**ARTIFICIAL INTELLIGENCE**

**Place Recommendation for Tourists based on Tourist Attraction**

**By:**

**Yugali Bafna**

**Student ID : 888261666**

**CPSC 481**

**Fall, 2018**

**Professor: Dr. Wenlin Han**

**Department of Computer Science**

**California State University, Fullerton**

**Table of Contents**

Abstract…………………………………………………………………………………………….3

1. Introduction……………………………………………………………………………….4
   1. Project Goal and Benefits………………………………………………………..4
   2. System Installation……………………………………………………………….4
2. Contribution to the project………………………………………………………………..5

2.1 User-user collaborative Filtering…………………………………………………5

2.2 Item-item collaborative Filtering………………………………………………….6

2.3 Non-Personalized Recommender System…………………………………………7

2.4 Model Evaluation and Model Comparison…………………………………………8

2.5 Accuracy of Algorithms……………………………………………………………9

2.6 Results………………………………………………………………………………10

3. Implications……………………………………………………………………………………14

4. Conclusion……………………………………………………………………………………..15

References……………………………………………………………………………………………16

Abstract

Recommendation systems are mainly designed to suggest items to users based on user interests. Place recommendation for tourists help user’s find most rated destinations by different users. Three different algorithms were used for finding the most accurate method for recommending place to the tourists. User-user collaborative filtering algorithm has highest accuracy of 95% as it calculates similar interest user’s whereas item-item collaborative filtering recommends places based on the place of interest of each user by giving 92% accuracy. Non-Personalized algorithm calculates the average of all the cities in the list by giving 90% accuracy.

1. **Introduction**

The biggest problem faced by the internet user is the overload of information. If we look at the past two-three decades, there has been an exponential growth in the amount of information available online. This huge pile of information, although marked a revolutionary change in the knowledge gathering capability of human beings, made it quite difficult to manoeuvre through the webs of internet and find relevant and reliable information. This is when the industry realized the need of recommender systems which can filter, prioritize and efficiently deliver relevant information to the user in a very short period of time.

Recommender system, in its most naïve definition, is a system that provides a user with the most relevant information that the user is seeking on the basis of user’s preferences, interest, or observed behavior about item. Initially, recommender systems started by recommending most popular or famous items and with the course of time developed into more sophisticated recommender systems utilizing collaborative filtering, content-based filtering and subsequently hybrid filtering.

While Implementing Place recommendation system, ratings given by the user were considered wherein each user rated the city he/she visited on a scale of 5. After recording the rating, the data was collected and stored in the database for further processing purpose. User-user collaborative filtering algorithm, item-item collaborative filtering algorithm and non-personalized algorithm were used to check for the accuracy during recommendation.

**1.1 Project Goal and benefits of this project**

Public opinion/ Friends opinion is always considered when it comes to seeking an opinion about holiday destination. The main goal of the project is to recommend most rated places to the tourists depending on their interest.

**1.2 System installation and implementation**

System installation:

1. Install PyCharm Community version.
2. Install Python 3.5 or later.
3. Install Flask library in PyCharm for integrating web in Python.
4. Install various libraries like Pandas, NumPy, scikit.
5. Install MySQL for database collection and storage.

**2. Contribution to the Project**

I implemented three algorithms:

* User-user collaborative filtering algorithm.
* Item-item collaborative filtering algorithm.
* Non-Personalized Algorithm.

**2.1 User-user Collaborative Filtering**

User-user Collaborative Filtering builds a system that uses profile of the given user and provide recommendation completely based on that user’s preference and liking. The similarity between the two users is calculated over the places that both users have rated in the past. The numerator of common ratings was calculated for all the ratings given by both users which makes it similar to cosine similarity.

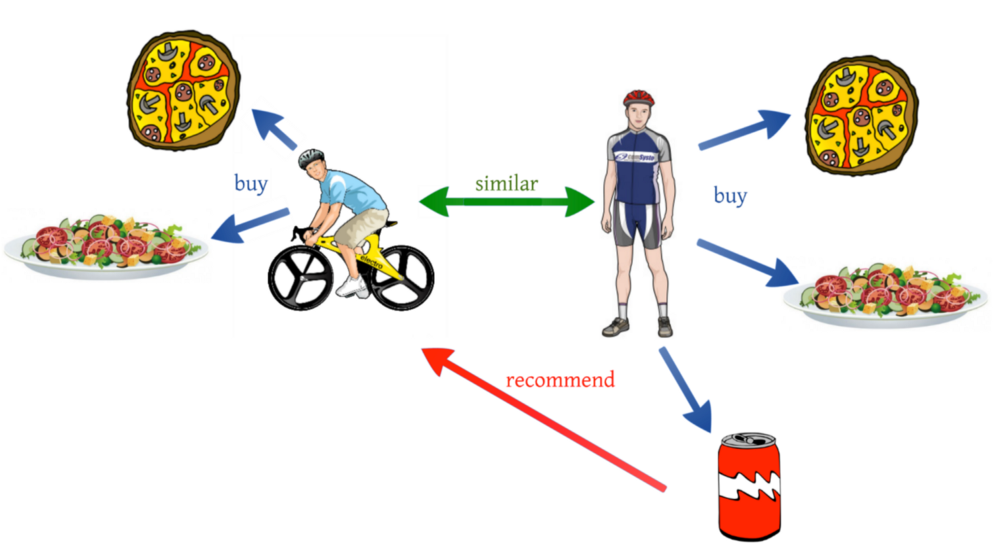
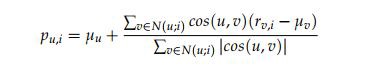


Fig 1: User user Collaborative filtering (Ankur Tomar, 2017)

For a user, an average place lies at a scale of 2 while another user rates an average place as 4. So first user’s 2 is similar to second user’s 4. To incorporate this inconsistency, we will calculate the mean of the ratings of the user and then subtract this mean from each of the ratings provided by the user. This will tell us how much above or below average a user rated the place. After incorporating this, the final rating formula looks like this :



**2.2 Item-Item Collaborative Filtering:**

ITEM-ITEM collaborative filtering look for items that are similar to the articles that user has already rated and recommend most similar articles. In this case we don’t mean whether two items are the same by attribute like Fountain pen and pilot pen are similar because both are pen. Instead, what similarity means is how people treat two items the same in terms of like and dislike.

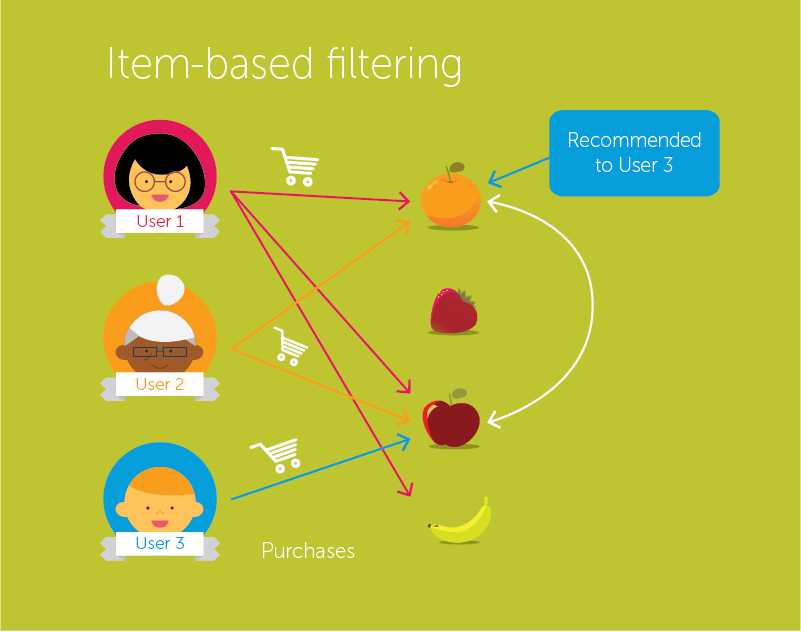
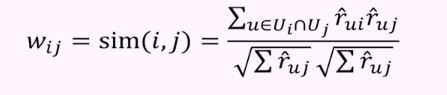


Fig 2: Item-Item collaborative Filtering

To calculate similarity between two items, we looks into the set of items the target user has rated and computes how similar they are to the target item i and then selects k most similar items. Similarity between two items is calculated by taking the ratings of the users who have rated both the items and thereafter using the cosine similarity function mentioned below:



Once we have the similarity between the items, the prediction is then computed by taking a weighted average of the target user’s ratings on these similar items. The formula to calculate rating is very similar to the user based collaborative filtering except the weights are between items instead of between users.

**2.3 Non-Personalized Recommender System:**

As the name suggests, this recommender system provides with general recommendations to the user without any context of what user wants or what is his preference. Let’s think of Amazon for example. When we visit the website, it provides us with a list of most popular item. These popular items can be based on different parameters like geography, age, sex etc. Based on these parameters, system calculates the mean of the product rating of all the users and provides us with the products with maximum mean. This is also called ***stereotyped recommender system.***



Fig 3: Non-Personalized Recommender System

But there is a problem in this approach of taking simple mean of the ratings. When there are very less number of ratings, we have a very less confidence of whether a good rating provided to an item is actually good or not. We do not want to be very much sanguine about an item being very good based on very few ratings. To overcome this challenge, we **damp** the overall mean. In this, we assume that without evidence, every item is average and every additional rating is the evidence if the given item is above average or below average. In simple language, it will not allow any item to have a very good rating on the basis of only few good ratings until sufficient number of good ratings is not provided.

To achieve this, we modify the formula of mean in the following way :



α : Strength of the evidence required to overcome the overall mean(Generally equals to 5 but can be changed as per business requirements)

µ : Global mean rating

|  |  |  |  |
| --- | --- | --- | --- |
| **SR.NO** | **USER-USER COLLABORATIVE ALGORITHM** | **ITEM-ITEM COLLABORATIVE FILTERING** | **NON PERSONALIZED RECOMMENDER SYSTEM** |
| 1. | User-User Collaborative Filtering Algorithm makes the recommendation based on the similar likes of the people. | Recommenders scale with the number of items or users they must deal with, so there are scenarios in which each type can perform better than the other. | Non Personalized recommender system recommends what is popular and relevant to all the users. |
| 2. | Similarity estimates between users are more likely to converge over time | Similarity estimates between items are less likely to converge over time | Similarity estimates are not taken into consideration |
| 3. | User based recommenders begin with the list of similar user’s preferences. | Item based recommenders begin with a list of a user's preferred items. | Non personalized based recommenders begin with a list of a user's preferred items. |

**2.4 Model Evaluation and Model Comparison:**

Fig 4: Comparison of algorithms

**2.5 Accuracy of the algorithms:**

The recommendation system provides a prediction of ratings to new cities for a particular user based on his ratings to similar cities and also on ratings given by other users.

The accuracy of the system is directly based on whether or not a recommended city is actually preferred by the user. The algorithm after processing gives a sorted list of cities as output. This list is sorted in descending order based on predicted ratings of the city. The list consists of all the cities present in the database. So the algorithm also predicts ratings for the cities that a user might have already rated. We can easily compute the accuracy of the system by evaluating or comparing these ratings.

User to user collaborative filtering is used for recommendation.

Accuracy =

As we can see the accuracy of the system is the number of correct recommendations divided by the number of total recommendations.

The database consists of 50 users totaling of 500 ratings. The algorithm is able to recommend 96% of the users cities which they value have already rated highly.

The other two algorithms are :

1. Non-personalized recommendation system

This algorithm is used to calculate the highest rated city among all the users irrespective of the current user likings which although has high accuracy to recommend the most rated city is not personalized and hence is not used in the system. (W, S, 2010).

2. Item-Item collaborative Filtering

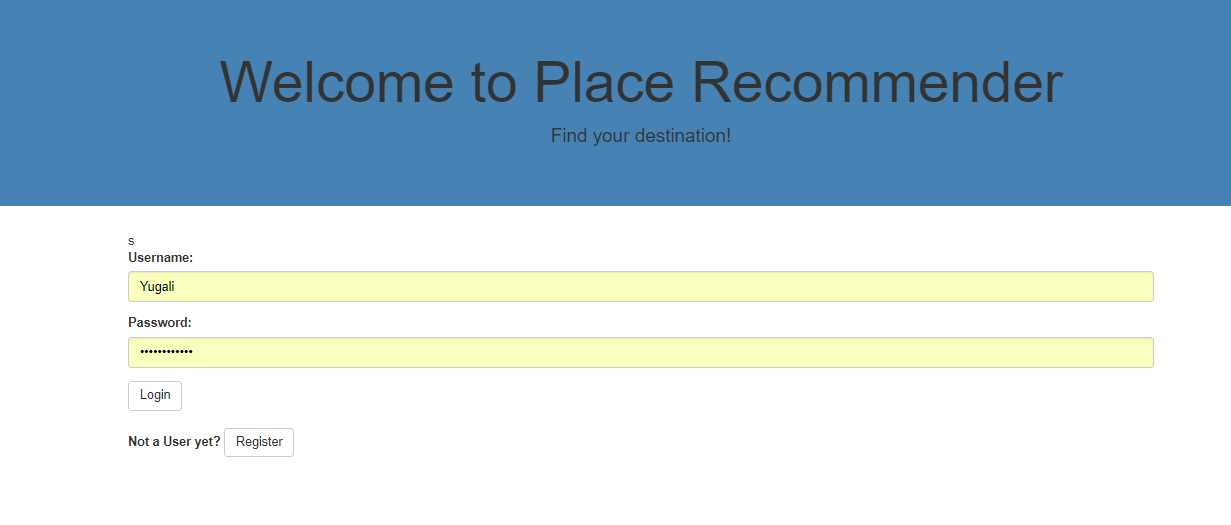
This algorithm can although be used to recommend personalized list of cities for the users but has a major issue which affects its accuracy.

Data Sparsity: In case of large number of items, number of items a user has rated reduces to a tiny percentage making the correlation coefficient less reliable.

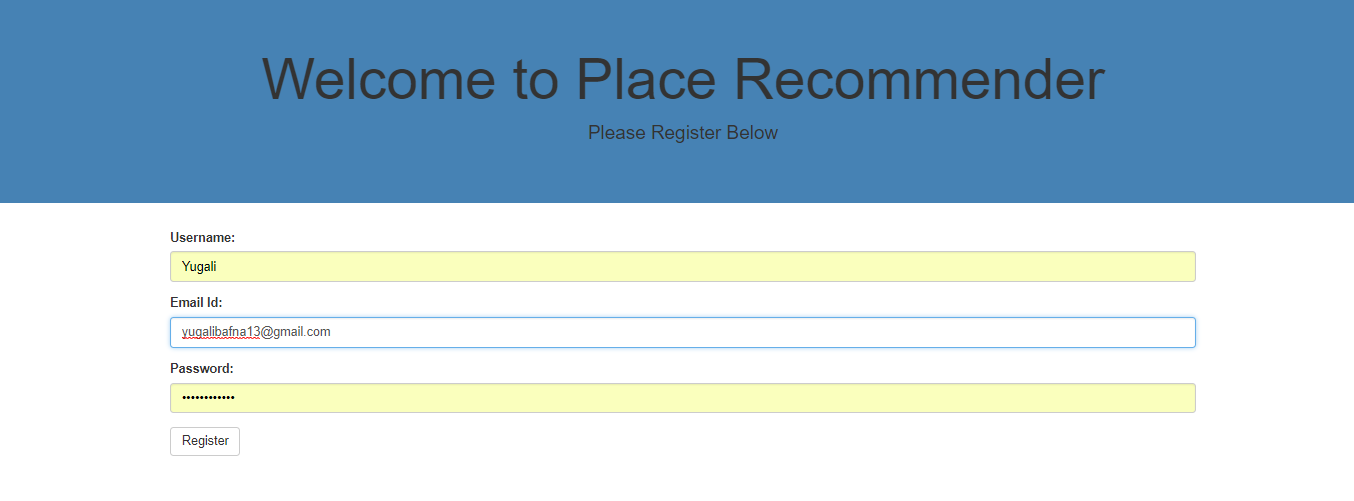
Therefore the accuracy of this algorithm is limited to 92%.

|  |  |  |  |
| --- | --- | --- | --- |
| Factor | Non Personalized | Item – Item Collaborative Filterting | User - User Collaborative Filterting |
| Personalized for user | No | Yes | Yes |
| Accuracy | 100% but 0% for respective user | 92% | 96% |

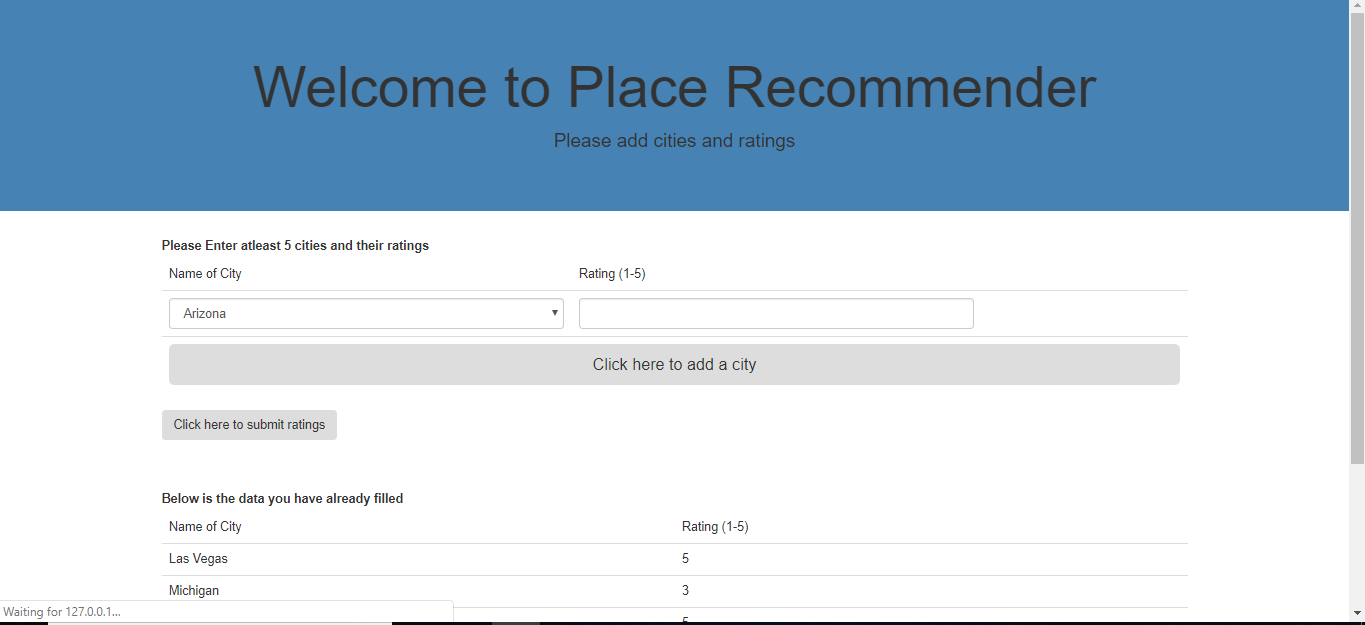
**2.6 Results**



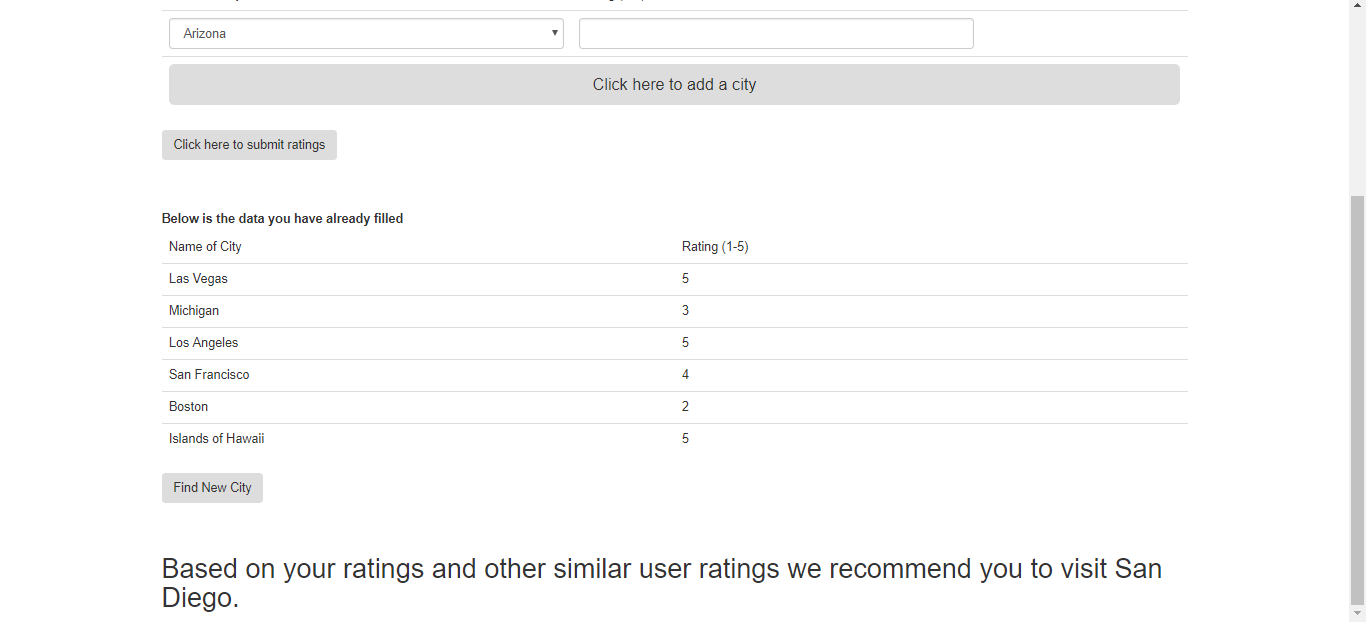
**Fig 5: Login Page**

****

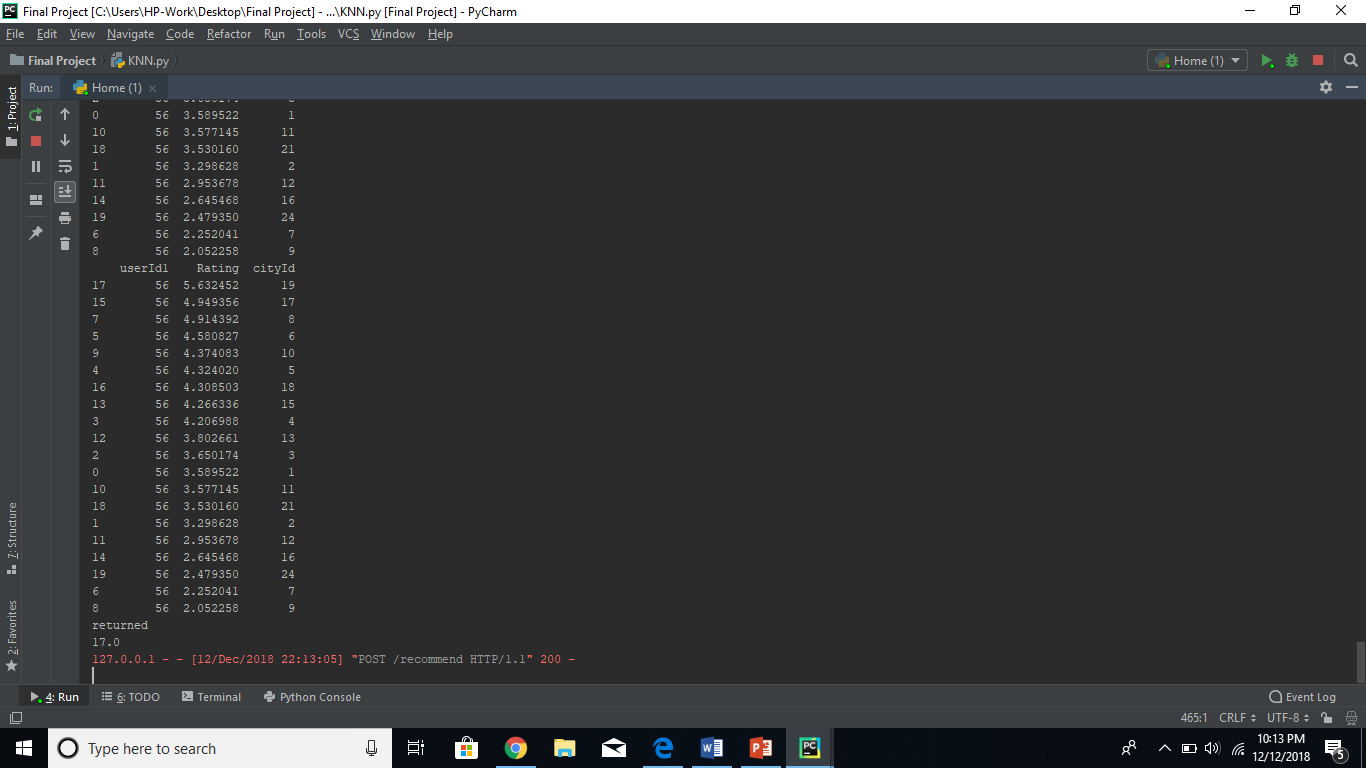
**Fig 6: Registration Page**

****

**Fig 7: Add the cities**

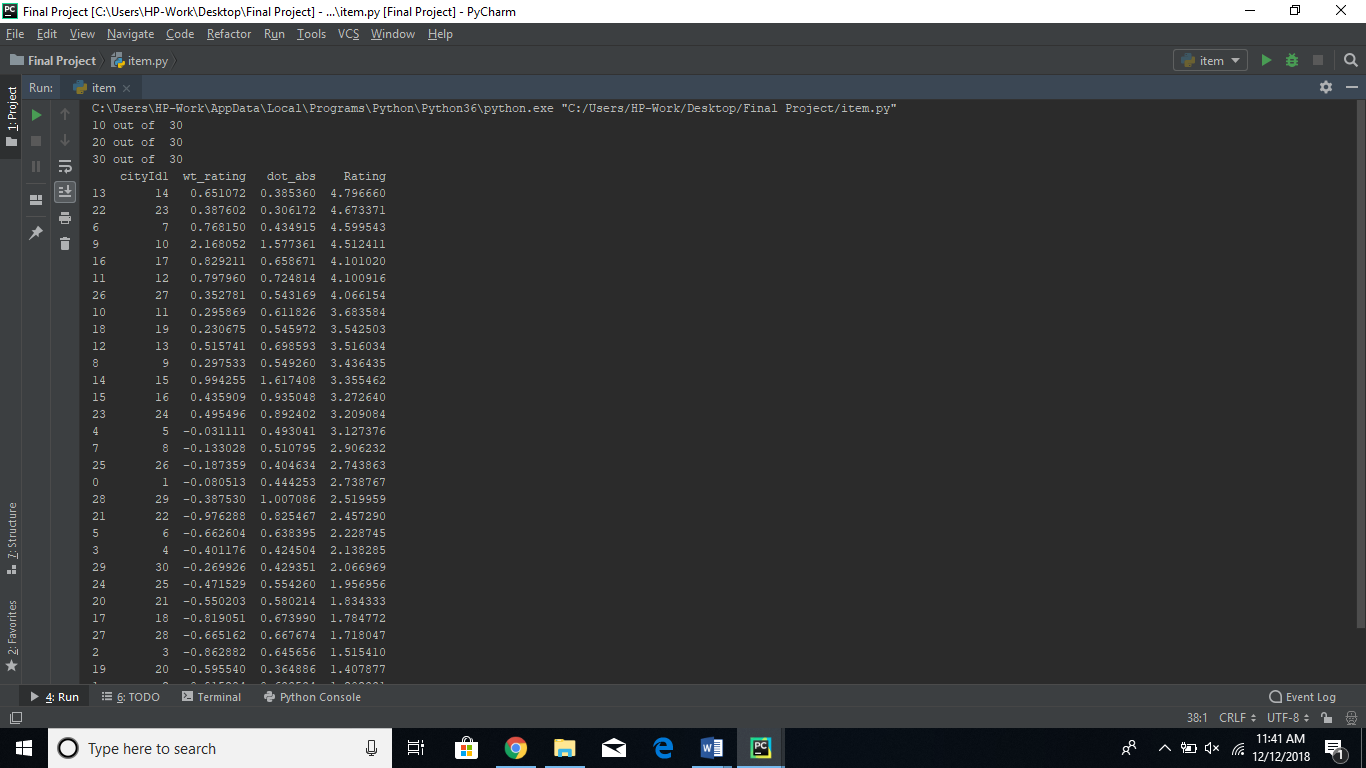
****

**Fig 8: Recommendation**

****

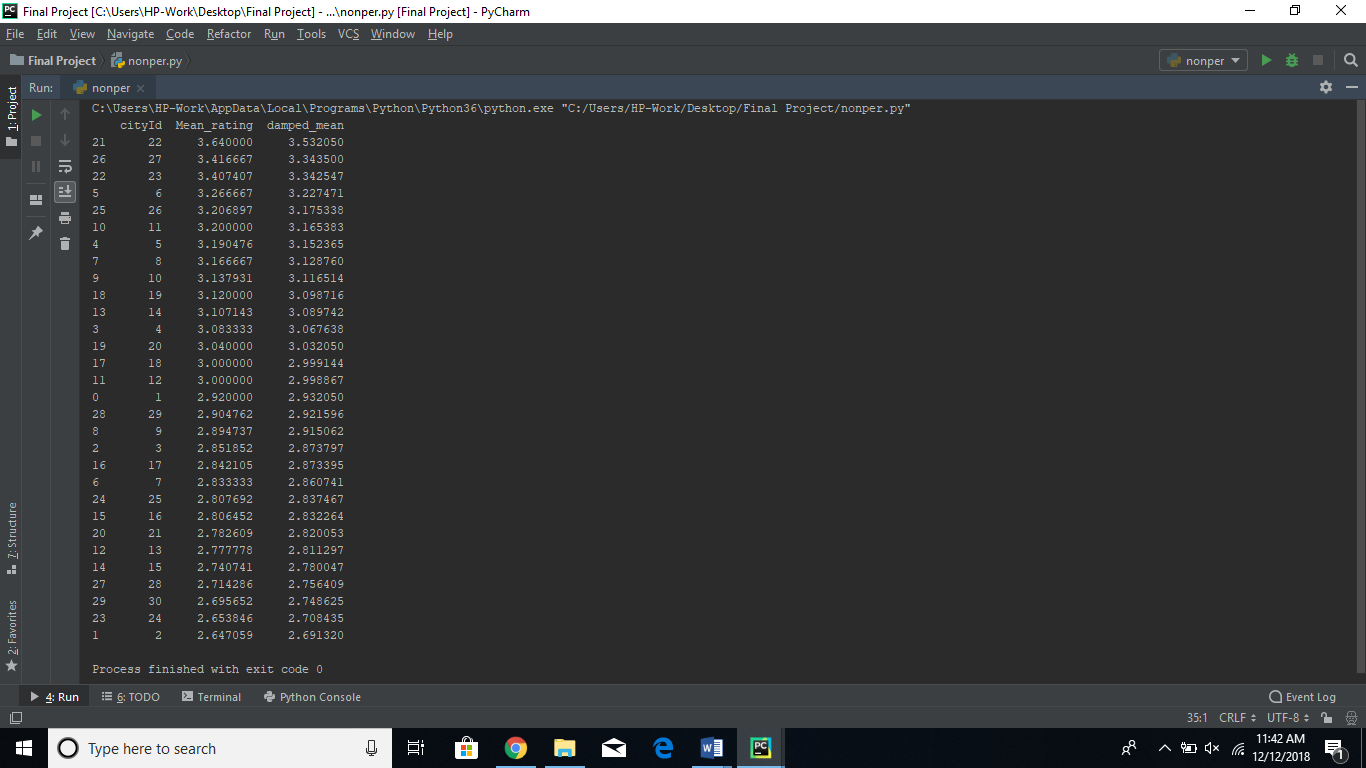
**Fig 9: Console Output of user-user collaborative filtering**

**Output of Item-item collaborative Filtering:**

****

**Fig 10: Console Output of item-item collaborative filtering**

**Output of Non-personalized Filtering:**

****

**Fig 11: Console Output of Non-Personalized algorithm**

**3. Implications**

Implications:

* Tourist place recommendation applications like MakeMyTrip, TripAdvisor etc. use the recommendation of tourists based on ratings given by travelers. Location based tourist place recommendation are also a good example of place recommendation system wherein users are suggested to visit the place near to their local area which is rated high by their friends.
* Today, we all know that Amazon a cross large and well-known company is using recommender system to recommend the most likely viewed items of the customer. This shows the impact of recommendation system.
* Other examples are like Netflix the video streaming service which is using AI’s recommendation system which is saving around $1billion each year.
* Then we have Spotify a music recommendation application, which uses Machine learning to handle the huge amount of music data arising day-by-day.

**4. Conclusion**

This system will be very useful for tourists to decide the most rated tourist destination for their holidays. Recommendation algorithms such as collaborative filtering methods were studied which recommend places based on user’s interests. User-user collaborative filtering algorithm recommend similar places to the tourist. Item based collaborative filtering recommends places based on the preference of the tourists. Non Personalized recommender system recommends frequently visited places by the tourists in comparison with other tourists. Among all the three algorithms, user-user collaborative filtering has highest accuracy of 95%. This system will help tourists go for a perfect holiday destination recommended by their friends however it failed to give location based suggestions to the tourists for visiting local areas. This project taught us to handle data coming from the database and integrate that data with python framework deepen our knowledge in the field of Big Data and Data Science.

**References**

Cheng W, Yin G, Dong Y, Dong H, Zhang W (2016) Collaborative Filtering Recommendation on Users’ Interest Sequences. PLoS ONE 11(5): e0155739. [*https://doi.org/10.1371/journal.pone.0155739*](https://doi.org/10.1371/journal.pone.0155739)

Chu. W, Park. T.S. et al, (2010), "Personalized Recommendation on Dynamic Content Using Predictive Bilinear Models" *Yahoo! Labs.,2821 Mission College Blvd, Santa Clara, CA95054*

Dang, T. A., & Viennet, E. (2013, January). Collaborative filtering in social networks: A community-based approach*. In Computing, Management and Telecommunications (ComManTel), 2013 International Conference on (pp. 128-133). IEEE.*

Farhan Hassan Khan, Saba Bashir. Collaborative Filtering based Online Recommendation Systems: A Survey*. International Conference on Information & Communication Technologies, At Karachi, Pakistan, Volume: 7th (2015)*

Rana, C., & Jain, S. K. (2012). Building a Book Recommender system using time based content filtering. *WSEAS Transactions on Computers, 11(2), 2224-2872.*

Sharma, R., Gopalani, D., & Meena, Y. (2017, February). Collaborative filtering-based recommender system: Approaches and research challenges. *In Computational Intelligence & Communication Technology (CICT), 2017 3rd International Conference on (pp. 1-6). IEEE*

Tomar Ankur (2018). User user Collaborative Filtering in Python. Retrieved from <https://medium.com/@tomar.ankur287/user-user-collaborative-filtering-recommender-system-51f568489727>

Yuri Stekh, Mykhoylo Lobur, Vitalij Artsibasov, Vitalij Chystyak (2015). Methods and Tools for Building Recommender Systems. *CADSM 2015, Polyana-Svalyava (Zakarpattya), UKRAINE.*