TELECOM Paris
IP PARIS

LAURYNE MOYSE
FARAH JABRI

SENTIMENT ANALYSIS ON IMDB

MODS 207

SUMMARY

INTRODUCTION

In the era of digitalization and the ever-expanding influence of social media, movie reviews have become an essential aspect of the film industry. Audiences often rely on these reviews to make informed decisions about which movies to watch. While movie reviews offer valuable insights into the quality and appeal of a film, the impact of prestigious accolades, such as Oscar nominations and awards, on the sentiments expressed in reviews remains an intriguing subject of investigation. This project aims to delve into this phenomenon by analyzing movie reviews before and after their nomination to the ceremony of Awards.

The purpose of this project is to explore how Oscar nominations and awards influence the sentiment expressed in movie reviews, shedding light on any discernible patterns or changes in audience perception. By employing sentiment analysis techniques on a dataset of movie reviews scraped from IMDb, one of the most prominent online databases for film enthusiasts, we seek to understand how Oscar nominations and awards impact the opinions and feelings expressed by the audience.

The first objective of this project is to collect and clean data on Oscar nominations and awards. We performed web scraping to extract movie reviews from IMDb from 1995 to 2019. This data will serve as a foundation for our analysis. Additionally, we performed data cleaning processes to ensure the reliability and consistency of the collected information.

In pursuit of a detailed analysis, we will focus on films that were either nominated for an Oscar or emerged victorious in the major award categories, commonly referred to as the "big 5." By compiling a substantial collection of reviews, we will have a rich corpus of textual data for the sentiment analysis.

To gain deeper insights into the sentiments expressed in movie reviews, we will employ Natural Language Processing (NLP) techniques. Our objective is to conduct an analysis of the scraped reviews, utilizing sentiment analysis techniques to understand and specify the overall sentiment and assess the content of the reviews both before and after the movies' Oscar recognition.

Building upon the results of our sentiment analysis, our objective is to investigate and compare how movie reviews change before and after a movie's association with Oscar nominations or wins. By analyzing the sentiment shifts, we aim to uncover any discernible patterns and examine the potential impact of critical recognition on audience perception.

Ultimately, this project contributes to a better understanding of how awards shape audience opinions and their role in the film industry.

BACKGROUND INFORMATION

IMDb, which stands for Internet Movie Database, is an online platform that offers a great deal of information regarding films, actors, directors, scriptwriters, and all the people and companies involved in the creation of movies, TV films, TV series, and also video games. One of the most important feature of IMDb is its collection of filmography of films of all types.

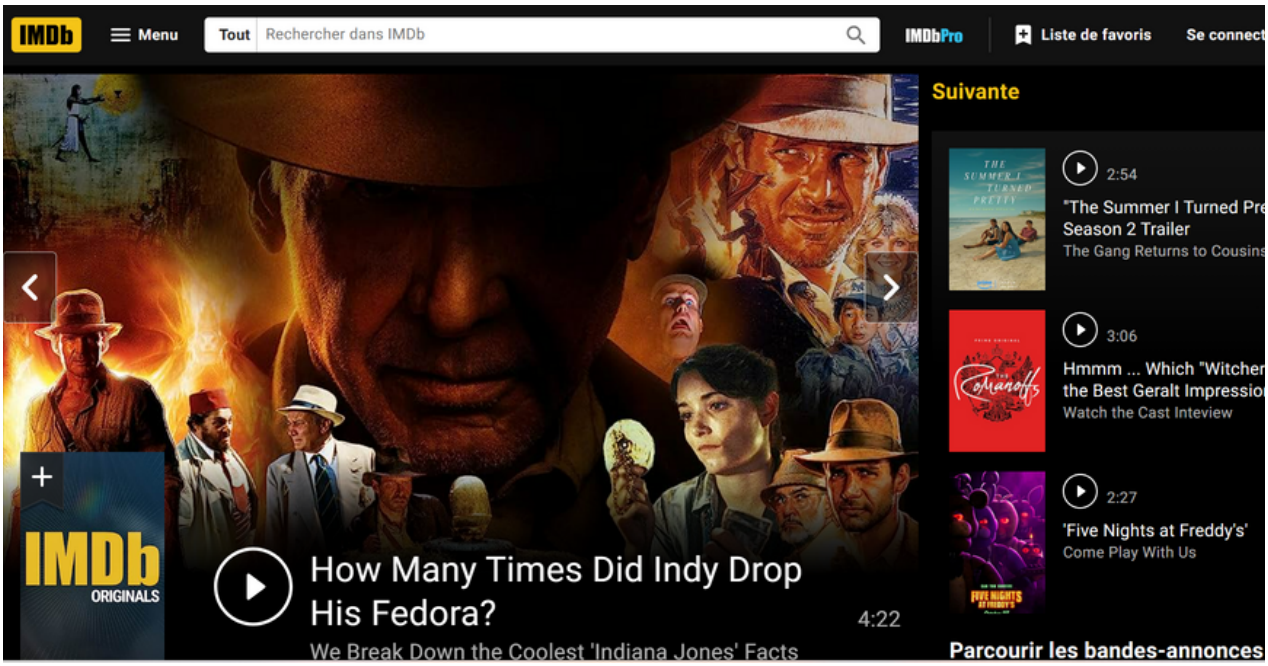

Figure 1-IMDb's welcome page

This platform allows users to access comprehensive information about movies including their titles, release dates, genres, plot summaries, production companies, and box office performance. IMDb also provides the ratings and reviews for each films from users and critics and this offers insights into the appreciation of a certain film.

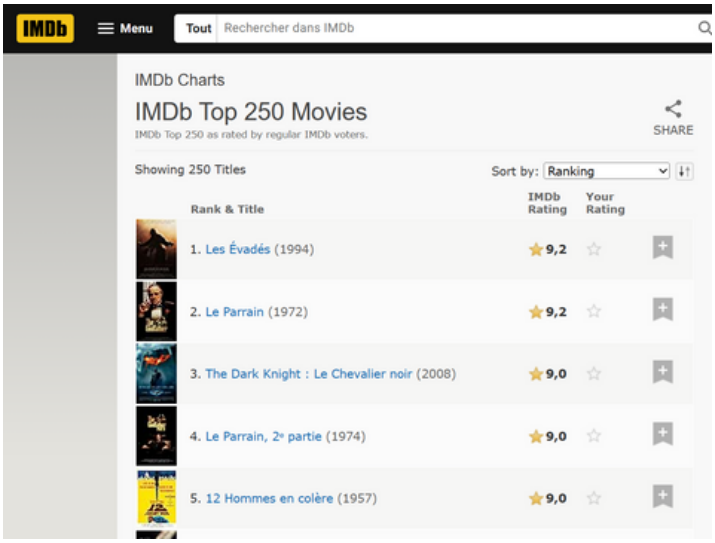

Figure 2-IMDb's top movies page

Users can find detailed profiles of these professionals, which include biographical information, filmographies, awards and nominations.

Users also have access to the profiles of other users. These user profiles show the number of films rated by the user over the years, how long the imdb member has been a member and the most recently rated films or series.

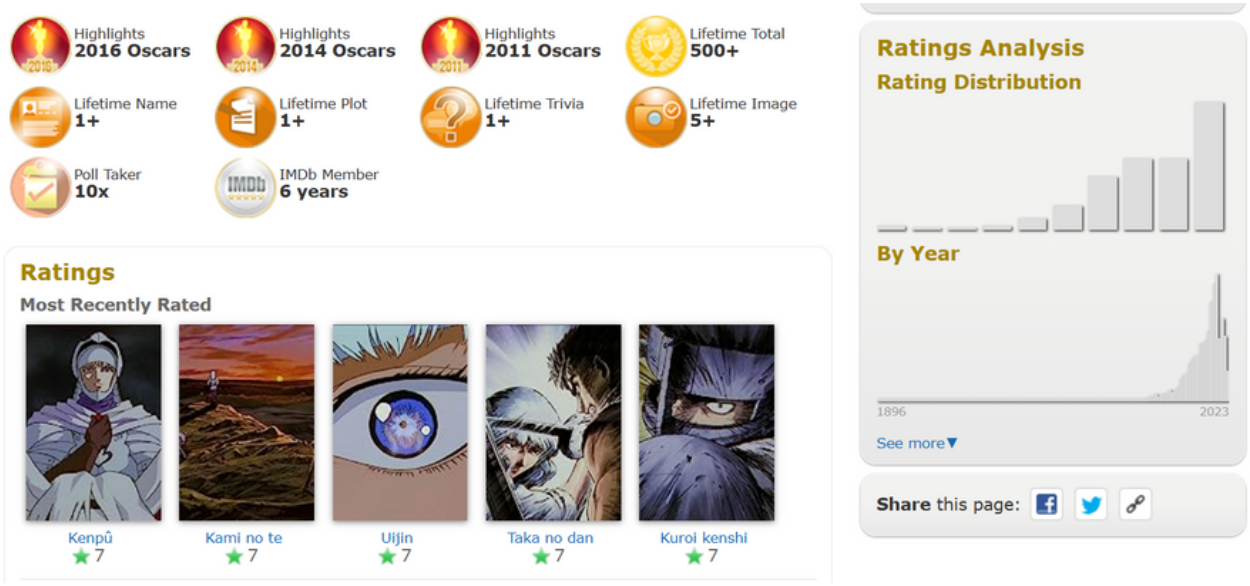

Figure 3-IMDb user profile

After examining the platform, we chose to analyse and scrape various features in this project. Firstly, all the data relating to the film: title, link/url, nomination date and award date if there is one. Secondly, the reviews associated with this film, for each review we will analyse: the rating given, the title, the body text of the review, the date of the review (if it was made before/after any nominations/awards).

We weren't able to use the information provided by the users during the scrapping because the html format of the page was different from one user to another, which made parsing difficult and also led to a large number of missing values.

Figure 4-Structure of a review

CONSTRUCTION OF THE DATABASE

For this project, a CSV file containing information on movies nominated for an Oscar and those that won awards in the major categories from 1995 to 2019 has been provided. This first file allowed us to continue the scrapping. The IDs of the movies were given, we were therefore able to reconstitute the URL of each movie which followed the same structure and to scrap the reviews.

To obtain the necessary movie reviews, we initially decided to use the Python's library Beautiful Soup. However, we encountered an issue: for one movie, all the reviews are not displayed on the screen and it is necessary to click several times on a button 'Load More' to obtain them all. The techniques used in Python to solve this problem not having been successful, we therefore opted for a code in Javascript.

Javascript being an asynchronous language allowing to send requests and to wait for their resolution, its use was much more adapted in our case. We noticed that the reviews were displayed in batches of 25, so all we had to do was retrieve the total number of reviews per film and click the necessary number of times to display them all. With Javascript, we were able to retrieve the HTLM of all the movie pages. Then, we came back to Python and used Beautiful Soup's parsing capabilities to programmatically navigate the HTML files we build and extract the desired review data. This involved capturing the review content, associated ratings, reviewer name, the title of the reviews. The use of Beautiful Soup ensured an efficient and structured approach to web scraping, facilitating the extraction of valuable textual data for subsequent analysis.

After the scraping of the website and the parsing of the html, we were able to constitute an organized database with all the desired information that we concatenated with the first database provided.

CLEANING OF THE DATABASE

They were many inconsistencies in the database that we first obtained. It was necessary to harmonize it as well as putting all the data in an exploitable shape. This step mainly implied putting the dates in a datetime format in order to be able to manipulate them. Moreover, most of the information we parsed with BeautifulSoup were stocked in our CSV file in the form of one tremendous string. So it was necessary to find the right delimitations and to reshape them in a form of lists of strings or lists of int.

An important part of this step was the adding of the column **'before_after'**. The goal of our study is to compare the sentiment expressed in the reviews on IMDb before and after the nomination/reception of awards by the movies. Therefore, for each review, we add a label belonging to {-2; -1; 0; 1; 2} indicating if it was posted respectively more than two months before the nominations to awards; two months before the nominations to awards; between the date of nomination and the date of reception of awards; two months after the reception of the awards; and more than two months after the reception of the awards.

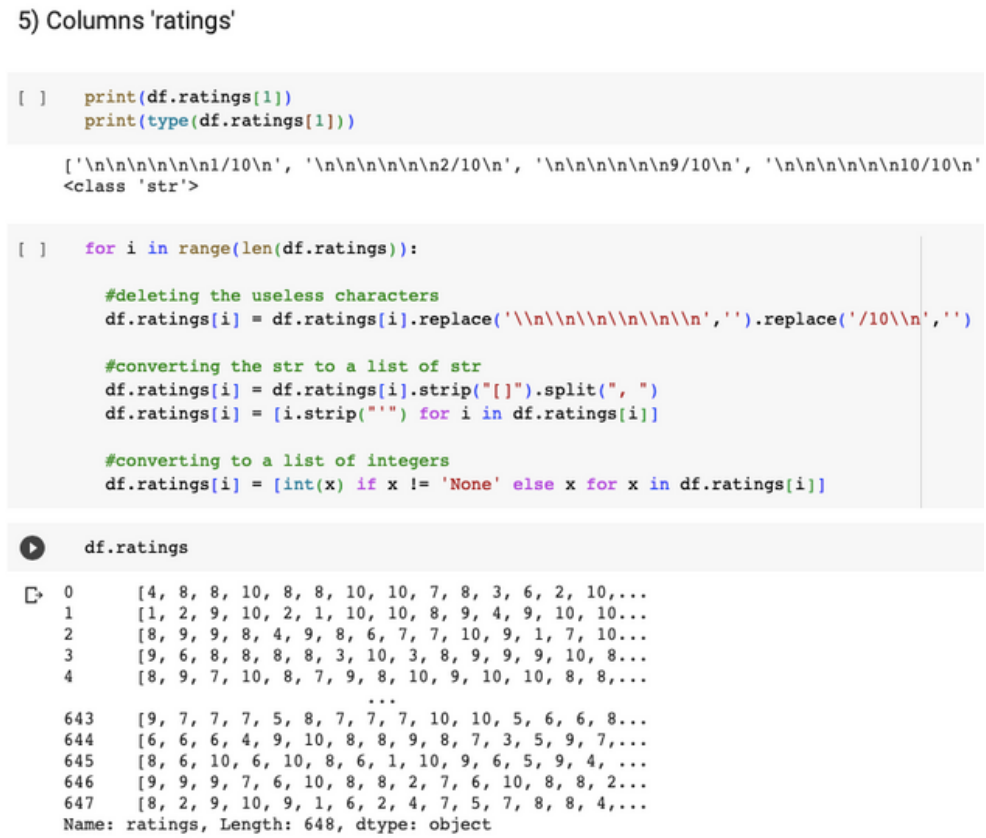

Figure 5- Example of a cleaning task

DATA DESCRIPTION AND FIRST ANALYSIS

1) Final structure of the database

Name	Type	Description
year	int	The year of the movie.
movie_title	string	The title of the movie.
movie_id	string	The id of the movie on IMDb.
date_nomination	datetime.date	The nomination date of the movie to the awards.
date_award	datetime.date	The date of the awards ceremony were the movie was nominated.
nom_actor	int	The movie was nominated in this categorie (0=False, 1=True).
nom_actress	int	The movie was nominated in this categorie (0=False, 1=True).
nom_anime	int	The movie was nominated in this categorie (0=False, 1=True).
nom_foreign	int	The movie was nominated in this categorie (0=False, 1=True).
nom_direct	int	The movie was nominated in this categorie (0=False, 1=True).
nom_doc	float	The movie was nominated in this categorie (0=False, 1=True).
nom_pict	int	The movie was nominated in this categorie (0=False, 1=True).
nom_screen	int	The movie was nominated in this categorie (0=False, 1=True).
tot_nom	int	Total number of nominations of this movie.
award_actor	int	The movie received an award in this categorie (0=False, 1=True).
award_actress	int	The movie received an award in this categorie (0=False, 1=True).
award_anime	int	The movie received an award in this categorie (0=False, 1=True).
award_foreign	int	The movie received an award in this categorie (0=False, 1=True).
award_direct	int	The movie received an award in this categorie (0=False, 1=True).
award_doc	int	The movie received an award in this categorie (0=False, 1=True).
award_pict	int	The movie received an award in this categorie (0=False, 1=True).
award_screen	int	The movie received an award in this categorie (0=False, 1=True).
tot_award	int	Total number of awards received by the movie.
total	int	Total of awards and nominations.
url	str	Url of the movie's revie page on IMDb.
nb_reviews	float	Number of reviews for the movie on IMDb.
ratings	list of floats	Ratings of the reviews.
dates	list of datetime.date	Dates of the reviews.
authors	list of strings	Authors' names of the reviews.
author_links	list of strings	Links towards authors' profile page.
review_titles	list of strings	Titles of the reviews.
texts	list of strings	Texts of the reviews.
spoilers_id	list of ints	Index of the movie reviews that contains spoiler.
before_after	list of ints	Period were the review was published (-2=before; -1=two months before nomination; 0=between nomination and awards, 1=two months after awards; 2=after.

Figure 5-Structure of the data base

2) Visualization of the number of reviews before, during and after the awards

The feature added "before_after" enabled us to evaluate, firstly, the change in the number of reviews as a function of the period. The graph below shows the evolution of the number of reviews as a function of the period, {-2,-1,0,1,2} as explained above.

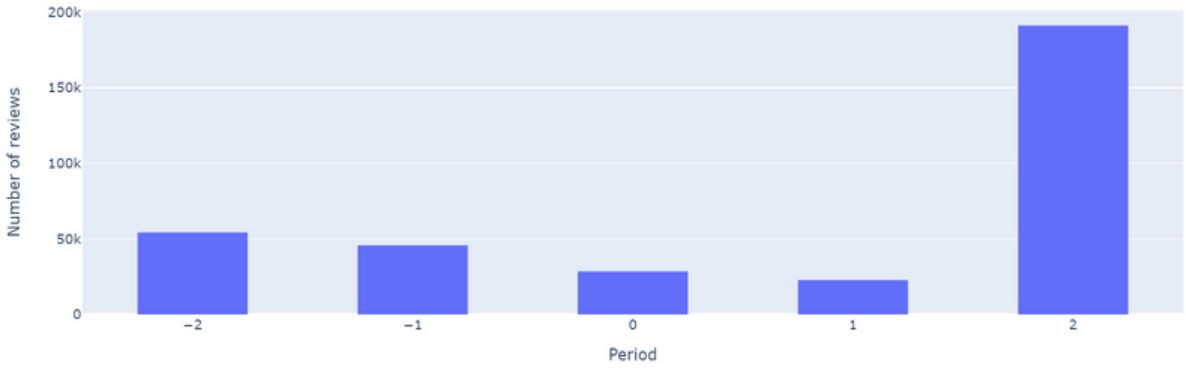

Figure 6-Number of reviews according to the period

As period 0 (between the date of nomination and the date of reception of awards) is very short, we can see that there is an increase in reviews during this period, which is fairly consistent.
To better visualize this, here is a serie displaying the number of days between the date of the nomination and the date of the award.

	0	1	2	3	4	5	6	7	8	9	...	638	639	640	641	642	643	644	645	646	647
0	41 days	41 days	41 days	41 days	41 days	41 days	41 days	41 days	41 days	41 days	...	33 days	33 days	33 days	33 days	33 days	33 days	33 days	33 days	33 days	33 days

1 rows × 648 columns

Figure 6- Number of days between awards and nomination

As a conclusion for this part we can assume that there are about half as many reviews in the period following the awards as in the period before.
There are also significantly more reviews in the 'during' period, whereas this is generally around one month long, i.e. twice as short as 'after' period. We can therefore assume that the fact that a film is nominated for an award stimulates criticism.

3) Visualization of the average rating reviews before, during and after the awards

We decided to carry out a first approach of the problem without sentiment analysis, but by performing a comparison of the different periods through the feature 'ratings'. This features is the evaluation that a user gave to a movie, along with the review posted.
First, we analyzed the average ratings gave by the users (average for all the movies of the database).

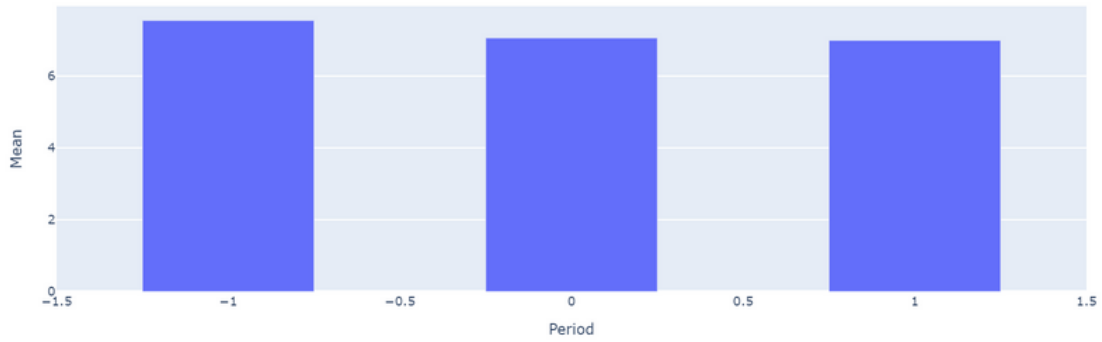

Figure 7- Average rating in function of the period

The averages are quite close but the value decreases over time. The average rating disminish of 0.55 points between before and after the awards. So it seems that once a film is nominated for the awards, the ratings are more severe.

4) Visualization of the bad ratings reviews before, during and after the awards

To confirm our precedent conclusion, we studied the ratings given to the reviews in a slightly more precise way. We wrote a function that returns, for a given period, the percentage of bad ratings obtained (we consider that a rating is bad below a certain threshold set initially).
Here, we considered as a bad review a review that has a rating stricly lower than 6. Then, we plotted the results in order to visualize these percentages.

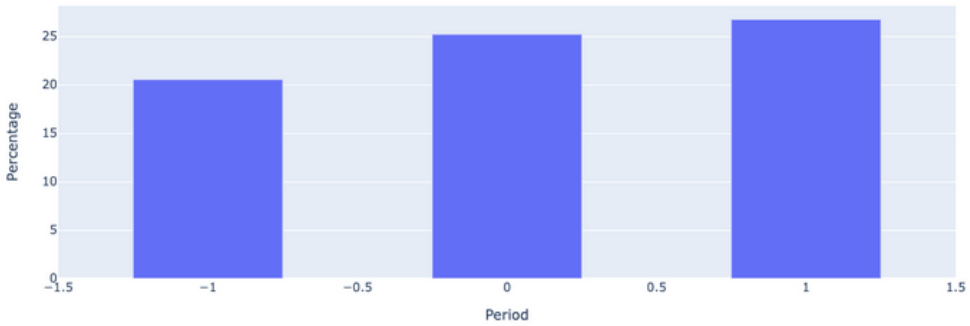

Figure 8- Percentage of bad ratings of movies according to the period

The difference between the periods is more marked. We can clearly see that the periods after the nominations and the awards are marked by a higher share of negative reviews : from 20% up to 26%.
And this significant increase takes place from the 'during' period, that is to say that only the nominations of the films for the awards has an impact and not their effective winning o or not.r not.

This first part of analysis is in line with the results we wanted to observe. Overall, these findings highlight the impact of awards nominations on both the quantity and sentiment of an increase in the number of reviews during the period between nominations and awards, indicating that nominations stimulate criticism. The average ratings given by users also decrease over time, suggesting that once a film is nominated, the ratings become more severe. This is further confirmed by the higher percentage of negative reviews after nominations and awards compared to before.

SENTIMENT ANALYSIS OF THE REVIEWS

We trained a classifier and use it to perform a sentiment analysis on the reviews. We were therefore able to no longer evaluate the positivity of a review by the rating that accompanies it but by the text content itself: by a sentiment analysis using NPL techniques.

1) Training a classifier

There is a public database of IMDb reviews labelled with either 'positive' or 'negative' available online. We imported it and trained a classifier with the training and validation sets that we constituted.

To analyze the reviews, we decided to use the "bag of words" technique. This technique relies on the fact that a word is represented by a vector where each coordinate x_{i} is the number of occurence of this word in the review number i. Thus, our classifier uses the frequency of the different words to determine the negativity or the positivity of a review

```
[ ]  #demonstration on a small corpus
     test_corpus = ['The movie was really bad.',
                    'The movie movie is phenomenal.',
                    'A really good film.',
                    'The film was not that bad.',
                    'I love this movie.']
     vectorizer = CountVectorizer()

     bow = vectorizer.fit_transform(test_corpus)

     print(vectorizer.get_feature_names())
     bow.toarray()
```

```
['bad', 'film', 'good', 'is', 'love', 'movie', 'not', 'phenomenal', 'really', 'that', 'the', 'this', 'was']
array([[1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1],
       [0, 0, 0, 1, 0, 2, 0, 1, 0, 0, 1, 0, 0],
       [0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0],
       [1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1],
       [0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0]])
```

Figure 9- Example of the bag-of-words technique

	precision	recall	f1-score	support
0	0.83	0.87	0.85	238
1	0.88	0.84	0.86	262
accuracy			0.85	500
macro avg	0.85	0.85	0.85	500
weighted avg	0.85	0.85	0.85	500

13

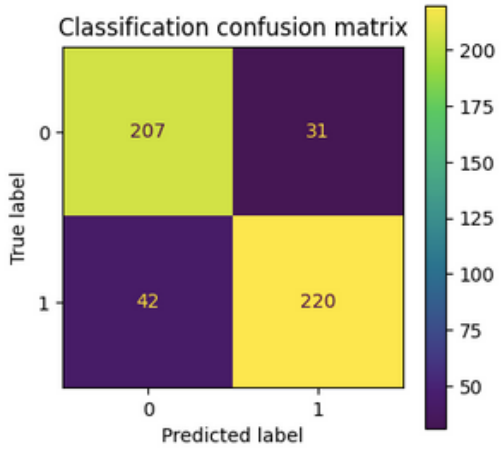

Figure 10- Final performance of our classifier after adjusting the parameters

After adjusting the parameters, the accuracy is quite satifactory and accounts for 85%. Now, we can perform a sentiment analysis on the data we scraped.

2) Testing the classifier on our reviews

We tested on our dataset the classifier that we previously trained. We first observed the number of reviews in function of the period with a color that indicates whether the sentiment expressed is positive or negative.

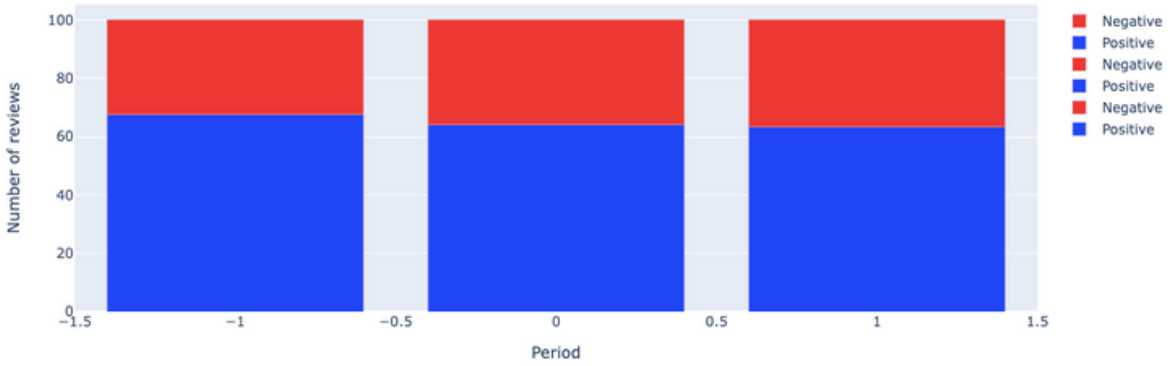

Figure 10- Main sentiment of the reviews according to the period

These results are in agreement with what we obtained previously. The classification by the texts is a little more pessimistic since there are generally more negative reviews. But the tendency of the share of negative reviews to increase between the periods is still observable going from 32.5% to 36.7%. And once again the increase is seen as soon as the films are nominated.

3) Quantifying the negativity

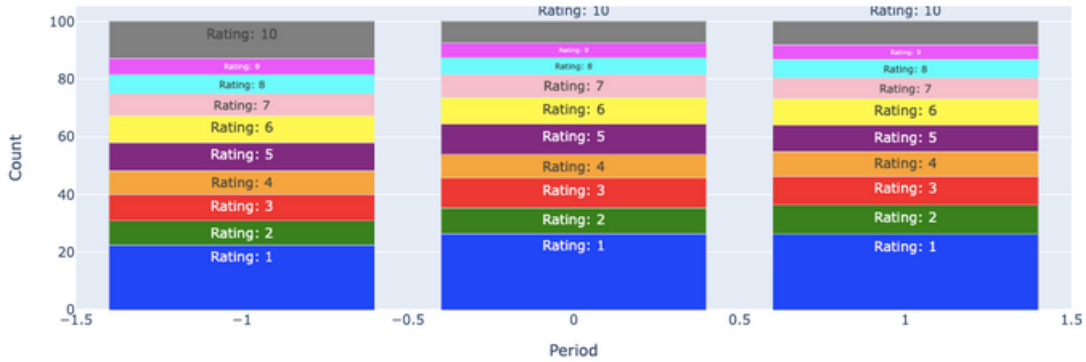

Figure 11- Ratings distribution of reviewq by period

We looked into more detail at the reviews that have obtained the qualification of negative. We did not have the necessary training set to carry out multi-classification, so we once again studied the ratings associated with the reviews.

We can see that there are a lot of false negatives. For the lowest rating, the difference is once again pronounced between 'before' and the 'during' and 'after' periods, those last having a higher percentage of really negative appreciations : 22,47% against 26,34% and 26,31% respectively. The gap fades as the ratings increase but becomes again marked for the highest rating (with the tendency inverse) : 12,78% against 7,49% and 8,09% respectively.

4) Logit regression

With the classification we obtained for our reviews, we tried to study the link between the label `positive` or `negative` of a review and the period during which it was posted. Since we obtained in our study, very similar results with the periods 'during' and 'after', we considered them in this part as one unique period. We were studying the links between two binary variables, so we performed a logistic regression.

prediction	0	1
period		
0	8991	18665
1	11456	20082

Cross-frequency table of the variables 'prediction' and 'period'

The number 1 is associated to a positive classification and the number 0 to a negative one. The number 1 is associated when the review was posted in the 'during' or 'after' period and 0 is associated for a review posted in the 'before' period

At first glance, studying the cross-frequency table, there is no particular trend between the period and the sentiment of the review, except what we have already observed. We then took a closer look.

```
Optimization terminated successfully.
         Current function value: 0.643765
         Iterations 4
                           Logit Regression Results
==============================================================================
Dep. Variable:                      y   No. Observations:                59194
Model:                          Logit   Df Residuals:                    59192
Method:                           MLE   Df Model:                            1
Date:                Sun, 02 Jul 2023   Pseudo R-squ.:                0.001244
Time:                        23:08:39   Log-Likelihood:                -38107.
converged:                       True   LL-Null:                       -38155.
Covariance Type:            nonrobust   LLR p-value:                 1.945e-22
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.7304      0.013     56.898      0.000       0.705       0.756
x1            -0.1691      0.017     -9.733      0.000      -0.203      -0.135
==============================================================================
```

Summary of the logit regression of 'prediction' according to 'period'

The pseudo-R2 is very weak: only 0.1% of the variations in sentiment classification are explained by the model. However, the p-values of the coefficients are null so they are statistically significant. The coefficient in front of X is negative, i.e. being during the 'after' period decreases the probability of obtaining a positive review. The final Z-index is : (0.7304 - 0.1691*period).
After applying the link function of the logistic regression, we obtain :

- P(predictiont=1|OO=0) = `0,67`
- P(prediction=1|OO=1) = `0,64`.

Thus, being during the 'after' period decreases the probability of obtaining a positive review by **4,5%**

CONCLUSION

Through the different approaches we have conducted to tackle the problem, we have obtained quite similar results.

01

Analysis of ratings

- Ratings tend to be worse afterwards
- This starts as soon as movies are nominated for awards

02

Sentiment analysis

- Confirmation of the previous findings
- The difference is mainly on the proportion of very bad reviews

03

Logistic regression

- Quantification of the difference between the periods
- When a movie is nominated for an award, the probability of receiving a negative review increases by 4.5%

There are several possible interpretations for this result :

• When a movie is nominated for an award, expectations become higher. If the movie fails to meet these elevated expectations, it can result in more negative critiques.

• The nomination brings visibility to the movie. Whereas before it may have only been watched by fans of its genre, its exposure opens it up to a wider audience. This can lead to people who wouldn't have normally watched the film to do so, and consequently, potentially more negative evaluations.

APPENDIX

About classifiers

A classifier in machine learning is an algorithm that automatically orders or categorizes data into one or more of a set of “classes.” Here we used it to classify a review as positive or negative, but in machine learning, it is also used to classify diverse data.
We have trained the classification algorithm to be able to classify a review by indicating whether the sentiment is positive or negative.

More explanations about bag-of-words technique

The bag-of-words technique is a common approach used in natural language processing (NLP) and text analysis. It represents a document as a "bag" or collection of words, disregarding grammar and word order, and focusing solely on the presence and frequency of individual words.
The process starts by creating a vocabulary, which is a unique set of all the words present in the entire corpus or collection of documents being analyzed. Each word in the vocabulary is assigned a unique index or identifier.

To represent a document using the bag-of-words model, the text is tokenized, meaning it is split into individual words or tokens.Then the text is lemmatized. Lemmatization in linguistics is the process of grouping together the inflected forms of a word so they can be analysed as a single item, iden tified by the word’s lemma, or dictionary form [2]. In other words, it consists in refer ring each word to its common lexical entry, often called canonical form. For example, all the words in following list [’change’,’changes’,’changing’,’changed’,’changer’] have the same lemma : change.
Then, a vector is created for the document, with each element of the vector corresponding to a word in the vocabulary. The value in each element represents the frequency or occurrence of that word in the document. This approach transforms the textual data into a numerical representation that can be utilized by machine learning algorithms.
One of the key advantages of the bag-of-words technique is its simplicity and computational efficiency. It allows for quick processing of large amounts of text data.

Bag-of-words is often used as a preprocessing step in various NLP tasks, such as sentiment analysis, text classification, and information retrieval. It provides a foundation for further feature engineering and statistical analysis of textual data.

BIBLIOGRAPHY

Harish, B. S., Kumar, K., & Darshan, H. K. (2019). Sentiment analysis on IMDb movie reviews using hybrid feature extraction method.

Zulfiqar, A., Xiao, C., Azeem, M., Mahmood, T., & Shaukat, Z. (2020). Sentiment analysis on IMDB using lexicon and neural networks. SN Applied Sciences, 2(2).

Kastrati, Z., Dalipi, F., Imran, A. S., Pireva Nuci, K., & Wani, M. A. (2021). Sentiment analysis of students' feedback with NLP and deep learning: A systematic mapping study. Applied Sciences, 11(9), 3986.