Product Recommender System Based on the Ratings

3rd May 2022

Abstract

The business of online shopping thrives on recommending the right product to the right user. Recommending products to an user based on the ratings he has given to similar products might entice the user to buy more Products. So we are trying to create a Recommendation System which with an acceptable accuracy predict the Beauty products a user might like and probably buy depending upon the ratings he/she has given to other Beauty Products. This will help E-Commerce companies(in our case Amazon) make better product placements. Better product placements inturn ensures their product marketing actually reach the corrected most probable potential buyers, inturn generating much more revenue and also better returns on money spent of marketing in per dollar terms.

Introduction

In today's world we are overloaded with data which provides us the useful information. But it's not possible for the user to get the information in which they are interested in from the available data. In order to help the user to find out information about the product, recommedation systems are developed.

Recommeder system helps in creating a relation between the user and items and utilizes the similarity between user/item to make recommendations. Many popular Ecommerce sites widely use Recommended Systems to recommend news, music, research articles, books, and product items. More the development of e-commerce websites more is the need emerged for providing recommendations compiled from filtering the whole range of available options.With a wide variety of options that are provided to the users from multiple websites, users find it very difficult to make the most appropriate choices.

Methods

Our goal of the project is to recommend products to users based on the ratings given to the products. Before diving into data modeling methods, we have to process and clean the data to prepare the dataset for modeling. We will go through the data cleaning, data analysis and data processing in this section.

1.Data Collection: We have used a dataset from kaggle. The dataset contains over 2 million customer reviews and ratings and 4 features. The entries are from May 1996 - July 2014.

2.Data Cleaning: We have checked for any null values in the data. We didn't find any null values or duplicates in this dataset. So the dataset is mostly clean. We also checked the datatypes of the features of the categorical variables and found 2 of them of type Object. We used LableEncoder in SkLearn library to convert them into numeric types. The ratings given by different users is different, we have normalized the ratings to make our model more robust.

3.Data Analysis and Visualization: As part of data analysis we analyzed the relationship between user and the products as this dataset is mainly based on the ratings that are provided by the user for every product. We analyzed scenarios like the number of rated products per user and the number of ratings for per product. From this we understood which user has given the max/min number of ratings and which product has the max/min number of ratings.

We have plotted the following plots to infer few observations

- Bar plot for unique users,products and total records

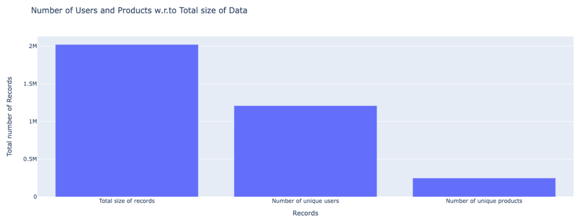


Figure 1: bar__plot__user

We can infer from the above graph how the data is distributed in the dataset. The total number of records is 2M and total number of users is 1.2M which makes us conclude that on an average every user rates at least twice. The total number of unique products is 250K. We can conclude that on an average, each product is rated at least 8-10 times. Here we can see that no of total records are significantly more than the not of unique products and unique users. That means there are a no of ratings for different combinations of user ids and product id, which makes this data set a perfect fit for user to user or item to item ccollaborative filtering techniques.

- Bar plot for ratings given by users

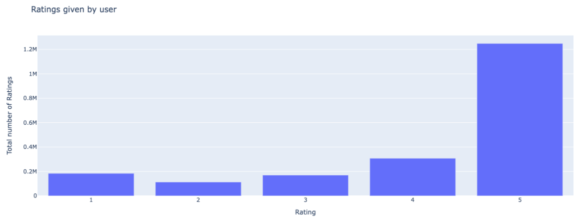


Figure 2: bar__plot__user__ratings

The above graph shows the distribution of various ratings. The rating 5 was given by 1.2M users which is highest and rating 2 is given by 113k users which is lowest. Here we can see that the no of ratings given by all the users on all the products are disproportionately on higher side. So to negate this bias we have normalized the ratings of the products by substracted average rating of each product from its orginal rating.

- Bar plot to show the most popular products

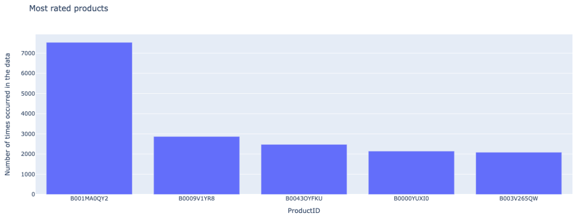


Figure 3: bar__plot__high__rated__products

The above graph shows the most popular products and their frequency. The most popular product is B001MA0QY2 which is rated by 7533 Users. The number of ratings for the first popular product and second popular product is very high. Here we can see that some Products have more no of reviews and some dont have any. So to take care of this disproportionality we have kept the average no of reviews as threshold and eliminated the rest of them.

- Bar plot to show the ratings range

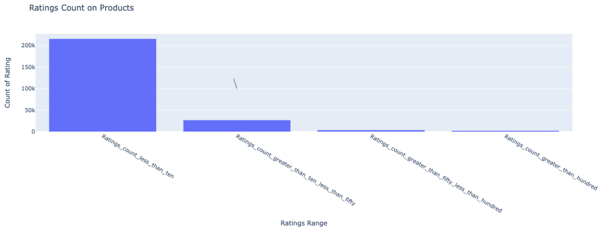


Figure 4: ratings__count__products

Most of the products have received less than 10 ratings. Out of 200k products, 2000 products have received more than 100 ratings.

- Violin plot to show the product ratings

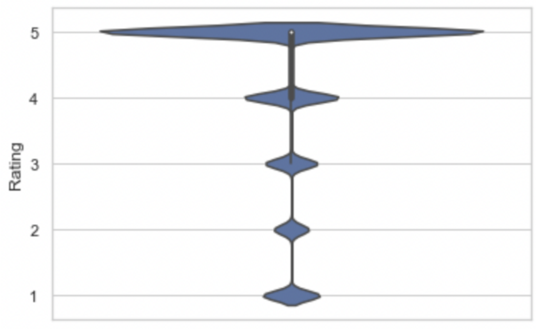


Figure 5: violin__plot__ratings

From the above Violin Plot we can conclude that the number of products with 5 star ratings is high and this number is greater than the sum of all other ratings given to other products. The number of products with ratings 2 stars is the least. The maximum rating given by any user is 5 and lowest rating given by any user is 1.

- Distplot to show the probability of total ratings

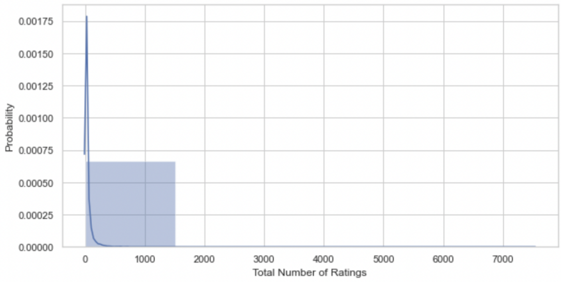


Figure 6: distplot__total__ratings

From the above graph we can see that the majority of the products have less than 100 ratings and the number of products having more than 100 ratings is very low.

- Jointplot for mean ratings and total ratings

Here, as you can see every Data Point represents a distinct product, with y-coordinate representing the total number of users which have rated that product and x-coordinate representing the mean of all the ratings of the corresponding users. Also you can see that there is a huge Density in the region corresponding to 0-1000 no of users and between mean rating 3.5-5

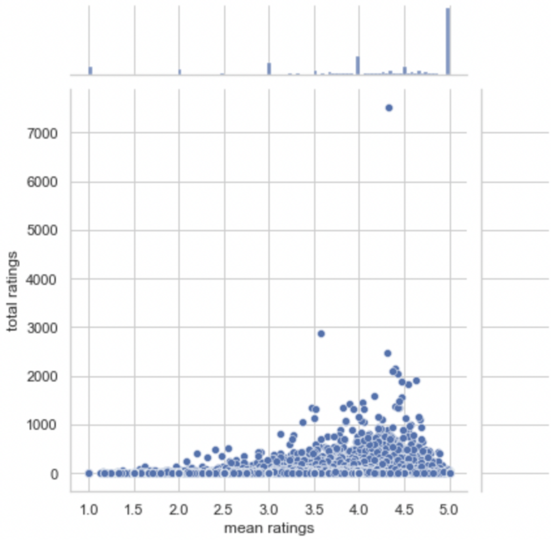


Figure 7: joint__plot__ratings

Models

Recommendations are mainly of two types: personalized and non-personalized. In personalized recommendations, different users receive different suggestions. In non-personalized recommendations, all the users get same suggestions.

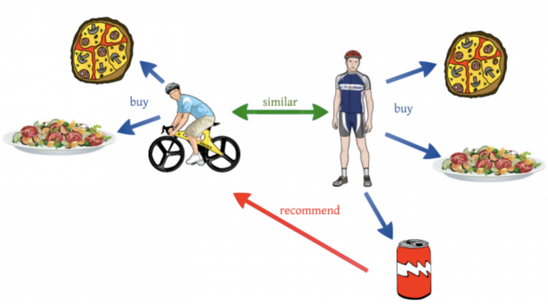
Recommended Systems are mainly classified into the 6 types:

1.Popularity based systems :- These systems provide items that are viewed and purchased by most people and which are highly rated. This system works on the basis of popularity or trend. These systems verify about the products or movies which are most popular among the users and directly recommend those.So if a product is often purchased by most people then the system will get to know that the product is most popular and recommends the same for new users. The chances that the new user will purchase that product will be more.In this way the products will be easily sold.

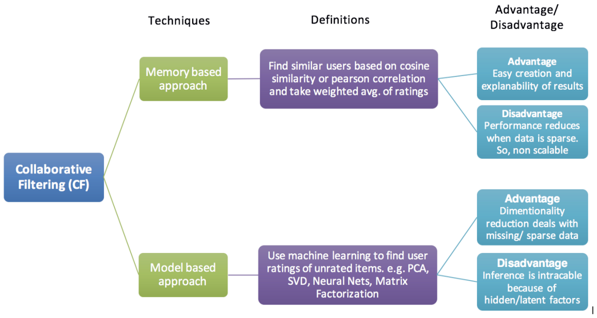
So as part of this method we are aggregating the count of users for every product. Once we have the count of the number of users product wise it becomes easy for us to identify the most popular products. The products are sorted based on the counts of users and a ranking is provided to each and every product. Now when a UserId is provided as input these popular products will be recommended to the users. If a new user signs in they will find these products as recommendations.But the main disadvantage with this method is this system would recommend the same sort of products which are solely based upon popularity to every other user which is not ideal. So it is not a personalized model.

2.Content based recommedations :- This model is mainly based on the information of the contents of the item rather than on the user opinions. Since our data set doesnt have any attributes which describe the data except for ratings. We decided that content based recommendation is not the best fit for our data set.

3.Collaberative Filtering :- Collaborative filtering is a technique to recommend items to a user based on the items liked by similar users.It is based on assumption that people like things similar to other things they like, and things that are liked by other people with similar taste.



The CF techniques are broadly divided into 2-types:


Techniques
Definitions
Advantage/ Disadvantage
Collaborative Filtering (CF)
Memory based approach
Find similar users based on cosine similarity or pearson correlation and take weighted avg. of ratings
Advantage Easy creation and explanability of results
Disadvantage Performance reduces when data is sparse. So, non scalable
Model based approach
Use machine learning to find user ratings of unrated items. e.g. PCA, SVD, Neural Nets, Matrix Factorization
Advantage Dimentionality reduction deals with missing/ sparse data
Disadvantage Inference is intracable because of hidden/latent factors

1. Memory based Collaborative Filtering: There are two types of memory-based collaborative filtering approaches: user-item filtering and item-item filtering. A user-item filtering takes a particular user, find users that are similar to that user based on similarity of ratings, and recommend items that those similar users liked. In contrast, item-item filtering will take an item, find users who liked that item, and find other items that those users or similar users also liked. It takes items and outputs other items as recommendations. The key difference of memory-based approach from the model-based techniques is that we are not learning any parameter using gradient descent. The closest user or items are calculated only by using Cosine similarity or Pearson correlation coefficients, which are only based on arithmetic operations.

2. Model Based Collaborative Filtering: In this approach, collaborative filtering models are developed using machine learning algorithms to predict user's rating of unrated items and recommends the items which are pridicted with high ratings. They can be further classified as :

SVD is a unsupervised machine learning algorithm which is popularly used dimension reduction, data-driven model which is used to predict the ratings and recommend the products based on the predictions made by it.

4.Hybrid Approaches :- This method combines collaborative filtering, content-based filtering, and other approaches.Here we have implemented the different methods which make a hybrid recommendation system. But not the one which take 2 different methods

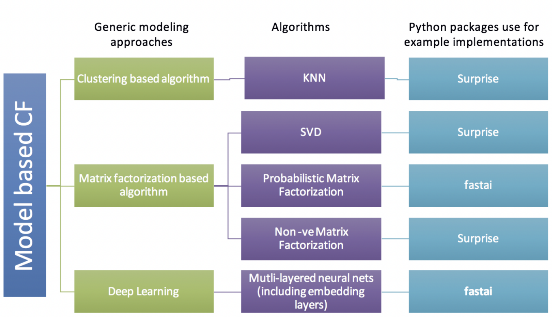

Generic modeling approaches
Algorithms
Python packages use for example implementations
Model based CF
Clustering based algorithm
KNN
Surprise
SVD
Surprise
Matrix factorization based algorithm
Probabilistic Matrix Factorization
fastai
Non -ve Matrix Factorization
Surprise
Deep Learning
Mutli-layered neural nets (including embedding layers)
fastai

Figure 8: Model_Based_CF

into consideration.

5.Association rule mining :- Association rules capture the relationships between items based on their patterns of co-occurrence across transactions.

From the above classified methods of Recommended Systems we decided to implement Popularity Based Recommended systems, User Based Collaborative Filtering and Item based Collaborative Filtering methods. We are not using Content based recommender systems because it recommends products or items based on their description or features. We don't have any features or descriptions provided in our dataset that can be used for content based.

Comparisons

The most important part in implementation of any machine learning model is to compare the model accuracy with the accuracy of other models. The Root Mean Square Error, Mean Square Error are some of the metrics which can be used to compare the accuracy of a model. Mean Square Error: It is the square of the difference between the predicted value and the actual value. Root Mean Square Error: It is the square root of Mean Square Error. This is the most easily interpreted statistics as it has the same units as the quantity plotted on the vertical axis.

Now we use the above metrics RMSE, and MSE to compare the different models which we implemented as part of this project. And we used only the unsupervised learning methods in to implement this system.

1.Popularity based systems:- This method is all about predicting the top rated products to any user.It is user cold-start resistant. The system can suggest products without any information about the user. As every user gets same recommendations, it is not personalized recommendation. This recommandation system is useful to recommend the new users as initially the user has low correlation with other users. This approach has very low accuracy and high errors but needs less compulational power.

2.Content based recommedations:- Content-based filtering methods are based on product descriptions and user preferences.As we can see from the data visualization above there are not enough attributes to the data, infact there no attributes to the product except for the ratings. So Content based recommendation cannot be a vaiable option for our recommendation system for this particular data set.

3.Collaborative Filtering:- As the content based filtering is not possible, the only way to make the predictions is by using Collaborative filtering. The Collaborative filtering is all about finding the similarity between items and users. We have calculated similarity using cosine function, jaccard function and Pearson function.

4.Model Based Collaborative Filtering: The User-Item matrix which is generated in Collaborative Filtering have very high number of dimensions.Calculating the similarity between users/items is highly time consuming. We have used existing machine learning models like KNN, SVD and NMF to reduce the dimensions and calculate the similarity in less time.

We have imlemented all the above methods and plotted a line graph to compare the Root Mean Square Error of each model.

On Comparing all the methods, the least RMSE value occurs in Model based Collaborative filtering model using SVD. # Example Analysis

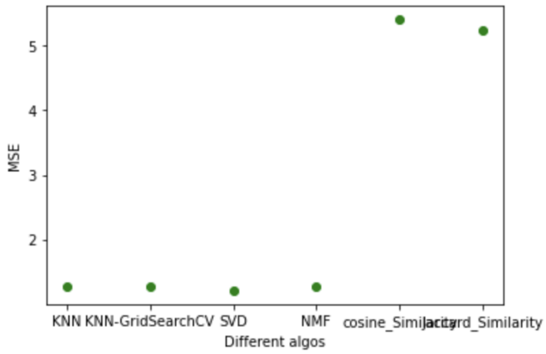


Figure 9: Comparison of different methods

Conclusions

Filtering the important data and predicting the right recommendations to the right user is beneneficial to both the user and the Seller. We have implemented couple of methods to predict the right recommendations. After analysing all the model, we came to conculsion that all the models are giving more Mean Square Error which implies the accuracy of the model is less. In future, we plan to build the recommendation system using Matrix Factorization which involves building couple of neural networks to filter the data with more accuracy.

References

1.N. Mustafa, A. O. Ibrahim, A. Ahmed and A. Abdullah, “Collaborative filtering: Techniques and applications,” 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), 2017, pp. 1-6, doi: 10.1109/ICCCCEE.2017.7867668.

2.J. Joy and R. V. G, “Comparison of Generic Similarity Measures in E-learning Content Recommender System in Cold-Start Condition,” 2020 IEEE Bombay Section Signature Conference (IBSSC), 2020, pp. 175-179, doi: 10.1109/IBSSC51096.2020.9332162.

3.A. Fanca, A. Puscasiu, D. -I. Gota and H. Valean, “Recommendation Systems with Machine Learning,” 2020 21th International Carpathian Control Conference (ICCC), 2020, pp. 1-6, doi: 10.1109/ICCC49264.2020.9257290.