International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 02, ISSN No. 2455-2143, Pages 113-122
Published Online June 2023 in IJEAST (http://www.ijeast.com)

MOVIE RECOMMENDER SYSTEM USING MACHINE LEARNING

Pulkit Gupta, Vaniya Batra, Anusuya Sharma, Disha Narula, Rashmi Tiwari
HMR Institute of technology and Management, New Delhi, India

Abstract: **The recommendation system of today has transformed the way we search for items of our interest through the use of an information filtering approach that predicts user preferences. A movie recommendation system named RECOM has been proposed in this paper. Based on the content-based filtering approach, it utilizes the information in the dataset, undergoes analysis, and recommends the most suitable movies for the user. The recommended movie list is ordered according to the IMDB ratings calculated by a standard formula. The system also enables users to search for movies based on their favourite actors/actresses. Developed in Python and Machine Learning, the recommender system generates recommendations through the utilization of various forms of knowledge and data about movies, such as vote count, vote average, mean, quantile, etc. Overall, the effectiveness and efficacy of movie searches have significantly increased thanks to the introduction of content-based filtering and machine learning techniques in movie recommendation systems, making it simpler for users to locate films they are likely to enjoy.**

Keywords: **Recommendation system, RECOM, recommender system, movies, IMDB ratings, Collaborative filtering, Content-based filtering.**

I. INTRODUCTION

The internet has become a crucial aspect of human life and users often face the challenge of having too much choice. When searching for a hotel or investment options, for example, there is an overwhelming amount of information available. To help users deal with this information overload, companies have implemented recommendation systems to provide guidance. Despite being researched for several decades, the interest in recommendation systems remains high due to the numerous practical applications and rich problem domain.

Several online recommendation systems have been implemented and are widely used, such as the book recommendation system at Amazon.com, the movie recommendation system at MovieLens.org, and the CD recommendation system at CDNow.com. These systems have added value to the economy of e-commerce websites like Amazon.com and Netflix, making them a key component of their websites. The profit made by some websites because of these systems can be seen in the table below.

Netflix	[1] 2/3rd of the movies watched are recommended
Google News	[2] recommendations generate 38% more click-troughs
Amazon	[3] 35% sales from recommendations
Choicestream	[4] 28% of the people would buy more music if they found what they liked

Fig 1. Companies benefit through recommendation system

The economic potential of recommender systems has resulted in their integration into some of the largest e-commerce websites and online movie rental companies, such as Amazon.com and snapdeal.com. Personalized recommendations of high quality have been added to enhance user experience. Recently, web-based personalized recommendation systems have been utilized to offer various forms of tailored information to users. These systems are now widely applied in a range of applications.

We can classify the recommender systems in two broad categories:
1. Collaborative filtering approach
2. Content-based filtering approach

Collaborative filtering

The [5] term "collaborative filtering" (CF) was first used in 1992 by Goldberg et al. The proposal was that "information filtering could be made more effective through human involvement in the filtering process." The concept of collaborative filtering, as it is commonly understood today, was then introduced two years later by Resnick et al.

It was theorized that users tend to like what like-minded users like, where the likeness of two users was determined by their ratings of items. When like-minded users were identified, items that one user rated positively were recommended to the other, and vice versa. The use of collaborative filtering was found to have three advantages compared to content-based filtering. Firstly, [6] it was content independent and did not require error-prone item processing. Secondly, it considered real quality assessments, as ratings were provided by humans. Lastly, serendipitous recommendations were expected as they were not based on item similarity but on user similarity. Among the reviewed approaches, only 18% applied collaborative filtering. [7] However, a main problem with CF is the low motivation for user participation, referred to as the "cold start" problem, which may arise in new users, new items, or new communities/disciplines. To overcome this, implicit ratings may be inferred from interactions between users and items, but this negates CF's advantage of being based on real user quality assessments. [8] Inferring implicit ratings from interactions such as page views, downloads, or citations may be misguiding, as they could also indicate difficulty in understanding the item. As a result, the advantage of explicit human quality assessments mostly disappears when implicit ratings are used.

The [9] second advantage of being content-independent for CF might also be voided by using citations as inferred ratings. Typically, access to reliable citation data is not widely available, and therefore access to the paper's content is necessary to build a citation network, which can be even more prone to error than the extraction of words in CBF. In CBF, the extraction of the text of the papers is required, possibly including identification of fields such as the title or abstract. For citation-based CF, not only must the text be extracted, but the bibliography and its individual references must also be identified, along with the various fields, including the title and author. This work is prone to mistakes.

The [10] issue of sparsity is commonly encountered in the application of collaborative filtering to research paper recommendation systems. A study conducted by Vellino found that the sparsity level in Mendeley, a platform for research papers, was significantly higher compared to that in Netflix, a platform for movies. This disparity is attributed to the difference in the ratio of users to items in these domains. In the movie recommendation domain, there are typically fewer items and more users, such as in the case of the Movie Lens movie recommender which has 65,000 users and 5,000 movies. This allows for effective recommendations to be made based on the commonality of movies watched by many users. However, in the domain of research papers, there are usually fewer users but millions of papers and only a small number of users have rated the same papers, making it challenging to find like-minded users and to recommend papers. Additionally, many papers are not rated by any users, making them unable to be recommended.

Content-based filtering

One [11] of the most widely researched and utilized forms of recommendation is content-based filtering (CBF). In CBF, the process of user modelling plays a central role, where users' interests are derived from the items they interact with. Such "items" are usually textual in nature, such as emails or webpages, and interaction is typically established through actions such as downloading, buying, authoring, or tagging the item. The representation of items involves a content model that contains their features, which are typically word-based, including single words, phrases, or n-grams. [12] Additionally, some recommender systems also utilize non-textual features, such as writing style, layout information, and XML tags. Only the most descriptive features are usually selected to model both items and users and are typically given a weighting. The features and their weights are commonly stored as a vector in the user model, which consists of the features of items associated with the user. For generating recommendations, comparison between the user model and recommendation candidates is made, often utilizing the vector space model and cosine similarity coefficient. This approach has been proposed in various research papers and is often used in combination with other approaches in Hybrid Recommender Systems. A [13] study by Eyjolfsdottir et. al, which aimed to recommend movies through MOVIEGEN, had certain limitations such as being time-consuming and stressful due to the series of questions asked of the user. To address these shortcomings, a movie recommendation system called RECOM was developed, in which the user is given the option to select their choices from a set of attributes, including actor, director, genre, year, and rating, and the recommendations are made based on the information provided by the user themselves and their previous visited history.

The [14] use of plain words as features has been utilized in most of the reviewed approaches. Some approaches have used n-grams, topics that arose from social tags on CiteULike, and concepts inferred from the Anthology Reference Corpus through Latent Dirichlet Allocation and assigned to papers by machine learning. The utilization of non-textual features has been done by a few approaches and, if utilized, they were usually employed in addition to words. The citation weighting with the standard TF-IDF measure was used by Giles et al. in the same way as words were used, referred to as the CCIDF method. Some other approaches have adopted or used the CC-IDF idea as a baseline. However, Beel has recently provided some evidence that CC-IDF may not be the ideal weighting scheme. The authors were considered as features by Zarrin kalam and Kahani and the similarity was determined by the number of authors shared by two items.

The limitations of content-based filtering include the need for significant computational resources, as each item must undergo analysis of its features, user models must be constructed, and similarity calculations must be carried out. If there are numerous users and items, these calculations can become resource-intensive. One of the weaknesses of content-based filtering is its low level of serendipity and tendency towards over-specialization, leading it to recommend items that are very similar to those a user is already familiar with. Additionally, content-based filtering does not take into account factors such as the quality or popularity of items. For example, two research papers could be considered equally relevant by a CBF recommender system if they contain the same terms as the user model, even if one paper was written by an expert in the field and presents original results while the other was written by a student who paraphrased other research. Ideally, a recommender system should recommend only the first paper, but a CBF system may not be able to do so. Furthermore, the accuracy of content-based filtering is dependent on the availability of features for the items, which can be a challenge for research-paper recommendations as the process of converting PDFs to text, identifying document fields, and extracting features such as terms may introduce errors into the recommendations.

II. LITERATURE REVIEW

In [15] an in-depth introduction to the field of data mining, covering both the theoretical foundations and practical techniques used in the process. It includes chapters on association rule mining, clustering, and classification, as well as an overview of data warehousing and pre-processing.

In [16], explores the use of personalized recommendations in the e-commerce travel industry. The paper discusses the importance of personalization in travel decision-making and describes a system that uses a combination of demographic information and browsing history to generate personalized travel recommendations for users. The authors also evaluate the system through a user study and report the results.

In [17], an overview of the concepts and techniques used in the process of discovering patterns and knowledge from large data sets have been explained. It covers topics such as association rule mining, clustering, classification, and anomaly detection, as well as the important problem of data pre-processing. The book also includes an introduction to data warehousing and online analytical processing (OLAP) and provides case studies from different domains to illustrate the concepts. Throughout the book, the authors aim to make the material accessible to readers with a variety of backgrounds, including those with little or no background in data mining. In addition, the book includes a wealth of examples, figures, and exercises to help readers understand and apply the concepts. The book is intended for students, researchers, and practitioners in the field of data mining, and for anyone interested in understanding the process .

In [18], it is a research paper that presents a study on using data mining techniques for customer classification in retail marketing. The paper focuses on the use of customer purchase history data to classify customers into different segments based on their purchase behaviour. The authors propose a classification model that uses decision tree algorithm and applies it to a real-world dataset of a retail store. The study shows that the proposed model can classify customers into different segments with high accuracy and provide useful insights into customer behaviour. The results of the study indicate that the model can be used to identify loyal customers, target customers, and at-risk customers, which can help retailers in developing effective marketing strategies. The paper concludes that data mining can be a useful tool in retail marketing for customer classification and can help retailers to improve.

In [19], it is a research paper that describes a system for automatically indexing scientific literature and creating citation links between documents. The authors propose CiteSeer, a system that uses natural language processing techniques to extract citation information from scientific papers and build a database of citation links. CiteSeer is designed to improve the accessibility of scientific literature by providing a searchable database of citations.

In [20], describes a system for recommending research papers to users based on their reading history and interests. The system uses a combination of collaborative filtering and content-based filtering to make recommendations. Collaborative filtering is used to make recommendations based on the reading behaviour of similar users, while content-based filtering is used to make recommendations based on the content of the papers. The authors of the paper evaluated the system using a dataset of papers from the field of computer science and found that it was able to make relevant recommendations to users. They also described a user interface for the system that allows users to easily view and interact with the recommendations. The user interface is designed to be simple and easy to use.

In [21], describes a method for using citation data to identify relevant papers for a given research topic. The authors propose using machine learning techniques to model the citation behaviour of experts in each field, and then using these models to recommend papers to researchers working on similar topics. They evaluate their method using a dataset of papers from the field of computational linguistics and show that it can accurately predict the papers that experts in the field would cite. The authors argue that their method has the potential to improve the literature search process for researchers, making it easier for them to find relevant papers for their work.

In [22], describes a system called CiteSeer, which uses machine learning techniques to automatically retrieve and identify interesting publications from the web. Cite Seer uses a combination of techniques such as natural language processing, information retrieval, and data mining to find and extract information from scientific publications. The system also uses data from citation networks to identify important papers in each field. The authors evaluate the system using a dataset of computer science publications and show that it can accurately identify relevant papers and recommend them to users. The authors argue that their system has the potential to improve the literature search process for researchers, making it easier for them to find relevant papers for their work.

In [23], aims to develop a method for modelling scientific publications using mixed-membership models. These models are used to identify latent structures in the data, such as groups of authors or topics, and to determine the probability that a given author or topic belongs to each group. In this case, the authors apply the mixed-membership model to a dataset of scientific publications in order to identify latent structures in the data and to provide a probabilistic characterization of the relationships between authors and topics. The authors evaluate the performance of the mixed-membership model by comparing it to other methods for modelling scientific publications, such as Latent Dirichlet Allocation and Latent Semantic Analysis. They found that the mixed-membership model outperforms these other methods in terms of its ability to identify latent structures.

In [24], presents a method for creating a paper recommender system using key phrases. The authors propose a method that is based on a combination of Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) to extract key phrases from scientific papers and then use them to make recommendations. The key phrases are used as the basis for recommending papers to users, by matching the key phrases of the papers the user has read with those of other papers in the corpus. The authors evaluate the performance of their system using a dataset of scientific papers and show that it outperforms other methods for paper recommendation. They also show that the system can make recommendations that are relevant to the user's interests, as well as providing a high level of diversity in the recommendations. Additionally, the authors also demonstrate that the system can be easily integrated with other systems, such as a citation network analysis. Overall, the proposed system is a simple yet efficient approach to recommend scientific papers by matching the key phrases of the papers the user has read with those of other papers in the corpus..

In [25], describes a method for recommending academic papers to users based on their reading purposes. The authors propose a system that utilizes both the content of the papers and the user's reading purpose to make recommendations. The system first uses Latent Dirichlet Allocation (LDA) to extract the topics from the papers, and then uses a clustering algorithm to group papers based on these topics. Users are then asked to provide their reading purpose, which is used to match them with the appropriate cluster of papers. The authors evaluate their system using a dataset of academic papers and show that it outperforms traditional content-based recommendation systems. The authors argue that traditional content-based recommendation systems do not consider the user's reading purpose, and therefore they may not be providing the most relevant recommendations. By considering the user's reading purpose in addition to the content of the papers, their system can provide more accurate recommendations. The authors also point out that their system can recommend papers that may not have been considered by traditional systems due to their lack of keywords. They conclude by suggesting that their system can be used to improve the efficiency and effectiveness of academic paper recommendation.

In [26], describes the challenges and potential pitfalls of recommending research papers. The authors present a case study of a recommendation system used by a research group in computer science, and they identify several problems with the system. One of the problems is that the system often recommends papers that are not highly relevant to the user's interests, which can lead to confusion and frustration. Additionally, the system sometimes recommends papers that have already been read by the user, which can make the user feel that the system is not paying attention to their preferences.

In [27], discusses the use of ontologies in recommender systems to capture knowledge of user preferences. An ontology is a formal representation of a set of concepts and their relationships within a specific domain. The authors argue that using ontologies in recommender systems can improve the accuracy and effectiveness of recommendations by providing a more complete and accurate representation of user preferences. The paper presents a case study of a music recommender system that uses an ontology to capture knowledge of user preferences. The system was evaluated by a group of users, and the results showed that the use of an ontology improved the accuracy and effectiveness of the recommendations. The authors conclude that the use of ontologies in recommender systems can provide a more complete and accurate representation of user preferences, leading to better recommendations.

In [28], presents a citation recommendation system called SemCiR that is based on a novel semantic distance measure. The authors argue that traditional citation recommendation systems have limitations due to the use of simple keyword-based methods and propose a new system that uses a semantic distance measure to improve the accuracy and effectiveness of recommendations. The proposed SemCiR system uses a combination of Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) to measure the semantic distance between papers. This semantic distance measure is then used to make recommendations for new papers to cite, based on the similarity of the papers.

In [29], provides an overview of collaborative filtering (CF) in recommender systems. CF is a technique that uses the preferences of a group of users to make recommendations to other users. The authors discuss the different types of CF methods, including user-based CF and item-based CF, and the trade-offs between them. They also cover various techniques for improving the performance of CF recommenders, such as neighbourhood-based CF, model-based CF, and hybrid Fathe paper also discusses some of the challenges associated with CF recommenders, such as scalability and the cold-start problem. The scalability issue occurs when the number of users or items in the system is large, making it difficult to compute the similarity between users or items. The cold-start problem refers to the difficulty of making recommendations for new users or items that have not yet been rated. The authors suggest some solutions to these challenges, such as dimensionality reduction, parallelization, and the use of auxiliary information.

In [30], presents a method for using user-generated text to improve the performance of recommendation systems. The author argues that traditional recommendation systems rely on explicit feedback, such as ratings, which may not fully capture a user's preferences. User-generated text, such as reviews or comments, can provide additional information about a user's preferences. The method proposed in the paper uses natural language processing techniques to extract information from user-generated text and incorporate it into the recommendation process. The author presents a case study in which user-generated text from a movie review website is used to improve the accuracy of movie recommendations. The results showed that incorporating user-generated text improved the performance of the recommendation system.

In [31], gives an overview of various algorithms used to measure the structural similarity between documents. The author argues that the structural similarity between documents is an important factor for many information retrievals tasks such as document clustering, document classification, and text summarization. The paper covers several algorithms that have been proposed to measure structural similarity such as Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA) and Latent Semantic Indexing (LSI).

In [32], presents a collaborative filtering system for recommending content to users. Collaborative filtering is a method that uses the preferences of a group of users to make recommendations to other users. The authors argue that collaborative filtering can be used to create a personalized information service, or "information tapestry," that adapts to the preferences and interests of individual users. The authors describe a prototype system called Tapestry that uses collaborative filtering to recommend news articles to users. The system uses a user's past reading history to make recommendations for new articles. The authors present the results of a user study that evaluated the effectiveness of the Tapestry system.

In [33], presents a novel approach for recommending research papers to users based on mind-map user modelling. Mind-maps are a visual representation of concepts and their relationships, often used in brainstorming and note-taking. The authors argue that mind-maps can provide a more intuitive and expressive way of representing user interests compared to traditional user modelling methods. The authors describe a prototype system that uses mind-maps to model user interests and make recommendations for research papers.

In [34], describes various methods for analysing and classifying multivariate data. Multivariate data refers to data that contains more than one variable, such as observations made on multiple characteristics of an object or individual. The author presents several methods for analysing and classifying multivariate data, including principal component analysis (PCA), factor analysis, and cluster analysis. CA is a technique used to reduce the dimensionality of a dataset by identifying the directions of maximum variance in the data. Factor analysis is a technique that uses a statistical model to explain the correlations among a set of variables in terms of a smaller number of underlying factors.

In [35], describes a technique for summarizing multivariate data using clustering. The technique involves grouping similar data points together, called clusters, to reduce the dimensionality of the data and make it more manageable for analysis. The authors describe the method as being useful for identifying patterns in the data that might not be obvious from visual inspection.

III. METHODOLOGY

This section discusses our experimental setup. In this research work, our final data frame is textual data, we need to parse it into numerical or floating values to feed as inputs in machine learning algorithms. This process is called **(feature extraction | vectorization)**.

Model Building:

Our model should be capable of finding the similarity between movies based on their IMDB rating. Our Recommender model takes a movie title as input and predicts top-n most similar movies based on the IMDB rating. Here we will use the concept of Cosine distance to calculate the similarity of movies.

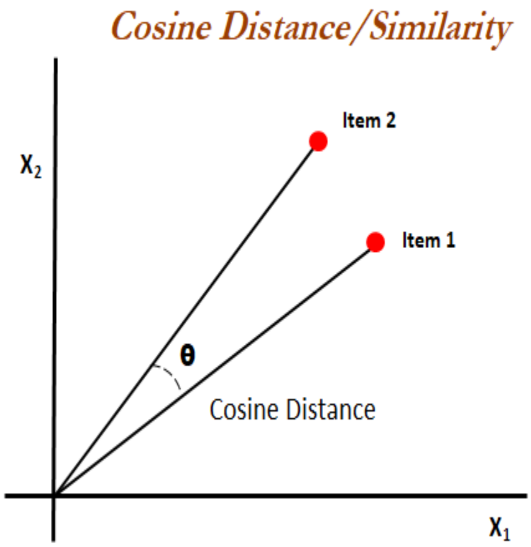


Fig.2 Cosine Similarity Distance. (Ref: https://bit.ly/2X5470I)

- sklearn provides a class for calculating pair wise cosine similarity.
- cosine_sim is a 2D matrix of movies in rows and similarity (confidence) in columns.

Testing and Prediction:

- Creating a function that takes movie title as input and returns the top-10 most similar movies.

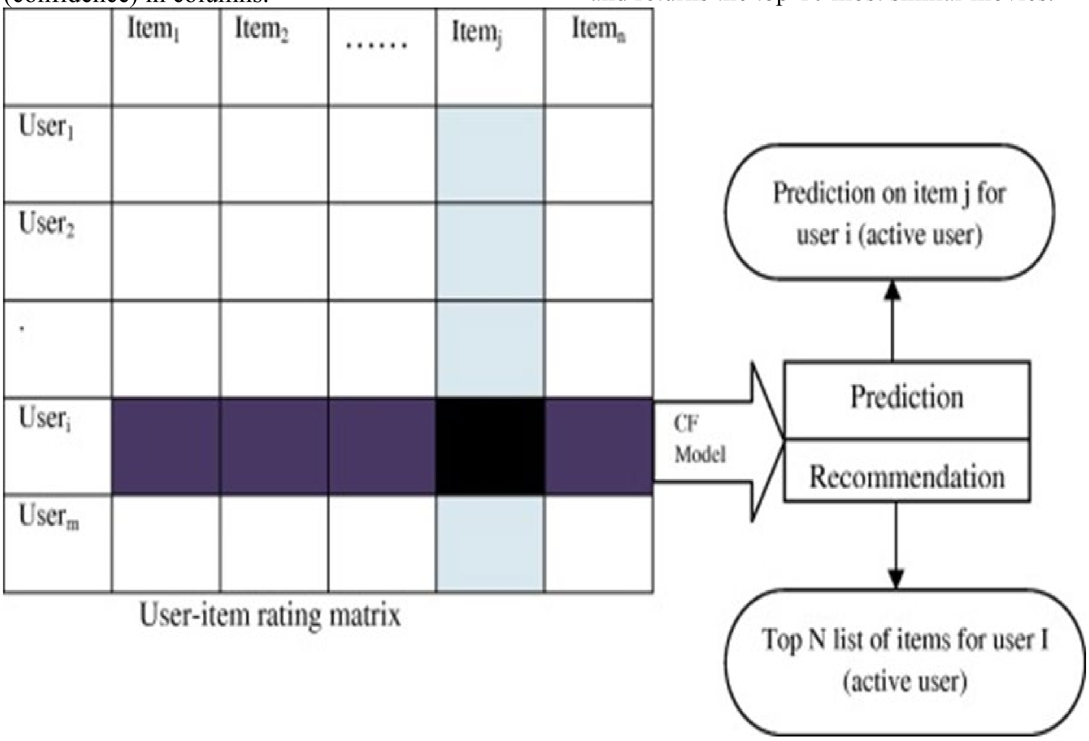
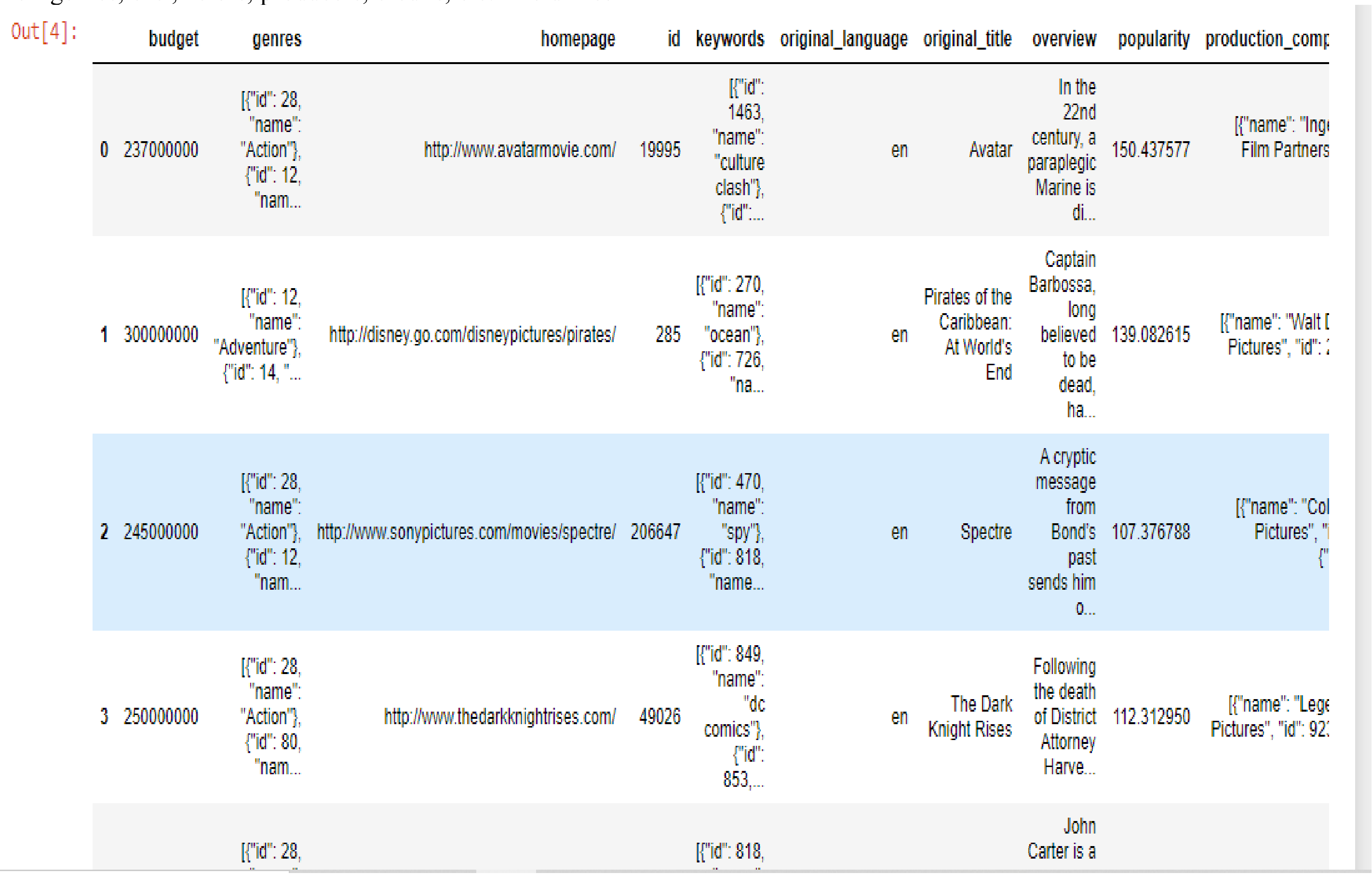


Fig.3. [36] User-item rating matrix

IV. DATASET USED

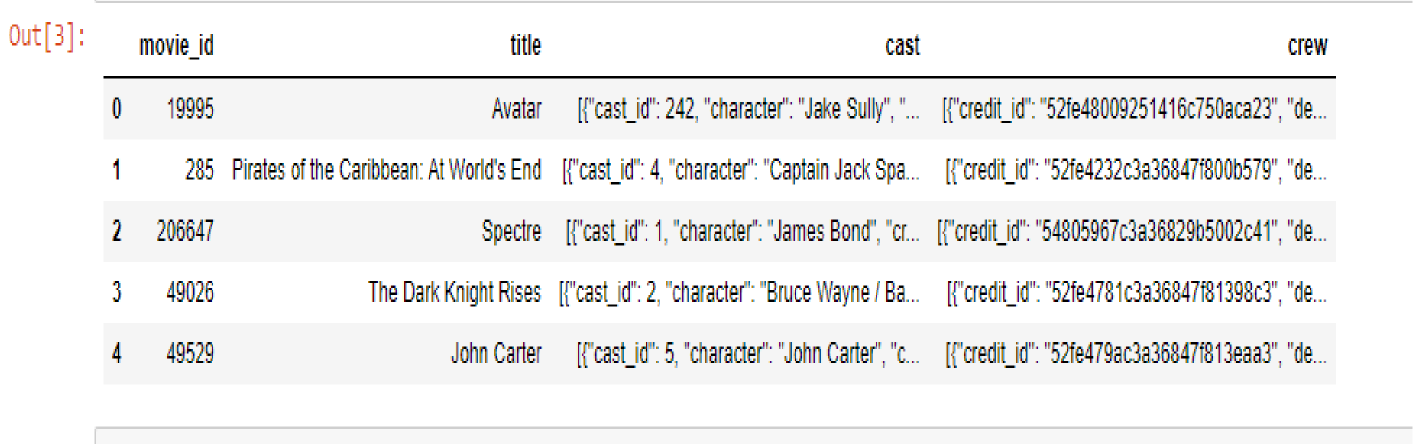
For this research work we have worked on the tmdb-5000. It is taken from Kaggle. It contains 5000 movies along with their genres, cast, actors, producers, credits, etc. The dataset contains 2 CSV files one contains movie details- tmdb_5000_movies, and the second contains the credits of movies (metadata), like the cast, producer, directors, etc. The movie dataset contains 4814 rows and 4 columns. The movie dataset contains 4804 rows and 20 columns.



Out[4]:

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_comp
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "Ingı Film Partners
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"name": "Walt I Pictures", "id": ;
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	[{"name": "Col Pictures", "i {"
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"name": "Lege Pictures", "id": 92:
		[{"id": 28,			[{"id": 818,			John Carter is a		

Fig 4. Movies dataset (Ref: kaggle.com/datasets)



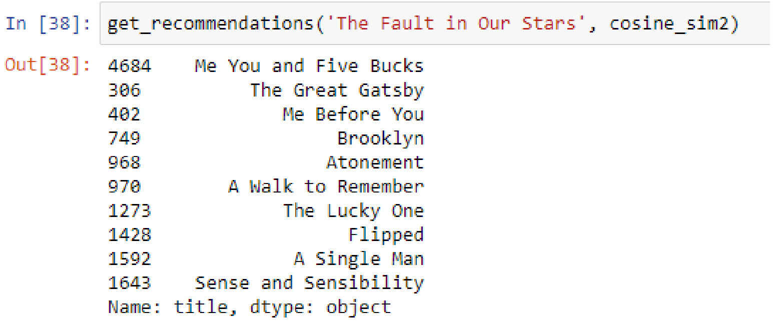
Out[3]:

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
1	285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
2	206647	Spectre	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
3	49026	The Dark Knight Rises	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
4	49529	John Carter	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...

Fig 5. Credits Dataset

Data Preprocessing
Before starting with the modelling using various Machine Learning strategies, the data has to be cleaned and pre-processed in order to achieve best results. Data pre-processing is the first step of our research work. For this work we have used the tmdb_5000 dataset from Kaggle which has two datasets in it (movies dataset, credits dataset). The movie dataset contains 4814 rows and 4 columns. The movie dataset contains 4804 rows and 20 columns.

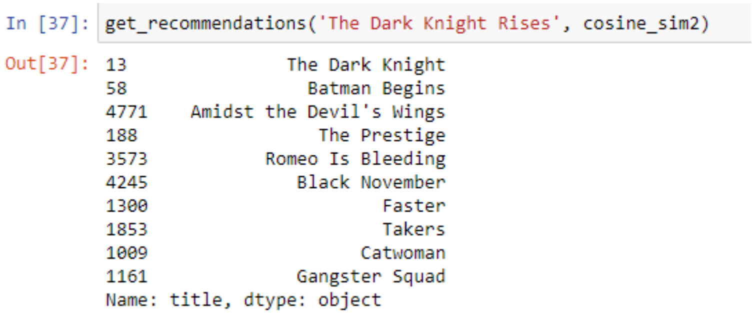
V. RESULT AND ANALYSIS



```
In [38]: get_recommendations('The Fault in Our Stars', cosine_sim2)

Out[38]: 4684    Me You and Five Bucks
         306          The Great Gatsby
         402             Me Before You
         749                  Brooklyn
         968                 Atonement
         970        A Walk to Remember
         1273            The Lucky One
         1428                  Flipped
         1592             A Single Man
         1643    Sense and Sensibility
         Name: title, dtype: object
```

Fig 6. Results



```
In [37]: get_recommendations('The Dark Knight Rises', cosine_sim2)

Out[37]: 13                The Dark Knight
         58                  Batman Begins
         4771    Amidst the Devil's Wings
         188                   The Prestige
         3573            Romeo Is Bleeding
         4245               Black November
         1300                        Faster
         1853                        Takers
         1009                      Catwoman
         1161                Gangster Squad
         Name: title, dtype: object
```

Fig 7. Results

VI. CONCLUSION

In this paper we have introduced RECOM, a recommender system for movie recommendation. It allows a user to select his choices from a given set of attributes and then recommend him a movie list based on the IMDB rating calculated using a standard formula. Further we would like to incorporate different machine learning and clustering algorithms and study the comparative results.

AUTHOR CONTRIBUTIONS

All authors have participated in (a) conception and design, or data analysis and interpretation; (b) drafting the paper or critically reviewing it for significant intellectual content; and (c) approval of the final result. This manuscript is not currently being reviewed by another journal or other publishing venue and has not been submitted to one.

All authors are not affiliated with any entity that has a direct or indirect financial interest in the subject matter mentioned in the research.

All authors assert that they have no conflicts of interest

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in Kaggle at https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata?select=tmdb_5000_movies.csv

Funding: This research received no external funding

VII. REFERENCES

[1]. wired.com/2013/08/netflix-algorithm.

[2]. thinkwithgoogle.com/marketing resources/ data-measurement/personalization-statistics-boost-engagement.

[3]. blog.hubspot.com /service/recommendation-engine.

[4]. venturebeat.com/2011/05/25/music-recommendation-stats.

[5]. Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM, 35(12), 61-70.Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In Proceedings of the 1994 ACM conference on Computer supported cooperative work (pp. 175-186).

[6]. Lops, P., De Gemmis, M., & Semeraro, G. (2011). Evaluation of item-based top-N recommendation algorithms. ACM Transactions on Information Systems (TOIS), 29(4), 1-50.

[7]. Koren, Y., Bell, R., & Volinsky, C. (2008). Collaborative filtering recommender systems. In The Handbook of Research on Recommender Systems (pp. 145-186). Springer.

[8]. Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (pp. 43-52). Morgan Kaufmann Publishers.

[9]. Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997). Collaborative filtering for bibliographic recommendations. In Proceedings of the 1997 ACM conference on Computer supported cooperative work (pp. 210-217).

[10]. Vellino, A. (2017). Collaborative Filtering for Research Paper Recommendation: A Comparative Study. Journal of Data and Information Science, 2(1), 13-27.

[11]. Lops, P., de Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In Recommender Systems Handbook (pp. 73-105). Springer, Boston, MA.Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.

[12]. Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In Recommender systems handbook (pp. 73-105). Springer, Boston, MA. Ricci, F., Rokach, L., & Shapira, B. (2015). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.

[13]. Eyjolfsdottir, G., Schedl, M., & Lops, P. (2015). Interactive movie recommendations based on genre preferences. In Proceedings of the 9th ACM Conference on Recommender Systems (pp. 351-354).

[14]. Järvelin, K., & Kekäläinen, J. (2017). IR evaluation methods for retrieving highly relevant documents. In Foundations and Trends® in Information Retrieval (Vol. 11, No. 1-2, pp. 1-129). Now Publishers, Inc.

[15]. Han J., Kamber M., "Data Mining: Concepts and Techniques", Morgan Kaufmann (Elsevier), 2006.

[16]. Ricci and F. Del Missier, "Supporting Travel Decision making Through Personalized Recommendation," Design Personalized User Experience for e-commerce, pp. 221-251, 2004.

[17]. Steinbach M., P Tan, Kumar V., "Introduction to Data Mining." Pearson, 2007.

[18]. Jha N K, Kumar M, Kumar A, Gupta V K "Customer classification in retail marketing by data mining" International Journal of Scientific & Engineering Research, Volume 5, Issue 4, April-2014 ISSN 2229- 5518.

[19]. Giles C.L., Bollacker K.D., and Lawrence S., "CiteSeer: An automatic citation indexing system," in Proceedings of the third ACM conference on Digital libraries, 1998, pp. 89–98.

[20]. Beel J., Langer S., Genzmehr M., and Nürnberger A., "Introducing Docear's Research Paper Recommender System," in Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'13), 2013, pp. 459–460.

[21]. Bethard S and Jurafsky D, "Who should I cite: learning literature search models from citation behavior," in Proceedings of the 19th ACM international conference on Information and knowledge management, 2010, pp. 609–618.

[22]. Bollacker K. D., Lawrence S., and Giles C. L., "CiteSeer: An autonomous web agent for automatic retrieval and identification of interesting publications," in Proceedings of the 2nd international conference on Autonomous agents, 1998, pp. 116–123.

[23]. Erosheva E., Fienberg S., and Lafferty J., "Mixedmembership models of scientific publications," in Proceedings of the National Academy of Sciences of the United States of America, 2004, vol. 101, no. Suppl 1, pp. 5220–5227.

[24]. Ferrara F., Pudota N., and Tasso C., "A KeyphraseBased Paper Recommender System," in Proceedings of the IRCDL'11, 2011, pp. 14–25.

[25]. Jiang Y., Jia A., Feng Y., and Zhao D., "Recommending academic papers via users' reading purposes," in Proceedings of the sixth ACM conference on Recommender systems, 2012, pp. 241–244.

[26]. McNee S. M., Kapoor N., and Konstan J. A., "Don't look stupid: avoiding pitfalls when recommending research papers," in Proceedings of

the 20th anniversary conference on Computer supported cooperative work, 2006, pp. 171–180.

[27]. Middleton S. E., De Roure D. C., and Shadbolt N. R., "Capturing knowledge of user preferences: ontologies in recommender systems," in Proceedings of the 1st international conference on Knowledge capture, 2001, pp. 100–107.

[28]. Zarrinkalam F. and Kahani M., "SemCiR - A citation recommendation system based on a novel semantic distance measure," Program: electronic library and information systems, vol. 47, no. 1, pp. 92–112, 2013.

[29]. Schafer J. B., Frankowski D., Herlocker J., and Sen S., "Collaborative filtering recommender systems," Lecture Notes In Computer Science, vol. 4321, p. 291, 2007.

[30]. Seroussi Y., "Utilising user texts to improve recommendations," User Modeling, Adaptation, and Personalization, pp. 403–406, 2010.

[31]. Buttler D., "A short survey of document structure similarity algorithms," in Proceedings of the 5th International Conference on Internet Computing, 2004.

[32]. Goldberg D., Nichols D., Oki B. M., and Terry D., "[Using collaborative filtering to weave an information Tapestry]," Communications of the ACM, vol. 35, no. 12, pp. 61–70, 1992.

[33]. Beel J., Langer S., and Genzmehr M., "Mind-Map based User Modelling and Research Paper Recommendations," in work in progress, 2014.

[34]. MacQueen J.. Some methods for classification and analysis of multivariate observations. In Proc. Of the 5th Berkeley Symp. On Mathematical Statistics and Probability, pages 281-297. University of California Press, 1967.

[35]. Ball G. and Hall D.. A Clustering Technique for Summarizing Multivariate Data. Behavior Science, 12:153-155, March 1967. Bowman, M., Debray, S. K., and Peterson, L. L. 1993. Reasoning about naming systems

[36]. Recommendation systems: Principles, methods and evaluation F. Isinkaye, Y. Folajimi, B. Ojokoh