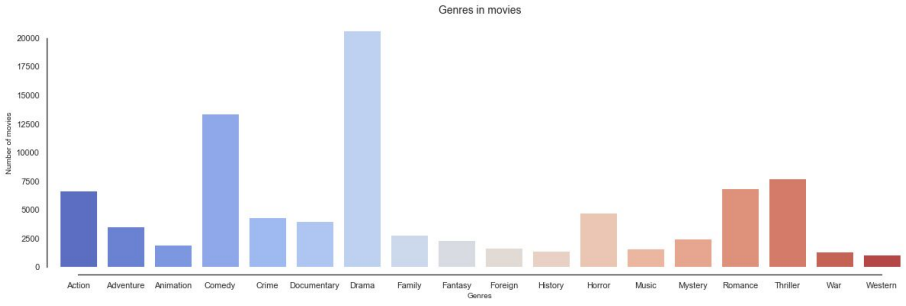

Fig. 14: Genres in movies

Genre	Movies
Comedy	Minions, Big Hero, Deadpool, Guardians of the Galaxy, Forest Gump, Ted, Pirates of the Caribbean
Drama	Gona Girl, War of the Planet of the Apes, Blade Runner, Whiplash, Logan, The Shawshank Redemption, Schindler's List, Life is Beautiful
Romance	Beauty and the Beast, The Twilight Saga, Fifty Shades of Grey, Titanic, Cinderella, La La Land
Thriller	John Wick, Gona Girl, The Hunger Games, Pulp Fiction, The Circle, Alien, Transformers, Get Out, Jurassic World
Horror	Alien, The Dark Tower, Get Out, World War Z, Don't Breathe, Rings, Saw, Annabelle, Black Mirror, Amityville
Action	Wonder Woman, Avatar, Captain America, The Avengers, Thor, Doctor Strange, Suicide Squad, Star Wars

Table 1: Examples of movies by genre

H. How likely are movies by genre?

In this section, we explore the average score of films by genre, displaying histograms and KDE to compare visually and then statistically the distributions of movies.

Action and Horror are the first genres chosen and as we note in Fig. 15a., distributions are negatively skewed. Action distribution is leptokurtic ($k > 3$) and Horror distribution is platykurtic ($k < 3$). We apply the Levene test, in which the null hypothesis is that **all input samples are from populations with equal variances**. Levene is used because we are dealing with significant deviations from normality (therefore, the assumptions of the ANOVA test are violated).

The first null hypothesis proposes that **Action and Horror movies are from populations with the equal variances** and we **fail to reject the null hypothesis** with a p-value $p = 0.26$

The second null hypothesis adds one of the less frequent genres, War. From Fig.15b., we chose previously two genres with similar frequency. How much likely are War movies? Apparently, they concentrate more positive scores because the distribution shows a more high skew than Action and Horror movies. And the same previous test, but adding one more genre **fail to reject the null hypothesis** with a p-value $p = 0.33$

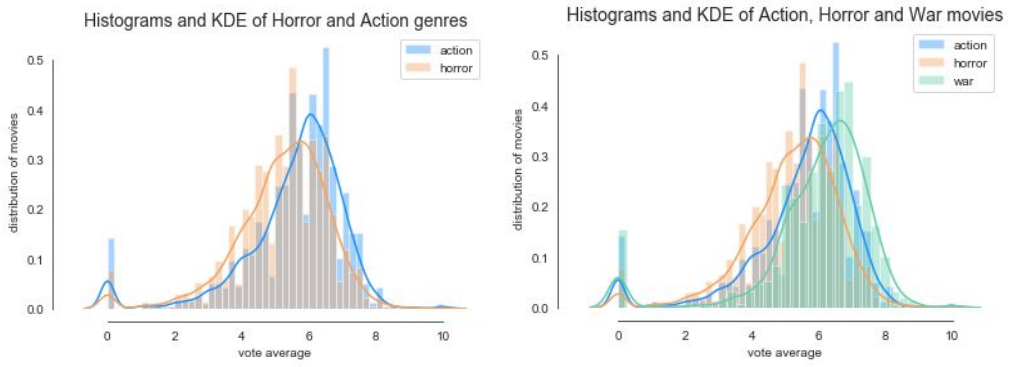

Fig. 15: Distribution of movies in genres Action, Horror (a) and War (b) by vote average

Finally, we apply the third test using Drama, Comedy, Romance. **The null hypothesis asserts that Drama, Comedy and Romance movies are from populations with the equal variances** and we **fail to reject the null hypothesis** with a p-value $p = 0.09$. Figure 15 shows that KDE of those genres is similar. If we include some of the previous genres to this list (action, horror or war) the result is to reject the null hypothesis. Therefore, we conclude that Drama, Comedy, and Romance belong to one cluster of liked movies and Action, Horror and War to another.

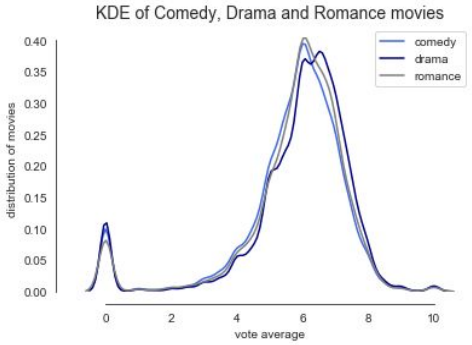

Fig. 16: Distribution of movies in genres Comedy, Drama and Romance by vote average

2. Exploratory Data Analysis for Books

"Books and movies are like apples and oranges. They both are fruit but taste completely different." (Stephen King).

In this section, we'll try to understand and extract the most relevant and useful information about books, preparing us for the next step, where we'll do some experiments to find the similarities between these kinds of apples and oranges. The book dataset contains more than 550 thousand items. To display some figures and answer proposal questions, samples of the whole data are used, chosen randomly and applying bootstrapping and statistical tests to assure the legitimacy of results.

A. The relation between publication year and average rating: along the time, could we detect some trends?

As we could expect, we find more book publications in recent years with respect to the past century. How is the evaluation of readers for books from the past century? How is right now? The average rating is measured between 1 and 5 scores. Older books published before 1960 have a good reception from the readers and from 1980 we observe a more wide range of scores, but the distribution of rating shows a clear more positive reception, with a concentration of books mainly between 3 and 5 scores. It means, that in general, books are qualified as Neutral, Good and Excellent moving scores to a categorical scale.

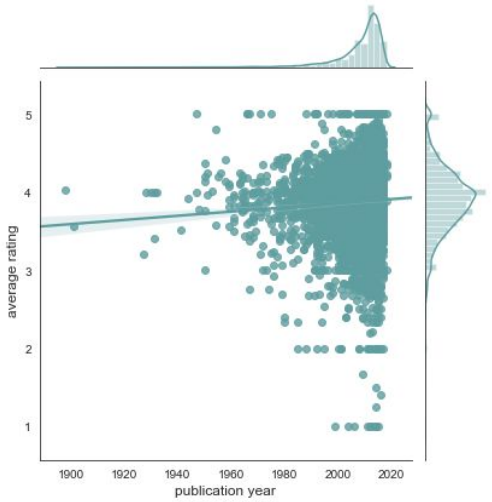

Fig. 17: Scatter plot average rating and publication year

To know if this sample is representative of the population, we compute the Kolmogorov-Smirnov statistic on 2 samples: the observed sample distribution and the alleged original distribution for the **null hypothesis that the rating average sample distribution is drawn from the population of 550k books and then is representative**. With a p-value $p = 0.32$ **we fail to reject the null hypothesis.** Furthermore, a pairs bootstrapping is applied to measure the Pearson coefficient between the rating average and publication year. The relationship found in this sample of the original population is a weak correlation of $p_{sample} = 0.0518$, but is it really illustrative? After 1000 trails of 5000 data points chosen randomly with replacement (the size of these samples is comparative with the size of the original sample analyzed), we calculate the Pearson coefficient and determine how many times we get a coefficient equals or higher than p_{sample} . We fail to reject the null hypothesis with a p-value $p = 0.358$.

B. Authors are writing more long novels?

Firstly, we inspect the histogram and KDE for the number of pages in every book (see Fig. 18). The mean is 260 pages and the percentiles 25% and 75% are 152 and 346 pages respectively. The maximum value is 1078 pages and corresponds to the books The Nietzsche Anthology, published in 2013, The Count of Monte Cristo by the french Alexandre Dumas, The Source, a historical novel about Jewish people before the monotheistic era, published in 1965, to mention just a few of them.

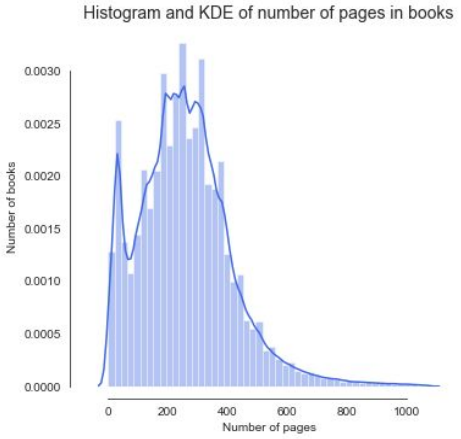

Fig. 18: Histogram of the length of books

Now, using the sample distribution previously exposed, we could graph a scatter plot of the length of books and publication year. And this time, we need to check, for the new variable studied (number of pages) if the sample randomly chosen is representative of the population. Therefore, the **null hypothesis**, in this case, is that t**he sample distribution of the number of pages belongs to the same distribution that the original population** and if we couldn't reject the hypothesis, we assert that the **sample is illustrative**.

Let's turn to the graph in Fig. 19. We detect that the range of length of books spreads out from the second half of the past century until now. Inspecting the longest books in this period, we found titles as Quixote by Cervantes published in 1605, 2004 and 2013; Ulisse by James Joyce published in 1922 and 2013. Then, if we found longer books in this century some of them could be new editions of older books, but it doesn't mean that authors are not writing long books right now. It only limits us in the sense that we can't assert that current books are longer or not. But certainly, according to the graph, long stories, anthologies, re-edition of classic books still being sold and read today.

Using a significance level of 0.05, and a Kolmogorov-Smirnov statistic on 2 samples, a p-value $p = 0.23$ was got and we fail to reject the null hypothesis. Therefore, the **sample is illustrative**.

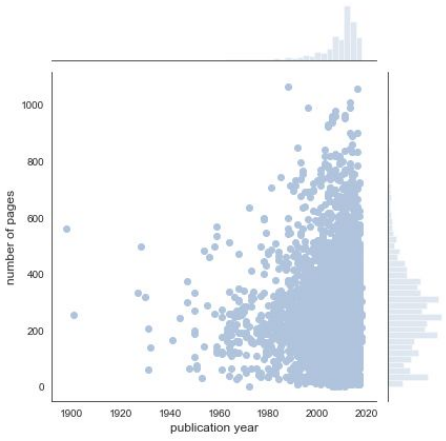

Fig. 19: Scatter plot number of pages and publication year

C. Do readers prefer the shortest or longest novels?

If you are thinking to write a book, this is an interesting question that you must be able to answer. People still love both, but certainly, we can find interesting results.

Applying **z score** criteria to the complete population of data, we could split the data points into two groups: regular and long books. Let's turn to the boxplot below, where every absolute rating is displayed per group. As we could expect, long books have higher means than regular books (around 800 vs 200 pages) and further, the mean number of pages (in regular and long books) increases according to the absolute rating. Observing the ratings 3, 4 and 5 (that means, neutral, good and excellent books on a categorical scale), it's clear that they include the books with a higher number of pages.

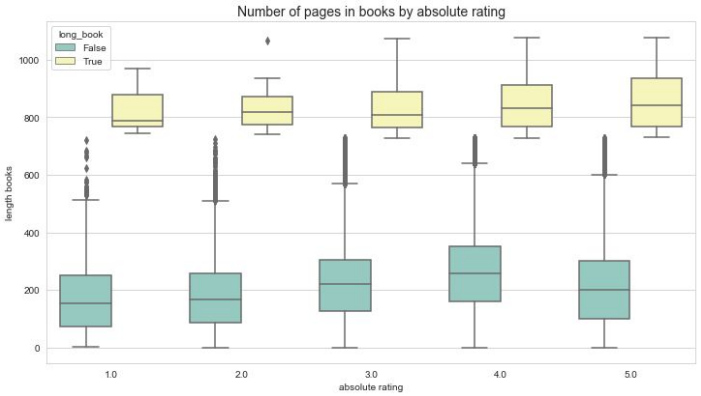

Fig. 20: Boxplot of books by absolute rating and type of book

D. What are the most popular languages in books?

We found 146 different languages in data and this is a small value thinking about there are more than 7 thousand languages in the world. Despite this variety, some of them are more frequent. To analyze them, we work with the complete population. As we can see in Fig. 21, English is the most popular language, followed by Italian, German, Spanish, French and finally, Portugal and Aragon. We use a logarithmic scale because English literature covers more than 60%. and is in a higher magnitude scale than the rest.

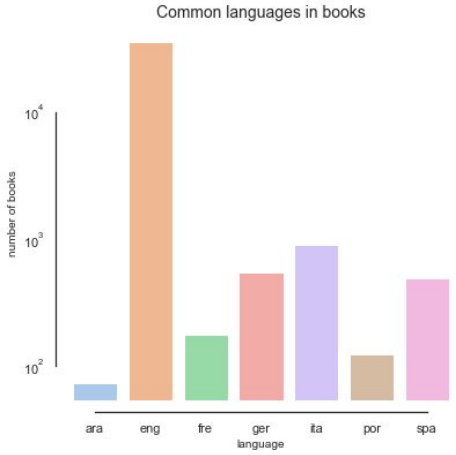

Fig. 21: Bar chart languages in books

To inspect books by language through the time and average scores, we get a sample of 20% of data. But again, how do we know that this sample is representative? We did a chi-square bootstrapping test, chosen randomly 1000 subsamples of data (with replacement) for getting the absolute frequency of the ten most popular languages and applying a chi-square test using these results as data expected and the frequency on the sample analyzed as data observed. Finally, we calculate how many times we fail to reject the null hypothesis about the **sample data of languages is representative of the population** with a confidence level of 0.05 and we get that 73% of tests fail to reject the null hypothesis.

As we mentioned early, the rating scores in books are concentrated in positive categories and, splitting them by languages, we can conclude the same. Fig. 22 displayed English books (including Canadien, British) and Romance languages books (including French, Spanish, Portuguese, Italian, Romanian, Catalan, Aragon languages). And it's interesting to note that distribution of data points is similar between categories and in both cases the population of Excellent and Good books still being higher than Fair and Poor items. Besides, this dataset, in particular, has older English books than Romance books. We highlight Romance languages because they have a particular interest in a lot of undergraduate and postgraduate programs now and certainly it’s because of their rich cultural heritage and because it’s a key to understand the multicultural societies in the world.

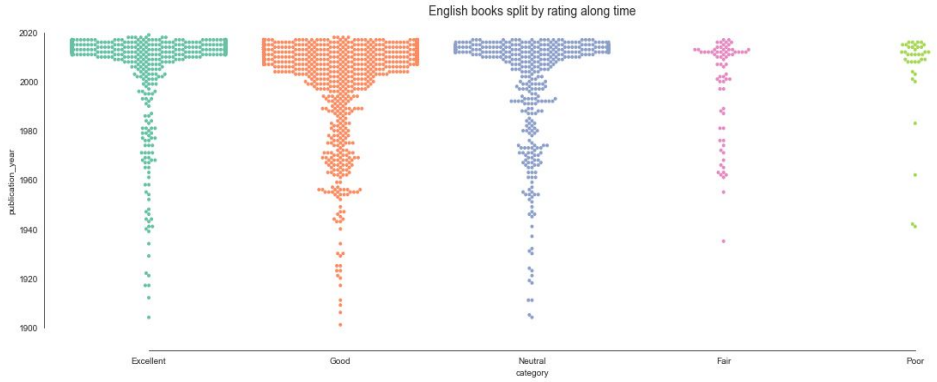

Fig. 22a: English books through time split by rating category

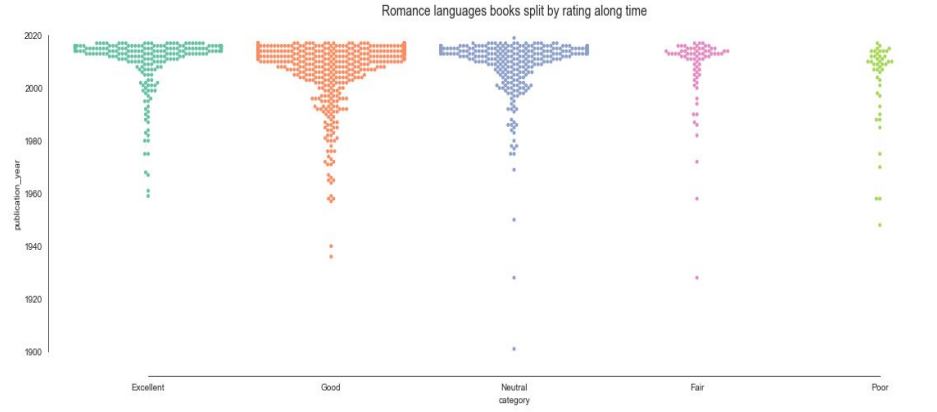

Fig. 22b: Romance books through time split by rating category

E. Evolution of publications over time

The following graph (see Fig. 23) shows the evolution of books belong to the English language, German language (except English) and Romance languages. The lasts groups have a similar growth through time. Germanic language family includes German, Swedish, Danish, Dutch, Norwegian, Afrikaans and Icelandic languages.

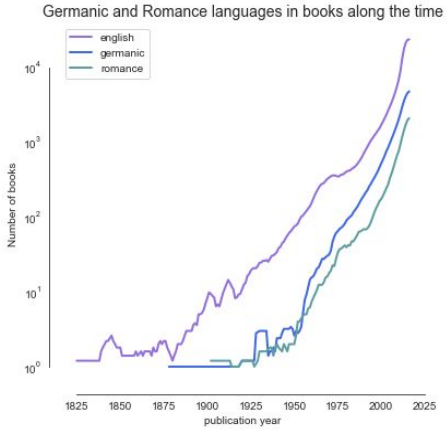

Fig. 23. Curves of growth in publications according to language families

F. Sentiment Analysis: AFINN lexicon versus TextBlob

Polarity and subjectivity patterns are some useful tools to analyze reviews. As we explained previously, we decide to try two techniques to measure the polarity of patterns: the **TexBlob** pattern analyzer and AFINN. The first one has the advantage of computing both (polarity and subjectivity pattern) of text, to study the text data from two perspectives: we determine how much positive or negative is data and additionally, the linguistic intention of the message or how much informative/descriptive or argumentative is. The second criterion only measures the polarity, but, as we'll check later, use a different and more wide-scale that allows us to split levels of positive and negative information.

AFINN patterns are around -200 and 200 scores and TextBlob between -1 and 1. We re-scaled both to put them together in the following polarity distributions graph. It seems to be that TextBlob patterns are more homogeneously distributed in positive and negative scores (see Fig. 24). Does it mean that the analyzer is better?

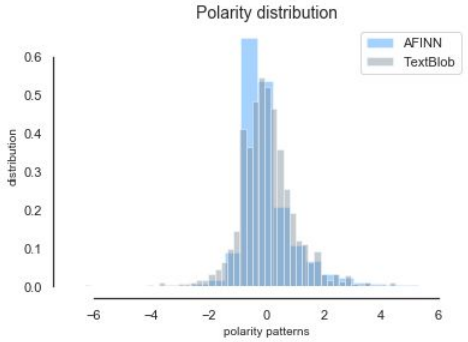

Fig. 24. Polarity distribution from AFINN and TextBlob analyzers

In Fig. 25, we identify positive, negative and neutral information. Some conclusions about these pictures are that AFINN criteria concentrate more positive messages on the highest scores and more negative messages on the opposite. Otherwise, the TextBlob pattern analyzer gets a coefficient along all the possible negative and positive scale independently of the rating score. Therefore, we'll use the polarity patterns from AFINN criteria and we'll study the subjectivity patterns measured by the analyzer of TextBlob.

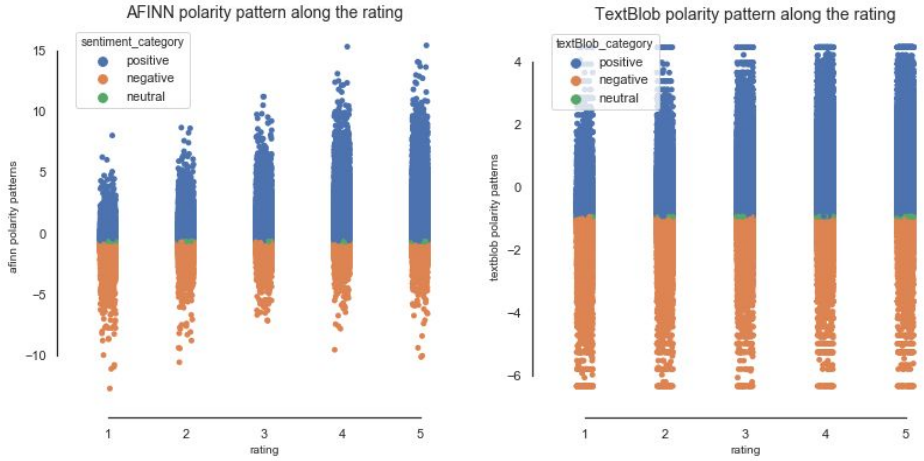

Fig. 25. Sentiment Analysis Comparative

G. Sentiment Analysis: subjectivity patterns

The polarity patterns for a sample corresponding to 1% of book reviews are displayed in Fig.26. The reviews qualified as neutral are more descriptive because they have fewer subjectivity scores (around 0) and the positive and negative reviews are in higher positions (between 0.4 and 1), which means, from mediumly to completely subjective. Furthermore, from rating 3 to 5, we see the increase of the subjectivity pattern.

Fig. 26. Subjectivity patterns from a sample of 1% data

The polarity pattern to the same sample is displayed in Fig. 27. The polarity score is ascending through the rise of the rating, as it can be expected. We check that this a representative sample of the population of reviews testing the distribution of polarity through

Kolmogorov-Smirnov statistic on 2 samples. We got a p-value $p >> 0.05$ we **fail to reject the null hypothesis** about **the sample is representative of the population.**

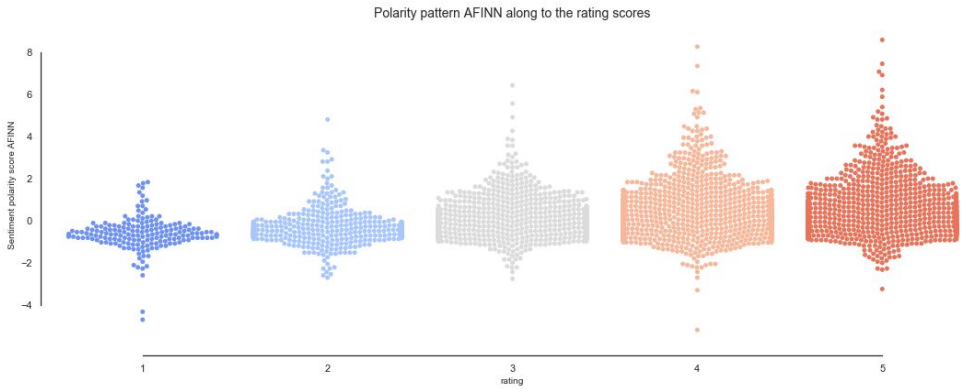

Fig. 27. Polarity patterns from a sample of 1% data

IV. In-Depth Analysis

The movie dataset contains overviews, key-words, and lists of genres. Every movie belongs to at least two genres and some movies have more than five. The book dataset offers a description of the different titles, among other features. How we could use this information to predict genres in books and create clusters of items? To resolve this challenge, we propose the following methodology:

a) Create a model to predict the movie genres using the keywords, titles, and overviews of movies.
b) Use the right genres to test and measure accuracy.
c) Apply the model to book descriptions to assign genres
d) Build feature vectors with genres to create clusters of items.

Besides, we'll build *cinephile* profiles, to find similarities between users and movie genres. To do that, we look for users with high participation in rankings or *cinephiles*.

Additionally, we' ll work a little more with books and their features like the number of reviews, pages, a number of rating counts and the new information about book genres and we' ll try to predict the average ranking. Finally, we'll inspect the description of books to get dates, places and relevant information to put them in pertinent historical contexts.

1. Predicting genres using the description of movies

As was early mentioned, every movie belongs to at least two genres. Eighteen genres were identified (*Action*, *Adventure*, *Animation*, *Comedy*, *Crime*, *Documentary, Drama*, *Family*, *Fantasy*, *Foreign*, *History*, *Horror*, *Music*, *Mystery*, *Romance*, *Thriller*, *War* and *Western*) and the possible combination of them, just considering 2 genres per movie is 153. In this scenario, we considered creating families of genres that group a couple of combinations or use a model to measure the membership of movies to each genre. The obstacles associated with

the number of combinations and the trade-off between to get enough accurate and general categories at the same time to describe genres, lead the effort toward getting coefficients of belonging to genres.

How to determine the grade of belonging of a movie to every genre?

A. Preprocessing to extract keywords

In this part, we use a different approach to the NLP preprocessing pipeline mentioned in prior sections to extract relevant words from overviews based on **part of speech** (**PoS**) tags. This arrangement will avoid using words that are not deleted as stop-words but they don't give enough meaning itself and only nouns, verbs, adjectives and cardinals (for retrieving dates that have particular importance in *History* and *Documentary* genres) are allowed. Comparing both approaches, according to the example exposed in Table 2, the length of the original text is reduced at 48% and 60% using the NLP preprocessing pipeline and part of speech respectively. Therefore, we proceed to apply the PoS filter in movie overviews.

	Text	Length
Movie Overview	*When siblings Judy and Peter discover an enchanted board game that opens the door to a magical world, they unwittingly invite Alan -- an adult who's been trapped inside the game for 26 years -- into their living room. Alan's only hope for freedom is to finish the game, which proves risky as all three find themselves running from giant rhinoceroses, evil monkeys and other terrifying creatures.*	67
Overview normalized	*sibling judy peter discover enchanted board game open door magical world unwittingly invite alan -- adult trapped inside game 26 year -- living room alans hope freedom finish game prof risky three find running giant rhinoceros evil monkey terrifying creature*	40
Overview filtered by PoS	*siblings judy peter discover board game door magical world invite alan adult game 26 years living room alan only hope freedom game risky three find giant rhinoceroses evil monkeys other terrifying creatures*	32

Table 2: Overview normalized vs filtered by Part of Speech

The original key-word list of every movie and the normalized titles are added to this new feature called *description,* as we can see in Fig. 28.

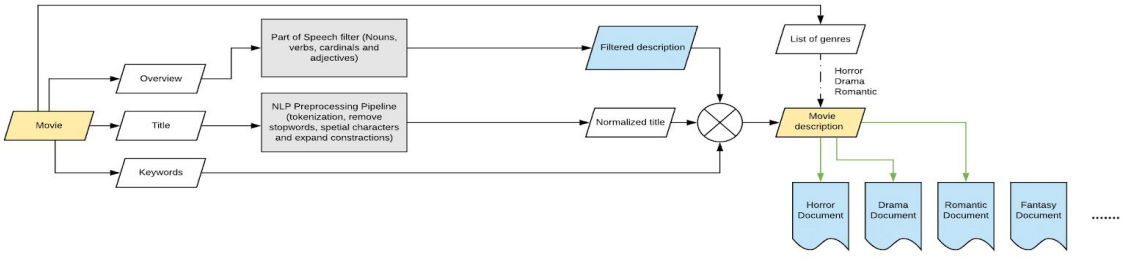

Fig. 28: Building the description of movies using filtered overviews by PoS, normalized titles and keywords

B. Documents by genres

Now, we create eighteen documents of genres and we insert every movie in the corresponding genre according to its own list. It means that if a movie belongs to *Drama* and *Romantic* genres, its description appears replicated in both documents. After to insert 80% of the movies in these documents, we inspect the number of overviews included in every genre.

According to Fig.29a (left), two genres are exceeding ten thousand overviews, *Drama* and *Comedy* and they represent **outliers**. The third highest genre is *Thriller*, with six thousand overviews and the rest are concentrated between 1000-5000. Trying to reduce this difference, we establish the maximum number of overviews as 5000 and all the genres that exceed this value are randomly sampled.

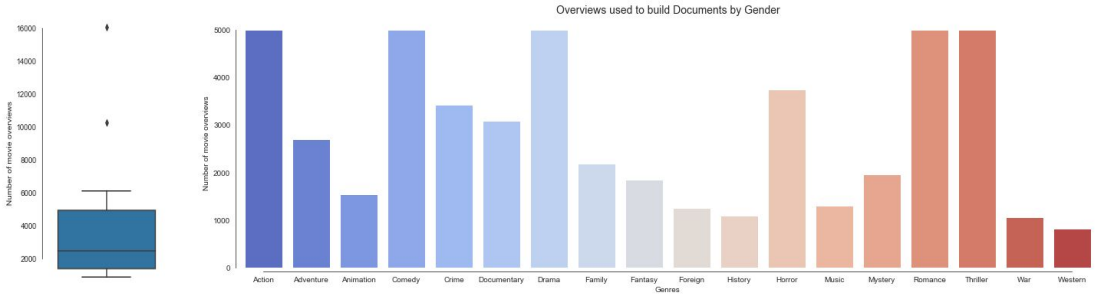

Fig. 29: (left) Boxplot of overviews used to build Documents by Genre and (right) number of overviews by genre after filtering the excess of overviews

Then, all the descriptions are aggregated, in huge corpora **by genre.** TF-IDF vectorizer is used to count words and the result is a TF-IDF matrix of size (18 × 105271) that contains every word put in documents and their normalized frequency. TF-IDF is appropriated considering that there are a lot of verbs in common between the different genres. The reason why we did not delete verbs in the part of the speech filter step is that we need some descriptive and particular verbs as *love*, *kill*, *scare* that has a strong relationship with a specific genre.

C. Cosine similarity

We are comparing a huge document, that represents a genre with descriptions of a movie in a couple of words. If every word in the document is a space, we create a vector representing the document in that multidimensional space and we measure cosine of the angle between the two vectors (documents). It means the orientation, not the magnitude. The cosine allows us to determine the similarity between both documents irrespective of their size, unlike the Euclidean distance. Fig. 30 shows the process and reveals that higher similarity coefficients are chosen as the predicted genres.

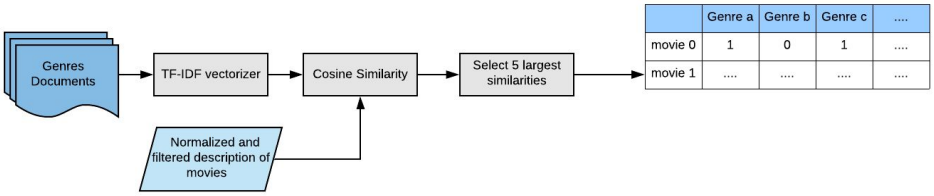

Fig.30: Cosine Similarity between Documents by Genre and movie descriptions

D. Grade of belonging of a genre

In consequence, we got a coefficient of membership in every genre. We choose the five largest coefficients as genre labels and compare these results with the right labels, defining as a success if at least one of the predicted genres belongs to the list. Using this metric, 91.81 % of the movies get success and comparing the number of genres correctly labeled with respect to the originals, the results are displayed in Fig. 31. For instance, if the original genres for a movie are three and two are correctly predicted, the accuracy is 0.666. Observing Fig. 31, we note that more than five thousand of genres are predicted with accuracies around 0.8 and 1, meanwhile, around two thousand predictions are between 0.4 and 0.8 scores.

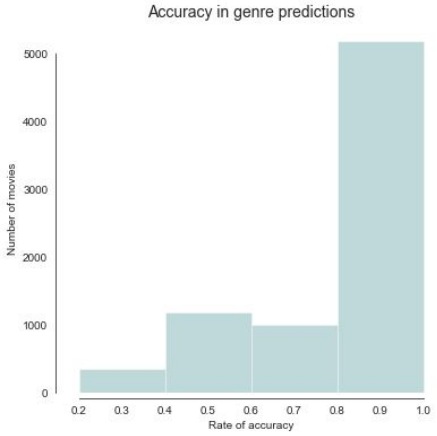

Fig. 31: Histogram of accuracy in predictions

Finally, two examples are used to illustrate fail and success in predictions. Table 3 contains the descriptions built for *The American President* and *GoldenEye* movies, original and predicted genres.

In the first case, the words ***president, washington***, **lobbyist** probably are leading the prediction towards ***History*** and ***Documentary*** instead ***Comedy*** and ***Drama***. Just using the description of the movie is hard to know the right labels. For *GoldenEye*, the words in bold are leading the prediction unto the right genres. ***Weapons*** are probably a recurrent word in the ***Western*** genre, therefore, the model suggests that the movie belongs to this genre.

Description	Genres	Predicted genres
*american **president** widower widowed u.s. president andrew shepherd one world powerful men anything sydney ellen wade **washington lobbyist** shepherd attempts spark wild rumors approval ratings (movie: The American President)*	Comedy Drama Romance	History Adventure Action Documentary War
*goldeneye cuba kgb satellite cossack **james bond mysterious** head janus syndicate leader goldeneye **weapons** system revenge britain (movie: GoldenEye)*	Adventure Action Thriller	Action Thriller Adventure Western

Table 3: Results for prediction of genres

2. Determine book genres

In this section, we calculate the cosine similarity between the documents of genres built in the previous section and the description of books. But before to do that, there are some issues to consider:

a) *Foreign* books contain descriptions in their original languages. Therefore, we are only including English books in the following varieties: Canadian, Britain, U.S, and Indian.
b) Some movie genres don't match appropriately with book genres. For instance, the *Western* or *Comedy* genres are not likely genres for books.
c) *Animation* genre is applicable to comics and manga books, but we are missing animation books excluding *Foreign* languages.

In consequence, *Animation*, *Comedy*, *Western* and *Foreign* are excluded as genres to label books. Then, the new set of documents has 14 genres and we proceed to measure the similarity between normalized and filtered descriptions (that include the normalized titles and description of books, filtered by **part of speech**, exactly as we did in the previous section). The new TF-IDF matrix has the shape (14 x 93112). In this way, every book is described as a combination of genres or **genres feature vectors.**

Adopting the same process of the earlier section, the highest scores of similarity determine the list of genres for books. Some results are exposed in Table 4.

Book title	Description (title normalized and text filtered by PoS)	Genres
Dark Matter	**dark** matter 10 hours 9 minutes author wayward pines trilogy mind-bending science-fiction **thriller** ordinary man unconscious awakens world different reality jason dessen chilly chicago one night quiet evening front fireplace wife daniela son charlie reality shatters man **mask gunpoint**...	Thriller, Drama, Mystery, Horror
Mystery on Southampton Water	**mystery** southampton water america '**crime** solent	Crime, Mystery, Thriller, Action
Sofia's Magic Lesson	sofia **magic** lesson cedric sofia apprentice **magical amulet magic**	Fantasy,

	friendship true lesson sofia cedric **tricky** situation	Family, Adventure, Drama
Agatha Raisin and the Quiche of Death	raisin 1 irascible personality **agatha** raisin heady dash curry may have serving detroit free press agatha picture-book english village swing quiche village quiche-making contest more judge hot trail **poisoner** agatha **fearless** while unaware next **victim**	Mystery, Horror, Fantasy, Thriller
Portugal's Guerrilla Wars in Africa	portugal three **wars** africa angola mozambique portuguese guinea-bissau today 13 years longer united states **army** fought vietnam are underreported **conflicts** modern era lisbon overseas **war** guerra do ultramar former **colonies** war liberation ...	War, History, Action, Documentary
Harry Potter and the Chamber of Secrets	(...) new collector edition j.k. rowling harry potter chamber secrets paint pencil pixels kay **wizarding** world have breathtaking scenes dark themes unforgettable characters dobby gilderoy lockhart await harry friends second year **hogwarts** school witchcraft wizardry **legendary** ...	Family, Adventure, Fantasy, Action
Batman: Detective Comics, Vol. 3	batman **detective** comic vol 3 league shadow batman team vigilantes gotham city next volume best-selling series batman detective comics vol 3 league shadows next big detective arc explodes league **shadows mysterious** rumor **deadly** fact two new members team azrael batwing **dark knight** squadron **crime-fighters** able **league plan** ...	Mystery, Crime, Action, Thriller
The Secret Life of a Dream Girl	secret life **dream girl** creative heart 4 dahlia keegan 1 disclaimer **teen** crush book adult language references **drinking drugs kiss** steamy ereader plain sight harder dahlia greene aka international pop **superstar** cherry undercover normal high school student real life hottie keegan matthews girl perfect opportunity real life...	Romance, Drama, Family, Fantasy

Table 4: Determining the genre of books

3. Predicting average rating in books

The following section is a more in-depth analysis of books, handling the new features about genres. We are wondering if it's possible to predict the rating of books based on the number of pages, number of ratings, text reviews counts and their belonging genres.

A. Classification approach

The average scores are transforming in absolute values and are managed as a binary classification problem according to the following categories: bad (1 and 2 scores) and good (4 and 5 scores). Fig. 32 displays the books by discrete rating. Books with score 0 are deleted (books not qualified yet) and score 3 could represent a good or bad result, depending on the perspective. In the case of 1 and 2 scores, they can be translated as not recommendable experiences; 4 and 5 are the opposite. Positive and negative qualifications are labeled as +1 and -1 respectively. The two highest scores also include huge number of books, thus, we manage a sample of 500 books from every one of these categories to balance the training data.

The genres are determined using the anterior process. A similarity matrix is constructed with all the possible genres and the similarity coefficients. Then, these values

are replaced by 0 and 1, depending on if, for a specific movie, the coefficient is among the four highest values. The rule is: assign a 1 if the coefficient is between 4 biggest or 0 otherwise.

The number of pages, number of ratings and text reviews counts are re-scaled between 0 and 1, avoiding slow or unstable learning process.

Next, a statistic chi-squared test provided by SelectKBest is applied to extract the features more strongly related to the label or output variable of the features vector. Tests measure dependence between the features and output variable and lower scores represent independence and then useless features to resolve the classification problem. The results are in Table 5. Filtering by score upper than 1, the 13 attributes selected to build the feature vector are the **number of pages**, **rating count, text reviews count** and **genres**, **except** for **Mystery**, **War**, **Thriller** and **Crime**. **History**, **Music**, **Family**, **Documentary**, and **Horror** are genres that we'll mention later in cinephiles profiles. These feature vectors are the inputs to different ensemble machine learning algorithms as Gradient Boost, Random Forest, the combinations Gradient Boost-Logistic Regression and Random Forest-Logistic Regression and XGBoost. Besides, Naive Bayes is used to comparing the performances of ensemble trees and other alternative models.

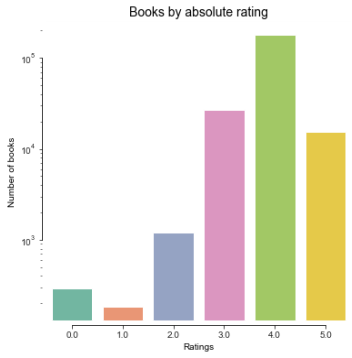

Fig. 32: Barplot of books by absolute rating

Random Forest is an extension of the bagging estimator algorithm. The base estimators, in this case, are decision trees. Random Forest selects randomly a set of features used to decide the best split at each node of the decision tree. On every stage, a new decision tree model is fitted and the final predictor is the average of the predictions from all decision trees.

Gradient Boost use boosting to convert weak learners into strong learners, through gradients in the loss function, the metric to know how good are model's coefficients are at fitting the data. It means every subsequent tree in series is built based on the prediction errors in the previous trees and the base learners are regression trees.

XGBoost offers some advantages with respect to the other ensemble models: high flexibility to define the optimization objective and evaluation criteria. Besides, reduce the overfitting through regularization terms and allow us to run cross-validation in every iteration of the boosting process (unlike Gradient Boosting).

Feature	Score	Feature	Score
number of pages	12.73	Action	2.09
rating count	9.26	Adventure	1.90
text reviews count	6.49	Drama	1.67
History	3.45	Horror	1.00
Romance	3.36	Mystery	0.47
Music	3.05	War	0.34
Fantasy	2.41	Thriller	0.13
Family	2.40	Crime	0.12
Documentary	2.15		

Table 5: Statistic test to select features strongly related to book ratings

Both ensemble models can be applied as **transformers of the features into a higher-dimensional space**, as a pre-processing step before training linear models. **Logistic Gradient Boost** and **Logistic Random Forest** fit features at tree methods and then each leaf in the ensemble is assigned an arbitrary index in a new feature space. Every leaf has its own index that is encoded using **One-Hot-Encoding**. In this way, every sample goes through the decisions of each tree and ends in one leaf per tree. These leaves are encoded to 1 an the rest to 0. This new representation of features trains a **Logistic Regression**, with new sparse, high-dimensional categorical embedding data.

We tuning three parameters of **Gradient Boosting**: learning_rate, max_features and n_estimators. For the last parameter, we decided to use the total number of features. Through GridSearch with 5 folds, the best parameters are 100 estimators and a learning rate of 0.1. On the other hand, **Random Forest** has 2 interesting parameters: the number of features and the number of trees. This time the max number of features will be the total number of them and the optimized number of trees using a GridSearch is 2bes00. For **XGBoost**, **AUC** and **Binary Error Rate** are evaluated using 25 estimators; gamma and max depth are adjusted to control the minimum loss reduction required to make a split into a node and overfitting by cross-validation. A prior, we know that **Naive Bayes** is best for categorical data than numerical (is assumed that data distributed normally), but it was included to compare the performance of different approaches.

Given these considerations, five ensemble models are fit: Random Forest, Gradient Boosting, Logistic Random Forest, Logistic Gradient Boosting and XGBoost. The **ROC curves** are displayed in Fig. 33. **AUC** and **Accuracy** are used as evaluation metrics; both, the area under the curve, AUC and Accuracy are highest applying **Logistic Gradient Boosting and XGBoost**, and the two high-dimensional ensemble trees approach overcomes performances that original ensembles. Both ensemble models are expensive computationally, hard to visualize and implement (related to KNN, for instance). The advantage of Random Forest over Gradient Boosting is less variance and reduction of overfitting (because we choose an average of all the trees implemented) and training takes less time because trees are built in

parallel, meanwhile, in Gradient Boosting, they are created sequentially. These drawbacks also mean that Gradient Boosting is more susceptible to overfitting but at the same time, Gradient Boosting has more hyperparameters for tunning the model and this sequential structure helpt to correct errors from the previous trees. As we could expect, Naive Bayes for this particular problem has the lowest performance. This classifier makes strong assumptions about the shape of distributions.

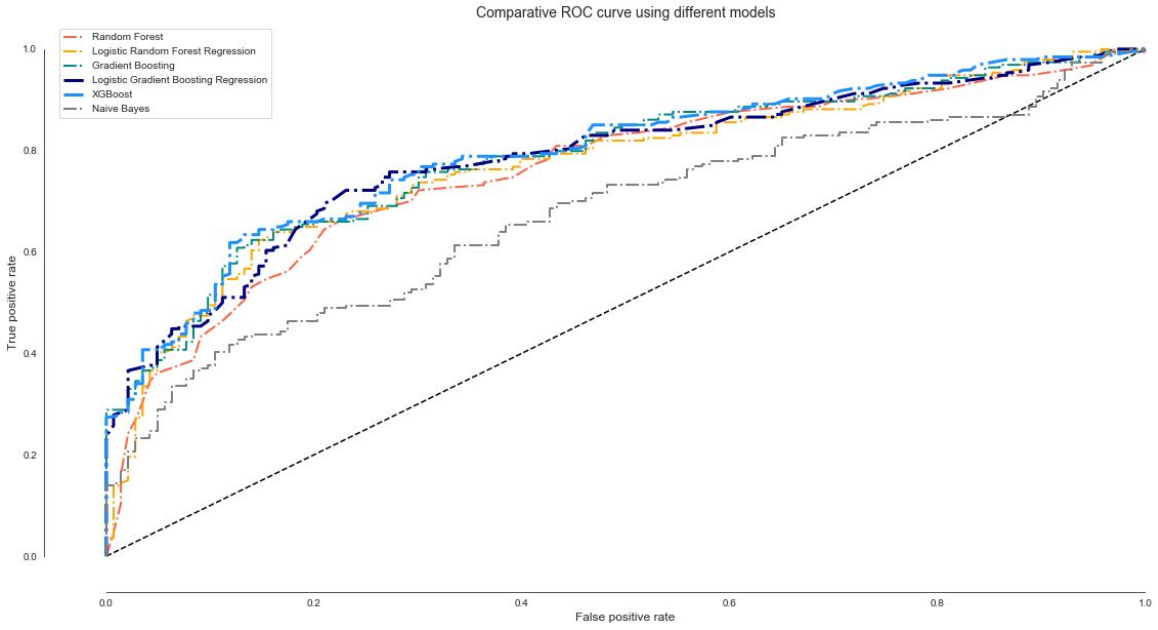

Fig. 33: ROC curve for the ensemble models combining Random Forest, Gradient Boosting and Logistic Regression

Model	AUC	Accuracy
RANDOM FOREST	0.76	71.81%
LOGISTIC RANDOM FOREST	0.77	70.62%
GRADIENT BOOSTING	0.79	71.81%
LOGISTIC GRADIENT BOOSTING	**0.79**	**74.48%**
XGBOOST	**0.80**	**73.29%**
NAIVE BAYES	0.67	52.52%

Table 6: AUC and Accuracy for the ensemble models combining Random Forest, Gradient Boosting and Logistic Regression and xgBoost

And the confusion matrix for the models with best performances are the following (Table 7).

XGBoost	Predicted Positives	Predicted Negatives
Actual Positives	97	46
Actual Negatives	44	150

Log GB	Predicted Positives	Predicted Negatives
Actual Positives	104	39
Actual Negatives	47	147

Table 7: Confusion Matrix for xgBoost and Logistic Gradient Boosting

4. Users profiles

In this section, we built *cinephiles* profiles, looking for users with high participation in rankings to predict user movie preferences based on genres through **Collaborative Filters**. Profiles are defined according to the historical information of users about their explicit rating votes. We face this through the user-based approach or UBCF.

To do that, we inspect the *ranking.csv* file. Grouping the users based on the number of evaluations done, (Fig. 34), we note that almost all the users are concentrated between 1 and around 5000 votes. We are focused on users with frequencies among percentile 75 and 95 movies voted, it means users with 120 and 400 movies voted.

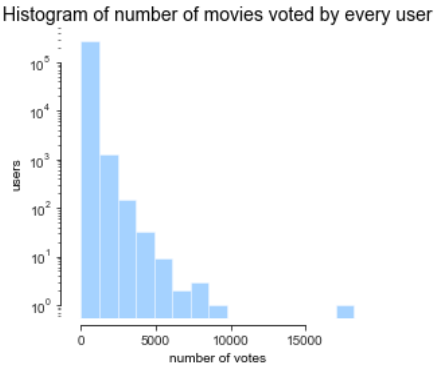

Fig. 34: Histogram of number of movies voted by every user

In Fig. 35, the preferences of two users are displayed in boxplots of scores between 1 and 5 to every genre. The user A have seen movies belonging to all the possible genres, but not enough *Foreign* movies, meanwhile in user B, *Music* movies could be more explored. In this case, we could to predict the level of acceptance of users A and B to uncover genres, through Collaborate Filters, that basically look for similar profiles. For user A, evaluations of *Drama*, *Fantasy, Comedy*, *Romance*, *Western* and *Music* genres are concentrated between 3 and 4 scores. *Family* and *Mystery* genres have better evaluations (among 4 and 5 scores) and other movies are distributed in 3, 4 and 5 scores, as *Crime*, *Thriller*, and *Action*. Additionally, evaluations of 5 scores as percentile 75 are could be found in *Crime*, *Thriller*, *Action*, *Adventure*, *Family*, and *Mystery* genres. User B, instead, concentrates punctuations between 2 and 4 scores. The genres with higher percentile 75 are *Thriller*, *Crime*, *History*, *War*, and *Animation*. *Family*, *Documentary*, *Horror*, and *Western* are not his favorites and the only genres with 5 scores as percentile 100 are *Action*, *Thriller*, *Drama*, *Comedy*, *Crime*, and *History*.

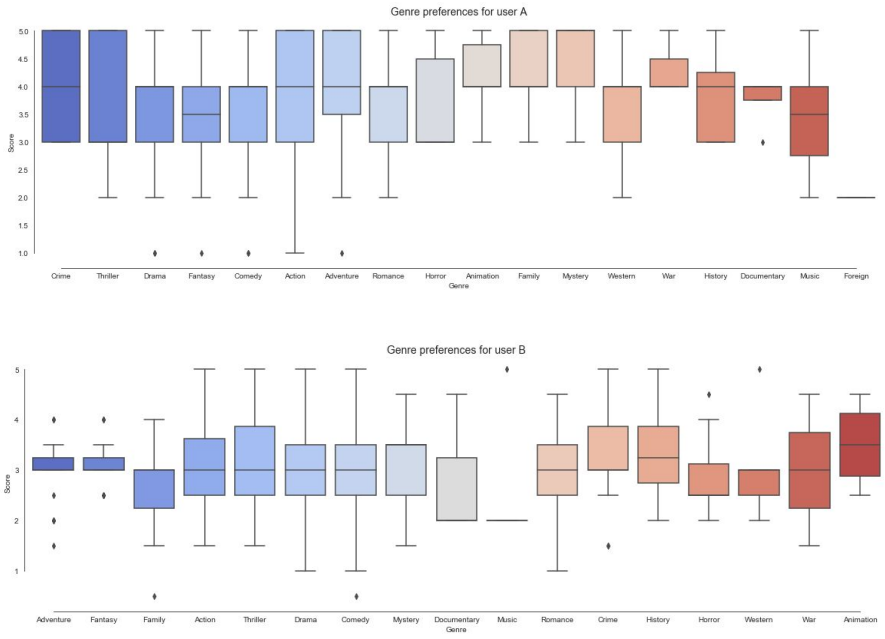

Fig. 35: User profiles: (up) User A, (down) User B.in these boxplots we can see how many explorations had had every user and his preferences

Using a sample of a thousand of *cinephiles*, we built a general profile of genres, according to the average rating of the movies belonging to every genre (see Fig. 36). Based on the mean of every genre in this general profile, the genres with more possibilities to get remarkable scores are *History*, *Documentary*, *Mystery*, *Family*, *Music*, *Drama, Crime* and *Horror*. Some of them are top of the list of best features to predict the scores in movies as we discussed in section 3.

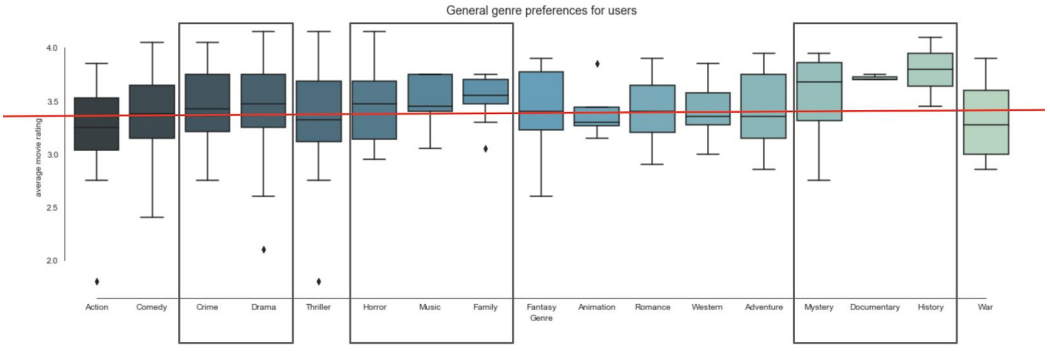

Fig. 36: Boxplots of the average score of movies belonging to different genres

To following, a function to calculate a new rating of genres based on the average score and the number of movies belonging to the genre is required, to correct the average scores of genres measure as the mean of all the movie scores in every genre. In other words, a genre integrated by just one movie with a high score and another genre with one hundred movies with the same high score can't be considered as the same. To do that, [1] was used as a reference to generate a scale between 0 and 5:

$$score = \frac{5p}{10} + 5(1 - e^{-\frac{q}{Q}})$$

where p, in this context is the average score for the genre and q is the number of movies in the genre. Q expresses the importance of the *number* of items. In consequences, we get, for a user C, the following results:

genre	average score	count	corrected score
Drama	3.16	100	3.93
Foreign	4.00	2	2.06
Comedy	3.04	57	3.04

Table 8: Examples of rankings to correct scores based on the number of movies

The distribution of corrected genre scores is moderately positive skewed (skewness 0.78) and mesokurtic (kurtosis 1.47), as we can observe in Fig. 37.

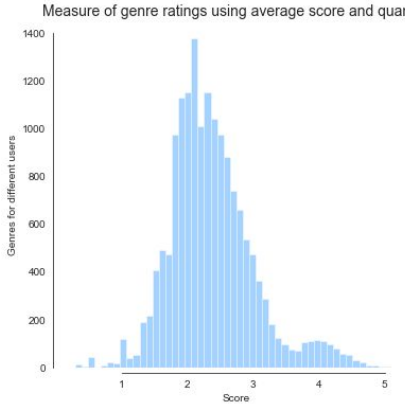

Fig. 37: Distribution of corrected genre scores

Adopting the corrected genre scores and an unsupervised **K-Nearest Neighbors** model with cosine similarity as the distance metric, we can find the most similar users in the neighborhood K. For instances, Fig. 38 displays the scores per genre for a user and his four most similar neighbors:

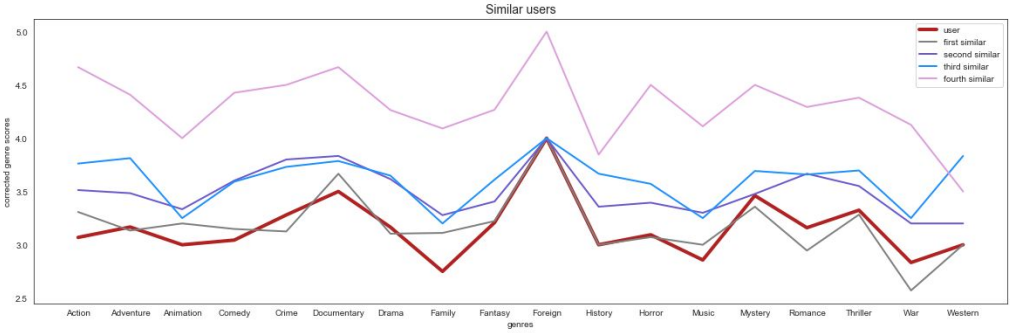

Fig. 38: Four similar users found by KNN

User Based Collaborative Filtering, UBCF is a memory based approach to resolve the recommendation problem and uses user rating data to calculate the similarity between different users. Given $\overline{r_u}$, the average rating of user u for all the items scored by u; $\omega_{u,v}$, the coefficient of similarity between the user u and v; $r_{v,i}$ and $\overline{r_v}$ the rating of user v for item i and the average rating of user v for all the items scored, predictor for user u and item i is calculated as the mean rating of u, plus a weighted average of deviations from neighbor's mean:

$$p_{u,i} = \overline{r_u} + \frac{\sum_{v \in K} (r_{v,i} - \overline{r_v}) \times \omega_{u,v}}{\sum_{v \in K} \omega_{u,v}}$$

where K represents the neighborhood of similar users. How does it work the UBCF to predict the complete profile of a user? In the following example, we iterate over the algorithm asking for different genre scores (because every prediction is about one pair user-item). Given four neighbors, the MSE for this experiment was 3.49% (Fig. 39)

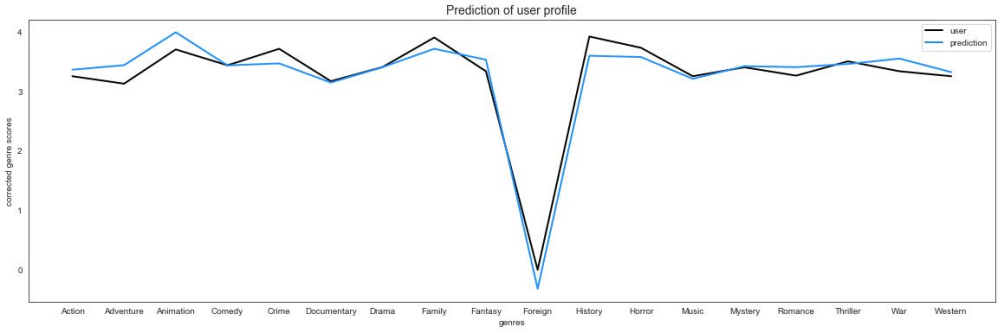

Fig. 39: Prediction of the genre scores through UBCF algorithm

5. Retrieve places and dates to put books in historical contexts

Coming back to books, to discover how powerful is ***spacy*** to get dates, places and other keywords to establish the context or create a general idea about a document. The model ***en_core_web_sm*** is used in this section. With a precision of 96.78% for Part of Speech, this English multi-tasking model assign context-specific token vectors, PoS tags, dependency parse, and named entities. This NLP model is applied to the original description of books and we recover some keywords as we can see below:

For the title ***Under Drake's Flag: A Tale of the Spanish Main***, *England*, and *Spain* are identified as **GPE**, *Ned Hearne* and *Francis Drake* are **PERSON** and the story happens in the sixteenth-century (**DATE**).

The struggle between England GPE and Spain GPE for supremacy of the high seas, as seen through the eyes of a sixteenth-century DATE teenager, Ned Hearne PERSON . Along with three CARDINAL friends, young Ned PERSON is swept up in one adventure after another as he accompanies the daring English LANGUAGE mariner Francis Drake PERSON on amazing voyages of discovery across the Pacific LOC . An eyewitness to the great naval battle between the English LANGUAGE fleet and the Spanish NORP Armada, Ned PERSON has firsthand views of England GPE 's rise as the world's most powerful sea-going nation.

What about ***The Young Hitler I Knew***? The story takes place in *Vienna* (**GPE**) in *1904* (**DATE**) and it is about *Adolf Hitler* (**PERSON**)

August DATE Kubizek met Adolf Hitler PERSON in 1904 DATE and over the next four years DATE they became close friends, eventually sharing a flat together in Vienna GPE . This book tells the story of their extraordinary friendship, and gives fascinating insight into Hitler PERSON 's character during these formative years DATE .

Remembering the Brontë sisters, ***The Secret Adventures of Charlotte Brontë***, we extract context through: *Victorian England* (**LOC**), *the British Empire* (**GPE**), *the legendary 19th century* (**DATE**) and the story is about the *Brontes*, especially *Charlotte Bronte* and *Rochester* (**PERSON**).

Laura Joh Rowland's PERSON San Ichiro GPE novels have enthralled thousands CARDINAL of readers. Now the author turns her gifts for historical fiction to Victorian England LOC and the famous and fascinating Bronte PERSON family.
THE SECRET ADVENTURES OF CHARLOTTE BRONTE, by Laura Joh Rowland PERSON (author of the Sano Ichiro mystery series) is an epic, world-at-stake thriller starring the legendary 19th century DATE author and her equally famous family. It's a tour of Victorian England LOC from gutter to palace, featuring a hero who combines Mr. Rochester PERSON with Agent 007 CARDINAL and a villain whose devious schemes threaten the very fabric of the British Empire GPE . Charlotte Bronte PERSON is plunged headlong into the sort of thrilling adventures and passionate romance she never actually experienced, but secretly craved. ( )

More examples could be found in *section4_extracting_names_places_books.ipynb*.

Finally, we use the book sample built before to create classes of books based on genres. Considering that we are managing 14 features per book, we analyze the variance between the variables. Plotting the Cumulative Summation of the Explained Variance, we note that with 12 features we get a variance of 0.98.

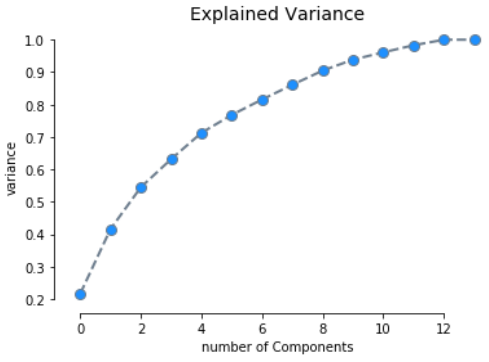

Fig. 40: Cumulative Summation of the Explained Variance

Thus, dimensionality is reduced by PCA selecting 12 components and creating a new representation of the features in other dimensional spaces. This step, followed by an **Agglomerative Clustering** for 9 clusters using **Euclidean distance**, lead us to classify books according to the classes mentioned in Fig. 41. Dominant genres make reference to the

number of books in every category. The two (or three, in case of duplicate values) higher population of books decide the genres that represent the cluster. Some titles, with words in bold, are exposed to prove the similarity of books into every cluster.

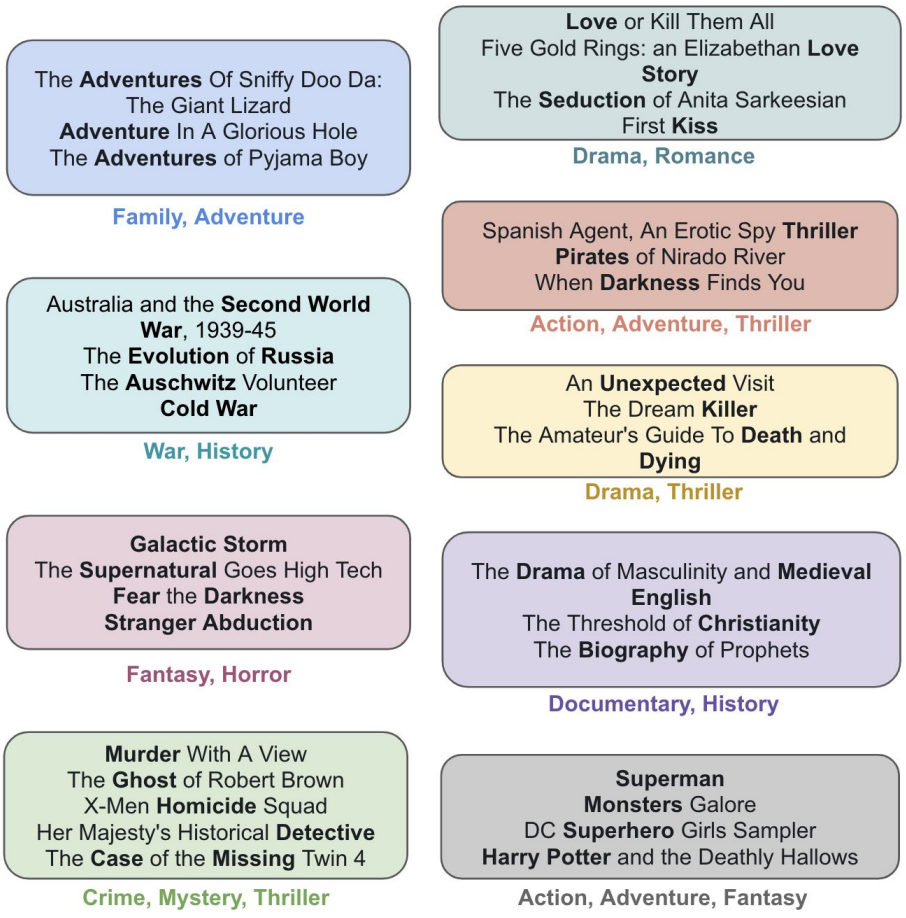

Fig. 41: Results of Agglomerative Clustering for books

V. Conclusions

In this project, movies and books are treated as two different worlds connected by fiction genres. Two worlds where English is one of the most popular languages (in production of movies and publication or translation of books) and where long productions have a good reception because they are pieces of art and history with valuable cultural meaning.

The **average runtime** of movies has not suffered considerable changes over the last decades, but we must **split regular** and l**ong movies** and analyze them separately. In this way, percentiles 25 and 75 to regular movies are 86 and 107 minutes and for longest movies, 230 and 350 minutes. Some of the longest films with historical contents are Empire, War, and Peace, based on Leo Tolstoy's equally long novel, Hitler: A Film from Germany, just to mention some of them.

ANOVA statistic tests are used to compare the distribution of movies by release/year for the most popular languages. **French and German distributions have the same population mean**, and the release of English movies has had a different and more speed up evolution. The same test for Japanese, German and Spanish films **fails to reject the null hypothesis of the same population mean.**

The genres more recurrent in movies are *Drama* and *Comedy* followed by *Action*, *Horror, Thriller*, and *Romance.* Levene statistic tests were applied to the distributions of vote averages based on genres, concluding that distributions of *Drama*, *Comedy* and *Romance* films **come from populations with equal variances.** As we discovered later, they could be perfectly part of the same cluster in the world of books.

The majority of analysis with books required sampling data because the original dataset has more than 550 thousand titles. We applied the **Kolmogorov-Smirnov** statistic test and bootstrapping tests to assert that the samples were representative of the population. Regarding the **length of books,** people still love long and short books, but the mean number of pages (in **regular** and **long** books, distinguished applying **z-score** to split them) increases according to the rating.

AFINN and **TextBlob** are applied to get sentiment patterns of readers' reviews. **AFINN** criteria concentrated more positive messages on the highest scores and more negative messages on the opposite, getting a **more suitable distribution of scores** than the TextBlob pattern analyzer.

Then, we use **overviews**, **titles**, and **keywords** of movies to **predict genres**, through **cosine similarity** between documents. The documents by genre included normalized titles (NLP preprocessing pipeline for deleting **stop-words**, **expanding contractions**, removing **special characters**, **tokenization,** and **lemmatization**), overviews filtered by **Part of Speech** and keywords. We validate the dictionary created using 80% of movies and testing with the rest of the films. 91.81% of trials at least one genre was successfully predicted, and the rate of prediction in almost all the cases was [0.4, 1].

The dictionary was used to label books for including more features to predict the rating of books based on the number of pages, the number of ratings, text reviews counts and their belonging genres. Managed as a binary classification problem, a **statistic chi-squared** test provided by SelectKBest allowed us to discharge the four less useful features to resolve the problem (the genres *Mystery*, *War, Thriller,* and *Crime)*, meanwhile *History*, *Romance* and *Music* played more relevant roles to predict the ratings. The ensemble models **Random Forest**, **Gradient Boost** and variations of them, **Logistic Gradient Boost** and **Logistic Random Forest** were applied to compare their performance by **ROC curve**, **AUC** and **Accuracy**. Aditionnaly, we include xgBoost and naive Bayes, to analyse the different performances. The variations models got better results for Gradient Boosting and similar results in Random Forest. **Logistic Gradient Boosting and xgBoost** achieved the highest scores in Accuracy, with 74.48% and 73.29% respectively.

User-Based Collaborative Filtering was used to resolve the problem of predict the vote of one user for an unexplored genre. We compared the right and predicted scores from

one user, getting a **mean squared error** of 3.49%. The feature inserted in every genre was calculated as a **corrected rating vote**, a mathematical formula that considers the votes of the user to the movies belonging to every genre and the number of movies voted per genre.

The NLP library ***spacy*** was applied to book descriptions for extracting dates, places and other keywords to establish the context or create a general idea about a document. And finally, the book feature vectors (containing genres) were reduced by **PCA** inspecting the **Cumulative Summation of the Explained Variance,** selecting 12 components and creating new representations of the features in other dimensional spaces. **Agglomerative Clustering** for 9 clusters using **Euclidean distance** applied to these feature vectors gave us the book clusters detailed in Fig. 41.

What books could be interesting purposes for cinematographic adaptations? Based on the general profile for a sample of one thousand *cinephiles*, the genres with more possibilities to get remarkable scores are *Comedy*, *Crime*, *Drama*, *Thriller*, *Horror*, and *History*. In books (Fig. 41), these genres are represented by clusters 3, 4, 6 and 8.

VI. References

1. Movies Dataset: Metadata on over 45,000 movies. 26 million ratings from over 270,000 users.Available in: movies_dataset
2. Goodreads Metadata of books. Available in: Goodreads
3. Understanding Gradient Boosting Machines. Available here
4. Named Entity Recognition with NLTK and Spacy. Available here
5. Collaborative Filtering Based Recommendation Systems Exemplified. Available here
6. Text Similarities: Estimate the degree of similarity between two texts. Available here
7. Visualize dependencies and entities in your browser or in a notebook. Available here
8. How to use Data Scaling Improve Deep Learning Model Stability and Performance here
9. Traditional Methods for Text Data (Dipanjan Sarkar). Towards Data Science. Available here
10. Practical Statistics for Data Scientist (Peter Bruce and Andrew Bruce). O'REILLY, 2017.
11. A Practitioner's Guide to Natural Language Processing (Part I) - Processing and Understanding Text (Dipanjan Sarkar). Towards Data Science. Available here