ISSN No:-2456-2165

Movie Recommendation System Using Machine Learning

Sai Rushyanth Vattikonda

Abstract:- User-generated content, such as reviews and comments, contains valuable information about products as well as the opinions expressed by users. With the rise of internet usage, there has been an influx of user-generated data in the form of reviews and comments. Individuals share their experiences, opinions, sentiments, and emotions by writing reviews and comments about products they have purchased online or their impressions after engaging with various media, such as movies or books. These user-generated data often encompass emotions such as happiness, sadness, and surprise, which can serve as crucial indicators for recommending new items based on users' emotional preferences.

In this study, we propose a method to extract emotions from user-generated data by utilizing lexical ontology, specifically WordNet, in conjunction with insights from the field of psychology. These extracted emotions can then be leveraged to enhance recommendations. To evaluate the effectiveness of our approach, we compare our emotion prediction model with the traditional rating-based item similarity model, as well as explore the impact of emotional fuzziness in the feature space.

In the realm of e-commerce, recommender systems play a vital role in guiding users toward interesting and useful products within a vast array of options. To deliver reliable recommendations, these systems need to capture customer needs and preferences accurately, thus creating a comprehensive user profile. However, when it comes to subjective and intricate products like movies, music, and news, user emotions surprisingly hold a significant influence over the decision-making process. Since conventional user profiles fail to account for the impact of user emotions, recommender systems struggle to understand and adapt to users' constantly evolving preferences.

To address this issue, we introduce a Movie Recommender System (MRS) as a solution. MRS aims to provide personalized and tailored suggestions to users by employing a combination of collaborative filtering and content-based techniques. The recommendations are based on inferences drawn from a user's preferences as well as the opinions of other similar users. Furthermore, this paper delves into the system design, implementation, and evaluation procedures of MRS. We strongly believe that our system yields superior recommendations by enabling users to grasp the relationship between their emotional states and the recommended movies.

I. INTRODUCTION

Recommender systems play a crucial role in analyzing user activities, studying patterns, and predicting their preferences among a range of items. These systems employ technology that can be broadly categorized into two types: content-based and collaborative filtering.

Content-based models (CBF) operate by examining and analyzing the properties of items to make predictions. By understanding the characteristics and features of items, CBF can recommend similar items to users based on their preferences.

On the other hand, collaborative filtering (CF) models focus on finding similarities between users or items to make predictions. CF approaches leverage the collective wisdom of users with similar tastes and preferences to recommend items. Traditional memory-based CF techniques identify similar users based on their ratings for different items, forming the basis of user-based recommender systems. Model-based CF approaches analyze rating patterns to pre-compute a model that can be used for recommending items.

In addition to CF and CBF, there are also case-based reasoning (CBR) systems that recommend new items based on the features and content of previously liked items. CBR systems identify similar items with comparable attributes or features to recommend to users.

However, both CF and CBR approaches face certain limitations. CF-based recommender systems often encounter sparse rating data, as not all users rate items, which can affect the accuracy of recommendations. Additionally, CF may face challenges with the "cold start" problem, where new users or items have limited data available for making accurate predictions.

- *Problem Statement :*

The rationale for building a recommendation system lies in enhancing the user experience and driving business goals. By recommending movies that align with customers' preferences and satisfaction (customer-driven goal), the system aims to improve user engagement and retention. Additionally, by presenting a list of movie recommendations that are likely to be selected as the next viewing choice (business-driven goal), the system can increase user engagement and promote content consumption.

To measure the success of the recommendation system, several metrics can be considered. Customer satisfaction can be evaluated through post-viewing ratings and feedback from users regarding their movie-watching experience. Engagement metrics such as click-through rates, time spent on recommended movies, and the frequency of returning users can indicate the system's ability to capture users' interests and preferences effectively.

When building the recommendation system, it is essential to consider various input factors. In addition to a single movie title, incorporating the user's viewing history (excluding recently watched choices) can help personalize recommendations. Demographic information, if available, can provide further context and improve the relevance of the recommendations.

As for the output, it is valuable to provide more than just a ranked list of choices. Including an "explanation" of why a user might enjoy a particular movie can add a personalized touch. This could involve highlighting relevant features such as genre, mood, or themes that resonate with the user's preferences. Mimicking a conversation with a friend sharing movie recommendations, the system can provide a brief description or enticing aspects of the recommended movie to pique the user's interest.

In summary, the goal of the project is to recommend movies to users by providing related content from a collection of items, ensuring relevancy and enhancing the user experience on online service platforms. The success of the recommendation system can be measured through metrics such as customer satisfaction, engagement indicators, and feedback from users.

➢ *Scope/Purpose:*

The project's objective is to enhance the accuracy, quality, and scalability of the movie recommendation system compared to traditional approaches. This will be achieved by employing a hybrid approach that combines content-based filtering and collaborative filtering. Additionally, the system will serve as an information filtering tool on social networking sites to address data overload. The project recognizes the importance of improving scalability, accuracy, and quality in movie recommendation systems, as pure collaborative approaches often suffer from poor recommendation quality and scalability issues.

➢ *Challenges:*

Recommendation systems face various challenges, including the Cold Start problem, Data Sparsity, and Scalability. The Cold Start problem occurs when there is insufficient data available for new users or items, making it challenging to provide accurate recommendations. Similarly, Data Sparsity refers to the scarcity of user ratings, making it difficult to find enough similarities among users or items for reliable recommendations. Scalability becomes an issue as the size of the dataset grows, requiring more resources for processing and potentially leading to inaccuracies in the recommendations. To tackle these challenges, the adoption of hybrid techniques can be beneficial in enhancing the performance and effectiveness of recommendation systems.

II. LITERATURE SURVEY

The motivation behind this improvement stems from the rise in popularity of organizations like Netflix, whose primary objective is to ensure customer satisfaction. In the past, individuals had to physically select movies from libraries, relying on user reviews or making random choices. However, this approach became impractical due to the vast number of viewers with unique movie preferences. Consequently, numerous recommendation systems have been developed over the last decade. These systems employ different approaches such as collaborative filtering, content-based filtering, and hybrid models.

By analyzing the behavior and history of different users, recommendation systems suggest movies to watch based on their ratings, eliminating the need for users to make extensive decisions. These recommendation systems can be broadly categorized into two types: collaborative filtering and content-based filtering. Collaborative filtering combines the ratings of users with similar tastes to generate recommendations, while content-based filtering focuses on a single user, utilizing their viewing history and ratings to provide personalized suggestions.

To implement these recommendation systems, various methodologies have been introduced, including techniques from the fields of Data Mining, Clustering, and Bayesian Networks. These methodologies aim to enhance the accuracy and effectiveness of the recommendation system by leveraging advanced algorithms and statistical techniques.

➢ *System requirement*

This involves both the hardware and software requirements needed for the project and a detailed explanation of the specifications.

➢ *Hardware Requirements:*

- A PC with Windows/Linux OS
- Processor with a 1.7-2.4gHz speed
- Minimum of 8GB RAM
- 2GB Graphic Card

➢ *Software Requirements:*

- *Django*
- *Visual Studio Code*
- *HTML,CSS*

➢ *Software Specification:*

- Text Editor (VS-code/WebStorm)
- Anaconda distribution package (PyCharm Editor)
- Python libraries **Design Flow:**

➢ *Using Content-based Filtering:*

The system takes into consideration the user's interests and profile features to provide recommendations. It assumes that if a user has shown interest in a particular item in the past, they will likely be interested in similar items in the future. Items are grouped together based on common features, allowing for more targeted recommendations. User profiles are constructed using past interactions or by directly asking users about their interests. Some recommendation systems also utilize user personal and social data to enhance the recommendation process.

However, one major drawback of this recommendation system is its tendency to make obvious recommendations due to excessive specialization. For example, if user A has only shown interest in categories B, C, and D, the system may not recommend movies outside of these categories, even if they might be of interest to the user. Another challenge is that new users do not have a well-defined profile unless they provide explicit information. Despite these limitations, incorporating new movies into the system is relatively straightforward by assigning them appropriate groups based on their features.

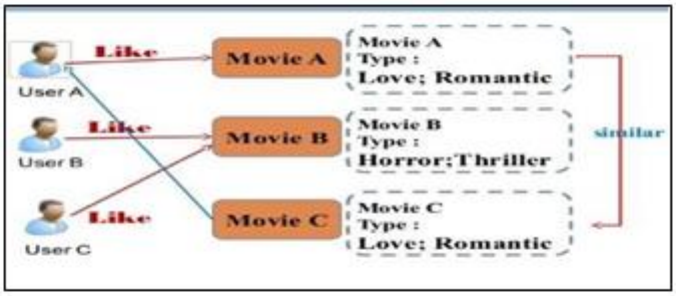


Fig 1 Content-based Filtering:

This approach to movie recommendations focuses solely on the similarity of movies and does not take into account the preferences of other users. The algorithm selects similar movies based on their content to provide recommendations. However, this method tends to result in limited diversity in the recommendations, as it primarily considers the specific preferences of the user. As a result, the recommendation system may suggest a narrow set of similar movies.

To address this limitation, it is possible to enhance the efficiency of the system by incorporating various categories such as subgenre, keyword, cast, director, and other relevant factors. By considering a broader range of attributes, the recommendation system can offer more diverse and varied movie recommendations to cater to the individual tastes and preferences of users.

➢ *Algorithm use: Cosine Similarity*

Cosine similarity is utilized to calculate the similarity between two movies in this recommendation system. The dataset is transformed into vector representations. The concept of similarity between two vectors, denoted as u and v, is determined by the ratio of their dot product to the product of their magnitudes. If the vectors are identical, the similarity score will be closer to 1. On the other hand, if the vectors are different or orthogonal, the similarity score will be closer to 0. By employing cosine similarity, the recommendation system can quantitatively assess the similarity between movies based on their vector representations, aiding in the generation of relevant and related recommendations.

III. LIMITATIONS

Content-based recommendation systems may exhibit poorer performance when the available content lacks sufficient information to accurately classify movies. This limitation can result in imprecise recommendations. Additionally, content-based systems may offer limited novelty since they rely on matching user profile features with available movies. These systems may not effectively cater to new users who have not yet rated any movies, as an adequate number of ratings are required for accurate content-based evaluation of user preferences. In the case of item-based filtering, only movie profiles are created, and users are suggested movies based on similarity to their rated or searched movies, without considering their past history. Consequently, a content-based filtering system may fail to offer unexpected or surprising recommendations. Furthermore, it is unable to evaluate the subjective quality of a movie. Different individuals have diverse tastes, perspectives, and frameworks, making it challenging for a machine to analyze and differentiate between good and bad movies solely based on similar words. The assessment of movie quality encompasses subjective elements that are difficult for a machine to accurately interpret.

➢ *Using Hybrid Approach:*

In a content-based recommendation system, recommendations are generated based on the similarities between movies and do not take into account individual user preferences. Therefore, all users would receive the same set of recommendations if they have watched similar movies. On the other hand, in collaborative filtering, there is no consideration of the relationships between the movies watched by a user, which means that the system may not capture the user's specific genre interests.

To overcome these limitations, the concept of a hybrid movie recommendation system emerged. The hybrid approach combines both content-based and collaborative filtering methods, leveraging the advantages of both and mitigating their disadvantages. By integrating the two approaches, the hybrid system can provide more personalized and accurate recommendations to users. It takes into account the similarities between movies as well as the user's preferences and behaviors, resulting in a more comprehensive recommendation process. The hybrid recommendation system aims to offer the best of both worlds, improving the overall recommendation quality and enhancing the user experience.

ISSN No:-2456-2165

Fig 2 Hybrid Approach

IV. RESULT ANALYSIS AND VALIDATION

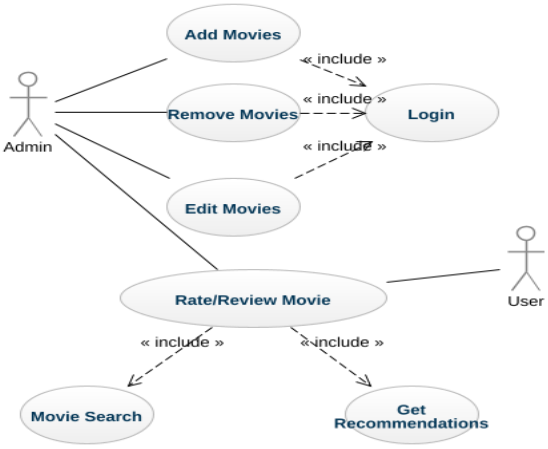


Fig 3 Result Analysis and Validation

➢ *Testing:*

System testing is a crucial process that involves a series of tests designed to thoroughly evaluate a computer-based system. Each test serves a specific purpose, but collectively they aim to ensure that all system components are properly integrated and perform their intended functions. The main goal of system testing is to confirm the quality of the project and ensure that the software functions as expected.

During the testing stage, several objectives are pursued:

1. Affirming Quality: The testing process aims to validate and affirm the overall quality of the project, ensuring that it meets the desired standards and specifications.
2. Error Elimination: System testing helps identify and eliminate any residual errors or bugs that may have been missed during earlier stages of development.
3. Solution Validation: The software is tested to validate its effectiveness in solving the original problem it was designed for.
4. Operational Reliability: The testing process ensures the operational reliability of the system, verifying that it performs reliably under different scenarios and conditions.

Functional testing, a type of software testing, is an integral part of system testing. It focuses on validating the software system against its functional requirements and specifications. This testing aims to test each function of the software application by providing appropriate input and verifying the output against the specified requirements. Functional testing includes various aspects such as testing the user interface, APIs, database, security, client/server communication, and other functionalities of the application under test.

During functional testing, the following aspects are typically evaluated:

- Mainline Functions: Testing the core functions of the application to ensure they work as intended.
- Basic Usability: Evaluating the system's usability by checking if users can navigate through screens without difficulties.
- Accessibility: Verifying that the system is accessible to users, including those with disabilities.
- Error Conditions: Testing error conditions to ensure appropriate error messages are displayed when needed.

Functional testing is performed manually or using automation tools depending on the requirements. Its main focus is to validate the functionalities of the software system and ensure its smooth operation.

V. CONCLUSION

The project focuses on enhancing the accuracy, quality, and scalability of the movie recommendation system by employing a Hybrid approach that combines content-based filtering and collaborative filtering. The proposed methodology utilizes Singular Value Decomposition (SVD) as a classifier and Cosine Similarity to generate recommendations.

To evaluate the effectiveness of the Hybrid approach, it is compared with existing pure approaches using three different Movie datasets. The results demonstrate that the proposed approach outperforms the pure approaches in terms of accuracy, quality, and scalability. Furthermore, the computational time required by the proposed approach is also lower compared to the other pure approaches.

By integrating the strengths of content-based and collaborative filtering techniques, the Hybrid approach aims to provide improved movie recommendations that cater to user preferences while overcoming the limitations of the individual approaches. The results highlight the potential of the proposed approach to enhance the overall performance of the movie recommendation system.

VI. FUTURE SCOPE

In addition to considering movie genres, the proposed approach can be further enhanced by incorporating the age of the user as a factor in movie recommendations. Age often plays a significant role in shaping movie preferences, as individuals tend to have different preferences at different stages of their lives. For instance, during childhood, animated movies might be more appealing compared to other genres. By incorporating age-related information, the recommendation system can provide more personalized and relevant suggestions to users.

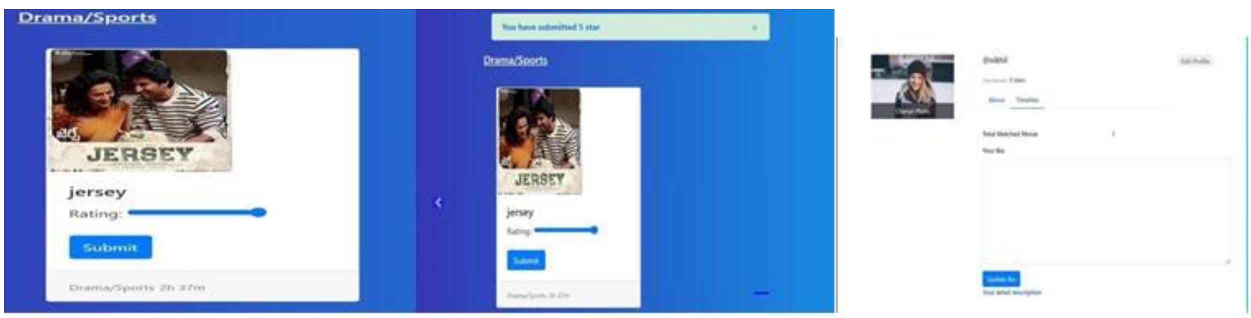
Furthermore, future work should also focus on addressing the memory requirements of the proposed approach. As the system expands and more data is accumulated, it is important to optimize memory usage to ensure efficient processing and scalability.

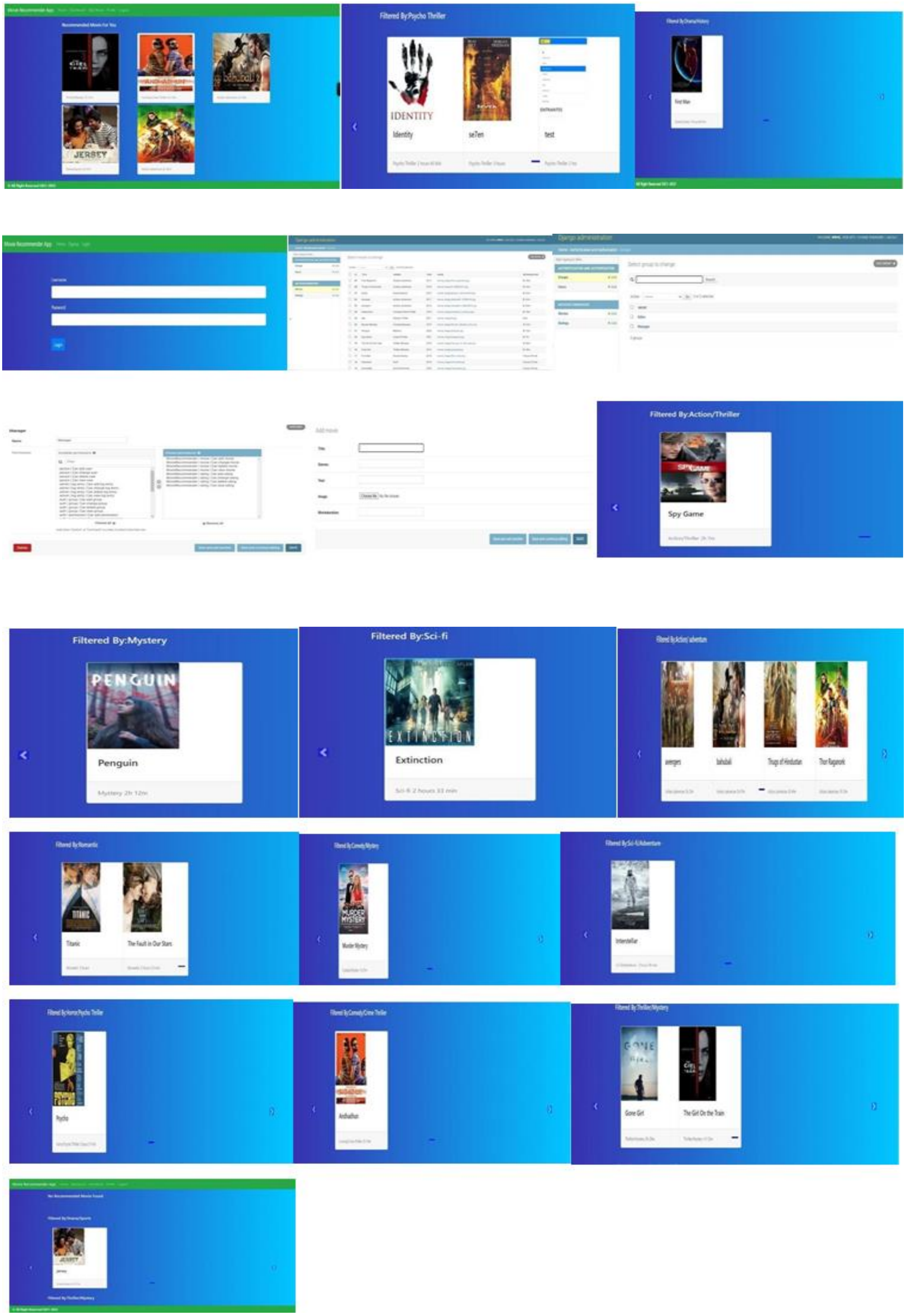
To further evaluate the performance and effectiveness of the proposed approach, future research can implement it on additional datasets such as Film Affinity and Netflix. By assessing its performance on different datasets, a more comprehensive understanding of the approach's capabilities and limitations can be obtained.
Overall, considering age as a factor and addressing memory requirements while expanding the evaluation to diverse datasets can contribute to the continuous improvement and advancement of the proposed approach in the field of movie recommendation systems.

REFERENCES

[1]. Hirdesh Shivhare, Anshul Gupta and Shalki Sharma (2015), "Recommender system using fuzzy c-means clustering and genetic algorithm-based weighted similarity measure", IEEE International Conference on Computer, Communication and Control.
[2]. Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), "A Movie Recommender System:
[3]. MOVREC", International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3.
[4]. RyuRi Kim, Ye Jeong Kwak, HyeonJeong Mo, Mucheol Kim, Seungmin Rho,Ka Lok Man, Woon Kian
[5]. Chong (2015),"Trustworthy Movie Recommender System with Correct Assessment and Emotion Evaluation", Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II.
[6]. Debadrita Roy, Arnab Kundu, (2013), "Design of Movie Recommendation System by Means of
[7]. Collaborative Filtering", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 4.
[8]. References:
[9]. Kim, Mucheol, and Sang Oh Park, "Group affinity-based social trust model for an intelligent movie recommender system", Multimedia tools and applications 64, no. 2, 505-516, 2013.
[10]. Noguera, José M., Manuel J. Barranco, Rafael J. Segura,and Luis Martínez, "A mobile 3D-GIS hybrid recommender system for tourism", Information Sciences 215, 37-52, 2012.
[11]. Nanou, Theodora, George Lekakos, and Konstantinos Fouskas, "The effects of recommendations" presentation on persuasion and satisfaction in a movie recommender system",
[12]. Multimedia systems 16, no. 4-5, 219-230, 2010
[13]. Hirdesh Shivhare, Anshul Gupta and Shalki Sharma (2015), "Recommender system using fuzzy cmeans clustering and genetic algorithm based weighted similarity measure", IEEE International Conference on Computer, Communication and Control.
[14]. Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), "A Movie Recommender
[15]. System: MOVREC", International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3.
[16]. RyuRi Kim, Ye Jeong Kwak, HyeonJeong Mo, Mucheol Kim, Seungmin Rho, Ka Lok Man, Woon
[17]. Kian Chong (2015), "Trustworthy Movie Recommender System with Correct Assessment and Emotion Evaluation", Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II.
[18]. Zan Wang, Xue Yu*, Nan Feng, Zhenhua Wang (2014), "An Improved Collaborative Movie
[19]. Recommendation System using Computational Intelligence", Journal of Visual Languages & Computing, Volume 25, Issue 6.

OUTPUT DIAGRAMS




Filtered By:Psycho Thriller
Identity
seTen
test
Filtered By:Action/Thriller
Spy Game
Filtered By:Mystery
Penguin
Mystery 2h 12m
Filtered By:Sci-fi
Extinction
Titanic
The Fault in Our Stars
Interstellar
Gone Girl
The Girl On the Train