Content Based Recommendation System on Movies

D. Phani Kumar(✉), Animesh Kumar Singh, Sai Neha Arepu, Manideep Sarvasuddi, Erragatla Gowtham, and Yalamanchili Sanjana

Godavari Institute of Engineering and Technology, Rajamahendravaram, India
phanikumar@giet.ac.in

Abstract. Today's competitive environment makes it necessary for suggestive advice to be made to the user for them to continue using the services they currently find enjoyable. There, the recommender system's function assumes a key role. Every service in today's world has a recommendation system for movies, music, e-commerce, etc. The Netflix recommender system is essential for increasing the customer experience when watching movies on the service. This research proposes a machine learning-based content-based recommender system for movie recommendations. Examining the movie-enabling recommendations using data from the Tmdb, movies dataset from Kaggle. We use algorithms like Count Vectorizer, Porter Stemmer, and Cosine Similarity to generate five similar movies closely related to the type of content the target movie has and how well our machine-learning approach is working.

Keywords: Movie · recommendation system · content-based · Count Vectorizer · Porter Stemmer · Cosine Similarity · Machine Learning Algorithms

1 Introduction

A type of information filtering system called a recommender system bases the filtering on how the information has been shaped. The amount of data being generated has increased dramatically in recent years. As a result, the recommendation system has grown to be a crucial component of social media, e-commerce, and other websites that offer user services. The variety of entertainment options has been expanding quickly due to improvements in the entertainment sector. There is a tonne of alternatives available to consumers, which can be overwhelming for any consumer. As a result, recommendation systems for any product have become more and more common in all areas of digital technology. In light of this, we suggest focusing on the issue of movie suggestions using content-based filtering on a natural language technique. Recommendation algorithms are crucial for helping customers find comparable types of material on movie streaming services like Netflix, Amazon Prime, etc.

A recommendation system that uses content-based filtering bases its recommendations on similar kinds of content that have received user endorsements. Many of the top businesses in the world currently use recommender systems and are working to improve

B. Raj et al. (Eds.): ICETE 2023, AER 223, pp. 462–472, 2023.
https://doi.org/10.2991/978-94-6463-252-1_49

them. One of the key issues is that the majority of recommender systems were created by numerous researchers and engineers to excel at a certain activity. Anyone searching for such a system to integrate will need to devote a significant amount of time and money as a result. Contrary to collaborative-based filtering, content-based filtering takes into account users with similar tastes because it would be detrimental to the user experience to just deliver recommendations without considering the user's interests (Fig. 1).

The goal of this paper is to present the five most pertinent films that are comparable to the topic of the triggering film. The user should be recommended a specific group of movies if the user is approving that particular genre of film, according to machine learning techniques. We used the Tmdb 5000 movie dataset, which consists of two files with movie metadata and credits, in the recommendation system. This data is freely accessible on Kaggle. To create a complete dataset, the movie data was divided based on credits and then combined with the property of the credit. Each tuple afterward had 23 attributes (Figs. 2 and 3).

The dataset was preprocessed to retain those attributes that are important for the type of recommender system that needs to be built, and then Porter Stemmer and Count Vectorizer were applied to turn it into the most favorable tags. Finally, Cosine Similarity was used to find the similarity between the movies in an n-dimensional space. Finally, a functioning app was created using the Streamlight library after the model had been built, allowing for a better user experience and graphical output.

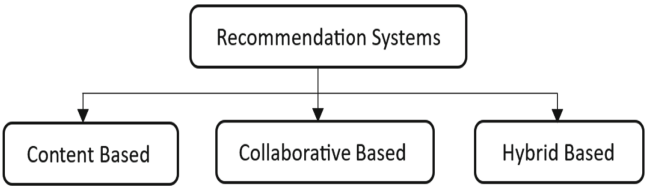

Fig. 1. Types of Recommendation System

	movie_id	title	overview	genres
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]

Fig. 2. Glimpse of Dataset (Tmdb Movie Dataset) Part-1

keywords	cast	crew	tags
[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCameron]	[In, the, 22nd, century,, a, paraplegic, Marin...
[ocean, drugabuse, exoticisland, eastindiatrad...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbinski]	[Captain, Barbossa,, long, believed, to, be, d...
[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMendes]	[A, cryptic, message, from, Bond's, past, send...
[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman]	[ChristopherNolan]	[Following, the, death, of, District, Attorney...
[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[AndrewStanton]	[John, Carter, is, a, war-weary,, former, mili...

Fig. 3. Glimpse of Dataset (Tmdb Movie Dataset) Part-2

2 Literature Review

Research on how to improve recommendation capacity by raising accuracy is ongoing in the field of recommendation systems. Through reciprocal cooperation of LSTM-CNN and a good recommendation, the work done with the aid of LSTM-CNN quickly increases the performance of the algorithm and offers a powerful tool. Given that CNN requires more processing [1], cluster computing is a good option for the algorithm's runtime.

The approach of developing the recommender system with the aid of ML algorithms also resulted in a lower RMSE value, as it was made to run multiple times to obtain the hyperparameters to tune the model to provide better accuracy with the aid of the ML framework, i.e., ML.NET, and another model built on using Microsoft Azure Machine Learning Studio, where it offered multiple ML solutions [2]. Making these systems constantly learn from the incoming new data is the best approach to enhance them. i.e., by regularly training the model to draw insights from recently received data. Another module, the big data module, can keep all of the user information and other relevant data for optimizing the model and delivering better results.

With the implementation of a straightforward neural network architecture model that performs well with root mean squared error and the retention of a collaborative filtering-based approach. The best and most effective tools for a variety of information retrieval are deep learning modules [3]. Here, regularisation was used to reduce prediction errors, and it worked well and produced improved suggestions. The deep learning module is substantially more scalable during testing than it is during training.

There has been and is still much effort being done to improve the recommender system used in movies, but this is not a static problem; with time and additional research, the efficiency and accuracy of the recommender system can be increased in a variety of ways. Here, when developing the system, consideration was given to rating, user consumption ratio, and user preference. K-means have been used to separate user tastes, and after doing so, the algorithm demonstrated 95% accuracy in predicting ratings from new users, which may be used to determine which movie should be recommended to new users [4]. The dataset from 12000 frequent customers has been employed in the system's workings to better comprehend and provide a variety of movie-going attitudes.

The movie recommender system can be built using a variety of datasets, and the model created using these datasets includes a dataset with numerous attributes that are

crucial for segmenting the movie. The collaborative recommendation system created for the chosen dataset uses a user-based co-coin similarity algorithm and singular value decomposition to provide individualized recommendations to active users [5].

As was previously noted, the subject of recommendation systems is always evolving. Designing a more powerful system requires multiple feature selection techniques with varied similarity measures, and enhancing the outcome requires feature selection filtering, says the report [6]. The value of support and confidence should be higher, such that few but effective recommendations should be displayed," according to the conditions for an effective suggestion [7].

Another strategy that might optimize the problem of poor accuracy suggestion can be overcome based on each user's interests such that it will lead to maximizing the accuracy and minimizing the topic difference between user interests is based on the user's interests [8]. One alternative strategy would thus be to get a large quantity of data from many sources, including the web, and apply techniques to it such as web scraping and extracting the integral data, and conducting filtering on it [9].

Another strategy for increasing performance is to integrate sequential recommendation models with content-based filtering and increase performance by highlighting deep connections between items by deconstructing content-based filtering [10]. Another way to increase its adaptability is to match the value of the centroid with the attribute value when a query is performed on Cluster Centroid in the database, and by any means, if that attribute value is not present, then according to the popularity the particular attribute can be recommended to the user make it a way to cold-start problem solution [11].

Finding the link between two attributes utilizing a set intersection between two things and predicting them for suggestions using content-based filtering is another creative method [12]. The function of the recommendation system is to open many new possibilities, such as in e-commerce, and many more, which leads to state-powerful marketing and advertisement which might lead to multiple sales and even more for the good of people and the owning firm[13]. It is true that recommender systems have become important information filtering systems and are still maintaining their supremacy with different types of alternatives such as content, collaborative, and hybrid case [14]. Other factors that might improve the model are user correlation and the kinds of phrases that users search for [15] in the accuracy of the system.

3 Proposed Methodology

Machine learning is an area of research that focuses on comprehending and developing techniques that use data to improve performance on a variety of tasks without explicitly following instructions. These techniques use algorithms and statistical models to analyze the data and identify trends and patterns. In many facets of computing, machine learning is commonly used to streamline and better facilitate operations. One machine learning method uses a recommender system, where a large number of algorithms work together to extract features from a large amount of data and create meaning from it. This ensures that the results produced are closely related to the data for which they were generated, ensuring that the model generates and maintains good accuracy. Contrary to that, it should not be maintained with the currently in use method, but rather, it should be continuously

upgraded by experimenting with new algorithms and comparing it to other algorithms for efficacy.

Here, in our model, we have used multiple algorithms to construct the model which includes NLP (Natural Language Processing) algorithms and distance algorithms i.e., Count Vectorizer, Porter Stemmer, and Cosine Similarity.

3.1 Count Vectorizer

It is a utility made available by the Python sci-kit-learn module. Natural Language Processing (NLP) and Text Analytics both employ the common feature extraction method known as the Count Vectorizer. It's a quick and easy method of turning a group of text documents into numerical feature vectors, where each dimension reflects the quantity (or count) of a certain word or token used in the document.

The Count Vectorizer algorithm operates as follows:

Text preprocessing*:* Cleaning and preparing the text data is the first stage in the text preprocessing process. To do this, all characters must be changed to lowercase, punctuation must be eliminated, and words must be stemmed or lemmatized to return them to their original form.

Tokenization*:* The text must then be separated into individual words or tokens, which can be accomplished using strategies like word or character tokenization. Building the vocabulary: Following that, the algorithm creates a vocabulary out of all the distinct words or tokens in the text input. A list of every word that appears in the book, coupled with an individual index for each term, makes up the vocabulary.

Encoding the documents: The next stage is to turn each written document into a numerical feature vector after the vocabulary has been established. This is accomplished by calculating the frequency of each word in each document's vocabulary and setting the frequency as the value of the relevant dimension in the feature vector.

Normalization: Using methods like TF-IDF (Term Frequency-Inverse Document Frequency), which gives uncommon words a greater weight and frequent words a lower weight, the counts may be normalized to avoid high-frequency terms from predominating the feature vectors.

Return the feature matrix: The feature vectors for all the text documents are then integrated into a single feature matrix, where each row corresponds to a text document and each column to a word in the vocabulary. To represent text data as numerical characteristics that can be utilized as input to machine learning algorithms, Count Vectorizer is frequently used in NLP applications such as sentiment analysis, document categorization, topic modeling, and more.

3.2 Porter Stemmer

The Porter-Stemmer method is employed in natural language processing for text normalization and word reduction. The Porter-Stemmer algorithm's major objective is to break down words into their stem, which facilitates word analysis and comparison. The Porter-Stemmer method eliminates affixes like prefixes and suffixes by applying several heuristics or rules to the ends of words. The word "running," for instance, would be

reduced to its stem, "run," via the Porter Stemmer algorithm. The "-ing" suffix and any additional affixes that can be eliminated based on the established guidelines are deleted to do this. To simplify words while maintaining their semantic meaning, the algorithm applies the rules in a certain sequence.

In NLP applications including document categorization, sentiment analysis, and text summarization, the Porter-Stemmer method is frequently employed. The technique can aid in lowering the dimensionality of the text data by breaking down words into their most basic forms, making them simpler to process and analyze. Furthermore, stemming can enhance the effectiveness of machine learning algorithms by lowering the number of unique words and data variance.

3.3 Cosine Similarity

A measure of similarity between two non-zero vectors in an inner product space is called cosine similarity. The quantitative comparison of two texts or collections of texts is frequently done in information retrieval (IR) and natural language processing (NLP). Measurement of the cosine of the angle between two vectors is the fundamental concept behind cosine similarity. The cosine value ranges from -1 to 1, with 1 denoting perfect similarity and -1 denoting complete dissimilarity.

The following is the formula for the cosine similarity between two vectors A and B: $\cos(\Theta) = (A * B)/(\|A\| * \|B\|)$where A and B are the vectors under comparison, ||A|| and ||B|| are the vector magnitudes, and "*" denotes the vectors' dot product.

The process by which cosine similarity operates is to first transform text data into numerical vectors, each of whose dimensions indicates the count or frequency of a particular word in the source material. The cosine of the angle between the vectors is then calculated by comparing them. The two vectors are more similar when the cosine value is higher, and more different when the cosine value is lower.

Cosine similarity is used in NLP and IR applications to compare the similarity between two documents by contrasting the vectors that describe the texts. As an illustration, given two documents D1 and D2, we may generate vectors for each one based on the frequency of terms in each document, and then use cosine similarity to assess how similar the two vectors are to one another (Fig. 4).

$$\text{cosine similarity} = S_c(A, B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

3.4 Category Distribution of Dataset

The Tmdb 5000 movie dataset offers a wide range of genres, allowing for the creation of an effective recommendation engine and the selection of films that are relevant to the content. The aforementioned graph shows how many different genres the movies in the Tmdb dataset contain, and it is obvious from the projection that the dataset's movies have the most drama, comedy, thriller, and action content (Fig. 5).

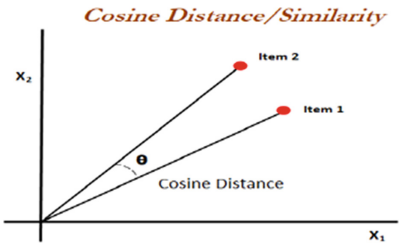

Fig. 4. Representation of Cosine Similarity algorithm

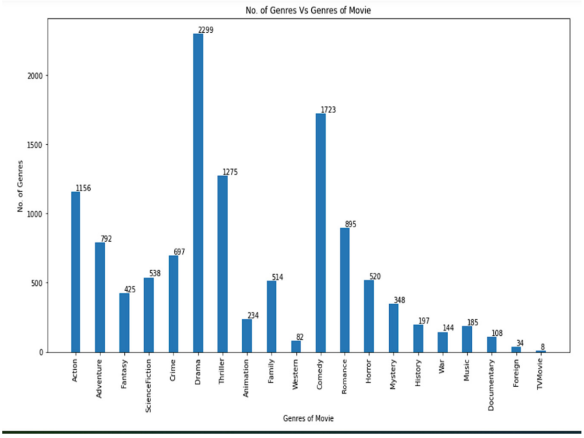

Fig. 5. Different Genres of Movies in the Dataset

Therefore, after closely examining the various film genres, we came to know that, aside from a few genres that the suggested films shared, the target film shared the same genre as the other top 5 films. Simply using the contradiction and the information on which we are working causes us to provide recommendations that also include additional variables like actor names, director names, and production names (Fig. 6).

4 Discussion

Content-Based Recommender System helps provide recommendations based on content that is being endorsed by the user. From a user's perspective, it cannot be guaranteed that he/she will like any other recommendation that is recommended based on assumption. So laying more stress on the user perspective, a content based recommender system can play a major role in the generation of content with holding minimum risk factors. In addition, content-based recommendation systems have inherent limitations, which is why collaborative and hybrid recommender systems are also available. However, in the beginning, stages of development, every tech service or product makes use of a content-based recommendation system to learn about users' preferences and mentalities. This is because, at the end of the day, users like to use the services, and if the user experience is poor, it will devalue the product or service, which could have unfavorable effects.

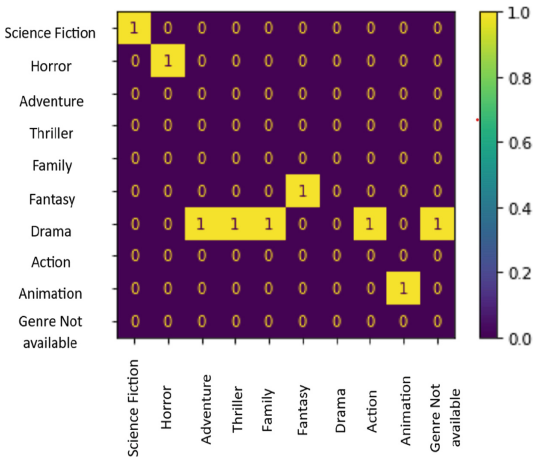

Fig. 6. Confusion Matrix on target and recommended genres on a particular movie "Avatar"

```
movie_recommendation('Avatar')
```

```
Target Movie Name ->  Avatar

-> Aliens vs Predator: Requiem
-> Aliens
-> Falcon Rising
-> Independence Day
-> Titan A.E.
```

Fig. 7. Recommendations similar to "Avatar"

```
movie_recommendation('Batman')
```

```
Target Movie Name ->  Batman

-> Batman
-> Batman & Robin
-> Batman Begins
-> Batman Returns
-> The R.M.
```

Fig. 8. Recommendations similar to "Batman"

Currently, recommendation systems are employed across all platforms, including those for entertainment, e-commerce, and technology, which improves user experience. Given this backdrop, it is obvious that recommender systems are crucial to improving the user experience and that they must continually improve to provide the best possible accuracy.

5 Results

To understand what happens when a movie is selected as the target and the recommended movies, i.e., the top 5 movies being recommended by the engine (Figs. 7, 8, 9 and 10).

5.1 Flow Architecture of Movie Recommendation System

The User Interface must first be launched to initiate the recommendation system. Here, you are prompted to choose a movie from the list of available films. The basic functioning of the system begins when the user selects a movie and clicks the "recommend" button. The system then analyses the type of content the user has chosen, generates its vectors and begins comparing that content with comparable movies. The output for movies is then shown, including movie names and their posters, on the U.I. The Streamlit framework is used to obtain the movie posters via the Tmdb API (Fig. 11).

5.2 Working Architecture of Content-Based Recommendation System

```
movie_recommendation('El Mariachi')
```

```
Target Movie Name ->  El Mariachi

-> Killers
-> Should've Been Romeo
-> I Am Sam
-> I Served the King of England
-> Take Me Home Tonight
```

Fig. 9. Recommendations similar to "El Mariachi"

```
movie_recommendation('The Dark Knight Rises')
```

```
Target Movie Name ->  The Dark Knight Rises

-> The Dark Knight
-> Batman Returns
-> Batman
-> Batman Forever
-> Batman Begins
```

Fig. 10. Recommendations similar to "The Dark Knight Rises Movie"

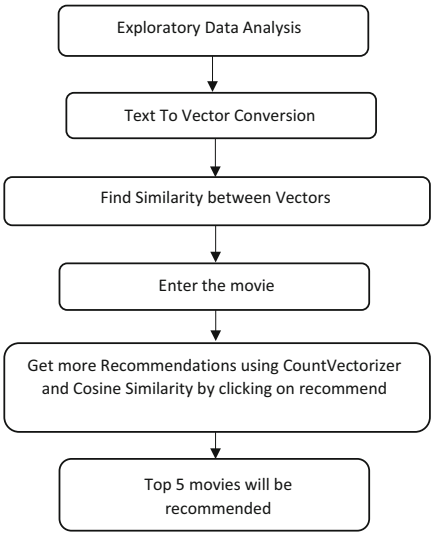

Fig. 11. Architecture of content-based recommender system

6 Conclusion

There has been a lot of progress made in the realm of product recommendations, such as for movies. The most effective recommendation algorithms are provided by digital giants like Netflix, Amazon, and YouTube. But since the process of making recommendations for any product is dynamic because every day more and more people sign up for the services and their needs can change, a recommendation system based on older algorithms won't be able to make recommendations or might make them, but only to a limited degree. So, with continued study, the effectiveness and accuracy may be enhanced.

In this study, we used one of the most popular datasets from the Kaggle library to develop a content-based recommendation system that accurately recommends movies based on factors like cast members, directors, and production names. We developed a user interface (UI) using Streamlit to replace this model and give a better perspective, clarity, and comprehension of what happens when a user selects a movie and the recommendations produced by this recommendation engine.

References

1. H. Wang, N. Lou, and Z. Chao, "A Personalized Movie Recommendation System based on LSTM-CNN," 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2020, pp. 485–490, doi: https://doi.org/10.1109/MLBDBI51377.2020.00102.
2. A. Fanca, A. Puscasiu, D. -I. Gota and H. Valean, "Recommendation Systems with Machine Learning," 2020 21th International Carpathian Control Conference (ICCC), 2020, pp. 1–6, doi: https://doi.org/10.1109/ICCC49264.2020.9257290.

3. 3. J. Lund and Y. -K. Ng, "Movie Recommendations Using the Deep Learning Approach," 2018 IEEE International Conference on Information Reuse and Integration (IRI), 2018, pp. 47-54, doi: https://doi.org/10.1109/IRI.2018.00015.
4. M. Ahmed, M. T. Imtiaz and R. Khan, "Movie recommendation system using clustering and pattern recognition Network," 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 2018, pp. 143–147, doi: https://doi.org/10.1109/CCWC.2018.8301695
5. Jose Immanuvel. J, Sheelavathi. A, Priyadharshan. M, Vignesh. S, Elango. K, "Movie Recommendation System."
6. Yassine Afoudi, Mohamed Lazaar, Mohamed Al Achhab 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS).
7. Tessy Badriyah, Sefryan Azvy, Wiratmoko Yuwono, Iwan Syarif 2018 International Conference on Information and Communications Technology (ICOIACT).
8. Qiusha Zhu, Mei-Ling Shyu, Haohong Wang 2013 IEEE International Symposium on Multimedia Year: 2013 | Conference Paper | Publisher: IEEE.
9. Mehmet Kayaalp, Tansel Ozyer, Sibel Tariyan Ozyer 2009 International Conference on Advances in Social Network Analysis and Mining Year: 2009| Conference Paper | Publisher: IEEE.
10. Yeongwook Yang, Hong-Jun Jang, Byoungwook Kim IEEE Access Year: 2020 | Volume: 8 | Journal Article | Publisher: IEEE
11. Parmar Darshna 2018 2nd International Conference on Inventive Systems and Control (ICISC) Year: 2018 | Conference Paper | Publisher: IEEE.
12. Ashish Pal, Prateek Parhi, Manuj Aggarwal 2017 Tenth International Conference on Contemporary Computing (IC3) Year: 2017 | Conference Paper | Publisher: IEEE.
13. Sanya Sharma, Aakriti Sharma, Yamini Sharma, Manjot Bhatia 2016 International Conference on Computing, Communication and Automation (ICCCA) Year: 2016 | Conference Paper | Publisher: IEEE.
14. Sarika Jain, Anjali Grover, Praveen Singh Thakur, Sourabh Kumar Choudhary International Conference on Computing, Communication & Automation Year: 2015 | Conference Paper | Publisher: IEEE
15. K. Funakoshi, T. Ohguro KES'2000. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies. Proceedings (Cat. No.00TH8516) Year: 2000 | Volume: 1 | Conference Paper | Publisher: IEEE.