Recommender Systems: An Analysis on content based recommending

LaTeX template adapted from:
European Conference on Artificial Intelligence

Mercy Nwabueze-nwoji[1]

Other group members:
Aasiya Jilani[2] **Alex McKenzie**[3] **Natasha Benjamin**[4]

Abstract.
Recommender systems have become rampant due to the information overload present in numerous domains on the internet. Thus, the tedious task of finding information which would have been considered time consuming and difficult once has become much simpler to even the layman. This system is important to us today as it helps internet users find the information, they need by providing personalized suggestions. In our case, the so called content based recommenders which we will be discussing offers suggestions similar to what users liked or viewed in the past as well as explicit feedback from other users. In this paper, we will be experimenting and discussing the mechanisms which content based filtering works as well as comparing it to collaborative filtering, all while making use of data from the company Netflix as well as data from the movie lens data set.

***Keywords*—** : content based filtering, collaborative filtering, Netflix, movie lens, experiment, TF-IDF vectorization, Cosine similarity

1 Introduction

The television and film entertainment industry has, in the last decade, gained much needed momentum through the reliance on recommender systems. The main objective of this algorithm being to sift through the data in accordance with the interests of the users who watch these content. Since Netflix has a very large collection of data, the recommender algorithm has been put in place for the users to be able to filter the information in favour of what they are looking for. Recommendation systems have become much more common over the years and are used almost everywhere from e-commerce websites, online article, to the music industry and the film industry. However, even for algorithms as great as content based filtering which suggests to user's products similar to those they viewed in the past, there are still limitations involved.

In this paper we will be testing various parameters in regard to content based filtering, we will also be comparing the time taken between collaborative filtering and content based filtering. We will also be testing two sets of data, one from the movie lens data set and another from the Netflix 2021 data set. This particular data sets were chosen because of it's impact on the daily life of everyone, as television and film not only relaxes individuals but also serves as a source of information and possibly, even education. Therefore, it is vital that the suggestions which a user is given is not only accurate and up to date but also reliable and trust worthy.

The sections of these paper are divided into the following sections: (II)Background in which we would be discussing the definition of the content based recommender system and it's advantages in comparison to the collaborative filtering algorithm, (III)experiments and results which would include a detailed analysis of the use case recommender system chosen as well as tests run on it, (IV)discussions of what could be the mechanisms used for content based recommendations, how it can be improved and limitations of this in relation to the experiment and results done in the third section of this paper and finally, (V)our conclusion which will also shed light to future works in the making.

2 Background

The definition of content based filtering according to (Google Developers, 2022)[1] is an algorithm that uses item features to recommend other items similar to what the user likes, based on their previous actions or explicit feedback. In other words, it makes use of various features from the database it is using, such as genres, cast, director for movies, and titles, ratings and descriptions for books in order to recommend what the user would like. (Ricci, Rokach and Shapira, 2022,) [6]state that this system learns to recommend items that are similar to the ones that the user liked in the past. Meaning that it's ability to suggest movies to the users can be attributed to the fact that it can make recommendations for people with unique tastes, however as good as this is, it has its down sides. This is due to the overspecialization nature of this recommendation system as it is unable to make recommendations outside what a user has seen in the past.

The internet has become a vast sea of data and information, we make use of recommendations to be able to navigate this hypothetical sea as it improves the "user to content" experience making it easier for the layman to find whatever they need on the internet. The recommendation systems are especially great for businesses as they can leverage data from users' history to be able to suggest products they could potentially use. A recent survey done by (Beel et al., 2015)[4] shows that in the last 16 years, through extensive research, they have been able to identify that over half of the recommendation approaches made use of content based filtering at a rate of 55%. While Collaborative filtering and graph-based filtering were used 18% and 16% of the time respectively.

This study shows that even though collaborative filtering has been deemed the most popular of the various recommendation systems,

[1] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: mn6529d@gre.ac.uk

[2] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: aj5266s@gre.ac.uk

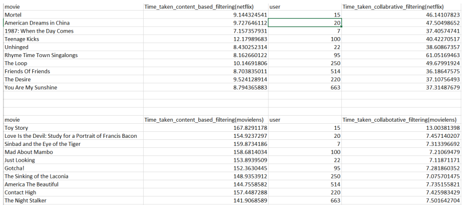
[3] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: am5636g@gre.ac.uk

[4] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: nb9652v@gre.ac.uk

it is by far not the most preferred means of recommending items to users due to shortcomings such as the cold start problem, the sparsity problem and the popularity bias problem. In the next section of this paper, we will make use empirical evidence to compare both the collaborative and content based recommender systems and determine which algorithm would do a better job at satisfying users.

3 Experiments and results

In the experiment which we carried out we made use of two data sets: Netflix and movie lens. In the evaluation of recommendation systems, we must think of evaluation metrics such as similarity metrics, predictive metrics, classification metrics and rank-based metric. However, we will be using a different approach of comparing these algorithms by looking at the time taken for these systems to function. It is important to note that although getting the time taken for the algorithm to function is an essential part of testing algorithms, it is not usually considered a prevalent method when testing recommender systems.This approach to testing the algorithm was chosen because, according to (Beel, 2017,)[3] it is not possible to present a comprehensive analysis of all variables effecting an algorithm's effectiveness. Meaning that evaluating our recommender systems based on precision and accuracy alone may not be enough to determine which recommender does best in the long run. This method of evaluating the recommender system might be overlooked but it is still important as users need to be able to get suggestions for items or movies as quickly as possible for them not to lose interest in the item. The table below shows a comprehensive list of the movies and users used to test the data as well as the time taken to run each recommender system.



movie	Time_taken_content_based_filtering(netflix)	user	Time_taken_collabrative_filtering(netflix)
Mortel	9.144324541	15	46.14107823
American Dreams in China	9.727646112	20	47.50498652
1987: When the Day Comes	7.157357931	7	37.40574741
Teenage Kicks	12.17989683	100	40.42270517
Unhinged	8.430252314	22	38.60867357
Rhyme Time Town Singalongs	8.162660122	95	61.05169463
The Loop	10.14691806	250	49.67991924
Friends Of Friends	8.703835011	514	36.18647575
The Desire	9.524128914	220	37.10756493
You Are My Sunshine	8.794365883	663	37.31487679

movie	Time_taken_content_based_filtering(movielens)	user	Time_taken_collabotative_filtering(movielens)
Toy Story	167.8291178	15	13.00381398
Love Is the Devil: Study for a Portrait of Francis Bacon	154.9237297	20	7.457140207
Sinbad and the Eye of the Tiger	159.8734186	7	7.313396692
Mad About Mambo	158.6814034	100	7.21069479
Just Looking	153.8939509	22	7.11871171
Gotcha!	152.3630445	95	7.281860352
The Sinking of the Laconia	148.9353912	250	7.075701475
America The Beautiful	144.7558582	514	7.735155821
Contact High	157.4487288	220	7.425983429
The Night Stalker	141.9068589	663	7.501642704

Figure 1. table with the different times taken for each recommender systems to function in accordance with the netflix and movie lens data set

In the analysis of the graph for time taken for the two recommender systems in terms of the Netflix data set, we discovered that it would take a shorter amount of time for the content-based recommender to recommend content to the user. An in depth analysis of these two graphs in figures two and three shows us that content based filtering for this data set may never exceed anything past 20milliseconds because, as shown in the graph the highest time taken that has been attained so far through our test data is at around 12.129milliseconds. Meanwhile the lowest time taken for the collaborative algorithm is about 36. 186milliseconds which is leaves a wide gap of a difference 26.056milliseconds between the highest and lowest time taken between the two algorithms respectively.

However, the analysis of the time taken for the two recommender systems using the movie lens data set proves otherwise as shown below. We find out immediately just by looking at the graph that the overall time taken for content based filtering is higher than that of the collaborative filtering. This is seen by just how consistent the time taken between user twenty and user two hundred and fifty is in the collaborative filtering graph with the highest time taken being 13.003milliseconds. Meanwhile the lowest time taken for the content based algorithm to function is 141.906milliseconds. Thus leaving an even larger difference of 128.903milliseconds between the two algorithms.

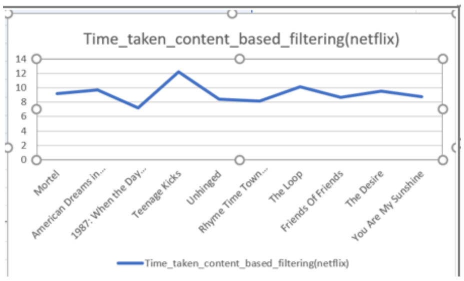


Figure 2. graphical analysis of the content based recommender systems in terms of the Netflix data

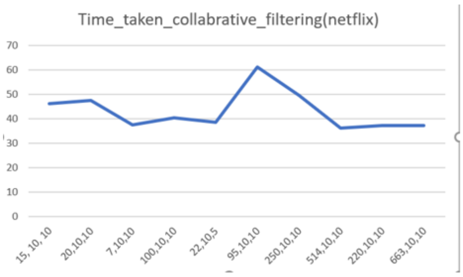


Figure 3. graphical analysis of the collaborative recommender systems in terms of the Netflix data

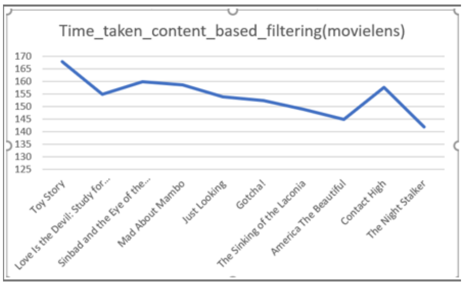


Figure 4. graphical analysis of the content based recommender systems in terms of the movie lens data

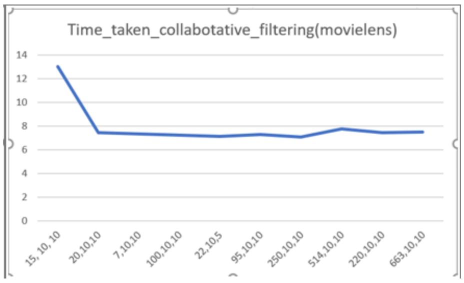


Figure 5. graphical analysis of the collaborative recommender systems in terms of the movie lens data

4 Discussion

In using the recommender systems on the Netflix database for our first comparison, we find that there is an anomaly with the recommender systems as they are doing the opposite of what they should be doing I.e., content based filtering should be slow while the collaborative filtering should be fast. The reason for this anomaly could also be due to scalability problems. As described by (Wikipedia, 2022)[2]due to the many users and products in the database, a large computational power is needed for this type of filtering to take place. This is the most probable cause of the collaborative system being much slower with the Netflix data set.

Another cause of this anomaly could be alluded to the fact that we are using very different data sets as the Netflix data set is different from the movie lens data set, therefore the tests are producing different results to what we should be expecting. There are also a handful of other reason which could hinder the time taken for these algorithms to suggest movies such as the sparsity problem of the collaborative system or even the limited ability of the content based algorithm to expand on the users' existing interests. These could prove a challenging setback when choosing the best algorithm to suggest movies/items to users.

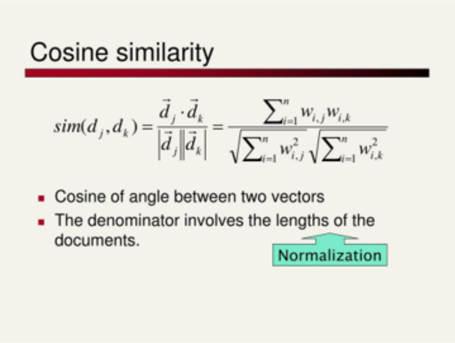
For the second comparison which painted a different story to that of when using the Netflix data set, we find that the slower time for recommendation to be produced could be occur due to various reasons. These reasons could be the use of descriptive data such as the summary, cast, crew etc. To be able to make accurate recommendations, the use of a larger data set as the movie lens data set can be quite large as well as others. Now this is to be expected as the content-based filtering makes use of TF-IDF vectorizer as well as the cosine similarity matrix to be able to make accurate suggestions based on what the user has liked in the past.

TF-IDF stands for Term frequency and inverse document frequency, and it is a formula used to evaluate how important a word is to a document in a documents body, or in this case the summaries of the movies. This means that it is expected for the time taken for a content-based filter to be slower than a collaborative filter, as it must weigh every word in the movie lens summary column to determine words are most important to the user so that it is used to make accurate recommendations. The cosine similarity is a mathematical formula (shown in figure 6)which is able to determine the similarity between movies using the data from the TF-IDF vectoriser. It calculates the similarity between two documents or in this case, the words in the overview which had been preprocessed by the TF-IDF vectoriser and measures how similar they are to each other. (Hakami, 2021,)[5]states that the cosine similarity mathematically, measures the cosine of the angle between two vectors projected in a multi-dimensional space. Meaning that even though two words maybe different when comparing them using other similarity, metrics it would still be considered close to each other in terms of the cosine similarity. Therefore, inability of the content based algorithm to make suggestions without the TF-IDF and cosine similarity, could be the reason why it takes a longer time for the this system to recommend suggestions to the user.

$$idf_t = \log \frac{1+n}{1+\mathrm{df}_t}$$

$$tfidf_{t,d} = tf_{t,d}\, idf_t$$

Figure 6. Formula of the TF-IDF vectorizer



Cosine similarity

$$sim(d_j, d_k) = \frac{\vec{d}_j \cdot \vec{d}_k}{\left|\vec{d}_j\right|\left|\vec{d}_k\right|} = \frac{\sum_{i=1}^{n} w_{i,j} w_{i,k}}{\sqrt{\sum_{i=1}^{n} w_{i,j}^2}\sqrt{\sum_{i=1}^{n} w_{i,k}^2}}$$

- Cosine of angle between two vectors
- The denominator involves the lengths of the documents.

Normalization

Figure 7. Formula of the cosine similarity.

This limitation of the content-based recommender can be alleviated by the collaborative filter as according to (Google Developers, 2022)[1] it needs no knowledge of the item to be able to recommend effectively and accurately. This could be the main reason why our results in figure three show the content-based recommender being slower than the collaborative recommender.

Therefore, the best and plausible solution or algorithm should be able to suggest accurate movies based on what a user would like would be the one which contains an aggregate of both these two algorithms so that the advantages of one algorithm could cover up for the limitations of the other algorithm and vice versa. This solution being the Hybrid recommender system which is able to combine the positive sides of both the content based and collaborative algorithms to create the best solution possible.

5 Conclusion and future work

As discussed in this paper, we were able to not only develop a content-based recommender, but we were able to give empirical evidence as to why this system of recommending may be better than collaborative filtering as well as its shortcomings in comparison to the collaborative algorithm. We did this by comparing content-based filtering to collaborative filtering using the time taken to run the program successfully.

As we move into a more technologically advanced age, it is essential to use the best algorithms to satisfy consumers in their quest for the best movie suggestions and doing this within a reasonable amount of time to prevent loss of interest in the movies.Therefore the content-based recommender system was the topic of discussion as it is one of the most popular algorithms used today amongst many others.

And in anticipation of our future work, we hope to be discussing how to evaluate the system based on coverage, Mean evaluation metric, the f1 measure and the cosine and euclidean similarity metrics. We will have a look at comparing the content based algorithm to the HMM algorithm, Pres algorithm and the Hybrid algorithms which is considered the best recommendations system.

REFERENCES

[1] Content-based filtering;— machine learning— google developers. https://developers.google.com/machine-learning/recommendation/content-based/basics, Jul 2022.

[2] Recommender system, Dec 2022. https://en.wikipedia.org/wiki/Recommender_system/.

[3] Joeran Beel, 'It's time to consider" time" when evaluating recommender-system algorithms [proposal]', *arXiv preprint arXiv:1708.08447*, (2017).

[4] Joeran Beel, Stefan Langer, Marcel Genzmehr, Bela Gipp, Corinna Breitinger, and Andreas Nürnberger, 'Research paper recommender system evaluation: a quantitative literature survey', in *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, pp. 15–22, (2013).

[5] Aisha Y. Hakami. Movie recommendation system, Mar 2021. https://medium.com/mlearning-ai/movie-recommendation-system-f2f57290b1b8/,.

[6] Francesco Ricci, Lior Rokach, and Bracha Shapira, 'Introduction to recommender systems handbook', in *Recommender systems handbook*, 1–35, Springer, (2011).