|  |
| --- |
|  |

### Project report

Sabir AKHTAR

Shivam Sarin

Siddhant Chauhan

A picture containing object

Description generated with very high confidence

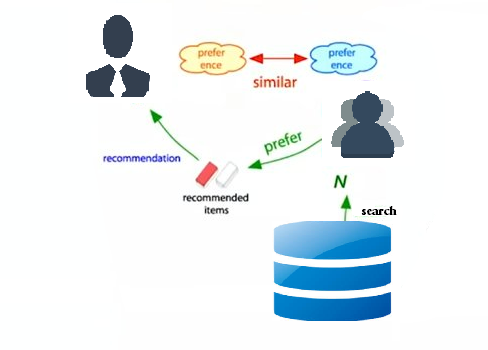
## Recommendation Tools

Preprocessing of Data:

* The pictureURL and URL columns were removed from the Artists data. We keep only the name and ID of the users.
* The columns userID, artistID, tagID were kept from the table user\_taggedartists data and the date column was removed.
* We ensured that in both the databases, the artists and users should be the same.
* We do a boxplot and observe that the weights in userArtists has many outliers.
* We need to make the data discrete. So, we decide to take deciles in percentile measure of 10 – 1 being the least listened and 1 being the most listened.
* After that, we change the column names to make it more intuitive. Each decile has now equal number of songs listened. So, there is no class imbalance.
* Next, we try to find that for each artist, the number of times he has been tagged.

For this project, we use **User based Collaborative Filtering** and **Item Based Collaborative Filtering**.

**User based Collaborative Filtering**

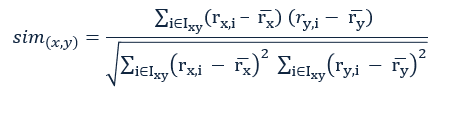


The basic idea behind collaborative filtering is very simple.

* We suppose that we have a user X to whom we want to make recommendations.
* So, at first, we try to find a group of other users whose likes and dislikes are similar to user X. For e.g. While recommending movies, this group of users you know, like the same movies that X likes *and* dislike the same movies that X dislikes. We call this set of users the neighborhood of user X.
* Next, we find other movies that are liked by a lot of users in the set N and recommend those items to the user X.
* So that's the basic idea behind collaborative filtering the key trick is to find the set of users that are similar to user X the neighborhood of user X and to do that we need to define a notion of similarity between users

Instead of using the cosine measure here, we decided to take the Pearson correlation approach:

Mathematically, Pearson's connection coefficient is the covariance of the two variables isolated by the product of their standard deviations.



Where x, y: users

rx,y: rating of user for item

Ixy: set of items, rated both by x and y

The possible similarity values lie between 1and 1

Disadvantages of User Based Collaborative filtering:

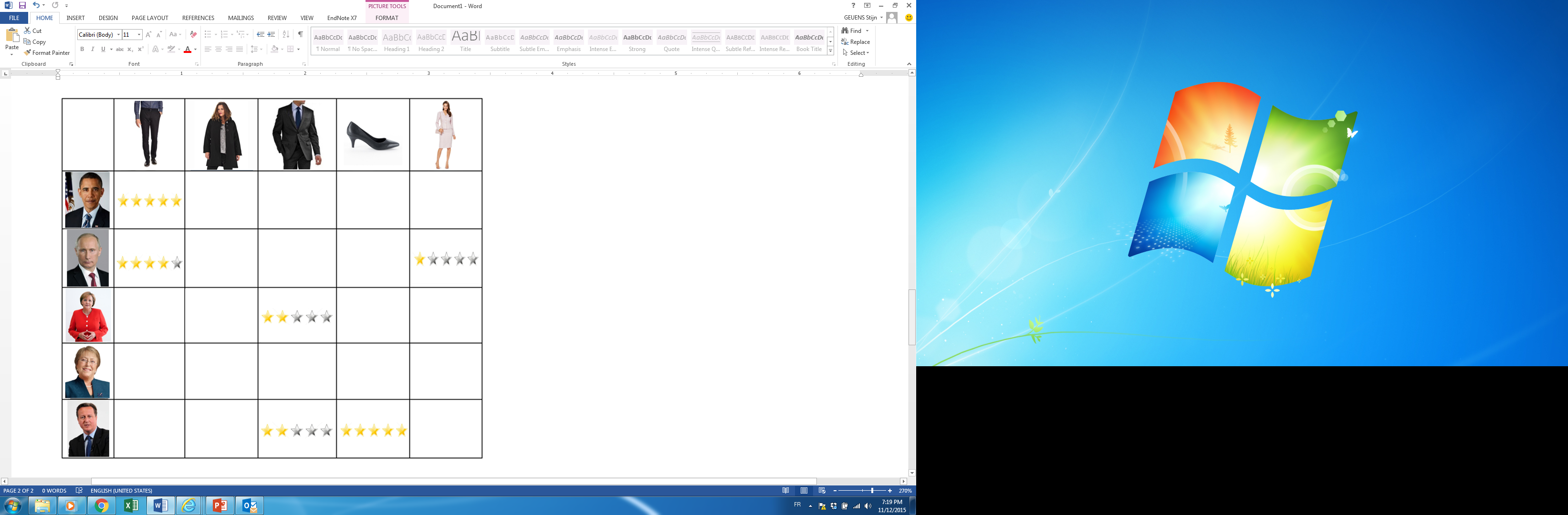
With a user-based way to deal with collaborative filtering for prediction, the framework can compute the likeness between sets of users by utilizing the cosine method, a method much like the item-based CF. Typically such computations:

* take more time to be completed
* should be registered more regularly, than those utilized in the item-based methodology.

That is since you'd have significantly a greater number of users than items (in an ideal case). The items are expected to change less frequently than users. In such a case, you can utilize a lot a bigger number of traits. A user-based framework can likewise utilize calculations to assemble all users who have demonstrated that they have similar tastes. The framework constructs neighborhoods of clients who have comparative profiles, buy examples, or rating designs. If an individual in an area purchases and likes a thing, the recommender framework can prescribe that thing to every other person in the area. Likewise, with item-based, the client put together methodology requires adequate information with respect to every client to be successful. Before suggesting, it is necessary to make a user profile — so it additionally necessitates that the user make a record and be signed in (or store session data in the program by means of treats) while seeing a site. At first the framework can ask the user to make a profile, detail the profile by making inquiries, and after that streamline its proposals after the user’s buying information has been collected.

**Item based Collaborative Filtering**

Instead of looking for other people like the user and recommending stuff they liked, here we look at the things the user liked and recommend stuff that's similar to those things. We call this Item based Collaborative Filtering.



Similarity between items are obtained by looking at who are the users, who have rated how, how the users have rated for the item there are two items which most which users have rated in a similar way then those items are considered to be similar, this is an alternative to content-based similarity of items. We can use this matrix, we can compare the column vectors to find out the pair wise item similarity and we can use it for the recommendation in this way for a user we look at those items the user has recommended user has liked and find out items which are similar to these items There are a few reasons why using similarities between items could be better than similarities between people.

1. The items tend to be of a more permanent nature than people. For e.g.: A rock song will always be a rock song, but an individual's tastes may change very quickly over the span of their lives. So, focusing on the similarities between unchanging objects can produce better results than looking at similarities between people who may have liked something last week and something totally different this week.
2. Another very important advantage to building item similarities is that we deal with fewer number of items to deal than the number of people. The product catalog is always smaller compared to the number of customers.

**Clustering Based Recommendation System**

Cluster-based recommendation is best idea of as a variation on user-based recommendation. Here, the items are prescribed to clusters/groups of likely clients. This involves a step, in which all users are initially divided into clusters. Each cluster is provided with a recommendation, with the motive that the recommendations are of maximum interest to the biggest number of users. The advantage of this methodology is that everything is pre-processed and hence, the run time is quick.

Information regarding ratings and interactions can be portrayed as a matrix or set of matrices, keeping the dimensions as users and items. Considering that the two matrices are almost same, we subtract the second from the first by changing existing ratings with 1 and missing ratings by 0. The resultant matrix is a truth table where:

* 1 represents whether there is interaction between user and product
* 0 represents no interaction between user and product

We use K-Nearest neighbor to calculate the set of individuals for recommendations dependent on the rating or item.

**Content-Based Recommender Systems**

Content-Based Recommender Systems are conceived from utilizing the content of everything for recommendations and attempting to take care of the issues faced by Collaborative Filtering. The three key parts are:

* A Content Analyzer, that give us an distribution of the items, utilizing a type of representation
* A Profile Learner: This involves a creation of a profile which depicts the taste and preference of every user.
* A Filtering Component: It serves as the source for each of the inputs and produces the recommendation list for each user.
* Once we have the content to be considered, we change it into a Vector Space Model.

Mostly, we do this with by using the Bag of Words model, by neglecting the word order. In this model, each document resembles a bag containing a few words. Subsequently, this approach permits word modeling, where each bag contains a couple of words from the dictionary.

A particular usage of a Bag of Words is the TF-IDF portrayal, where TF is for Term Frequency and IDF is Inverse Document Frequency.

Here are a few advantages and disadvantages from this new strategy:

Advantages:

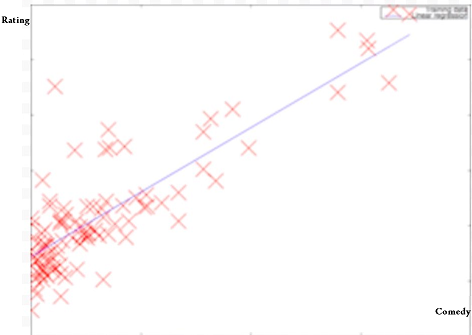
* If the items have adequate depictions, we stay away from the "new item issue".
* Content depictions change, and they open up the choices to utilize diverse methodologies like: text processing, the utilization of semantic data, inductions, etc.
* A simple framework can be constructed. We can utilize the same content to define the recommendations.

Disadvantages:

The method does not provide precision as good as Collaborative Filtering.

**Linear Regression**

A linear regression in general is using one variable like Hours of sleep explain another variable like Exam grades. So here what we are trying to do is explain a user’s rating by the comedy movie the user has watched.



So, we have comedy and ratings as dimensions. And since I really like comedy, I’m going to give that a higher rating, compared to movie with little comedies. My rating should be positively correlated with the level of comedy of that movie. The more comedic that movie is, the higher I will rate that movie.

IN THIS SECTION:

* List your contents here
* List your contents here
* List your contents here
* List your contents here

# 

# Department

# Name

Report Notebook

Tab 1

IN THIS SECTION:

* List your contents here
* List your contents here
* List your contents here
* List your contents here

# 

# Department

# Name

Report Notebook

Tab 2

# 

# Department

# Name

IN THIS SECTION:

* List your contents here
* List your contents here
* List your contents here
* List your contents here

Report Notebook

Tab 3

IN THIS SECTION:

* List your contents here
* List your contents here
* List your contents here
* List your contents here

# 

# Department

# Name

Report Notebook

Tab 4

IN THIS SECTION:

* List your contents here
* List your contents here
* List your contents here
* List your contents here

# 

# Department

# Name

Report Notebook

Tab 5