Sentiment Analysis on Movie Reviews with NLP and Creating a Recommendation System

Amin Khoeini

Media streaming platforms are getting more and more popular in our time. People prefer watching movies with family in their homes with their big screen TV's. In this era, each subscription platform tries to get more subscribers and encourages users to engage with the specific platform. One of the ways they engage the viewers is by asking them if they liked the movie and right away recommending a list of movies for them to watch. Movies to watch stay on the platform more often and this is what the company wants. Netflix uses thumbs up and double thumbs up for this feature and then creates a list specific to the user's liking to watch.

While IMDB is not a streaming website (well, YET!), it has the most users among movie database websites. Thousands of users write reviews for popular movies on IMDB and therefore, this website has the largest database of audience reviews. If IMDB applies the same recommendation system that Netflix has, it can suggest a list of the movies that users might like to watch based on previous reviews they have written. This means that people have more clicks and spend more time on the website which further suggests more user engagement and ad revenues for IMDB.

1. Data

The data for this project consists of two datasets, both gathered from Kaggle's website. The first set includes the 500,000 reviews data scraped from IMDB's website for movies released from 1920 to 2016. This set includes:

- **Review ID:** Unique identifying number for each review given by IMDB website.
- **Reviewer:** The username of the person who wrote the review.
- **Movie Title:** The title of the movie accompanied by release date
- **Rating:** The rating that the reviewer gave to the movie, it is a number between 0-10.
- **Review Summary:** The title that the reviewer gets to the review, usually its between 4 to 10 words.
- **Review Date:** The date that the review was written.
- **Review Detail:** The actual review that was written by the reviewer.
- **IMDB Link:** the link to the IMDB page of the movie

The second set consists of all the movie's information from the IMDB website which includes lots of data about movies. From this set, IMDB ID, title, genre, director, actors, year and description will be used in the content-based filtering as a part of the recommendation system.

1.1 Data Wrangling

There are two different steps of data manipulation for this project, one related to NLP and review sentiment analysis, and the other for the recommendation system. First, there are some rows that have no reviews as data and instead, the user just gave a vote number to the movie. Because the project focused on the review detail, these rows were dropped from the dataset. The rest of the columns need to be processed separately based on their use in the project:

- Reviewer: this column contains the user name, which was created by each reviewer and is very inconsistent. To have a unique and consistent ID for each user, a unique ID number was created for each user. This number starts from 1 and ends at 11,256 which is the number of unique reviewers.
- Rating: To predict the sentiment of the review, there needs to be a label based on the rating that the user gave to the movie. This label was created by using the rating column and the threshold that was decided. So each review that has a rating of 7 and above was labeled as ‘LOVE IT’, which means that the reviewer loves the movie, and ‘NOT LOVE IT’ for any rating given below 7.
- Review Detail: The actual corpus of the review needs most of the manipulation to make it ready for the NLP process.
    - Clean punctuation from the corpus
    - Remove the stop words from the corpus
    - Remove any numeric and non english words from the corpus.
    - Lemmentize and stemm the corpus.

For the recommendation system, both the Review and IMDB dataset need to be cleaned too. First for the review dataset, the unique IMDB ID needs to be extracted from the IMDB link columns presented in this dataset. This way, the review dataset has a unique identifier for each movie besides the user ID, which is necessary for the Collaborative filter. At the same time, the IMDB data can be filtered using the IMDB ID, in order to have only the data of the movie that is presented in the review dataset which is around 4000 unique movies. The final step of the data manipulation would be creating a column which was called Soup, that contained all the data about content of the movies, so a content filter can be created from that column.

- Year that the movie was created.
- Director: The director column needed to be lowercase, remove any punctuation and be separated in the case that there was more than one director. In order to give more weight to the director 's name in the soup, this column was added twitch in the soup.
- Genre: The genre columns needed to be lowercase, and all the dashes were removed.
- Actors: The actors columns needed to be lowercase, and all the dashes were removed.
- Description: This column contains the brief summary of the movie. In order to be able to use it in the soup, these columns needed to be lowercase, the stop words needed to be removed, and also lemmentized.

2.Methods

This is a combination of two models; the first will be a NLP model to predict the sentiment of the review. For this, the different NLP models were trained with the review and the label columns and predicted if it is a positive or negative review. Each model was evaluated on

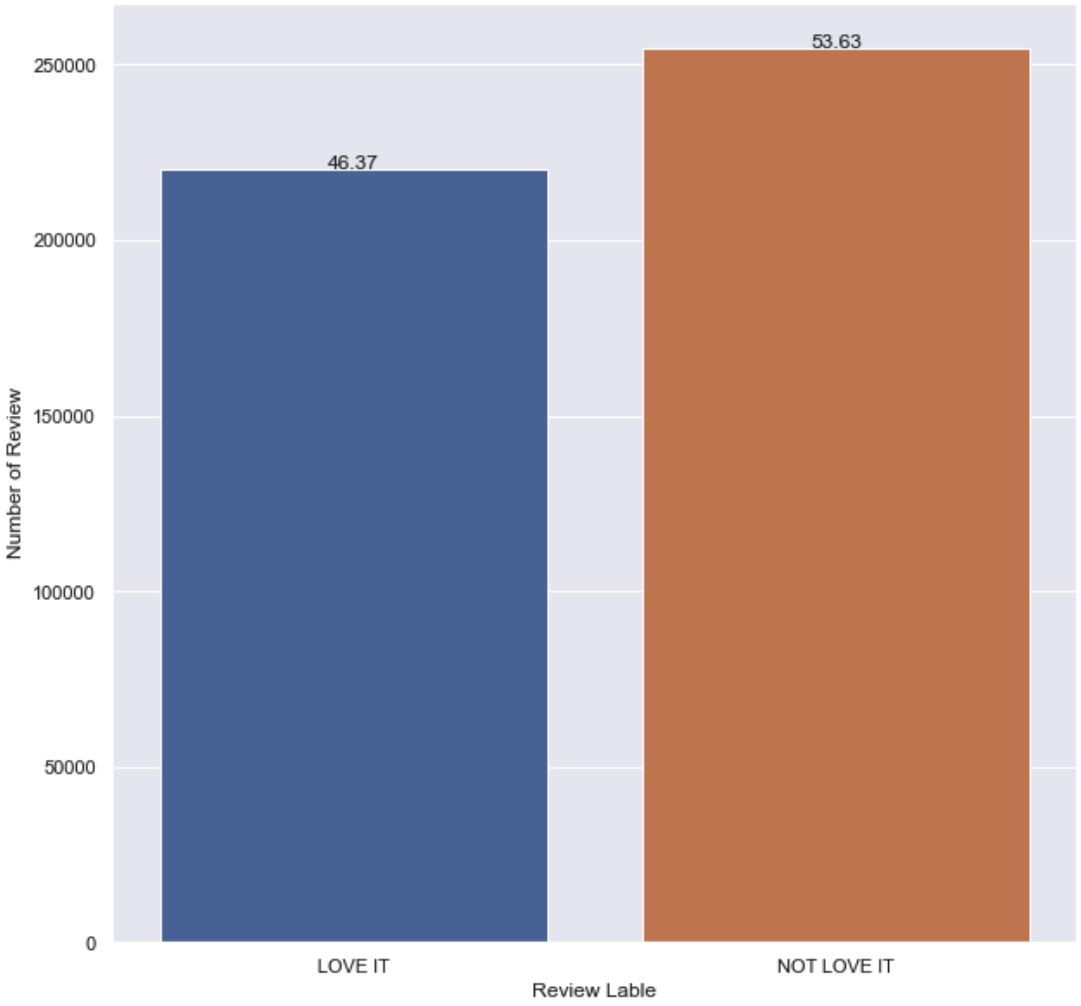
its accuracy and fitting time, and the most optimized one was chosen as the final model for the review sentiment analysis.

The recommendation system contains two filters. First, a content base filter and then the collaborative filtering recommender was used. The content filter used the content soup columns that was created before, and by using a Sklearn countvectorizer and cousin similarity, a similarity dataframe was created. By using this data frame, a list can be created of the movies that are similar to the targeted movie. The collaborative filter is entirely based on past behavior and not on the content of the movie. More specifically, it analyzes how similar the tastes of one user is to another and makes recommendations on the basis of that. For this filter, the Surprise package was chosen and trained the data on the different model that Surprise provides to see which can predict the target user vote with the least RMSE. The first filter created a list of fifty movies that content wise are similar to the target movie, then by using the collaborative filter, it can predict the vote that the target user would give to each of the movies. Then by sorting the movie list based on this estimation, the top 10 can be chosen as the final recommendation.

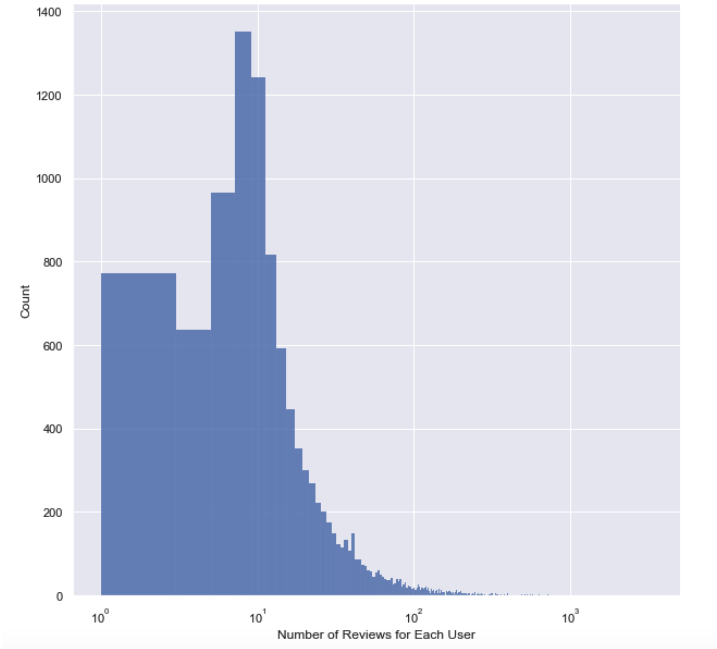
3. EDA:

3.1 - Reviews label:

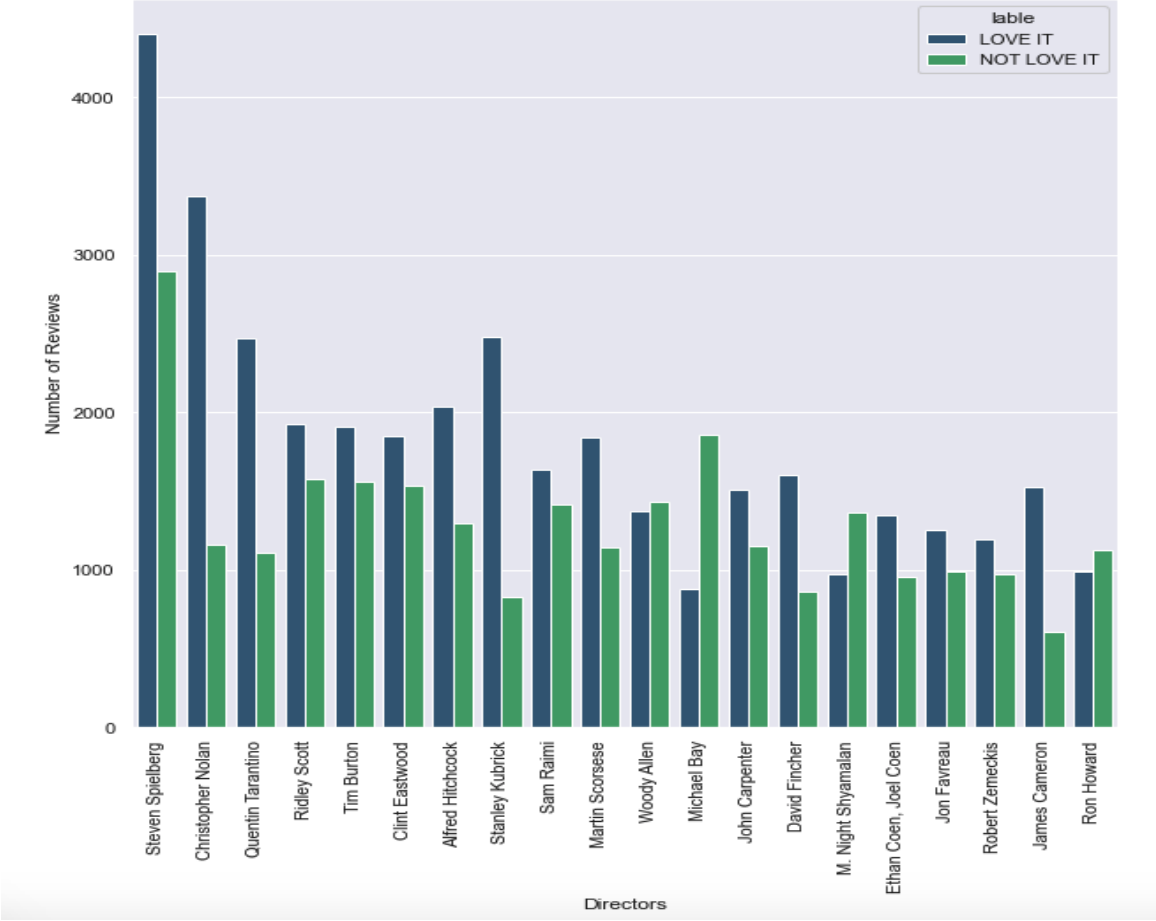
A considerably high number was chosen as the threshold for the sentiment label. Only ratings 7 or higher are considered to be movies that the user loved, but a good distribution of the label in the data was still seen. 46 percent of the reviews were labeled as positive while 53 had a rating below 7 and labeled as negative.



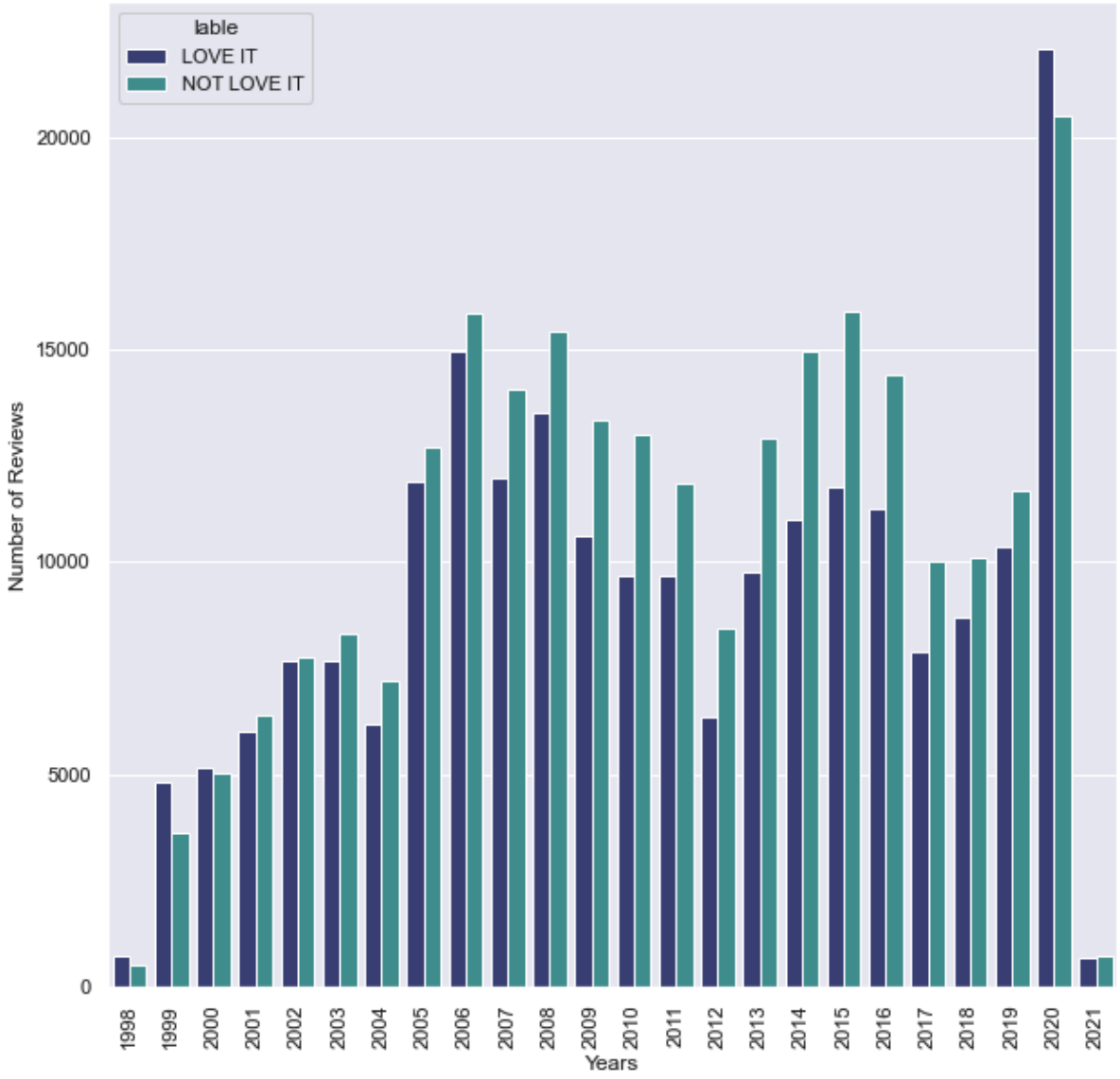
It can be observed that most of the users wrote less than the 10 reviews on the website.



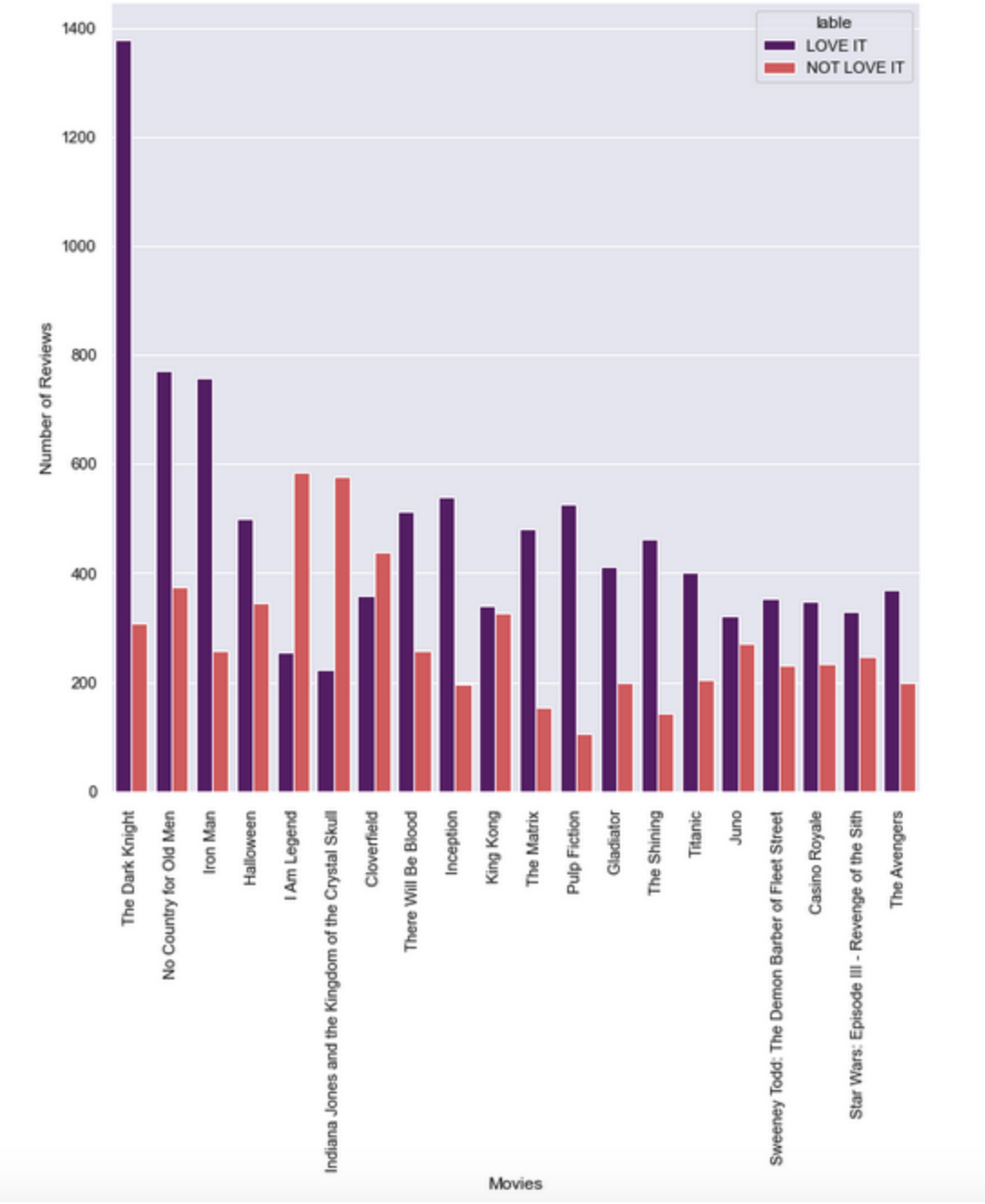
3.2 - Director, year and movie title:



Obviously, famous directors have the most reviews about them, but if we look at the distribution of the label, there is an interesting insight. Micheal Bay and M. Night Shyamalan are notorious for making a bad blockbuster and here it can be seen that they have more negative reviews. Besides this, Woody Allen also has more negative reviews than positive and this might be because of the drama that he faced in the media recently. The rest of the directors have more positive than negative reviews and this is expected because these are famous directors that received most of the reviews.



In 2020, users of IMDB wrote the largest number of reviews. People were at home because of the pandemic and therefore, watched more movies on streaming services and wrote more reviews. Also, for the first time after IMDB was created, there were more positive reviews than negative. Most likely because a more regular audience was using IMDB and writing reviews in comparison to before when only cinephiles used to write reviews on IMDB.



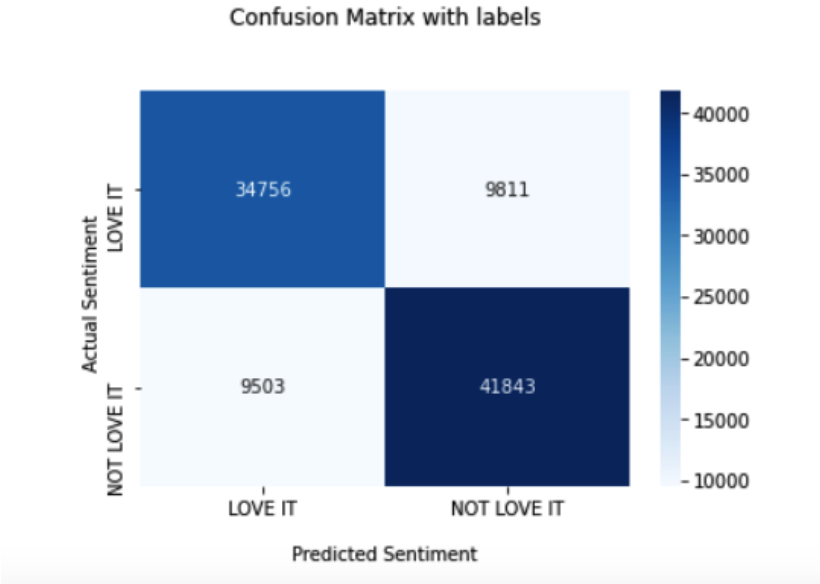
Popular movies have considerably more positive reviews, especially the top three movies. But at the same time, there are a large number of negative reviews for I am Legend, Indiana Jones and the Kingdom of the Crystal Skull, and Cloverfield. Indiana Jones was the reboot of the

famous series which apparently viewers did not like. This is the case for most of the reboots for successful series' in the past, as it has been seen for the Matrix series.

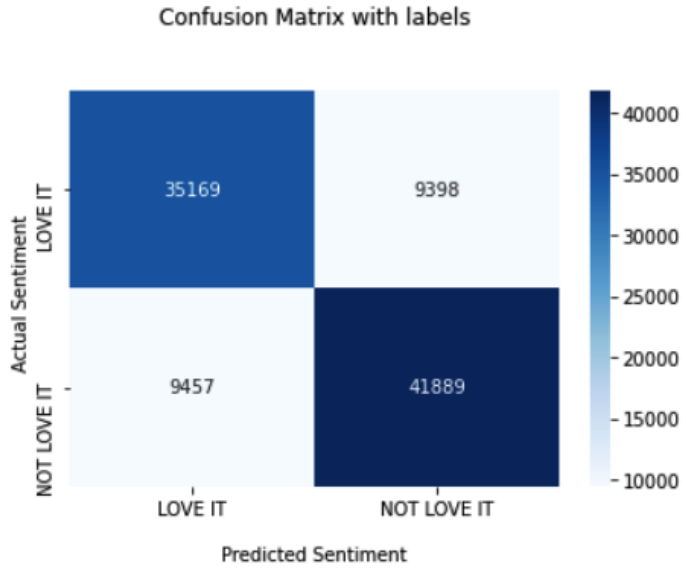
4. Algorithms & Machine Learning

4.1 - NLP and review sentiment analysis

For the sentiment analysis of the review, the processed review columns must be trained with a different classification model. But first with the help of nltk library, we must tokenize the review corpus and then make a Tfidf vector matrix with Sklearn feature extraction. Now this matrix can be used to train a classification model. MultinomialNB, LogisticRegression and Linear SVC, were chosen from the Sklearn library and after doing the Hyperparamater tuning for each model and getting the best possible performances, results were evaluated based on their accuracy and fitting time. The initial test showed the Linear SVC provided the best accuracy around 79%.



With the Hypertunning on Liner SVC, and setting the model parameter to the suggested value (C = 1, loss = hinge, penalty = l2) , the performance became slightly better and raised the accuracy to 81%, which is not perfect but acceptable for classification.



	precision	recall	f1-score	support
LOVE IT	0.79	0.79	0.79	44567
NOT LOVE IT	0.82	0.82	0.82	51346
accuracy			0.80	95913
macro avg	0.80	0.80	0.80	95913
weighted avg	0.80	0.80	0.80	95913

4.2 - Recommendation System:

Recommendation system includes two steps. As discussed earlier, the content filter used the soup columns, Count Vector and similarity matrix to create a similarity dataframe for all of the movies. Therefore, only the target movie in the similarity dataframe needed to be searched, then sort the result and pick the top 30, as the movies that content wise are closest to the target movie. For example if we try to find movies similar to Gang of New York directed by Martin Scorccesie, the following suggestion is given. The top picks are all directed by the same directors and the rest of the suggestions have the same genre as target movies. Content wise, it seems to be a very good suggestion list but it doesn't consider the taste of each viewer, and suggests the same list for all the users.

```
['The Departed (2006)',
 'The Aviator (2004)',
 'The Wolf of Wall Street (2013)',
 'Mean Streets (1973)',
 'The Last Temptation of Christ (1988)',
 'Shutter Island (2010)',
 'After Hours (1985)',
 'Casino (1995)',
 'New York, New York (1977)',
 "Alice Doesn't Live Here Anymore (1974)",
 'Bringing Out the Dead (1999)',
 'Hugo (2011)',
 'Midnight Cowboy (1969)',
 'The Bounty (1984)',
 'The Tailor of Panama (2001)',
 'Dead Man Down (2013)',
 'Catch Me If You Can (2002)',
 'There Will Be Blood (2007)',
 'Public Enemies (2009)',
 'Key Largo (1948)',
 'American Psycho (2000)',
 'London Boulevard (2010)',
 'Kalifornia (1993)',
 'Extremely Loud & Incredibly Close (2011)',
 'The Taking of Pelham 123 (2009)',
 'The Disappearance of Alice Creed (2009)',
 'A Most Violent Year (2014)',
 'Casualties of War (1989)',
 'Calvary (2014)',
 'The Shadow (1994)']
```

The Collaborative filter will predict the rating that each user might give to the movies. To create that filter, Surprise provided a variety of the algorithms that predicted the user ratings. So

by using the previous similar content list for the targeted movie, the prediction can be obtained on how the target user would vote for them. First, all the available algorithms on Surprise would give the performance result and see which one has the least RMSE.

	test_rmse	fit_time	test_time	Algorithm
0	1.854455	38.425230	2.319381	SVD
1	1.912588	2272.251957	56.917552	SVDpp
2	1.843742	9.421812	46.156351	SlopeOne
3	2.274557	40.471097	2.107328	NMF
4	3.132171	0.577075	2.039930	NormalPredictor
5	1.804845	60.256454	55.050260	KNNBaseline
6	1.935067	69.662538	58.847921	KNNBasic
7	1.865102	64.828813	49.204878	KNNWithMeans
8	1.867365	60.622549	51.948352	KNNWithZScore
9	1.841107	0.421204	1.647150	BaselineOnly

KNNBaseline has the least RMSE among the algorithms, and although it has one the longest performance times, it will be picked as our best model. Next, we will see if a better performance can be obtained by doing a hyperparameters tuning. The Hyperparameters tuning lowers the RMSE by 0.2, while we train the model Using Stochastic Gradient Descent method with learning rate of 0.01, 40 neighbors and regularization parameter set to 0.1.

	reg	learning_rate	k	train_rmse	test_rmse
17	0.10	0.010	40	1.039012	1.796238
8	0.10	0.005	40	1.010634	1.796835
11	0.02	0.010	40	1.058071	1.797147
2	0.02	0.005	40	1.026944	1.797261
14	0.05	0.010	40	1.050509	1.798137
5	0.05	0.005	40	1.020813	1.798795
4	0.05	0.005	20	0.876147	1.807240
16	0.10	0.010	20	0.897017	1.807699
1	0.02	0.005	20	0.884633	1.808392
10	0.02	0.010	20	0.923078	1.808613
7	0.10	0.005	20	0.863475	1.809698
13	0.05	0.010	20	0.912575	1.809842
12	0.05	0.010	10	0.787320	1.842183
9	0.02	0.010	10	0.798293	1.842209
6	0.10	0.005	10	0.728527	1.843196
0	0.02	0.005	10	0.753954	1.843601
15	0.10	0.010	10	0.768834	1.844367
3	0.05	0.005	10	0.743277	1.844763

The final recommendation engine can be created by combining these two filters in one function. First the content similarity list was created for the target film, then by using the collaborative system, the estimated score was predicted for each film. By sorting this list based on the score estimation, the final top ten is chosen as the final recommendation. We can see the result of this combination for the Gangs of New York below. Compare to content base filter result, There are still a lot of movies with the same director, however by using the collaborative filter and consider the user's taste, it can be seen that the top picks have similar genre, but don't have the same director.

Title	director	year	genre	est
Key Largo (1948)	john huston	1948	action crime drama	9.282466
The Departed (2006)	martin scorsese	2006	crime drama thriller	9.017104
Casino (1995)	martin scorsese	1995	crime drama	9.001346
The Last Temptation of Christ (1988)	martin scorsese	1988	drama	8.878417
Midnight Cowboy (1969)	john schlesinger	1969	drama	8.782507
Calvary (2014)	john michael mcdonagh	2014	comedy drama	8.721642
After Hours (1985)	martin scorsese	1985	comedy crime drama	8.721172
Hugo (2011)	martin scorsese	2011	drama family fantasy	8.687102
Casualties of War (1989)	brian de palma	1989	crime drama war	8.523942
Alice Doesn't Live Here Anymore (1974)	martin scorsese	1974	drama romance	8.517068

Now if we get the recommendation for the next user, a different list is produced and the predicted score is much lower, and it can be guessed that this user does not like the action drama and is not a fan of Scorsese.

Title	director	year	genre	est
Key Largo (1948)	john huston	1948	action crime drama	5.757901
Midnight Cowboy (1969)	john schlesinger	1969	drama	5.647158
Calvary (2014)	john michael mcdonagh	2014	comedy drama	5.350856
Alice Doesn't Live Here Anymore (1974)	martin scorsese	1974	drama romance	5.337787
The Last Temptation of Christ (1988)	martin scorsese	1988	drama	5.066978
Casino (1995)	martin scorsese	1995	crime drama	5.003399
Casualties of War (1989)	brian de palma	1989	crime drama war	4.890082
The Aviator (2004)	martin scorsese	2004	biography drama	4.810042
After Hours (1985)	martin scorsese	1985	comedy crime drama	4.805793
Catch Me If You Can (2002)	steven spielberg	2002	biography crime drama	4.719803

5. Modeling :

The final stage of this project is to combine the two previous models in one function and create the final engine. This NLP/Recommendation engine is designed to get the review of a movie from a specific user, and create a recommandation list based on that review for that user.
So, the final function receives the review, user id and movie title, feeding it to the NLP model that was previously created. Based on the NLP result, there are two options:

- If the NLP predicts a negative label, there is going to be a printed out stating that the user did not love the movie.
- In the case that the prediction was positive, the user id and movie title were fed to the hybrid recommendation system and the recommendation movie list printed out as a final result.

As an example, the performance of the final product can be evaluated on a real world review from the IMDB website for the **Superman IV: The Quest for Peace** movie.

```
'Superman III (1983)',
'Superman II (1980)',
'Iron Man 2 (2010)',
'The Ipcress File (1965)',
'The Entity (1982)',
'Batman v Superman: Dawn of Justice (2016)',
'The Amazing Spider-Man 2 (2014)',
'Vice (2015)',
'2001: A Space Odyssey (1968)',
'Flash Gordon (1980)',
'Iron Man Three (2013)',
'The Core (2003)',
'Iron Man (2008)',
'The Avengers (1998)',
'Terminator 3: Rise of the Machines (2003)',
'Captain America: Civil War (2016)',
'Outlander (2008)',
'The 6th Day (2000)',
'The Last Man on Earth (1964)',
'Dead Man (1995)',
'Raiders of the Lost Ark (1981)',
'Mission to Mars (2000)',
'Spider-Man (2002)',
'War of the Worlds (2005)',
'Rumble in the Bronx (1995)',
'Predator (1987)',
'Hot Shots! (1991)',
'Star Trek VI: The Undiscovered Country (1991)',
'Battlefield Earth (2000)',
'Hardware (1990)']
```

Content filter recommender returns movies from the same Superman movie series and some other superhero movies. The content recommendation also has movies from the same director, group of actors and relatively the same genre as the target movie. This list can be used to see how how the final NLP/recommender engine was performed.

First the relatively negative review with the score of 5 of 10:

```
#5/10 review.
user_id= 9877
movie_title= 'Superman IV: The Quest for Peace (1987)'
review = "Superman IV is not nearly as bad as the reviews suggest. The actors try really hard, \
particularly Christopher Reeve, Gene Hackman, and Margot Kidder, to make it work. The movie is watchable \
and the musical score is good. The movie is an improvement over the disappointing Superman III. \
However, Superman IV has major problems. The movie has obviously been cut from its original length \
make it incoherent at times. The special effects are below the standards set in the first two movies \
(even the third movie had decent effects). Maybe if the movie were restored to its original length, \
it would be better. I can only give this movie a 5/10. I wished it were better and hope someday they \
do restore this movie to its original length."


sentiment_recommender(user_id,movie_title,review)
```

```
The user did not love Superman IV: The Quest for Peace (1987) therefor the system has no recommandation based on th
at review.
```

Let's try a positive review now:

```
#7/10 review.
user_id= 5675
movie_title= 'Superman IV: The Quest for Peace (1987)'
review = "This is the fourth and final Superman film with Christopher Reeve taking on the role, \
where he tries to stop the spread of nuclear weapons and battles Lex Luther (Gene Hackman) and his \
super-powered sidekick, Nuclear Man (Mark Pillow).Though the level of excitement and intrigue of this \
film doesn't match the first two, it is almost on par with the third and is still fun and fast-paced \
with neat special effects that showcase Superman's powers and great action scenes, something that sorely \
lacks in Superman Returns.The nuclear weapons plot I thought was clever and unique to a Superman movie. \
Total disarmament of nuclear weapons may not be feasible in the real world, but Superman's quest for \
peace among all countries is well-intended in the film. Reeve is great as Superman and Gene Hackman gave \
us another thrilling performance as super villain Lex Luther. In addition, the fight scenes between \
Nuclear Man and Superman were awesome - Nuclear Man looked like a force to be reckon with as he possessed \
the same level of powers and same physique as Superman.Overall, I think this film is a fitting ending to \
the Christopher Reeve saga."

sentiment_recommender(user_id,movie_title,review)
```

```
Based on the review, This user loved Superman IV: The Quest for Peace (1987) therefor the system recommand the belo
w list for the user to watch:

 ['Raiders of the Lost Ark (1981)', '2001: A Space Odyssey (1968)', 'Predator (1987)', 'Iron Man (2008)', 'Spider-M
an (2002)', 'Captain America: Civil War (2016)', 'Dead Man (1995)', 'The Ipcress File (1965)', 'Star Trek VI: The U
ndiscovered Country (1991)', 'Hot Shots! (1991)']
```

The review was positive so there was a recommendation list. The target film is a classic superhero movie from 1987 and in the result there were Iron Man, Spider-Man, Capitan America and Star Trek which are among the superhero movies. There were also some classic action sci-fi movies, Raiders of the Lost Ark, Predator, and A Space Odyssey. The Ipcress File which was directed by the same person as the target movie and Hot Shots that has two similar actors as the target movie.

Let's now see the result for a positive review by different use ID:

```
# Same Review but changing the user_id to see a diffrence in the recommendation.
user_id= 3434
movie_title= 'Superman IV: The Quest for Peace (1987)'
review = "This is the fourth and final Superman film with Christopher Reeve taking on the role, \
where he tries to stop the spread of nuclear weapons and battles Lex Luther (Gene Hackman) and his \
super-powered sidekick, Nuclear Man (Mark Pillow).Though the level of excitement and intrigue of this \
film doesn't match the first two, it is almost on par with the third and is still fun and fast-paced with \
neat special effects that showcase Superman's powers and great action scenes, something that sorely lacks\
in Superman Returns.The nuclear weapons plot I thought was clever and unique to a Superman movie. \
Total disarmament of nuclear weapons may not be feasible in the real world, but Superman's quest for \
peace among all countries is well-intended in the film. Reeve is great as Superman and Gene Hackman \
gave us another thrilling performance as super villain Lex Luther. In addition, the fight scenes between \
Nuclear Man and Superman were awesome - Nuclear Man looked like a force to be reckon with as he possessed \
the same level of powers and same physique as Superman.Overall, I think this film is a fitting ending to the \
Christopher Reeve saga."

sentiment_recommender(user_id,movie_title,review)
```

```
Based on the review, This user loved the Superman IV: The Quest for Peace (1987) therefor the system recommand the
below list for the user to watch:

 ['2001: A Space Odyssey (1968)', 'Raiders of the Lost Ark (1981)', 'Captain America: Civil War (2016)', 'Predator
(1987)', 'Iron Man (2008)', 'Spider-Man (2002)', 'Star Trek VI: The Undiscovered Country (1991)', 'The Ipcress File
(1965)', 'Dead Man (1995)', 'Rumble in the Bronx (1995)']
```

Another User ID, same result, but this user got the recommendation to watch the Superman film that was made before this one.

```
# Same Review but changing the user_id to see a diffrence in the recommendation.
user_id= 28
movie_title= 'Superman IV: The Quest for Peace (1987)'
review = "This is the fourth and final Superman film with Christopher Reeve taking on the role, \
where he tries to stop the spread of nuclear weapons and battles Lex Luther (Gene Hackman) and his \
super-powered sidekick, Nuclear Man (Mark Pillow).Though the level of excitement and intrigue of this \
film doesn't match the first two, it is almost on par with the third and is still fun and fast-paced with \
neat special effects that showcase Superman's powers and great action scenes, something that sorely lacks\
in Superman Returns.The nuclear weapons plot I thought was clever and unique to a Superman movie. \
Total disarmament of nuclear weapons may not be feasible in the real world, but Superman's quest for \
peace among all countries is well-intended in the film. Reeve is great as Superman and Gene Hackman \
gave us another thrilling performance as super villain Lex Luther. In addition, the fight scenes between \
Nuclear Man and Superman were awesome - Nuclear Man looked like a force to be reckon with as he possessed \
the same level of powers and same physique as Superman.Overall, I think this film is a fitting ending to the \
Christopher Reeve saga."

sentiment_recommender(user_id,movie_title,review)
```

```
Based on the review, This user loved Superman IV: The Quest for Peace (1987) therefor the system recommand the belo
w list for the user to watch:

 ['2001: A Space Odyssey (1968)', 'Raiders of the Lost Ark (1981)', 'Predator (1987)', 'Spider-Man (2002)', 'Captai
n America: Civil War (2016)', 'Dead Man (1995)', 'Star Trek VI: The Undiscovered Country (1991)', 'Iron Man (200
8)', 'Superman II (1980)', 'The Ipcress File (1965)']
```

7. Future Improvements:

- In NLP model and sentiment analysis, it is optimal to have an accuracy above 90% and minimum number of false positives and negatives. For this project accuracy is 80% and there is obviously room for improvement. Having more reviews, better hardware to run hyperparameters tuning for even more different algorithms, would help to improve the NLP performance.
- There is only data about 4000 movies in the dataset. Because of that the recommendation engine was limited to those only. Having score and user data created more reliable recommendations for the user.