Product Based Recommendation System On Amazon Data

Product Based Recommendation System On Amazon Data

Rohit Dwivedi
Galgotias University, Greater Noida, India
rohitdwivedi16111998@gmail.com

Abhineet Anand
Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab
abhineet.anand@chitkara.edu.in

Prashant Johri
Galgotias University, Greater Noida, India
Johri.prashant@gmail.com

Arpit Banerji
Galgotias University, Greater Noida, India
Arpitbanerji9161@gmail.com

NK Gaur
Galgotias University, Greater Noida, India
Nit.gaur1982@gmail.com

***Abstract*— Data have grown up uncontrollably due to which there are large number of products that are listed on e-commerce website today. In a way through which users can quickly find a favorite item from big resources, the user requires such technology by which it can be automated so recommendation systems were introduced. Recommendation systems are mainly used by companies like e-commerce to help the user discover items they have to found out by them and increase the sales of the company. A recommender engine can recommend user interest products. In this paper the discussion various different recommendation system, evaluation techniques and also the challenges and problem in the system are discussed. Also using Amazon electronics data building of popularity-based recommender engine and recommender engine using collaborative filtering that is based on singular value decomposition is discussed. The goal of the model is to recommend users 5 top products to the user and performance of each model is evaluated.**

Keywords*— Popularity-based recommendation, Collaborative Filtering, Singular Value Decomposition.

I. INTRODUCTION

Recommendation systems had changed the way of interaction between user and websites and are increasingly important today. Recommender system enhance accesses and take charge to recommend appropriate items to users by in view of the users raised choices and objective behaviors.
Being an online advertisement or e-commerce websites, recommender system cannot be avoided today. Every other company is trying to use the power of recommendation system. These systems have huge application in different sectors that are education, economy and researches, like much other work[1]–[5]. The rate of digital information is increasing rapidly due to rapid growth of information technology. The recommendation system has attained great results solving the problem of data overloaded. There is a very large number of products that are listed today on e-commerce websites like Flipkart, Amazon. Recommender system helps user when they face huge amount of choices[6]. There are almost more than 30 million products present on Flipkart today. Due to which it has become tough for customers to choose their desired choice. The recommender system deals with many data present by filtering the most important information based on historical data of a user which takes care of the user's preference and interest. Recommender system can predict whether a particular user would prefer an item or not based on the user's profile. It can predict whether a user would prefer a product or not based on the user's historical data[7]. Recommender systems are beneficial to both service providers and users. The quality and decision-making process has also improved through these kinds of systems[6]. Recommender systems result in mainly things stated below:

- Benefits users in finding items of their interest.
- Help item providers deliver their items to the right user.
- Identity products that are most relevant to the user.
- Personalized content.
- Help website improves user engagement.

There are two books that are listed on Amazon. The first book is 'Touching the Void' which was written by a mountaineer in 1988. At that time Amazon started as online book seller. But unfortunately, the book did not get much attention from the people and as a result, there were not many sales for that book. After a few years in 1996, another book came up with the title 'Into thin Air' which had similar story and got popular and received huge sales. That was the time when Amazon started with off implementing recommender system. So, whosoever bought 'Into thin Air' book, Amazon started recommending them 'Touching the void' book, as a result, the sales of the old book surpassed the new book. This magic happened because of recommendation. This technology invention has proved to be a blessing for e-commerce retail stores as it is a proven tool for adding up and multiplying sales like never [5].

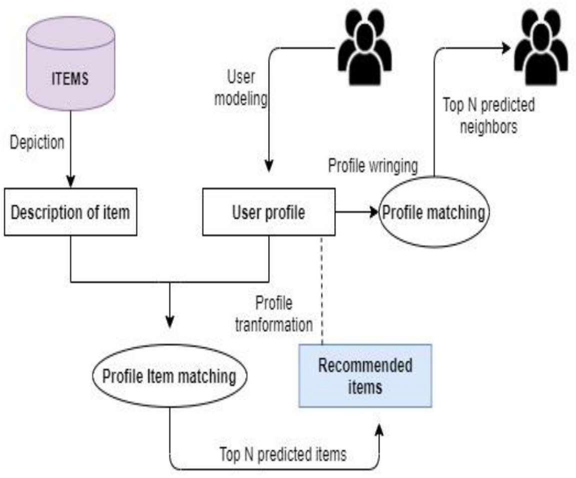

Figure 1: Basic Recommender Process

II. LITREATURE SURVERY

Many good numbers of researches have been performed in this domain of product recommendation. In previous works, collaborative filtering has been used in various different application which is solely based upon user-user and item-item interaction. Some of the areas of recommendation using collaborative filtering are movies, articles and product recommendation [8]–[10]. As, CF methods had some limitations there have examination to make use of web mining techniques that ae based of web usage mining. It is defined as the process of applying data mining techniques to find out different patterns from the web data. These pattern findings include association rules, page clusters and other different pattern discovery methods[11]–[13]. There was a recommendation system that was built based on web usage mining that can be seen in[14]. The technique was to recommend products based upon the web usage data as well as taking in consideration about product purchase data. The idea behind the methods was to recommend product of user's interest. For this, they proposed a list of top-N recommended products for different customer at particular period of time.

There was different experiment that were done with the web usage for the internet shopping mall for validation their proposed recommendation system. During evaluation they found that taking the right level of product and customer increases the quality of recommendations.

There was another product recommendation technique that has combination of previous suggested system based on data mining techniques and group-decision making methodologies. They made use of analytic hierarchy process (AHP) to compute the relative weights of frequency, monetary (RFM) variable in making evaluation of the customers loyalty. Then they made use of clustering techniques to make a cluster of customers on the basis of the weighted RFM value. At last, they recommended product to each customer group using association rule mining. The conclusion was made that recommending many numbers of items helps to enhance the quality of recommendation system for loyal customers, but not for people having less loyalty. Zeng also talked about the making of personalized product recommendation system in[15]. The recommendation system made use of mining methodologies to track the customer's purchasing behavior. The customer choice and product association were automatically taken from customers clock streams. Then there was the algorithm that used to combine the product association and customer preference that use to score individual product. Therefor system used to make a product lists for respective customer to do recommendation. In the experiment part, they were able to show their system is saving customers time for internet shopping and was giving good recommendation.

III. RECOMMENDER SYSTEM AND TYPES

A recommender system is used by every e-commerce website these days like Flipkart, Amazon. Also, companies Netflix and YouTube uses such technology to gain more engagement by the users. There are many different types of recommender system that are used at different platforms which are build according to their architecture. Different types of recommender systems are discussed below:

A. Popularity-based recommender system

This type of recommender system is a system that has been designed in such a way that it recommends the most popular products to a user. It checks about the products which are in trend. For example, if there is a product which is usually purchased by every new user then there is possibility that the user who just signed will also buy the same product. Generally, most popular products can be found based on several filters like user ratings, different locations, etc. Popularity based recommender system does not suffer from cold start problem because they can even recommend the most popular product to every new user based on different filters. There is no need for the user's historical data. There are still few of the problems in these systems which includes not personalized system[16]. They would recommend the same products to every new user who visited the website for the first time. The example of the popularity-based recommender system is Google news where news is filtered by top news that means the most popular news *and* trending videos in YouTube which are most popular videos[17].

B. Classification model

It is type of model that makes prediction whether a user will like a item or product or not considering features of both users as well as products. The outcome of the model can be either 1 or 0 that means it the user likes the product then it comes out to be 1 or else 0. Classification model are considered to be very tough task because there is need to have large amount of data of products as well as different users. Moreover, it this is also achieved then also it becomes difficult to generalize. Classification model suffers from flexibility issues as well[18].

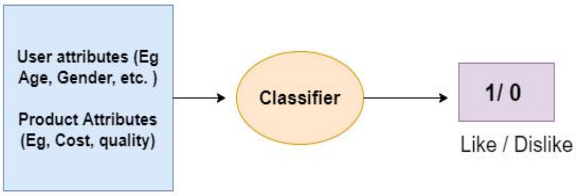

Figure 2: Classification model

Above image shows the representation of a classification model.

C. Content-based recommender system

It is a type of recommender engine which works on the principle of similar content. If a user is looking for a product the system will find similar products on the basis of different metrics and will recommend those products to the user. These systems are considered to a bit of personalized system because it recommends only those products which have similar content that means whatever the users will be recommended the product of their interest. Consider an example of one plus 7 mobile phone that has two more variant that are one plus 7T and one plus 7T pro listed on Flipkart website. If you looking at One plus 7 variants, Flipkart starts recommending you other variants of the mobile phones that are 7T and 7Pro. It is done by computing the similarity between the products on different factors. To compute similarity between one plus variant let's take factors that are camera and ram.

TABLE 1. COMARISION OF THREE MOBILES PHONES

ONE PLUS 7	ONE PLUS 7T	ONE PLUS 7T PRO
6GB ram	6GB ram	8GB ram
32MP camera	32MP camera	32MP camera
64GB storage	64GB storage	64GB storage

To compute the similarity between all the variants of one plus 7 we make the use Euclidean distances. If the distance comes out to be 0 then they are similar or else not. If the Euclidean distance is calculated between one plus 7 and one plus 7T on the basis of ram and camera then it comes out to be that means they are similar phones. But if you will compute the similarity between one plus 7 and one plus 7T on the basis of ram and storage it will not come out be 0 but if the distance is computed on the basis of ram and only storage then it will be 0.

These are the basic metric that are used to check the similarity between the products and hence the most similar one's are recommended to the user. This can be calculated for any feature of a product. In the above example camera and ram are taken. But it is not restricted to camera and ram distance can be calculated for any features of a product. For checking similarity between numeric data Euclidean distances are used, for textual data cosine similarity is used and if the data is categorical then Jaccard similarity is used for computing the similarity. Different metrics are discussed below.

i. Euclidian distance: Distance between two points.

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

ii. Cosine distance: Cosine of the angle between the 2 vectors of the item vectors of A and B is calculated for imputing similarity. If the vectors are closer, then small will be the angle and large will be the cosine.

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\|\vec{b}\|}$$

Perfect matching of the product can never be recommended. Consider a example where a person is

Figure 4: Item based collaborative filtering

looking for laptop with 1 TB HDD, 4GB RAM, i5 processor and budget is 40K. There never be a scenario where he will get recommended with exactly the same specifications. The recommended laptop will have similarities like it might of 40K but the specification is not the same like the person is looking for. Recommendation system cannot recommend exactly the same products.

D. Collaborative system

It is a very intelligent recommender system. It works on the principle of similar users or similar items. This type of recommender system is widely used by YouTube, Netflix, and Amazon, etc. Performance of this system is more efficient if we have more information about items and users[17]. There are various methods used for finding similar users and similar items. Mainly there are two approaches which are used in collaborative filtering stated below:

i. User based nearest neighbor collaborative filtering:

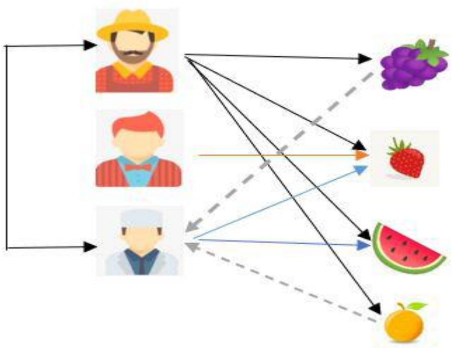

Figure 3: User based collaborative filtering

It finds out the users who have a same sort of taste of purchasing products. Similarity between user is computed based upon the purchase behavior. User A and User C are similar user because they have purchased similar products.

ii. Item based nearest neighbor collaborative filtering:

It recommends items that are like the items the user bought. Similarity between different items is computed

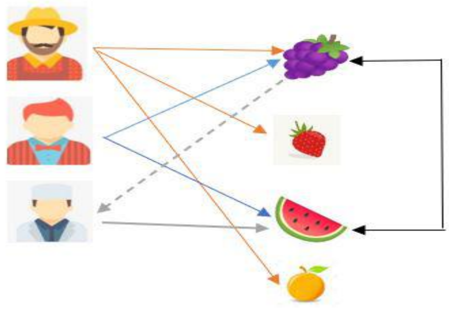

Figure 3: User based collaborative filtering

based on the items and not the users for the prediction. Users x and y both purchased items A and B so they are found to have similar taste.

E. *Singular value decomposition and matrix-factorixation*

It is a matrix factorization technique that reduces the features of the dataset by reducing dimensions from N to K (K<N). For recommendation only the matrix factorization is taken care of which is done on the user-item rating matrix. Vectors are used to represent item q_i and user q_i such that their dot product is out expected rating.

$$\text{Expected rating} = \hat{r}_{ui} = q_i^T p_u$$

q_i and q_i can be calculated in such a way that the square error difference between dot product of user and item and the original ratings in the user-item matrix is least.

$$min(p,q) \sum_{(u,i)\epsilon k} \left(r_{ui} - q_i^T \cdot p_u\right)^2$$

A very important aspect of any machine learning model is to avoid overfitting and for that purpose regularization is used. It removes out the risk of the model to get overfitted. A penalty term that is λ is used which is multiplied by user and item vectors and square sum of the magnitudes. SVD algorithm minimize the error between the actual value and the predicted value.

F. *Hybrid Recommeder systems*

Hybrid recommendation systems are those system that are build using combination of different recommender system. Studies shows that combination of collaborative filtering and content-based filtering could be more efficient in some of the cases. These systems can be built by many different methods like by making content based and collaborative prediction individually and then combining them by adding each potential to each other's approach. Studies shows that hybrid models have good performance if we compare it to pure content based or collaborative based approaches. Hybrid methods also get the better of common problems like cold start and sparsity problem in recommendation systems. Netflix is one of the good examples of hybrid models. Their system does recommendation by comparing the watching and taking in consideration about similar users i.e. collaborative filtering as well as by recommending movies that have similar content with the film rated more by the user.

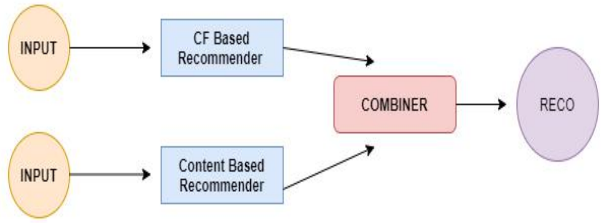

Figure 4: Hybrid Recommender System

IV. Challenges and Problems in Recommeder Systems

A. *Cold Start*

This is the problem that comes when a new user signs in or a new item gets a place into the system. In these sorts of cases, the new items cannot be rated or purchased nor the taste of the new users can be computed[4]. This sort of problem can be resolved by taking the ratings of the user in the beginning or to ask about the user taste and recommending the users items on the basis of demographic information.

B. *Synonymy*

It is the problem that arises when a single item is displayed with different names or entities having similar meaning. [10] In these sorts of cases recommender system is not able to understand whether it's the same item or different item. For example, a CF model will understand "comedy movie" and "comedy film" different. The usage of synonym words reduces the performance of the recommender system. The recommender system does not consider the latent association between items because item content is not considered. This is main reason if their ratings are present for the new product it would not be recommender. To conquer the problem, Singular value decomposition as well as Latent Semantic Indexing can be used[19].

C. *Shilling Attacks*

Consider a case if a spiteful user comes into system and start giving wrong ratings as to increase the popularity of an item or vice versa then this type of process if called Shilling Attacks [11] . These types of attack can disturb the quality as well as performance of the system. This is more seen in CF based recommender systems. There are various attack models like bandwagon, random, average, and reverse bandwagon attack which can detected by different methods like prediction shift and hit ratio. More metrics and methods for shilling attacks can be studied in [20].

D. Privacy

Providing personal information to the recommender engine will definitely give you good recommendations but it might lead do data security. Recommender system that ache from privacy issues, still User are resistant to put their data. So, recommender system should build a trust between the users. In CF based systems, the user's data that includes rating are kept in a centralized repository that can be compromised ensuring in data misuse. For this cryptographic method should be taken in use so that personalized recommendations can be made without using third parties.

E. Grey Sheep

Grey sheep occurs in pure CF systems where opinions of a user do not match with any group and therefore, is unable to get benefit of recommendations[21]. These types of user can be recognized by clustering techniques like K means clustering can be used so as to get enhanced performance and minimized recommendation error[22].

F. Evaluation and Dataset

Evaluation of recommendation system and Dataset
The evaluation metrics tells you about the performance of the model. Selecting evaluation techniques is considered to be problem in the case of recommender systems. Metrics like MAE, precision and F measures are used for evaluation by traditional recommender system[11], [16] . One more challenge with recommendation system is no benchmark dataset that can be used to evaluate the recommender system. These types of dataset inspire research and development of recommender.

EVALUATION IN RECOMMENDER SYSTEMS

The evaluation of recommender algorithm can be done be checking its performance on different datasets[23]. In this paper mean absolute error and root mean square error is computed to evaluate the recommender system. The prediction efficiency of the algorithm can be measured with different metrics. What evaluation metrics is to be used to compute to validation depends upon the goal of measurement[11], [15].

Mean Absolute Error (MAE)

It is the metric that is used to take average of all the value differences between original and predicted ratings [5]. The accuracy is better if the low is the MAE. It mainly has a range between 0 to infinity. Calculation of the MAE is done using the following formula[20]:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |x_i - x|$$

where: x_i is the original rating, x is the predicted rating by the system and n is the amount of ratings.

Root mean squared error (RMSE)

It is calculated by the mean value of all the dissimilarity squared between the original rating and the predicted rating which then move forward to compute the square root of the result[24]. Large errors might influence the RMSE rating, furnishing the RMSE metric most treasure when notable large errors are unwanted. The RMSE value between predicted ratings by the model and the original rating can be given by the formula[16]:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (p_i - o_i)^2}{n}}$$

where: p_i is the original rating, o_i is the predicted rating by the system and n is the amount of ratings.

Offline Experiments

In order to decide good algorithm to predict certain information offline experiment can be conducted on a dataset. From the time the data was collected that is still accurate, the offline experiments take assumption of the user behavior. Or else this sort of process is more or less illogical. These experiments include calculation of the neighborhood clusters, prediction of similarity between the items, and among other things[23][25].

Cross-validation

Cross-validation is a method to evaluate the model and check its performance on the unseen data. The model is created and evaluated multiple times. How many times the evaluation will be done will be dependent on the user. User needs to decide a value that is called as "k" which is an integer value. The sequence of steps used is repeated as many times as the value of 'k'. Initially, for doing cross validation you need to use random functions to divide the original data into different folds. Cross-validation splits dataset into x equal big portions, on the partition is used for testing and x-1 for training partitions. Training of the model is done through the algorithm and when it completes it is tested over the test partition that was built by dividing. This

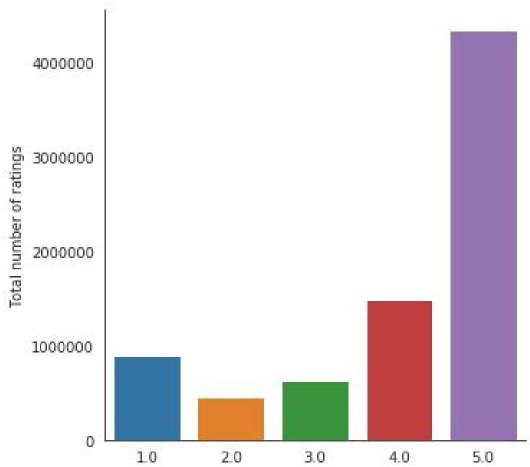

Figure 5: X-axis represent ratings and Y-axis represent users.

process keeps on going on until every partition has been the test partition[11].

VI. DATASET AND PROPOSED MODEL

The data was taken from Amazon's open source repository which holds several different data about the Amazon product. The data was about electronics products. The first three columns in the data are userId, productId, and ratings and the fourth column timestamp respectively. The timestamp was useless so it was discarded. There was a total of 7824481 rows and 3 columns. Missing value checking was done and was found that there were no missing values. Plots was plotted for the rating respect to users and it was found ratings ranged from 1-5. The subset of the data was taken to make it less sparse and dense and only those users who have rated more than 50 products was taken in consideration. There were about 12587 users who have rated more than 50 products. The final data was divided into 70:30 training and test respectively. Test data shape:(37762, Train data shape:(88109, 4).

A. *Recommender System - I*

Popularity-based recommender system where we first calculated how many users have rated a product using python package for data manipulation which is pandas and NumPy. Stored the number of ratings corresponding to a respective product in a column and gave it index as 'score'. The score described count of the rating users has rated to the particular product. Then it was grouped in descending order with respect to products and printed the most popular products which had highest ratings which were then recommended to every user. Prediction of the products was done based on popularity.

```
Here is the recommendation for the userId: 121

       user_id     prod_id  score  Rank
30847      121  B0088CJT4U    133   1.0
30287      121  B007WTAJTO    124   2.0
19647      121  B003ES5ZUU    122   3.0
8752       121  B000N99BBC    114   4.0
30555      121  B00829THK0     97   5.0

Here is the recommendation for the userId: 200

       user_id     prod_id  score  Rank
30847      200  B0088CJT4U    133   1.0
30287      200  B007WTAJTO    124   2.0
19647      200  B003ES5ZUU    122   3.0
8752       200  B000N99BBC    114   4.0
30555      200  B00829THK0     97   5.0
```

Figure 6: Represents the recommendation made by recommender (I) for user 121 & user 200

B. *Recommender System – II*

Using matrix-factorization build another model using a user-based nearest-neighbor collaborative filtering approach. Initially, using pivot functional pivot table was made based on the user which meant wherever the user has rated a product the respective ratings is present and those products where the user has not given the rating, it was filled with 0. So, matrix of user and product was obtained. Replaced the userID by creating a different column which was user index using np. arrange () function in NumPy giving step size of 1 and till the last row in the training data. So, finally, in the data frame, there were user from 0 to last row index of the training set, respective product, and respective rating. Used the SVD algorithm to train the training set where it was seen what each user has rated to each product. After, training our algorithm predicted each user rating with each product and compared it with the original rating. Created a function and recommend the product to users who had given similar ratings to the product. In short, the system was made in such a way that based on similar ratings the system reverts the similar users and recommendation of the product is done. Calculated the top 5 recommendations using the same function i.e. (k=5). The system was found to bit personalized system.

VII. EVALUATION DONE ON RECOMMENDER II

The models were evaluated based on the different metric like RMSE, mean square error, mean absolute error. The below figure shows the *Avg_actual_ratings* and *Avg_predicted_ratings*.

i. Root mean squared error (RMSE): RMSE measures the average magnitude of the error. It is calculated by taking the squared differences between original values and predicted values.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(p_i - o_i)^2}{n}}$$

ii. Mean Absolute Error (MAE): Without worrying about the direction MSE signifies average magnitude of the errors in a set of prediction. Also, its average over the test sample of the absolute differences between the original and predicted value where all individual differences have equal weight.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|x_i - x|$$

RMSE value given by the model was **0.00275** and MAE was **0.0019**. Both metrics signify the performance of the model. Less the score better the model is.

To visualize the same, the scatter plot was plotted to better analyze our model and check the predictions. The Scatter plot plotted the regression line between both the predicted and original data points. More, closer these data points are towards the line more the model is performing good and vice versa. There was a bit of variance which we is clearly seen in the plot.

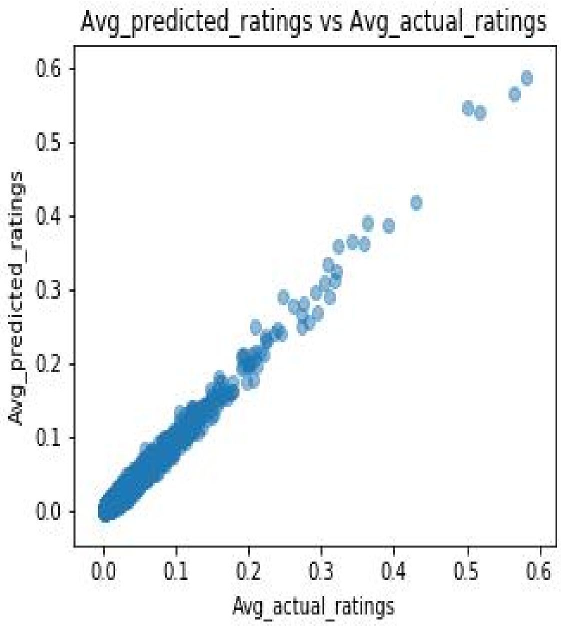

Figure 7: Scatter plot between predicted rating and actual rating.

VIII. CONCLUSIONS AND FUTURE SCOPE

Model-based Collaborative Filtering is a personalized recommender system, the recommendations are based on the past behavior of the user and it is not dependent on any additional information.

The popularity-based recommender system is non-personalized, and the recommendations are based on frequency counts, which may be not suitable to the user. For enhancing more, the evaluation of the model "timestamp" column that was discarded can be made in use so that it can be checked whether the user have purchased the product or not.

In present day, deep learning technology has remarkably doing great in the domains like speech recognition, computer vision and natural language processing. Also, recommendation can be benefited from deep neural networks. One of the main advantages of using deep neural network is their ability to lean or representation learning. So, for recommender system to perceive latent factors from complex data source, deep neural networks can a play a vital role. Deep neural network has the ability to model the linear interaction in the data with non-linear activation functions that are Relu, Sigmoid and Tanh. It can easily learn the important representation from the input data and factors that are explanatory. It has the ability of being powerful when it comes to sequential modeling tasks and also possesses high flexibility.

Deep neural networks can be used to improve the recommendations systems. YouTube used deep neural network architecture for doing recommendation.

IX. REFERENCES

[1] X. Kong, M. Mao, W. Wang, J. Liu, and B. Xu, "VOPRec: Vector Representation Learning of Papers with Text Information and Structural Identity for Recommendation," *IEEE Trans. Emerg. Top. Comput.*, p. 1, 2018, doi: 10.1109/TETC.2018.2830698.

[2] S. Yu *et al.*, "PAVE: Personalized Academic Venue recommendation Exploiting co-publication networks," *J. Netw. Comput. Appl.*, vol. 104, pp. 38–47, 2018, doi: https://doi.org/10.1016/j.jnca.2017.12.004.

[3] R. Kumar and A. Anand, "Internet Banking System & Security Analysis," *Int. J. Eng. Comput. Sci.*, vol. 6, no. 6, pp. 2319–7242, 2017, doi: 10.18535/ijecs/v6i4.43.

[4] A. J. C. Trappey, C. V Trappey, C.-Y. Wu, C. Y. Fan, and Y.-L. Lin, "Intelligent patent recommendation system for innovative design collaboration," *J. Netw. Comput. Appl.*, vol. 36, no. 6, pp. 1441–1450, 2013, doi: https://doi.org/10.1016/j.jnca.2013.02.035.

[5] J. Son and S. B. Kim, "Academic paper recommender system using multilevel simultaneous citation networks," *Decis. Support Syst.*, vol. 105, pp. 24–33, 2018, doi: https://doi.org/10.1016/j.dss.2017.10.011.

[6] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261–273, 2015, doi: https://doi.org/10.1016/j.eij.2015.06.005.

[7] J. Beel, "Towards Effective Research-Paper Recommender Systems and User Modeling based on Mind Maps," *CoRR*, vol. abs/1703.0, 2017.

[8] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An Open Architecture for Collaborative Filtering of Netnews," in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, 1994, pp. 175–186, doi: 10.1145/192844.192905.

[9] U. SHARDANAND, "Social Information Filtering: Algorithms for Automating."

[10] Y. H. Cho, J. K. Kim, and S. H. Kim, "A personalized recommender system based on web usage mining and decision tree induction," *Expert Syst. Appl.*, vol. 23, no. 3, pp. 329–342, 2002, doi: https://doi.org/10.1016/S0957-4174(02)00052-0.

[11] A. M. Rashid, S. K. Lam, A. LaPitz, G. Karypis, and J. Riedl, "Towards a Scalable kNN CF Algorithm: Exploring Effective Applications of Clustering," in *Advances in Web Mining and Web Usage Analysis*, 2007, pp. 147–166.

[12] B. Mobasher, H. Dai, T. Luo, Y. Sun, and J. Zhu, "Integrating Web Usage and Content Mining for More Effective Personalization," in *Electronic Commerce and Web Technologies*, 2000, pp. 165–176.

[13] B. Mobasher, R. Burke, R. Bhaumik, and J. J. Sandvig, "Attacks and Remedies in Collaborative Recommendation," *IEEE Intell. Syst.*, vol. 22, no. 3, pp. 56–63, May 2007, doi: 10.1109/MIS.2007.45.

[14] A. Gunawardana and G. Shani, "Evaluating Recommender Systems," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, and B. Shapira, Eds. Boston, MA: Springer US, 2015, pp. 265–308.
[15] L. Lü, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, and T. Zhou, "Recommender systems," *Phys. Rep.*, vol. 519, no. 1, pp. 1–49, 2012, doi: https://doi.org/10.1016/j.physrep.2012.02.006.
[16] Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan, "Large-Scale Parallel Collaborative Filtering for the Netflix Prize," in *Algorithmic Aspects in Information and Management*, 2008, pp. 337–348.
[17] K. Singhal and A. Anand, "Monetisation Of Youtube Content Using Data Mining Techniques," *Int. J. Sci. Technol. Res.*, vol. 9, p. 2, 2020.
[18] D. Anand and R. Scholar, "A Survey And Learning Techniques On Access Controls In Cloud Computing," *Int. J. Res. Advent Technol. Spec. Issue*.
[19] D. Kotkov, S. Wang, and J. Veijalainen, "A survey of serendipity in recommender systems," *Knowledge-Based Syst.*, vol. 111, pp. 180–192, 2016, doi: https://doi.org/10.1016/j.knosys.2016.08.014.
[20] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating Collaborative Filtering Recommender Systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, Jan. 2004, doi: 10.1145/963770.963772.
[21] I. Gunes, C. Kaleli, A. Bilge, and H. Polat, "Shilling Attacks against Recommender Systems: A Comprehensive Survey," *Artif. Intell. Rev.*, vol. 42, no. 4, pp. 767–799, Dec. 2014, doi: 10.1007/s10462-012-9364-9.
[22] G. Schr�der and et al., "Setting Goals and Choosing Metrics for Recommender System Evaluations." .
[23] X. Shi, X. Luo, M. Shang, and L. Gu, "Long-term performance of collaborative filtering based recommenders in temporally evolving systems," *Neurocomputing*, vol. 267, pp. 635–643, 2017, doi: https://doi.org/10.1016/j.neucom.2017.06.026.
[24] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Incremental Singular Value Decomposition Algorithms for Highly Scalable Recommender Systems," in *Fifth International Conference on Computer and Information Science*, 2002, pp. 27–28.
[25] N. Agarwal, E. Haque, H. Liu, and L. Parsons, "Research Paper Recommender Systems: A Subspace Clustering Approach," in *Advances in Web-Age Information Management*, 2005, pp. 475–491.