Movie Recommendation System using Hybrid filtering

MORRIS DARREN BABU,
M.S Data Science,
B.E Computer Science,
Department of
computer,Friedrich-
Alexander-University
Erlangen-Nürnberg Germany

Abstract: The amount of data on the World Wide Web is growing exponentially. Users often get lost in this vast ocean of data. The recommendation system is used to filter out valuable information from a large amount of data. According to the user's choice, we propose a movie recommendation system in this document. The purpose of the movie recommendation system is to provide personalized movie recommendations selected by users. Generally, the underlying recommendation system uses the following factors to make recommendations: user preferences, called content-based filtering, or similar user preferences, called collaborative filtering. In this document, we use hybrid filtering, which is a combination of content and collaborative filtering technology, to improve and expand user recommendations. ...

Keywords: recommendation systems, content based filtering, collaborative filtering, hybrid recommender

Ⅰ. Introduction

The recommendation engine uses intelligent algorithms to provide users with recommendations according to their needs. Recommendation systems can be used in any field, from e-commerce to network security in the form of personalized services, and they provide advantages for users and service for their needs, by suggesting items to users based on their known preferences. The recommendation system works by acquiring user knowledge (explicit or implicit) of items , and has the following categories:

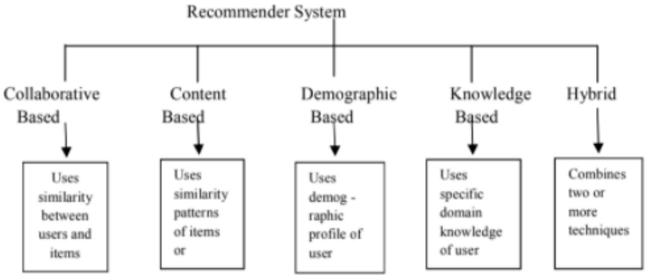

Recommender System
Collaborative Based
Content Based
Demographic Based
Knowledge Based
Hybrid
Uses similarity between users and items
Uses similarity patterns of items or
Uses demog - raphic profile of user
Uses specific domain knowledge of user
Combines two or more techniques

Fig.1. Classification of Recommender System

The main idea behind the recommendation systems is to rank the information according to the user's interest domain which is learned from the past actions of the user. To achieve the goal, recommender systems are built on three main filtering approaches. These are collaborative filtering, content-based filtering, and hybrid methods. Collaborative filtering is a technique that has been used in recommendation systems to predict and recommend items that users may like based on their known preferences. Content-based algorithms provide recommendations or suggestions based on similar types of user input.

Hybrid models combine these two approaches to overcome the disadvantage of both methods like (Cold start problem, Sparsity, Scalability). Since everyone is using the MovieLens dataset as the dataset for their recommendation system but it has many disadvantages. First, the reliability of the rating information provided by them is questionable, another drawback of movielen, it is biased to movies with a high rating.so instead we are using a dataset from IMDB and which is further verified by Wikipedia and by using this approach it can even be segregated based on age and different genres to give an even more personalized movie recommendation.

II RELATED WORK

Grupta (2015) proposed a collaborative filtering approach based on hierarchical clustering. In this work, users are divided into clusters according to hierarchical clustering and Recommendation is done by cluster voting. Chameleon is chosen over K-Means for hierarchical clustering. Rating recommendations are evaluated by mean square error.

Although optimization for hierarchical clustering and voting scheme is conducted, generalization of the system over different movie datasets raises some questions. The hierarchical clustering algorithm is mostly based on relative interconnectivity. Single, average and complete linkage must be considered for interconnectivity. Intra cluster similarity must also be calculated to identify the similarity of the items in a specific cluster. If this similarity is low that means clusters are poorly formed and recommendations cannot be trusted. The algorithm uses 25 partition points to divide or combine the existing clusters. These points are data dependent and can be high or low for the smaller or larger datasets. Instead of a fixed partition number, a dynamic optimization based on data volume could be conducted.

Cami et al., (2017) proposed a Content based filtering, a more personalized approach. Rather than similar users, preferences of a specific user are used for the recommendation. For this aim, a user profile vector is constructed. In user profile vector covers the activities of the user over a specific movie. Bayesian network algorithm is used to form clusters of similar movies. Probability of assigning a movie to a specific cluster is calculated. Then, the recommendation list is formed by the conditional probability calculations. Time and complexity of the probability calculations are the main disadvantages of this method.

Tuysuzoglu (2018) proposed a graph based hybrid recommender system. This work combines the collaborative filtering with graph theory. Movies and users are represented as verticals and edges represent the ratings. Statistical calculations are done list the most viewed movie, genre, etc. Unfortunately the effect of using graph theory over the recommendation performance is not given.

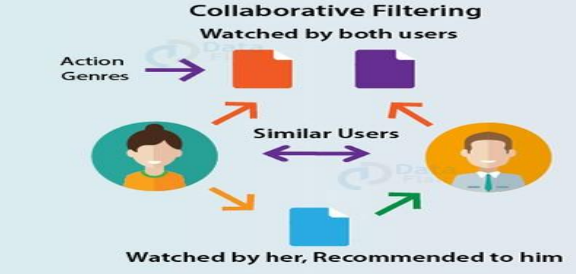
The conclusion drawn from the literature research is that sometimes these suggestions have nothing to do with our daily life, just a preliminary analysis of the suggestions, and data recovery is not fast. This work proposes a hybrid recommendation system that provides more detailed recommendations based on relevance, and uses the system to conduct fast data search for movie recommendations after collecting user and items data.

Ⅲ Methodology Background

In this section, a brief methodology background covering collaborative and content-based filtering is given

1.Collaborative Filtering

A widely used method for developing recommendation systems is collaborative filtering. Collaborative filtering is based on the assumption that people who agreed in the past will also agree in the future, and they will like item types similar to those in the system. The recommendations generated by the system only use the profile rating information of different users or elements. When they find a user/article of the same level with a similar rating history to the current user or , item they will use that neighborhood to generate recommendations. User-driven methods are user-driven algorithms, and model-driven methods are Kernel-Mapping guidelines.



Collaborative filtering methodologies are based on the user-item matrix. This matrix stores the rating information of each item for each user. This matrix is formed by the past actions of the user thus, in some studies this collaborative approach is named Memory-Based Collaborative Filtering. Based on the item–user matrix, recommendations are given according to the similarity of the users or the similarity of the items. The user-based approach tries to group the users with similar preferences on a specific domain. If a product is ranked high among most of the users in the group, it is recommended to the users who did not rank the product yet. The item-based approach focuses on item ratings. For a specific user, a rank of a new item is calculated according to its similarity to previously ranked items. In either case, the similarities must be calculated, a neighborhood must be formed, and ratings must be assigned.

Cosine similarity, Euclidian distance, Pearson correlation, and Jaccard similarity are used to can be used to calculate the similarity of the products or the users. Among them, cosine similarity and Pearson correlation is popularly used algorithm.

Suppose that there are two users (u1, u2) and n products where P = {p1, p2, ..., pn}. The rating vector of the products is given R = {r1, r2, ..., rn}. Then the rating vector for u1 is defined as ru1 = {ru1p1, ru1p2, ..., ru1pn} and rating vector for u2 is defined as ru2 = {ru2p1, ru2p2,..., ru2pn}.In this condition, similarity between rating vectors ru1 and ru2 can be calculated by cosine similarity as follows:

$$\cos(r_{u1}, r_{u2}) \frac{\sum_{i=1}^{n} r_{u1pi} \cdot r_{u2pi}}{\sqrt{\sum_{i=1}^{n}(r_{u1pi})^2} \sqrt{\sum_{i=1}^{n}(r_{u2pi})^2}}$$

Pearson correlation is the sum of dot products of the difference between a specific product rating and an average rating of the products (, divided by their sum of root product. More
formally:

$$sim(r_{u1}, r_{u2}) \frac{\sum_{i=1}^{n}(r_{u1pi}-\overline{r_{u1}}) \cdot (r_{u2pi}-\overline{r_{u2}})}{\sqrt{\sum_{i=1}^{n}(r_{u1pi}-\overline{r_{u1}})} \cdot \sqrt{\sum_{i=1}^{n}(r_{u2pi}-\overline{r_{u2}})}}$$

Neighborhood generation is generally based on
K – Nearest Neighbor (KNN) algorithm.This algorithm calculates the distance between the active user (uA) and other users (u). Then according to the predefined number k, the nearest k users form the neighborhood of the active user. Different metrics, such as Euclidian distance, Manhattan distance, and Minkowski distance, are used for distance calculations.

$$Euclidian\ (u_A, u) = \sqrt{\sum_{i=1}^{n}(u_A - u_i)^2}$$

$$Manhattan\ (u_A, u) = \sum_{i=1}^{n}|u_A - u_i|$$

$$Minkowski\ (u_A, u) = (\sum_{i=1}^{n}|u_A - u_i|^x)^{1/x}$$

After the user similarities and neighborhood is determined, the rating prediction of the active user (uA) for a new item, I, is calculated according to the formula below:

$$r_{u_{A_i}} = \bar{r}_{u_A} + \frac{\sum_{u \in N} sim(u_A, u)\ (r_{ui} - \overline{r_u})}{\sum_{u \in N}|sim(u_A, u)|}$$

Instead of similarity, neighborhood, and rating prediction calculations, some collaborative filtering approaches use machine-learning methods to construct a decision model. In this scheme, the model learns the preferences of similar users for a specific product. The learning scheme can be based on classification, clustering, and matrix factorization

The advantages of a collaborative filtering system:

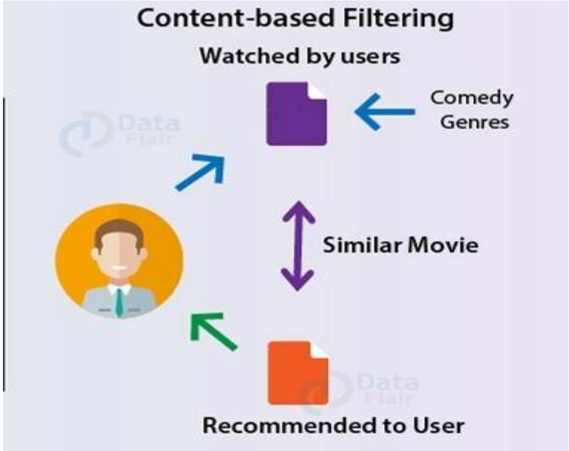
- It depends on the relationship between users, that is, they are independent of content.
- The CF recommendation system can suggest random items by observing the behavior of like-minded people.
- You can truly evaluate the quality of the object based on the experience of other people.

Disadvantages of collaborative filtering:

- Early estimation problem: The collaborative filtering system cannot make recommendations for new projects because there is no user score as a basis for prediction.
- Gray Sheep: CF-based systems require teams with similar characteristics. Even if there are such groups, it is difficult to recommend users who always disagree or disagree with these groups.
- Out of stock problem: In most cases, the number of articles clearly exceeds the number of users, and it is difficult to find articles that have been rated by enough people.

2. Content Based Filtering

This filtering method is widely used for text retrieval, news, book, or other textual data recommendation. The methodology identifies the content of the item and this content is matched to the user's profile. If the user is interested in the same or similar content, the item is recommended to the user. This scheme requires two important steps. To identify the content of a given item and to form a user profile.



This filtering method is widely used for text retrieval, news, book, or other textual data recommendation. The methodology identifies the content of the item and this content is matched to the user's profile. If the user is interested in the same or similar content, the item is recommended to the user. This scheme requires two important steps. To identify the content of a given item and to form a user profile.

Suppose that document set D consists of n documents D= {d1, d2, ..., dn} and terms are given in the term set T={t1, t2, ..., tm }, then TF for term t1in d1 is calculated as follows:

$$TF(t_1, d_1) = \sum_{i=1}^{m} \frac{t_1 \epsilon d_1}{ti \epsilon d_1}$$

$$IDF(t_1, D) = \log \sum_{i=1}^{n} \frac{d_i}{t_1 \epsilon d_i}$$

Then the weight of the term t1 is calculated as:

$$W_{t_1} = TF_{t_1} * IDF_{t_1}$$

These weights are used to construct the content vectors. These vectors are then used to learn the user preferences. Similarity calculations and/or model-based approaches can be used in this learning process.

The advantages of content filtering:

• Can recommend unrated items
• We can easily explain the recommendation system by listing the characteristics of the content of the items.
• Only the qualifications of the corresponding user are required to use the content-based recommendation system. And there are no other users of the system.

Disadvantages of content filtering:

• It is not suitable for new users who have not yet rated an items, because sufficient ratings are required. The content consultant will evaluate the user's preferences and make specific recommendations.
• Lack of suggestions for random items.
• Limited content analysis: If the system does not distinguish between items that the user likes and items that he does not like, the recommender will not work.

3. Metadata Based Filtering

By using more accurate metadata and capturing more small details, the quality of your referrals will be improved. A recommendation system is constructed based on the following metadata: 3 main characters (main protagonist, protagonist and comedian), director, related genres and plot keywords.

4.Hybrid Based Filtering

Hybrid recommendation systems are becoming more and more popular today. According to recent research, a combination of collaborative filtering, content-based filtering, and metadata-based filtering may be more effective. There are many ways to implement a hybrid recommendation system: combine the results of CF and CB recommendations, and add CF capabilities to the CB method. There are six hybrid methods:
• Weighting-adding ratings from the various components of the recommendation.
• Modification: The method is selected by modifying the various components of the proposal.
• Mixed: View recommended results from different systems.
• Features Combination: Extract features from different sources and combine them into one input.
•Feature Augmentation: Calculate performance based on recommendations and move it to the next step.
• Cascade: Use recommendation techniques to find approximate results and recommend the previous result.
Although there are many combinations theoretically, it is not always efficacious for a specific problem. The most important principle of hybrid recommendation is to avoid or make up the weakness of every single recommender technique.

Ⅳ Proposed Model

This paper proposes a hybrid recommendation system that provides content-based movie recommendation as well as collaborative filtering and metadata techniques.

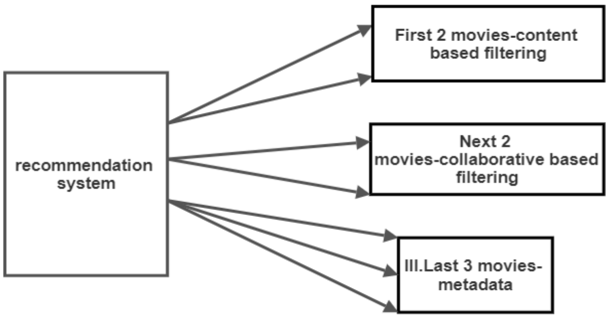
Hybrid filtering method by using IMDB dataset and Wikipedia, it can be segregated year-wise to give even more personalized movie recommendations.

In this hybrid movie recommendation system, we are going to recommend 7 movies in total, which follows:

I. First 2 movies-content based filtering

II. Next 2 movies-collaborative based filtering

III. Last 3 movies- metadata


recommendation system
First 2 movies-content based filtering
Next 2 movies-collaborative based filtering
III.Last 3 movies-metadata

Firstly in content-based movie recommendation, we are going to use the TFIDF matrix in the linear kernel and recommend a movie that is based on the same genre.

Secondly, in collaborative-based movie recommendations, we are going to use K nearest neighbor and recommend a movie that is based on similar user ratings.

lastly, in metadata-based movie recommendation, we are going to use K nearest neighbor and recommend a movie that is based on similar user ratings.

Ⅴ Working

This paper proposed a hybrid recommendation system that provides movie recommendations based on content and collaborative filtering techniques.

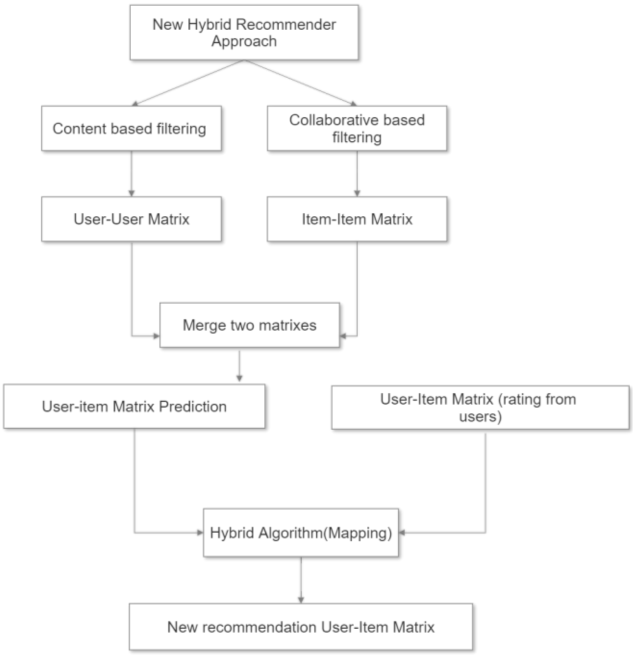
The methodology to recommend movies has the following steps with explanation:

a) Collect user information For the new user, the system requests to register him an account to gather his personal information and also the ratings of a set of movies.

b) In the second step the system will fetch the ratings of the movies and store this information in the movies database.

c) In the next step relevant features have to be gathered to use in the movie recommendation system.

d) Then we collect those users who shared common movies with the active user i.e. determine the neighbor set for an active user.


New Hybrid Recommender Approach
Content based filtering
Collaborative based filtering
User-User Matrix
Item-Item Matrix
Merge two matrixes
User-item Matrix Prediction
User-Item Matrix (rating from users)
Hybrid Algorithm(Mapping)
New recommendation User-Item Matrix

Techniques:

Techniques used for content and collaborative approaches are:

Content-oriented methods and techniques:

TF-IDF: Terms that frequently appear in documents (TF = term frequency), but rarely appear in the rest of the corpus (IDF = inverse frequency document) are used more frequently. , The normalized weight vector can prevent longer documents from being searched for a better chance. The TF-IDF function illustrates these assumptions well.

Naive Bayes: Naive Bayes is a probabilistic method of inductive learning, which belongs to the general category of Bayesian classifiers. These methods create probabilistic models based on previously observed data. The model estimates the posterior probability P (c | d) of document d, which belongs to class c. Based on the prior probability, P(c), the probability of observing the document of type c, P(d|c), for a given c, the probability of observing the document d, and P(d) ,Using these probabilities, the Bayes theorem is applied to calculate P(c|d).

Techniques of collaborative Based Approach:

K-MEANS CF: K-Means grouping is used to identify segments. K-Means is a grouping method widely used in data mining, statistics and machine learning. Among grouping elements, the distance indicates the difference of the elements. The number of groups k is also an input parameter. This is an iterative algorithm that first randomly divides the elements into k groups. , Calculate the center of gravity of the cluster, and redistribute each element to the cluster with the nearest center of gravity.

2.Matrix decomposition: The algorithm decomposes the matrix of user interaction with the movie into the product of two low-dimensional rectangular matrices, such as U and M. Decompose so that the result of the product is almost similar to the interaction between the user and the movie.

3. CLUSTER MODEL:

In order to find customers of similar users, the clustering model divides the customer base into many market segments and treats the problem as a classification problem. The goal of the algorithm is to assign users to the market segments that contain the most similar customers. The cluster model divides the customer base and users into many parts, and treats tasks as classification problems.

4. BAYESIAN CLASSIFIER:

Bayesian statistics that our prior beliefs are conjugate priors on μ and Σ: Σ ~ Inv-Wishartν0(Λ−10) μ|Σ ~ N μ0, ∑ 9 where ν0, Λ0, μ0, k0 are hyper-parameters of the model, that is, parameters specifying our prior belief about parameters Σ and μ before observing the data. The scalar hyper parameter ν0 describes the degrees of freedom and the matrix Λ0 describes the scale of inverse-Wishart distribution. The vector hyper-parameter μ0 is the prior mean and the scalar k0 is the scaling of prior variance.

Similarity:

There are various similarity measures to find out the similarity between user and item. Some of them are discussed here.

To calculate the similarity between movies the below methods are used

1.Jaccard Similarity:

It is the ratio of common items rated by the user to the total number of items rated by both the users.

2. Cosine Similarity Cosine similarity:

It is the measure of similarity between two non-zero vectors in the inner product space. It measures the angle between these two vectors. A cosine of two non-zero vectors can be calculated using the dot products of these two vectors.

3. Pearson Correlation Similarity;

It uses Pearson Correlation Coefficient to determine the similarity between users. The higher the coefficient the two users are more closely related.

4.Tanimoto coefficient.

It is a similarity between the two sets. It is a ratio of intersections. Assume that set X is {B,C, D} and set Y is {C, D, E}. The Tanimoto coefficient T of two set A and B is 0.5. This metric does‟t consider the user rating but the case of a very sparse data set is efficient.

5.Spearman Rank Correlation: Spearman Rank Correlation also measures the strength of the linear relationship between two variables. Unlike Pearson's correlation, this indicator considers many estimates. Appropriate outside the range of standardized preferences. Since the range of preference estimation for CF is normalized, the Spearman rank correlation in the CF field shows properties equivalent to Pearson correlation.

6.Mean Square Distance:The square of the difference between an item rated by one user and the common item rated by two users. Then calculate the similarity of the square by subtracting 1 .

7.Relevant Jaccard Mean Square Distance (RJMSD): Relevant Jaccard mean square distance is obtained by multiplying relevant Jaccard and mean square distance.

Performance:

Rating indicators for recommender systems There are many ways to evaluate recommender systems. These indicators are a way to determine the accuracy of the recommendation system. Accuracy is a prerequisite for any recommendation system to work. The estimated indicators are Mean Absolute Error (MAE), Mean Square Error (RMSE), accuracy, recovery, F1 measurement, and cumulative diversity.

Two commonly used methods are:

1.Mean Absolute Percentage Error

$$MAPE = \frac{1}{N}\sum\frac{|r-r^{\wedge}|}{r} X\,100$$

2.Root Mean Square Error:

$$\text{RMSE} = \sqrt{\frac{(r-r^{\wedge})^2}{N}}$$

Other performance metrics are:

3.Mean Absolute Error: The lower the MAE, the more accurate the recommendation engine’s prediction of user ratings.

4.Accuracy: Accuracy indicates the corresponding result, that is, the article correctly recommended by the system.

5.F1-Measure: To find F1-Measure, we must first find the accuracy and recall first.

6. Diversity:Where u is a specific user, U is a total user in the record, and Ln(u) is a list of matching items recommended to user u.

VI Future Work

1. To make an even more personalized recommendation, need to work in an, even more, larger dataset.

2. Create an API and make an interconnection to YouTube, to give trailers of the recommended movies

3. Create a user login page and store user details in the database and create a user profile in which a user can make a dashboard to add which movies to watch next and give recommendations for the previously watched movies also.

4. Allow users to add or remove movies in the dashboard and create a profile that can be displayed publicly for peers to follow and use the movies list (like Spotify playlist).

5. Display movie posters, images, and cast for the recommended movies.

6. Create an option to give ratings and write reviews for the movies by using a text-mining algorithm we can even predict movies based on the written reviews.

7. Make an analysis of ratings and visualize it and show the top movies of each genre category.

VII Conclusion

Recommendation systems are used to filter a vast amount of data and present relevant information according to user preferences. These systems can be applied to many different domains and movie recommendation is among them. Movie recommendation systems use three main approaches. These are collaborative filtering, content-based filtering, and metadata-based systems.

Collaborative filtering mainly suffers from data sparsity and content-based filtering suffers from cold start problems.

To overcome the disadvantages hybrid systems are proposed. These systems combine the three main filtering approaches by different methodologies such as weighting, clustering, regression, and augmentation. Some hybrid systems combine a machine learning model along with a filtering method. In such systems, the main problem is the number of decision models. Often, a decision model is built for every specific user or item. This increases complexity as well as computation time.

So, combining 3 filtering methods in the proposed order and make a recommendation using an IMDB dataset and combine it with the Wikipedia dataset will give an even more personalized movie recommendation.

By using this method we can even segregate more movies for different age of people and various genres according to user likes. The main advantage we can recommend movies in a user-defined particular year also ranging from the 1850s.

VIII REFERENCE

1.Gupta U., Patil N., 2015. Recommender System Based On Hierarchical Clustering Algorithm Chameleon. IEEE International Advance Computing Conference (IACC). DOI: 10.1109/IADCC.2015.7154856.

2.Cami B. R., Hassanpour H., Mashayekhi H., 2017. A Content-Based Movie Recommender System Based on Temporal User Preferences. Third Iranian Conference on Intelligent Systems and Signal Processing (ICSPIS). DOI: 10.1109/ICSPIS.2017.8311601.

3.Tüysüzoğlu G., Işık Z., 2018. Hybrid Movie Recommendation System Using Graph-Based Approach. International Journal of Computing Academic Research (IJCAR), 7(2): 29-37.

4. K. Soni, R. Goyal, B. Vadera, and S. More, “A Tree Way Hybrid Movie Recommendation Syste,” International Journal of Computer Applications, vol. 160, no. 9, pp. 29–32, 2017.

5. Chen R., Hua Q., Chang Y.S., Wang B., Zhang L., Kong X.A, 2018. Survey of Collaborative Filtering-Based Recommender Systems: From Traditional Methods to Hybrid Methods Based On Social Networks. IEE Access. DOI: 10.1109/ACCESS.2018.2877208.

6. Rombouts J., Verhoef T., (Date of access: July 2019). A Simple Hybrid Movie Recommender System. http://www.fon.hum.uva.nl/tessa/Verhoef/Past_projects_files/Eind_Rombouts_Verh oef.pdf.

7. Xiao T., Shen H., 2019. Neural Variational Matrix Factorization with Side Information for Collaborative Filtering. In: Yang Q., Zhou Z.H., Gong Z., Zhang M.L., Huang S.J., Eds. Advances in Knowledge Discovery and Data Mining. PAKDD 2019. Lecture Notes in Computer Science. Springer, Cham.

9. Xiao T., Shen H., 2019. Neural Variational Matrix Factorization with Side Information for Collaborative Filtering. In: Yang Q., Zhou Z.H., Gong Z., Zhang M.L., Huang S.J., Eds. Advances in Knowledge Discovery and Data Mining. PAKDD 2019. Lecture Notes in Computer Science. Springer, Cham.

10. Bobadilla, Jesus, Santiago Alonso, and Antonio Hernando, ”Deep Learning Architecture for Collaborative Filtering Recommender Systems”, Applied Sciences 10(7):24-41, 2020. DOI: http://dx.doi.org/10.3390/app10072441

11. Virk H.K., Singh M., Singh A., 2015. Analysis and Design of Hybrid Online Movie Recommender System. IJIET.

12. S. Bag, S. K. Kumar, and M. K. Tiwari, ”An efficient recommendation generation using relevant Jaccard similarity”, Information Sciences 483:53-64, 2019.

13. S. Bansal, C. Gupta, and A. Arora, ”User tweets based genre prediction and movie recommendation using LSI and SVD”, IEEE Ninth International Conference on Contemporary Computing , 1-6, 2016. DOI: http://dx.doi.org/10.1109/IC3.2016.7880220.

14. Singh, Tarana, Anand Nayyar, and Arun Solanki, ”Multilingual Opinion Mining Movie Recommendation System Using RNN”, First International Conference on Computing, Communications, and Cyber-Security, Springer, Singapore, 2020.