An Enhanced Recommendation System Using Ensemble Learning Techniques (Aersuelt)

Aman Kumar Sharma, *[1] Harsh Tyagi, [2] Rajat Moundekar, [3] and Parthasarathy G [4]
[1,2,3] M. Tech Scholar, School of Computer Science and Engineering, Vellore Institute of Technology
Vellore – 632014, Tamil Nadu, India
[4] Assistant Professor, School of Computer Science and Engineering, Vellore Institute of Technology
Vellore – 632014, Tamil Nadu, India
E-mail: amankumar.sharma2022@vitstudent.ac.in

Abstract

The recommendation system is one of the most prominent ensemble learning applications, attracting numerous researchers from all over the world. The Internet age has resulted in the widespread use of recommendation system in our daily lives. The Recommendation System indicates what is most comparable to a user's selection. It seeks to anticipate and filter preferences based on the user's preferences. The recommendation system may be realized using a variety of ensemble learning approaches. Choosing the optimal ensemble learning algorithm to give a product or service to consumers is therefore the most difficult challenge in the recommendation system sector. Among the three basic strategies used to develop a Recommendation System, we are working with a Hybrid Content-Collaborative based approach among all of them. This project's objective is to scientifically design a user-recommendation system and determine if we can establish a strategy that can deliver more accurate recommendations to users via a filtering process based on the user's previous history and surpass existing basic strategies. This project is intended to assist us in knowing how a recommendation system works.

Keywords: Ensemble learning, Recommendation system, Content-Based, Collaborative-Based, Hybrid-Based.

1. Introduction

Generally, recommendation systems have been understood to be those in which human inputs (recommendations) are aggregated and sent to relevant end users [4]. The recommendation system's objective is to provide customers with personalized items. The system of recommendations has been implemented in several fields. Many studies have looked at the topic of recommending movies, music, news, hotels, books, online stores, and vacation spots [13]. Numerous approaches and procedures have been developed alongside the recommendation system's growth. The three major techniques used to build a Recommendation System are content-based, collaborative-based, and hybrid-based [15].

1.1Content-Based Filtering

This kind of recommendation system relies on data that users have already submitted. Google, Wikipedia, and other sources are examples of content-based suggestions [15].

1.2 Collaborative-Based Filtering

Collaborative filtering is the approach used most often. The main benefit of collaborative methods is that they are not dependent on any kind of machine-readable representation of the suggested things. In general, collaborative filtering algorithms gather user ratings for products in a certain area and then calculate the likelihood that many users would suggest a product based on their ratings. [15].

1.3 Hybrid Filtering

The hybrid approach is implemented, by combining collaborative filtering and content-based filtering techniques [15].

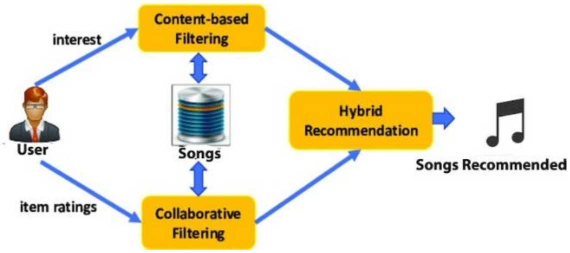


Fig. 1. Hybrid Filtering [14]

2. Literature Survey:

In A Comparative Study of Music Recommendation Systems by Ashish Patel and Dr. Rajesh Wadhvani, dealt with in the context of an incremental regression tree model and a graph-based approach. The quantity, accuracy, and speed of music recordings produced as a result of user input are crucial considerations. The majority of the suggested solutions largely depend on client-server architecture, necessitating user online presence to get suggestions. [1].

Elena Shakirova proposed Collaborative Filtering for Music Recommender System, she estimated the performance of recommender systems using collaborative filtering approaches and assessment measures. She developed a theoretical foundation for using collaborative filtering approaches in a music recommender system [2].

Thomas Hornung et al. proposed the evolution of a hybrid music recommender system, that provides a prototype of the TRecS (Track Recommender System) that integrates many different recommender methods into a single predictive score. Track similarity, tag similarity, and the listening profiles through time are the three main factors used to determine how similar two songs are to one another on Last.fm (time similarity). The TRecS recommender is a hybrid with a weighting system. A survey involving more than 140 people shows that the quality of the suggestions people assess increases with time [3].

Xixi Li et al. proposed a hybrid recommendation method based on features for offline book personalization, they use a qualitative technique to calculate the weighted similarity between consumers, which has to be refined further. In this article, they apply the average approach to establish the weight to change the anticipated rating, which has to be improved further. Furthermore, they employ word2vec to determine consumer preference on book kinds, with the dimension set to 50, which requires additional debate. Furthermore, the performance of their strategy has yet to be evaluated on additional data sets [4].

M.Sunitha Reddy, T.Adilakshmi, and V. Swathi proposed a hybrid recommendation system that incorporates association mining and clustering. The authors' primary goal was to resolve the CBF difficulty, often known as the cold start problem. This is how the suggested algorithm operates: User clusters are created in the initial stage. These clusters are created using collaborative filtering and similarity measurements made using the cosine similarity approach. Each cluster is transformed into a transactional database in the subsequent stage. Using an expanded FP tree, this transactional database is used to locate common item sets. When necessary, the tree is traversed to create strong association rules from the often occurring item sets. This process raises the quality of suggestions. [5].

Monali Gandhi, Khusali Mistry, and Mukesh Patel discuss the justification for utilizing hybrid algorithms. The writers discussed the various recommendation techniques. When the data set is insufficiently accessible, association mining alone is ineffective. The suggested system breaks down each of its task modules into four segments. Database preselection, association rule mining, and collaborative filtering are all done in the first stage. Finally, the consumer is recommended the top N items. [6].

Satya Prakash et al., presented a comparative analysis of machine learning algorithms. All the algorithms described in this paper are compared concerning their precision rates. This detailed evaluation shows the advantages and disadvantages of each of the many variants of the Movie Lens dataset. The experimental results testify to the algorithm's proficiency with sparsity [7].

Mohammadsadegh et al. proposed a link-based hybrid recommender system for analyzing movie baskets, They made an effort to get beyond the problem of a cold start in online movie networks and accurately and pertinently introduce movies to new users. To do this, clustering methods including DNN, CBRSs, and collaborative filtering were used. In order to provide new users with suitable movies that were far more accurate than any other previously utilized methods, the researcher in this work used clustering algorithms, DNN, hybrid similarity criteria, and an improved Friend link algorithm. The proposed methodology takes longer to execute and process than other approaches do. The suggested method cannot be used on all websites and systems. Big data strategies need to be updated. [8].

Pasquale et al., suggested, a content-based recommender system which is implemented based on the Learning of profile, Quality improves over time, Considers implicit feedback. The limitation of this method is this does not completely overcome the problem of over-specialization and serendipity [9].

3. Dataset:

We are using a million songs dataset [12] which contains a count_data.csv file and a song_data.csv file. The "count_data.csv" contains 720945 records of user ID, corresponding to song ID and the number of times the song was played. It means we have the list of who was listening to this or that song and how many times.

Table 1 Dataset [12]

Dataset	**Users (Unique)**	**Songs (Unique)**	**Total data**
Million Songs	71798	3778	720945

4. Proposed Model

The three major techniques used to build a Recommendation System are CBF, CF, and hybrid-based. According to the literature review, the following difficulties still persist in Recommendation System. First Scalability (Unable to handle large-scale datasets) in Collaborative Filtering, Second Cold Start (Arises when no information is found about the user or item in the system) in Content-Based Filtering. So, A hybrid recommendation approach is proposed to address the aforementioned concerns. So, we are dealing with Hybrid Content-Collaborative based technique as shown in the following figure.

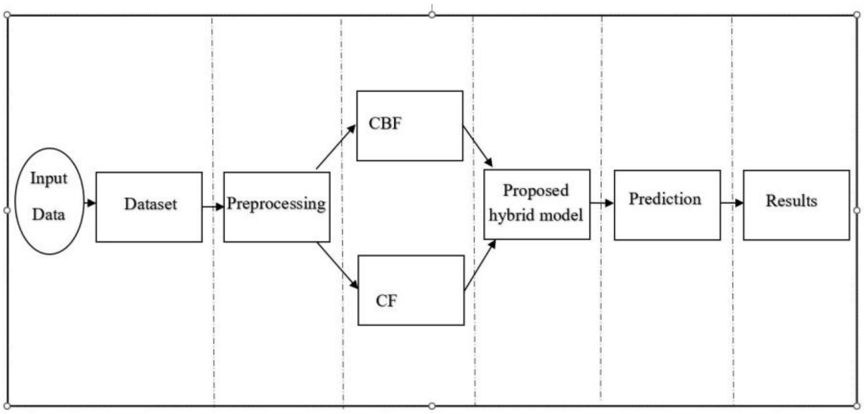


Fig. 2. Proposed Model

4.1 Preprocessing

We are using the Million songs Dataset which has two files since both the datasets have the column song_id, we can merge these datasets using this column. The user_id and the song_id is encrypted and these encryptions do not provide information about the user or song. So, we transform these variables using label encoding to ease the processing

4.2 Content-Based Filtering

Content-Based Filtering is dependent on inputs provided by users in past. This can be implemented by using two methods first is the vector space method and second is classification method. Here we are dealing with the vector space method [15]

4.3 Collaborative Filtering

In general, collaborative filtering algorithms locate products that are likely to be recommended by many users by gathering user ratings for those items in that field. Collaborative filtering falls into two categories: the first is user-based, which gauges how similar target users are to other users. The second method, known as item-based analysis, quantifies how comparable the objects that target users evaluate or engage with are to other items. User-based Collaborative Filtering is used here. [15].

4.4 Hybrid Filtering

The hybrid approach uses both collaborative filtering and content-based filtering techniques. This method overcomes the limitations of each particular algorithm and enhances the system's performance. [15].

5. Experiments and Results

5.1 Data Visualization

The ability to see, interact with, and gain insight from data is a key function of data visualization. The perfect graphic can get everyone on the same page, regardless of their knowledge level, for any topic, no matter how basic or complicated [16]. So here we visualize our processed and cleaned dataset and plotted some graphs which are following.

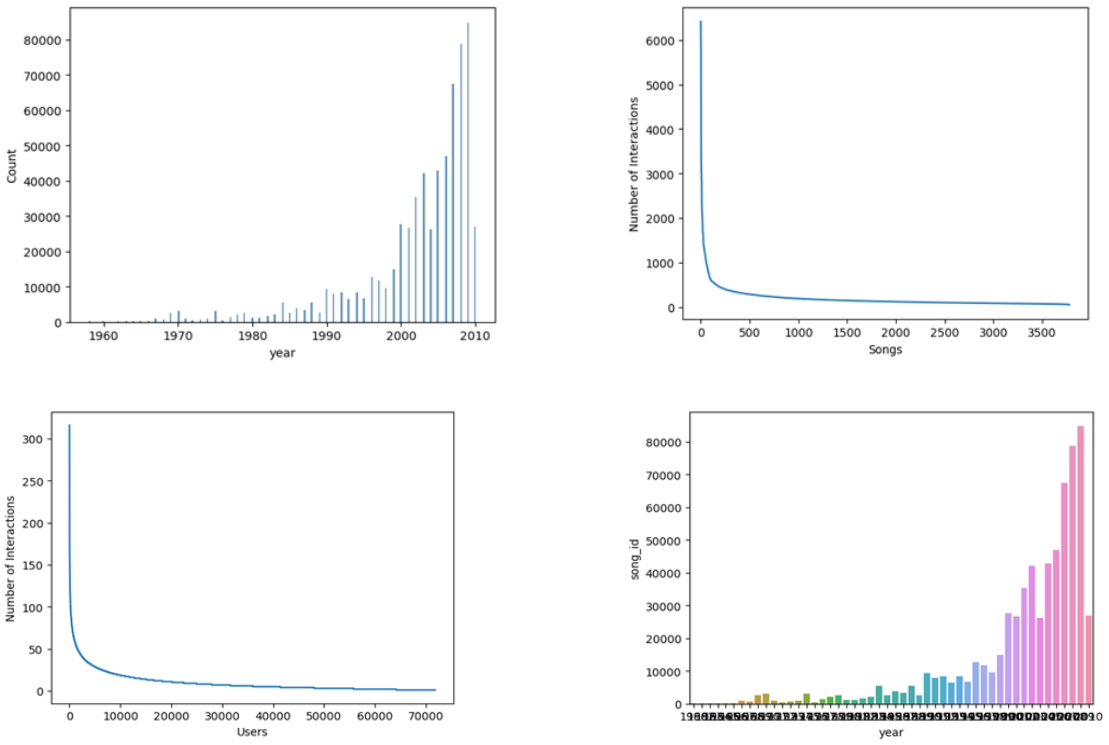


Fig. 3. Data Visualization

5.2 Implementation of Collaborative Recommendation System

Model-based collaborative filtering finds groups of users with similar tastes to predict the songs they like based on their previous behavior (what songs they listened to and how many times). For collaborative filtering to complete matrix estimation, we will use Singular Value Decomposition (SVD). SVD is fundamentally a matrix factorization algorithm that decomposes any matrix into three general and known matrices. The SVD of mxn matrix A is given by the formula:

A = UWV^{T} (1)

For validation of the model, we can use a simple random 80/20 split for training/testing sets or use k-fold Cross-Validation. Model-based Collaborative Filtering is a customized suggestion system that is not reliant on any extra information and is based on the user's prior behavior. To find suggestions for each user, we analyze latent characteristics. Latent Features are the features that are not present in the empirical data but can be inferred from the data. SVD is used to compute the latent features from the user-item matrix. But SVD does not work when we miss values in the user-item matrix [17].

Now, we will apply SVD and use latent features to find recommendations for each user. For SVD Algorithm we are using a reduced 5000 x 5000 matrix, for faster calculations It is a supervised learning method [17]. We need to check that we are not guilty of overfitting. complexity - the size of these matrixes. Need to check RMSE (Root Mean Square Error). If k is very high, we are likely to overfit. If k is very low we are likely to underfit. We need to decide; what k should be to avoid overfitting and underfitting. Splitting the dataset and selecting optimal latent variables. Now, we need to find the appropriate K (the number of latent features) to use in order to re-generate the interaction matrix and make predictions. We will choose the K which gives good performance on the train and test data. By changing k we potentially change the accuracy of our predictions on the test data. We have calculated U and Vt matrices for the train as well as test data. Now, we need to find the number of latent features that give us the lowest RMSE on the train and the test data.

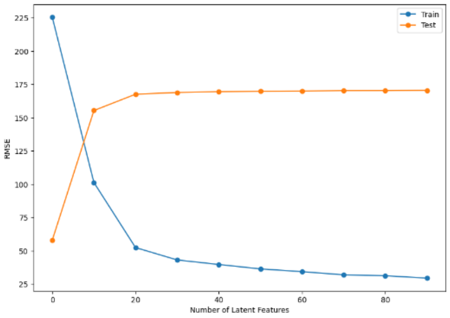


Fig. 4. RMSE v/s Number of Latent Features for SVD Model-Based Collaborative RS

From the data science point of view, key measures of success are:

1. Root Mean Square Error (RMSE).

The RMSE, is a prominent way of measuring accuracy metrics that was adopted from the literature on regression modeling. [2]:

$$RMSE = \sqrt{\frac{1}{|k|}\sum_{(u,i)\in k}(p_{ui} - r_{ui})^2} \qquad (2)$$

Where p_{ui} is the predicted rate, k — is the quantity of testing rates. The smaller value RMSE is better.

2. Precision, recall

Precision at K is defined as the fraction of accurate suggestions inside the projected top-k positions. Let y denote a ranking over items Y ↔ I: y(p) = i, which means that item i is ranked at position p [2].

$$p_k(u,y) = \frac{1}{k}\sum_{p=1}^{k} r_{uy(p)} \tag{3}$$

3. Mean Average Precision at K (MAP@K) where K is the Number of latent features

$$AP(u,y) = \frac{1}{\tau_u}\sum_{p=1}^{\tau} p_k(u,y) r_{uy(p)} \tag{4}$$

The idea of mAP is compute AP for each user:

$$mAP(u,y) = \frac{1}{N}\sum_{p=1}^{N} AP(u,y_u) \tag{5}$$

Table 2 RMSE Score for SVD Model

Model	**Dataset**	**RMSE Score**
SVD Model-Based Collaborative	Million songs	0.0296

The RMSE is low which implies that the majority of predicted ratings are close to the actual ratings.

Table 3 Mean Precision and Mean Recall for SVD Model-based Collaborative

Model	**Dataset**	**K**	**Mean Precision**	**Mean Recall**
SVD Model-Based Collaborative	Million songs	10	0.1776	0.0400
SVD Model-Based Collaborative	Million songs	20	0.1433	0.0640
SVD Model-Based Collaborative	Million songs	30	0.1298	0.0865
SVD Model-Based Collaborative	Million songs	40	0.1170	0.1034

5.3 Implementation of Content-Based Recommendation System

In a content-based recommendation system, we would be using the feature - text. In this dataset, we don't have any song reviews but we can combine the columns - title, release, and artist_name to create a text-based feature and apply the tf-idf feature extraction technique to extract features, which we later use to compute similar songs based on these texts.

TF-IDF is a measure for determining the relative importance of string representations (words, phrases, lemmas, etc.) inside a document or set of documents [18]. For measuring similarities, we will use cosine similarity here. Cosine similarity is a statistical metric for comparing the closeness of two number sequences. This similarity is defined as the cosine of the angle between two sequences or the dot product of two vectors divided by the product of their lengths in an inner product space. Therefore, the angle of the vectors rather than their magnitudes determine the cosine similarity. Similarity calculated using the cosine function is constrained to be between -1 and 1 [19].

$$\text{cosine similarity} = S_C(A,B) := \cos(\theta) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}}, \tag{6}$$

Table 4 Mean Precision and Mean Recall for Content-Based Model

Model	**Dataset**	**K**	**Mean Precision**	**Mean Recall**
Content-Based Model	Million songs	10	0.0546	0.0618
Content-Based Model	Million songs	20	0.0521	0.1183
Content-Based Model	Million songs	30	0.0496	0.1677
Content-Based Model	Million songs	40	0.0461	0.2068

5.4 Implementation of Hybrid Recommendation System

A hybrid approach is designed by combining the CBF and CF results. This approach overcomes the drawbacks of each individual algorithm and improves the performance of the system. So here we are implementing the hybrid approach. First, we Split the dataset into train/test (80/20) and then apply the following steps:

Step1. Apply Content Based. -> Get Interactions_Matrix_Prediction_CONTENT

Step2. Hybrid = Content Based + SVD ->

# -> Interactions_Train_Actual - Interactions_Matrix_Prediction_CONTENT = Ldiff (we need to estimate with SVD)

Step3. Hybrid_Prediction = Interactions_Matrix_Prediction_CONTENT + Estimation of Ldiff

We use the SVD method on the Ldiff training interaction matrix. Splitting the dataset and selecting optimal latent variables. Now, we need to find the appropriate K (the number of latent features) to use in order to re-generate the interaction matrix and make predictions. We will choose the K which gives good performance on the train and test data. By changing k we potentially change the accuracy of our predictions on the test data. Now we build Hybrid_predicted from Content_Predicted and Ldiff_Predicted and create a function to recommend the top songs. For user_index = 71750, the num_recommendations = 10 is shown in Figure 6.

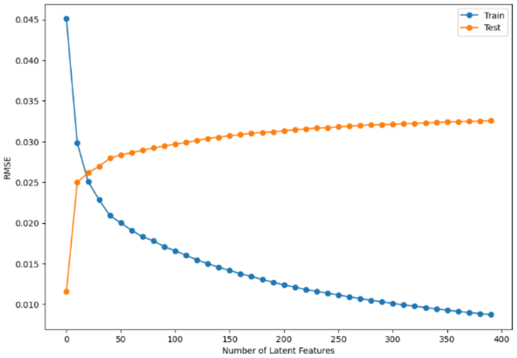
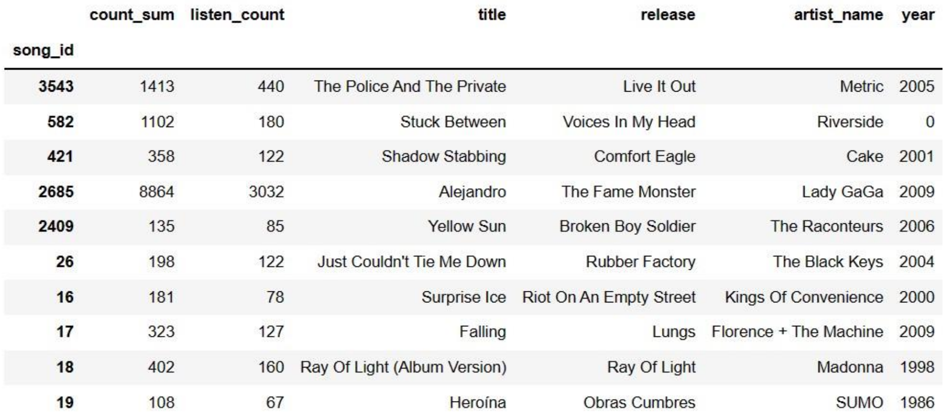


Fig. 5. RMSE v/s Number of Latent Features for Hybrid RS



	count_sum	listen_count	title	release	artist_name	year
song_id						
3543	1413	440	The Police And The Private	Live It Out	Metric	2005
582	1102	180	Stuck Between	Voices In My Head	Riverside	0
421	358	122	Shadow Stabbing	Comfort Eagle	Cake	2001
2685	8864	3032	Alejandro	The Fame Monster	Lady GaGa	2009
2409	135	85	Yellow Sun	Broken Boy Soldier	The Raconteurs	2006
26	198	122	Just Couldn't Tie Me Down	Rubber Factory	The Black Keys	2004
16	181	78	Surprise Ice	Riot On An Empty Street	Kings Of Convenience	2000
17	323	127	Falling	Lungs	Florence + The Machine	2009
18	402	160	Ray Of Light (Album Version)	Ray Of Light	Madonna	1998
19	108	67	Heroína	Obras Cumbres	SUMO	1986

Fig. 6. Recommendations for User_index=71750

Table 5 RMSE Score for Hybrid Model

Model	**Dataset**	**RMSE Score**
Hybrid Model	Million songs	0.0299

The RMSE is low which implies that the majority of predicted ratings are close to the actual ratings

Table 6 Mean Precision and Mean Recall for Hybrid Model

Model	Dataset	K	Mean Precision	Mean Recall
Hybrid Model	Million songs	10	0.016186	0.0174
Hybrid Model	Million songs	20	0.013225	0.0277
Hybrid Model	Million songs	30	0.010544	0.0327
Hybrid Model	Million songs	40	0.008829	0.0359

6. Conclusion

As per the Proposed System, we have implemented a CBF, CF, and hybrid recommendation system in our million songs dataset. In the hybrid, we have combined content-based and collaborative filtering and also, we have tested all models with our dataset. The results show hybrid model performed well and also, we addressed the issues of scalability through collaborative recommendation system and cold start through a content-based recommendation system.

References

[1] A. Patel and R. Wadhvani, "A Comparative Study of Music Recommendation Systems," 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2018, pp. 1-4

[2] E. Shakirova, "Collaborative filtering for music recommender system," 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), St. Petersburg and Moscow, Russia, 2017, pp. 548-550

[3] T. Hornung, C. -N. Ziegler, S. Franz, M. Przyjaciel-Zablocki, A. Schätzle and G. Lausen, "Evaluating Hybrid Music Recommender Systems," 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Atlanta, GA, USA, 2013, pp. 57-64

[4] Li, Xixi et al. “A Hybrid Recommendation Method Based on Feature for Offline Book Personalization.” ArXiv abs/1804.11335 (2018)

[5] M.Sunitha Reddy, T.Adilakshmi, V.Swathi, “A Novel Association Rule Miming and Clustering based hybrid method for Music Recommendation System”, International Journal of Research in Engineering and Technology ,Vol 3,Issue 5 ,May 2014, pp 55-59.

[6] Monali Gandhi , Khushali Mistry, Mukesh Patel , “ A Modified Approach towards Tourism Recommendation System with Collaborative Filtering and Association Rule Mining”, International Journal of Computer Application, Vol 91, No 6, April 2014, pp 17-21.

[7] Anand Nautiyal & Satya Prakash Sahu, Mahendra Prasad, “Machine Learning Algorithms for Recommender System - a comparative analysis,” 2017 International Journal of Computer Applications Technology and Research. 6. 97-100. 10.7753/IJCATR0602.1005.

[8] Vahidi Farashah, M., Etebarian, A., Azmi, R. et al. A hybrid recommender system based-on link prediction for movie baskets analysis. J Big Data 8, 32 (2021).

[9] Lops, P., de Gemmis, M., Semeraro, G. (2011). Content-based Recommender Systems: State of the Art and Trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds) Recommender Systems Handbook. Springer, Boston, MA

[10] Aggarwal Charu C.. 2016. Recommender Systems the Textbook. Springer International Publishing.

[11] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011

[12] http://millionsongdataset.com/ last accessed on 20.10.2022

[13] https://towardsdatascience.com/recommendation-systems-explained-a42fc60591ed last accessed on 05.11.2022

[14] https://analyticsindiamag.com/a-guide-to-building-hybrid-recommendation-systems-for-beginners/ last accessed on 08.11.2022

[15] https://towardsdatascience.com/brief-on-recommender-systems-b86a1068a4dd last accessed on 12.12.2022

[16] https://www.tableau.com/learn/articles/datavisualization#:~:text=The%20importance%20of%20data%20visualization,of%20their%20level%20of%20expertise. last accessed on 21.12.2022

[17] https://towardsdatascience.com/recommender-system-singular-value-decomposition-svd-truncated-svd-97096338f361 last accessed on 08.01.2023

[18] https://www.capitalone.com/tech/machine-learning/understanding-tf-idf/ last accessed on 08.01.2023

[19] https://en.wikipedia.org/wiki/Cosine_similarity last accessed on 08.01.2023