Analysis of Different Movie Recommendation Systems Used In a Movie Catalogue

Anuradha Yadav[1] Abhishek Kumar[2] Rohit Ram[3] Aman Kumar[4]
[1]Assistant Professor
[1,2,3,4]Department of Information Technology
[1,2,3,4]Greater Noida Institute Of Technology, Greater Noida, UP, India

Abstract — There are lots of content available on the movie recommendation systems. Movie recommendation systems plays an important role in finding the content of user's choice. Various recommender systems are used which provides useful recommendations for product and services. Movie recommendation systems are machine learning based approach which use information filtering to foretell the preferences and rating of users for a particular product or movies. Recommender systems are bitterly known as recommendation systems. Basically, there are four type of recommendation systems: Simple recommendation System, Collaborative Filtering, Content based and hybrid approach.

Keywords: Movie Recommendation System, Content Based Approach, Collaborative Filtering, Hybrid Approach

I. INTRODUCTION

Movie recommendation systems is a machine learning based approach which uses information filtering to predict and foretell the preferences and rating of users for a particular movie or film. The main idea behind the recommendation systems is to recommend the products or movies of user' choice and interest. In other word we can say that recommender systems are the systems which suggest appropriate product or items to the consumers. In current scenario recommendation systems are used on a daily basis for finding "which movie to watch", in online shopping and "which text to read". Required data for the recommendation systems is collected from in various ways and from various sources. These ways and sources can be user's rating about a movie, purchase history, and from the other's knowledge about the users/items themselves. Because of the huge demand of the latest content on the web the need of the recommender systems has increased a lot. These systems are highly useful to fulfill the remands of the users in providing the content of their choice. Although different users have different taste and different users like different movies and actors. So, it becomes challenging to seek out the perfect movies for a particular user. It is important to find a way to seek out a way to filter the irrelevant and relevant movie for a user. There comes the role of automated recommendation systems and services which make this task a way easier. Recommender systems are of four types: Simple recommendation systems, Collaborative filtering, content based and hybrid approach. Simple recommendation systems recommend movies based on the popularities and genre. The idea behind is this that the movie that is more popular and have more rating has higher probability to be liked by the user/audience. In Content based filtering the assumption is made that if someone has interest in an item or movie, then other will once again be interested in it in the future. Item features are used in content-based filtering to recommend other items to the users based on their previous feedbacks. The function of collaborative filtering can be subdivided into two ways, the first one is user-based collaborative filtering in which filtering is done on the basis of the taste of similar users. Similar user shares the taste, interest in movies and multimedia. The other one is item-based collaborative filtering in which filters similar items. A hybrid movie recommendation system uses both content based and collaborative filtering recommender system. It combines the advantages of both the recommender system, now user interference is possible and also the system will recommend the movies of the same genre in which the user is interested based on his previous experiences.

II. PROBLEM STATEMENT

Nowadays, there are lots of movies and web series available on the web and theaters. So, it becomes difficult to find one's favorite movie in a large number of options available. Most of the time consumers do not have a particular movie in their head. There comes the role of movie recommendation systems. Movie recommendation system predict and foretell various movies to the consumers based on their taste and interest. There are four methods implemented in this paper which are Simple recommendation system, Collaborative filtering, content based filtering and Hybrid filtering.

III. RELATED PAST WORK

Ramni Harbir Singh et al. presented the modelling of a movie recommendation system by using content-based filtering. It implements the KNN algorithm along with the cosine similarity principle, which results in increased accuracy compared to other distance metrics, as well as low running complexity.

The MOVREC movie recommendation system is based on collaborative filtering and was developed by D.K. Yadav et al. It utilizes user-provided information to make recommendations, prioritizing the movie with the highest rating.

According to Luis M Capos et al., content-based filtering and collaborative filtering are the most commonly used recommendation systems, but they both have drawbacks. Capos proposed a different system that combines Bayesian network and collaborative filtering to overcome these limitations.

Harpreet Kaur et al. have developed a hybrid recommendation system that combines content and collaborative filtering algorithms. Context is also considered during the recommendation process, and recommendations are based on both user-user and user-item relationships.

Utkarsh Gupta et al. utilized Chameleon to cluster user or item-specific information for recommendation purposes. This method is highly useful and relies on hierarchical clustering. A voting system is used to predict item ratings, leading to fewer errors and better clustering of comparable articles.

Urszula Kuelewska et al. proposed clustering as a method for dealing with recommender systems. They computed cluster representatives using two methods, presentation and evaluation, and compared centroid-based solutions and memory-based collaborative filtering. The use of clustering resulted in a significant improvement in accuracy compared to using only centroid-based recommendations.

Costin-Gabriel Chiru et al. developed a movie recommender system that ignores the user's specific data to create unique recommendations. The system is collaborative filtering-based and disregards an individual's psychological profile, watching history, and movie scores from other websites.

A paper on surprise library is proposed by Nicolas Hug using python. In this program, rating prediction algorithms are built and analysed. Python is used to optimize the computationally intensive parts of surprise, whereas it is primarily written in python. Surprise uses both built-in Python data structures and NumPy arrays internally. For predict ratings, Surprise provides a set of estimators or prediction algorithms. There are classical algorithms that can be executed, such as the most similarity-based algorithms and custom prediction algorithms can also be developed.

In this presentation, G. Linden et al. discussed the most popular recommender system Amazon. Amazon is one of the top sales companies in the United States. Being an online retailer, it has an integrated recommendation system as part of its marketing strategy. Basically, item-to-item Collaborative Filtering allows the website to be customized according to the interest of each customer. Its special design allows it to scale too many customers and products in real time, providing high-quality recommendations.

IV. METHODOLOGY

A. *Simple Recommender System*

The Simple Recommender system recommends movies on the basis of movie popularity and genre. The popularity and rating of the movie plays the most important role in this. Movies which are more popular and have more rating are likely to be liked by the respective audience. Therefore, the fundamental concept behind this system is that it offers movies to the audience on the basis of popularity and rating of the movie. This model doesn’t support personalized recommendation based on the user.

The implementation of this model is very trivial. Parameters like ratings, popularity and genre plays most important role in this model. The movies need to be sorted on the basis of these parameters. Genre parameter is used to categorize and fetch movies from different genre.

Steps to sort movies:

– Two parameters, score and ratings, are required for the recommender system.
– The score and rating are computed for each movie.
– The movies are then ranked based on their scores, and the top to bottom list of the best to worst movies is recommended.

IMDB weighted rating are used in Simple Recommendation System to recommend movies to the users.

$$\text{Weighted Rating(WR)} = (V.R + m.C)\ (V + m)$$

Here,

– V is number of votes for the movie
– m is the minimum number of votes required to be listed in the chart
– R is the average rating of the movie
– C is mean vote

1) *Advantages of Simple Recommender System*

Basically, this model is used to get the list of trending movies. Here not only the votes but the average ratings are also considered. That is, movies are not listed on the basis of votes only but also on the basis of their average rating as it will help the new movies which are watched less and voted much to be in the list. For instance, a movie with a mean rating of 9 but only 6 votes cannot be ranked higher than a movie with an average rating of 8 and 50 votes.

2) *Disadvantages of Simple Recommendation System*

Simple recommender systems have some terrible limitations. It offers common recommendation to every user without considering and knowing their personal taste. Suppose one likes Science-fiction movies and hates horror movies, if he goes through the top 10 chart, he probably not going to get the movie of his taste. If he looks at the chart by genre, then he might not find the simplest recommendation.

Consider a person who like movies like Interstellar, Inception, Passengers, Avatar then we can simply find out that he likes Sci-fi movies. But when he goes though the top recommendation chart, he finds romantic movies. In that case the recommendation system does not works for him properly.

Personalization is the most important factor for making recommendation engine. The engine should be capable of measuring the certain parameters and should recommend movies similar to those movies which suits of the user’s taste and interest. These limitations are solving in Content based filtering.

B. *Content-Based Filtering*

Recommendation systems often utilize item features to suggest products or items that are similar to those that a user has shown interest in or provided feedback on. User’s previous actions plays a vital role in this. This model is an upgrade from Simple Recommender System. The basic idea behind this model is that if one has interest in a movie or product, then one will again be interested in future. Items or movies are grouped together on the basis of their common features. The system asks the interest of the users on their very first login, then this data is saved and utilized for future use. The system has the ability to suggest items to the user in the future, taking into account their past interactions and feedback. In other words, we can say that in Content based filtering user profiles are constructed using the past interaction and feedback of the users. This model is different from the other models as it does not utilize user’s personal and social data.

Content-based filtering approach categorize and recommend movies on the basis of their similarity. For example; it will group movies on the basis of their genre, all the action movies are group together, and adventure movies will fall in another group. The table 1 given below shows how movies are categorized in content-based filtering. Here, the movie genre can be used as the movie feature. Features could include categories (such as action, adventure, romance, fantasy, science fiction), the rows represent the movie features and the column represents the movie title. The last row represents the User's interest. Some user-related features can be provided by the user.

	Action	Sci-fi	Fantasy	Horror
Harry Potter			*	
Interstellar		*		
Annabelle				*
Uncharted	*		*	
User's Interest	*	*		

Table I: Categorization of movies on the basis of their genre

The model should foretell the movies according to the interest of this user. But as this model do not have any information about the other users that's why it will be specific to this user only.

There are also some limitations with this system like the user is only interested in action and Science-fiction movies. So, the system will not recommend movies other than these categories even if the user might be interested about the other categories. Also, when new users are enrolled the system don't have any information about their interest. So, they are first asked for the information, only after this the system can recommend them some movies. But it is easy to feature new movies to this system, only we have to assign them according to their genre or features.

The model for this recommendation system does not involve individual users, but instead suggests movies based on common features between them. The algorithm selects similar movies based on their content and genre, offering recommendations based on the user's preferences. A similar set of movies are recommended through this system as the diversity is least because it only recommends what one specifically likes. The efficiency of the system can also be increased by creating different categories like subgenre, cast, awards, cast, director and so on.

– Algorithm used: COSINE SIMILARITY

The cosine similarity method is utilized to compute the similarities between two movies in this model. The dataset is converted into two vectors, and similarity is measured as the ratio of the dot product of the two vectors P and Q to the product of their magnitudes. When two vectors are identical, the similarity will be 1 or very close to it. On the other hand, if two vectors are dissimilar, the similarity will be 0 or close to it.

Cosine Similarity is given as:

$$\boldsymbol{Similarity\ (P,Q) = \frac{P.Q}{||P||\,||Q||} = \frac{\sum_{i=1}^{n} Pi \times Qi}{\sqrt{\sum_{i=1}^{n} Q_i^2}}}$$

Here are the steps to find two similar movies using cosine similarity:

– Obtain the title and index of the movie you want to find similar movies for.
– Find the list of cosine similarity for that particular movie along with the list of all the other movies.
– Convert the list into a tuple data structure where the first element represents the position and the second element represents the similarity score.
– Sort the list of tuples based on the similarity score.
– Ignore the 1st element of the list of top 10 elements since it represents the movie itself with the highest similarity score.
– The most similar movie is located in the 2nd place of the list.

1) Advantages of Content-based filtering

– It can easily recommend items by listing them on the basis of their features.
– It recommends unrated items.
– This approach only considers the ratings of the specific user in question, making the ratings of other users irrelevant.

2) Limitation of Content Based Recommender System

– It is sometimes imprecise.
– This system performs worse when there is not enough information available in the content.
– It is not feasible to generate recommendations for a new user who has not yet provided any ratings for any movie.
– It may suggest nothing if it does not have enough information.
– Only movie profiles are created in item-based filtering, where users are suggested movies only what they are rated and searched for, instead of their past history.

C. Collaborative Filtering System

Further, the earlier created engine was not truly personal as it didn't capture a user's preferences or biases. The system will provide the same recommendations for any query for that movie, no matter who is asking for it. Unlike the previous approach, Collaborative filtering systems involve other users. Using this methodology, users are recommended movies that are watched and liked by similar users. This method employs user-to-user collaborative filtering, which uses a collaborative filtering approach to identify users who are similar to the active user. As it is more effective than content-based recommendations, it is one of the most common approaches used. YouTube, Netflix and Spotify all use this type of recommendation system. The similarity of two users can be determined by comparing their ratings and watching habits of movies, which helps in determining whether they have similar taste or not. As a result, we can predict the rating a user will give to a movie based on its similar user ratings, even though they haven't seen it yet.

Collaborative filtering system is divided into two categories depends on their methods

a) Memory based
b) Model based

1) Memory-based

Memory-based have two perspectives, the first one determines the prediction of an individual user with the other users by using their interactions activity within the clusters of users. The second one collects the rating given by any particular user and use it to predicting the reaction of same user with different movie but similar to previously rated movie. Because of the portion of user-item interaction are very much low to generate the high qualities cluster, these methods help in confronting the major issues with large sparse matrices.

2) Modal-based

Machine learning and data mining technology is used in these methods. The main aim is to make these models capable to make predictions. In order to predict users rating, model-based algorithms firstly generate a model to identify the behaviour of that user. By using the information from the rating matrix, the model's parameters are evaluated. As an example, we can use past interactions between the users and items to develop a model that can predict the top ten items that a user may like. Comparing these methods to others such as memory-based methods, these methods have a significant advantage in that they can recommend a greater variety of items to a greater variety of users. Even while dealing with huge sparse matrices, they are said to have greater coverage.

There are two methods of collaborative filtering:

a) User-based Filtering
b) Item-based Filtering

– User-based Filtering

Based on the preferences of similar users, these systems recommend items to any user. With help of either Pearson correlation or cosine similarity, we can calculate the similarities between two users. An example will help to clarify this method. In table 1and 2, each row corresponds to a different user, and each column represents a movie, except the last column which indicates how much similarity there is between the user and the target user. The user's rating for particular movie is shown each cell. Suppose john is the target user. Despite how easy and convenient it is to calculate user-based collaborative filtering, it has a number of flaws. The fact that a user's interest can vary with time is a major concern. It also shows that pre-calculating the matrix on the basis of their fellow users may result in poor performance. To overcome this challenge, we can use item-based collaborative filtering.

	Harry potter	The Witcher	Game of Thrones	Wednesday	Stranger Things	Vikings	Similarities (p, John)
Sam	2		2	4	5		N.A
Tom	5		4			1	
Harley			5		2		
Charley		1		5		4	
John			4			2	1
Chris	4	5		1			N.A

TABLE II: Similarities between users with the help of pre-calculation of matrix

	Harry potter	The Witcher	Game of Thrones	Wednesday	Stranger Things	Vikings	Similarities (p, John)
Sam	2		2	4	5		N.A
Tom	5		4			1	0.87
Harley			5		2		1
Charley		1		5		4	-1
John			4			2	1
Chris	4	5		1			N.A

TABLE III: Similarities one the basis of person correlation

In the table given above, the similarities of Sam and Chris with John is not defined in person correlation because they both doesn¡¦t shares any movie ratings in common with him.

– Item based filtering.

To measures the common thing between the user in the item based collaborative filtering like if we have two movie and there is four people who have to rate the movie but out of four people only three people are rated the movie then we take the average of two people and then after we give the recommendation to the fourth one this is how item-based filtering recommend the movie wo doesn¡¦t rate the movie. And if we have the millions of user then we have to find the similarity between them also so in this case we use Euclidean distance between two user method.

The item based collaborative filtering will help to maintain the stability and also helping to avoid the moveable user preference. But this type of problem has a solution. First, the main problem is expandability. With increase of customer and the product the computation will also increase. The time complexity for this model is O(p q), where p is the number of users and q is the number of items. However, the sparsity of the data is an important consideration. For instance, in the table above, there are only a few people who rated both "Matrix" and "Titanic", and the common rating between them is 5. sometimes we have big number of people likes millions and the common part of rating the movie between them is very high because of the people are rated the same movies with the same number.

3) Singular Value Decomposition

To address scalability and economic challenges associated with collaborative filtering, a latent factor model can be employed to capture similarities between users and items. This approach aims to optimize the recommendation problem by predicting how well the model will predict a user's item rating. Root mean squared error (RMSE) is typically used to assess performance, with lower RMSE values indicating better performance. Latent factors refer to the characteristics or concepts that users or objects possess. For example, in the case of movies, the latent factor may refer to the genre of the movie. SVD is used to extract latent factors and reduce the dimensionality of the utility matrix, allowing for a direct comparison between users and items in a latent space of dimension r, which enhances our understanding of the relationship between users and items.

4) Advantage of collaborative filtering-based system.

- It means content agnostic as it relies on relationships between users.
- A recommender system with collaborative filtering can suggest unique article by observing the behaviour of like-minded people.
- A true quality assessment of an object can be made taking into account the experience of others.

5) Limitation of Collaborative filtering

- Rating matrix sparsity: In most recommender systems, each user only rates a small subset of the available items, leaving most cells in the rating matrix empty. In cases like this, it is difficult to find similarities between different user her groups and items. It can be challenging to recommend items to users who consistently disagree with or have different tastes from the majority of the users in a group, if such groups exist.
- Cold boot: Related to the previous issue, this issue addresses an issue with making recommendation for users who have recently been introduced to the system. In such case, the recommender system cannot infer the user¡¦s interests because the user has not rated the movie for the required threshold. One approach to address this issue is to require users to rate a specific set of movies initially. However, this may introduce bias into the system. It should also be noted that the cold start problem usually only affects newly released movies. This is because it is not recommended until a sufficient threshold of users has been evaluated.
- Spam attacks: Recommender systems are subject to spam attacks, primary from users¡¦ interest in tricking the system into recommending certain products.

D. Hybrid Movie Recommendation System

There were various limitations in both Content-based recommendation system and Collaborative filtering. User interference was missing in content-based filtering which means it recommends same set of movies to the user based on his/her previous interaction and feedback. Similarly, the relation between the movies watched by the user is missing in collaborative filtering. The system will recommend movies watched by the similar users. For example, suppose if the user likes Sci-fi movies but the system is not going to recommend movies from the sci-fi genre. It will only recommend those movies which were watched by the other similar users like him.

The concept of a hybrid recommendation system is developed to address the limitations of both Content-based filtering and Collaborative filtering approaches. This approach combines the advantages of both methods while minimizing their drawbacks. By using hybrid recommendation systems, we can make more accurate and personalized recommendations to users. A hybrid movie recommendation system is a combination of multiple recommendation techniques or algorithms, used to make personalized movie recommendations to users. These systems are capable of generating more accurate and diverse recommendations. The idea behind a hybrid movie recommendation system is to leverage the strength of different techniques and overcome their limitations, in order to create a more effective and efficient recommendation system.

1) Steps involved in working of a Hybrid movie recommendation system

- Data collection: The system gathers various information about the movie ratings, user¡¦s interest, movie genre, demographic information, etc.
- Data processing: The gathered data is converted into useful format to be used by the recommendation system.
- Algorithm selection and combination: A hybrid recommendation system combines different movie recommendation algorithms such as collaborative filtering, content-based filtering, and demographic-based filtering to provide better recommendations.
- Recommendation generation: The system generates personalized recommendation for each user based on the combined algorithms
- Evaluation and improvement: The performance and accuracy of the systems is evaluated using matrix and improvements to the algorithms are made.
- Hybrid movie recommendation system combines Content based and Collaborative filtering methods to provide more diverse and accurate recommendations.
- It weighs the recommendation from both the methods and then combines them to create the final list of recommendation.
- It uses one method to filter the users and the other to rank them
- Also, it uses one method to provide the set of recommendations and the other to refine them.

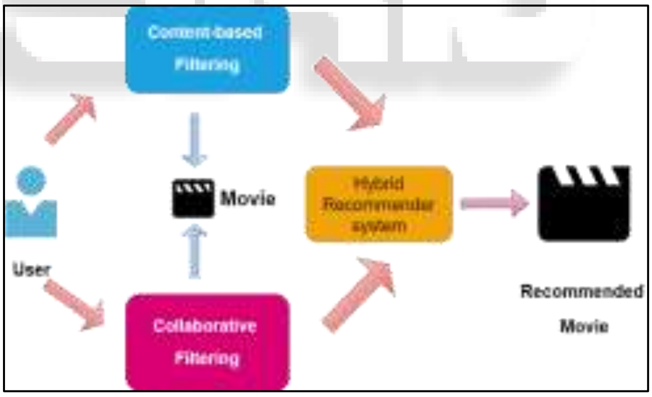

Fig. 1: Hybrid Movie Recommender System

2) Algorithms used in Hybrid Movie recommender systems

The following algorithms are used in Hybrid movie recommender systems.

- Matrix Factorization: It is a technique in which the user-item rating matrix is decomposed into low-dimensional latent feature representations.
- Cosine Similarity: The idea of calculating similarity between two movies using cosine similarity is a commonly used approach in recommendation systems. In this approach, the dataset is represented in the form of two vectors, and similarity is calculated as the cosine of the angle between these two vectors.
- Nearest Neighbours: The algorithm can be used for both the user-based collaborative filtering and item-based

collaborative filtering. It find the most similar users or items to a particular user to item.

- Decision trees: This algorithm creates a tree which can be used to make predictions. It is used for content-based filtering.
- Neural Networks: It creates a models that can learn from the data and make predictions. It can be used for both content based and collaborative filtering.
- Singular Value Decomposition (SVD): The algorithm factorizes a matrix into singular value and vectors. It can be useful to reduce the proportionality of the data and to make the data more acceptable for analysis.

3) Steps to recommend a movie in Hybrid Movie recommendation system by using Cosine Similarity and Singular Value Decomposition (SVD) algorithms.

- Identify the title and index of the target movie.
- Calculate the cosine similarity of the target movie with all other movies in the dataset.
- Convert the list of cosine similarity scores into a tuple data structure, where the first element is the movie index and the second element is the similarity score.
- Sort the list of tuples by similarity score in descending order.
- Remove the first element from the sorted list, as it corresponds to the target movie.
- Apply Singular Value Decomposition (SVD) on the user-movie rating matrix to extract latent factors.
- Use the latent factors to predict the user's rating for unrated movies.
- Sort the predicted ratings in descending order and recommend the top-rated movies.
- Return the recommended movie titles corresponding to their indices.

4) Advantages of Hybrid Movie Recommendation System

- Improves Accuracy: Combining multiple methods in a hybrid movie recommendation system has been shown to enhance the accuracy of recommendations compared to using a single method.
- Increases Diversity: It provides wider range of recommendations as compared to a traditional single method system.
- Improves Scalability: It combine the memory-based and model-based approaches so that the system can handle larger datasets.
- Improves reliability: Hybrid system are capable of using multiple methods to provide recommendations, which increases the reliability of the overall system.
- Handling of the sparsity problem: A hybrid system can use a combination of collaborative and content-based filtering to make recommendation even when the user-item rating matrix is sparse.

5) Limitation of Hybrid Movie Recommendation systems

- High complexity: Hybrid systems are more complex as compared to traditional recommendation systems.
- Data requirements: A huge amount of data is required to train the system with different algorithms and to improve accuracy of the overall system.
- High computation resources: To work on the larger datasets hybrid system requires high and expensive computational power and resources.
- Data quality: A hybrid system requires high quality of data to provide more accurate result.
- Human bias: As like any other system the hybrid system can be human bias in the data which may results in unfair and biased recommendations.

V. CONCLUSION

In conclusion of this paper, we have analyzed that movie recommendation systems plays a crucial role in providing personalized and relevant recommendation to the users. There are various movie recommendation systems, which we have analyzed in this paper, including Simple Movie Recommendation system which recommends movies based on their popularities. Content-based filtering which is based on the attributes of movie itself, Collaborative filtering which is based on the preferences of similar users, and Hybrid Movie recommender system which combine the advantages of multiple systems by eliminating their limitations. Each system has its advantages and disadvantages and the choice of the system depends upon the requirements of the applications. However, we have analyzed that Hybrid systems are most advanced in between all the systems, these systems are capable of handling scalability, sparsity and improves accuracy and diversity. At last, we conclude that the use of movie recommendation systems can enhance the user experience up to a great extent by providing more relevant and personalized recommendations.

REFERENCES

[1] M J. Pazzani, D. Billsus, “Content-Based Recommendation,” The Adaptive Web Springer Link, vol. 4321, pp.325–341.
[2] X. Zhou, J. He, G. Huang, Y. Zhang, “SVD-based incremental approaches for recommender systems,” Journal of Computer and System Sciences, vol. 81, Issue 4, pp. 717-733, June 2015.
[3] Rajarajeswari, S., et al. "Movie Recommendation System." Emerging Research in Computing, Information, Communication and Applications. Springer, Singapore, 2019. 329-340.
[4] Hrisav Bhowmick, Ananda Chatterjee, jaydip Sen, “Comprehensive Movie Recommendation system”.
[5] Yeole Madhavi B., Rokade Monika D., Khatal Sunil S. “Movie Recommendation System using Content-based filtering” IJARIIE-ISSN(O) 2395-4396, Vol-7 Issue-4 2021
[6] Mohammadsadegh Vahidi Farashah, Akbar Etebarin, Reza Azmi and Reza Ebrahimzadeh Dastjerdi “A hybrid recommender system based-on prediction for movie baskets analysis, Journal of Big Data, VAhidi Farashah et al. J Big data (2021) 8:32
[7] Najafabadi MK, Mohamed AH, Mahrin MN. A survey on data mining techniques in recommender systems. Soft Comput. 2019;23(2):627–54.
[8] K. Wakil, R. Bakhtyar, K. Ali, K. Alaadin, ” Improving Web Movie Recommender System Based on Emotions,”

International Journal of Advanced Computer Science and Applications, vol. 6, No. 2, pp. 218- 226.
[9] S. Kamta, and V. Verma, "Accuracy Analysis of Similarity Measures in Surprise Framework," Evolutionary Computing and Mobile Sustainable Networks, Springer Link, pp. 861-873, August 2020.
[10] X. Guan, C. T. Li, Y. Guan, "Matrix Factorization with Rating Completion: An Enhanced SVD Model for Collaborative Filtering Recommender Systems," IEEE Xplore, vol. 5, pp. 27668-27678, November 2017.
[11] P. H. Aditya, I. Budi, Q. Munajat, "A comparative analysis of memorybased and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: A case study PT X," 2016 International Conference on Advanced Computer Science.
[12] N. Immaneni, I. Padmanaban, B. Ramasubramanian and R. Sridhar, "A meta-level hybridization approach to personalized movie recommendation," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 2193-2200, doi: 10.1109/ICACCI.2017.8126171.