Movie Recommendation System Using Content-Based Filtering

Pranshu Mishra
Department of Computer Science and Enginnering
Sharda School of Enginnring and Technology
Sharda University
Greater Noida,UP,India
2019649669.pranshu@ug.sharda.ac.in

Vishal Jain
Department of Computer Science and Enginnering
Sharda School of Enginnring and Technology
Sharda University
Greater Noida,UP,India
vishal.jain@sharda.ac.in

***Abstract*— The movie industry has seen a tremendous increase in the amount of content being produced in recent years, making it difficult for viewers to navigate and find movies that match their preferences. Recommendation systems have been developed in order to solve this problem, and one of the most effective approaches is content-based filtering. This research paper proposes a movie recommendation system using content-based filtering, the system utilizes the features of movies such as genre, director, actors, and plot summaries to recommend similar movies to users. The system is implemented using Python and utilizes a dataset of movies collected from Kaggle called The TMDB 5000 Movie Dataset and model is train using using cosine similarity. It demonstrates that the content-based filtering approach provides accurate and relevant recommendations to users. The results of this study suggest that content-based filtering is a promising approach to movie recommendation systems and can significantly improve the user experience in the movie industry.**

Keywords*— Recommending system for movies, TDMB Dataset, Python, Content-Based Filtering, Cosine Similarity.

I. INTRODUCTION

The movie industry has observed a significant rise in theamount of content being produced in recent years, making it difficult for viewers to navigate and find movies that match their preferences. This is a problem that has been plaguing the industry for some time now, as the amount of content being produced is far more than any one person can possibly keep up with. The solution to this issue has been the creation of recommendation tools that enable users to locate films that are pertinent to their interests.. One of the most effective approaches to this problem is content-based filtering. It is a approach that utilizes the features of items to make recommendations. In the case of movie recommendation systems, these features can include genre, director, actors, and plot summaries. By using these features, content-based filtering can identify similar movies that match the user's preferences and recommend them to the user. The effectiveness of this approach lies in the fact that it focuses on the content of the movies, rather than user behavior or social context. As a result, content-based filtering can provide more accurate and relevant recommendations to users. This study aims to present a content-based filtering-based movie recommendation system.. The proposed system will utilize the features of movies such as director, genre, actors, and plot summaries to recommend similar movies to users. The system will be implemented using Python and will utilize a dataset of movies collected from Kaggle called The TMDB 5000 Movie Dataset and model is train using using cosine similarity. This research paper is organized as follows. In the next section, we will review the literature on movie recommendation systems, including content-based filtering. In section three, we will describe the methodology used to implement the proposed system, including the dataset used and the algorithms employed. In section four, we will present the results of our evaluation of the system. In section five, we will discuss the implications of our results and the potential for further research in this area. Finally, in section six, we will conclude the paper and summarize our findings.

II. RELATED WORK

The author proposed work which includes making of Collaborative filtering-based movie recommendation system combining clustering and neighbour voting to predict user cosine similarity in a cluster dataset [1]. In this research, a movie recommendation system is developed using a Convolutional Neural Network model. The system is further enhanced by incorporating content-based recommendation and system filtering techniques. [2]. Recommendation systems help users choose movies based on genre and category, saving time and allowing them to narrow down potential movies to fit their tastes.[3]. This study proposes a Recurrent Neural Network (RNN) technique to improve movie recommendation systems, using parameter adjustment, early halting, and dropout regularization to increase accuracy and provide customized recommendations. [4]. This study focuses on a hybrid technique of content-based and collaborative filtering to improve movie sales by providing users with a selection of movies to watch [5]. DNN with Trust is the best model for re commending movies to users, overcoming cold start, malicious attacks and data sparsity of RS [6]. The aim of this research was to develop a movie recommendation system using a collaborative filtering approach. The proposed system utilizes the alternating least squared model to predict the most highly rated movies with a remarkable accuracy of 97% [7]. This study develops a recommendation system for movies using clustering techniques such as K-Means, birch, mini-batch mean-shift, K-Means, affinity propagation, spectral analysis and agglomerative clustering. The goal is to extract emotions from user-generated data using WordNet, lexical ontology, and psychology. The efficacy of the emotion prediction model is assessed and contrasted with an item similarity model based on ratings [8]. Recommendation System is a popular tool used by companies like Facebook, LinkedIn, Pandora, Netflix, and Amazon to increase their revenue and benefit their customers. This paper reviews state-of-the-art methods for movie recommendation [9]. This work makes use of Twitter to process multilingual tweets in real-time to create a movie recommendation system. The system achieves 91.67 percent accuracy, 92 percent precision, 90.2 percent recall, and 90.98 percent f-measure, all outstanding results. To improve the system's performance, sentiment analysis techniques are used.. [10]Using real-time, multilingual tweets and sentiment analysis, Twitter is used to build a movie recommendation engine. [11]. Clustering techniques such as mini-batch K-Means, K-Means, birch, mean-shift, affinity propagation, spectral analysis and agglomerative clustering are used to develop a recommendation system for movies.[12].An explanatory computational model was developed to apply social explanations, which can enhance how a system and interaction are viewed. [13]. Utilizing measures like execution time, RMSE, and rank, a movie recommendation system based on ALS is evaluated for performance. [14]. This work uses the Movielens dataset from Kaggle to create a movie recommendation system utilising the K-Means Clustering and K-Nearest Neighbor algorithms.[15]. A recommendation system uses genre correlation and content-based filtering to propose resources based on a dataset, using R to analyse the Movie Lens dataset.[16]. A hybrid recommendation system combining collaborative filtering, content-based filtering, and sentiment analysis to improve movie selections.[17]. Collaborative filtering based on cosine similarity and K-NN algorithms is used to improve the effectiveness and precision of a standard filtering technique.. [18]. Virtual opinion leaders and Weighted Slope One-VU approach cluster users according to attributes. [19].Movie recommendation systems aim to help movie buffs by combining content-based and collaborative approaches.. [20]. Recommendation systems have been developed to help locate services or items based on user interests, with three methods suggested: straightforward, content-based filtering, and also collaborative-based filtering [21].

III. Methods, Background and Details

A. Algorithm

- *Content-Based Filtering*

Content-based recommendation systems generate recommendations based on the user's profile and preferences. They strive to connect customers with products that were loved previosly. The characteristics of products that the user likes serve as a broad basis for determining the degree of similarity between items. Content-based models put more emphasis on the ratings supplied by the target user themselves than the majority of models for collaborative filtering that leverage user ratings for the goal and other users. In essence, the content-based strategy makes use of many data sources to provide suggestions.

The following sources of data are important for the most fundamental kinds of content-based systems (the complexity of the system you're trying to construct may raise these requirements):

1. Item level data source — It is necessary to have a trustworthy source of data about the item's characteristics. We have information for our situation, such as book price, number of pages, published year, etc. The more details you have on the thing, the better for your system it will be.
2. User level data source — For the product suggestions are made, it is required to have some kind of customer feedback. Either implicit or explicit input is possible at this level. User reviews of books they've read are the source of our sample data. It will be better for your system if you can track more user input.

- *Advantages*

Whenever there is a dearth of rating information, content-based models are most useful for making recommendations. This is due to the possibility that the user has rated other things with comparable characteristics.A model should be able to produce suggestions by utilising both the ratings and the item properties even in the absence of abundant data.

- *Disadvantages*

Content-based systems have two key drawbacks.

1. Based on the things or materials the consumer has consumed, the suggestions are kind of "obvious." Thus this has the drawback of never recommending a certain sort of item to the user if they have never interacted with that item. For instance, with this strategy, you will never be suggested mystery books if you've never read them. This is due to the model's user specificity,

which prevents it from utilising information from comparable users. This has a negative impact on many businesses since it causes the diversity of the proposals to decline.

2. For making suggestions to new users, they are useless. To create a model, the objects must have an explicit or implicit history of user-level data. A big ratings dataset should often be available in order to generate accurate forecasts without overfitting [16].

- *Cosine-Similarity*

The cosine of two non-zero vectors which are Y and Z may be calculated using the Euclidean dot product formula (Equation 1)

$$\boldsymbol{Y}.\boldsymbol{Z} = ||\boldsymbol{Y}||\ ||\boldsymbol{Z}||\ cos\theta$$

Equation 1.

The cosine similarity, cos(), is represented by a linear combination and magnitude in Equation 2, where Y and Z are two vectors of characteristics.

$$cosine\ similarity = S_C(Y,Z) \coloneqq \cos(\theta) = \frac{\boldsymbol{Y}.\boldsymbol{Z}}{||Y||||Z||}$$

$$= \frac{\sum_{i=1}^{n} Y_i Z_i}{\sqrt{\sum_{i=1}^{n} Y_i^2}\sqrt{\sum_{i=1}^{n} Z_i^2}}$$

Equation 2.

Where Y_i and Z_i are the corresponding components of vectors Y and Z.

B. Framework – Streamlit.

Open source, Python-based framework called Streamlit is available which facilitates the rapid development of web applications for machine learning and data science. It works with several popular Python libraries, including PyTorch, scikit-learn, SymPy (latex), pandas, Keras, Matplotlib and NumPy, among others [19].

C. Dataset

The TDMB movie dataset is a collection of information about movies and the characteristics of such films, including the director, actors, genre, and year of release. It has more than 10,000 films from diverse nations and tongues. The collection also contains statistics on box office earnings and reviews from many websites, including IMDb and Rotten Tomatoes. The TDMB dataset is frequently utilised in studies on sentiment analysis, box office forecasting, and recommendation systems for films. It is an open-source dataset that can be used for non-commercial uses by scholars and data scientists. It is a useful tool for research on movies due of its accessibility and comprehensiveness. A preview of the dataset is shown in Fig.1.

5.00 - 45953.30 Count: 3,405 5 459k	4800 unique values	4761 unique values	4776 unique values
19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "credit_id": "5602a8a7c3a3685532001c9a", "gender": 2, "...	[{"credit_id": "52fe48009251416c750aca23", "department": "Editing", "gender": 0, "id": 1721, "job": ...
285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Sparrow", "credit_id": "52fe4232c3a36847f800b50d", "gende...	[{"credit_id": "52fe4232c3a36847f800b579", "department": "Camera", "gender": 2, "id": 120, "job": "D...
206647	Spectre	[{"cast_id": 1, "character": "James Bond", "credit_id": "52fe4d22c3a368484e1d8d6b", "gender": 2, "id...	[{"credit_id": "54805967c3a36829b5002c41", "department": "Sound", "gender": 2, "id": 153, "job":

Fig.1(Dataset)

IV. Implementation

The suggested movie recommendation system will suggest movies based on user interests and preferences. It will have a user query input, a movie database, a content-based filtering algorithm, a recommendation engine, a user interface design, a web platform deployment, and relevant libraries. The algorithm will calculate the similarity between different movies and recommend movies that are most similar to the user's preferences. The recommendation engine will use the output of the content-based filtering algorithm to provide personalized recommendations. The system will be deployed on a web platform, allowing users to access it from anywhere with an internet connection.. Fig 2 shows the flowchart of the whole process.

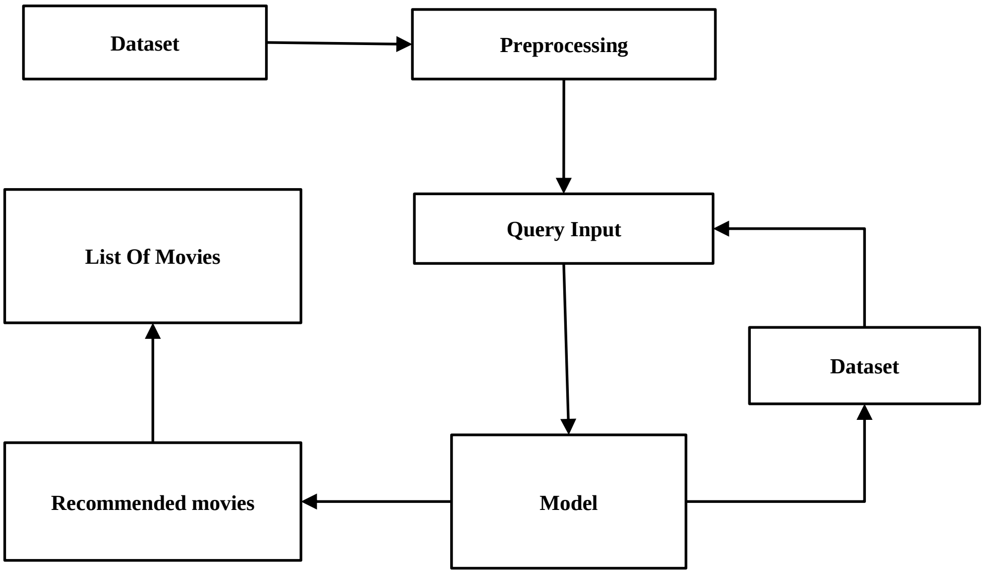

Fig 2.(Flowchart – Content Based Filtering)

V. Experiments And Result Analysis

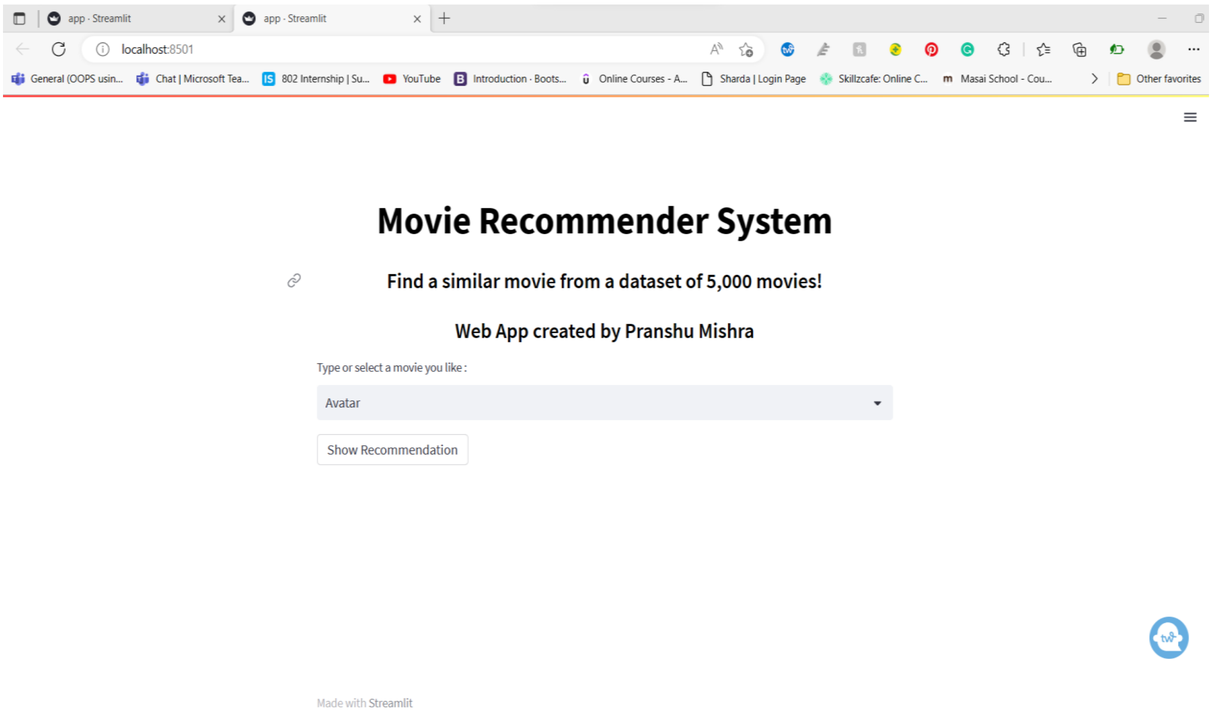

Fig.3(Course Recommendation App – Home Page)

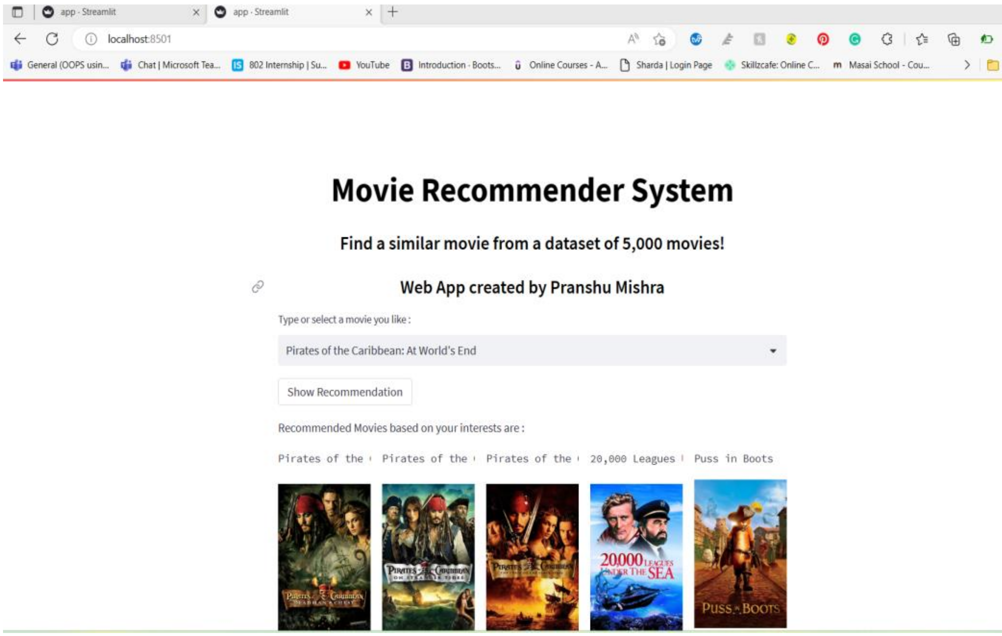

Fig.4(Course Recommendation App – Recommend)

The illustrated figure 3 shows the homepage of the streamlit app and figure 4 shows the recommendation page, here the result from the Streamlit web application, where the recommendations related to the query is input. In this case the query input is Python, the query is not can sensitive i.e. if written as PYTHON, python, PyThOn, it will always yield the same result.

VI. CONCLUSION AND FUTURE SCOPE

Personalized recommendations based on user preferences and movie attributes.. It uses a mathematical approach to extract features from the movie database and user profile, calculate similarity scores, and produce suggestions depending on the user's preferences. However, the system has limitations, such as the lack of diversity in recommendations and the inability to capture serendipitous recommendations. The proposed model has several future scope opportunities, such as hybrid approaches, deep learning, real-time recommendations, and sentiment analysis. Hybrid approaches can improve the accuracy and diversity of the recommendations by combining the strengths of different recommendation approaches, deep learning can extract features from the movie database and user profile, real-time recommendations can improve the user experience by providing instant and relevant recommendations, and sentiment analysis can identify the user's emotions and mood.

VII. REFERENCES

[1] Venugopal, M. B. (2023). COLLABORATIVE FILTERING OF MOVIE RECOMMENDATION SYSTEM. In Dogo Rangsang Research Journal UGC Care Group I Journal (Vol. 13, Issue 2).

[2] He, H., Shang, Z., Wu, M., & Zhang, Y. (2023). Movie Recommendation System Based on Traditional Recommendation Algorithm and CNN Model. In Highlights in Science, Engineering and Technology CSIC (Vol. 2022).

[3] Iyer, P. K., & Pokharel, R. (2023). Personalized Movie Recommendation System. International Journal of Information Technology, 2(1). https://doi.org/10.5281/zenodo.7779051

[4] MOVIE RECOMMENDATION SYSTEM USING OPTIMIZED RNN APPROACH. (2023). International Research Journal of Modernization in Engineering Technology and Science. https://doi.org/10.56726/IRJMETS34674

[5] Sheikh, A., Anjum, I., & Shaikh, A. A. (2022). A REVIEW STUDY ON MOVIE RECOMMENDATION SYSTEM. In International Journal of Creative Research Thoughts (Vol. 10). www.ijcrt.org

[6] Choudhury, S. S., Mohanty, S. N., & Jagadev, A. K. (2021). Multimodal trust based recommender system with machine learning approaches for movie recommendation. International Journal of Information Technology (Singapore), 13(2), 475–482. https://doi.org/10.1007/s41870-020-00553-2

[7] Awan, M. J., Khan, R. A., Nobanee, H., Yasin, A., Anwar, S. M., Naseem, U., & Singh, V. P. (2021). A recommendation engine for predicting

movie ratings using a big data approach. Electronics (Switzerland), 10(10). https://doi.org/10.3390/electronics10101215
[8] Saraswat, M., Chakraverty, S., & Kala, A. (2020). Analyzing emotion based movie recommender system using fuzzy emotion features. International Journal of Information Technology (Singapore), 12(2), 467–472. https://doi.org/10.1007/s41870-020-00431-x
[9] Goyani, M., & Chaurasiya, N. (2020). A Review of Movie Recommendation System: Limitations, Survey and Challenges. Electronic Letters on Computer Vision and Image Analysis, 19(3), 18–37. https://doi.org/10.5565/rev/elcvia.1232
[10] Singh, T., Nayyar, A., & Solanki, A. (2020). Multilingual Opinion Mining Movie Recommendation System Using RNN. In Lecture Notes in Networks and Systems (Vol. 121, pp. 589–605). Springer. https://doi.org/10.1007/978-981-15-3369-3_44
[11] Singh, R. H., Maurya, S., Tripathi, T., Narula, T., & Srivastav, G. (2020). Movie Recommendation System using Cosine Similarity and KNN. International Journal of Engineering and Advanced Technology. https://doi.org/10.35940/ijeat.E9666.069520
[12] Putri, D. C. G., Leu, J. S., & Seda, P. (2020). Design of an unsupervised machine learning-based movie recommender system. Symmetry, 12(2). https://doi.org/10.3390/sym12020185
[13] Pecune, F., Murali, S., Tsai, V., Matsuyama, Y., & Cassell, J. (2019). A model of social explanations for a conversational movie recommendation system. HAI 2019 - Proceedings of the 7th International Conference on Human-Agent Interaction, 135–143. https://doi.org/10.1145/3349537.3351899
[14] Aljunid, M. F., & Manjaiah, D. H. (2019). Movie Recommender System Based on Collaborative Filtering Using Apache Spark. In Advances in Intelligent Systems and Computing (Vol. 839, pp. 283–295). Springer Verlag. https://doi.org/10.1007/978-981-13-1274-8_22
[15] Ahuja, R., Solanki, A., & Nayyar, A. (2019). Movie recommender system using k-means clustering and k-nearest neighbor. Proceedings of the 9th International Conference On Cloud Computing, Data Science and Engineering, Confluence 2019, 263–268. https://doi.org/10.1109/CONFLUENCE.2019.8776969
[16] Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S., & Venkatesh, B. (2019). Content-based movie recommendation system using genre correlation. Smart Innovation, Systems and Technologies, 105, 391–397. https://doi.org/10.1007/978-981-13-1927-3_42
[17] Kumar, S., Halder, S. S., De, K., & Roy, P. P. (2018). Movie Recommendation System using Sentiment Analysis from Microblogging Data. http://arxiv.org/abs/1811.10804
[18] Zhang, J., Wang, Y., Yuan, Z., & Jin, Q. (1007). Personalized Real-Time Movie Recommendation System: Practical Prototype and Evaluation. http://121.42.174.147:8080/
[19] Furtado, F. (n.d.). Movie Recommendation System Using Machine Learning. https://doi.org/10.22105/riej.2020.226178.1128
[20] Nagarjuna College of Engineering and Technology, & Institute of Electrical and Electronics Engineers. (n.d.). 2019 Global Conference for Advancement in Technology (GCAT) : Bangalore, India, Oct 18-20, 2019.
[21] Gupta, M., Thakkar, A., Gupta, V., & Rathore, D. P. S. (2020, July). Movie recommender system using collaborative filtering. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 415-420). IEEE.