Restaurant Recommendation System

Abdullah Parves

Department of computer Science Engineering, Primeasia University, Bangladesh
abdullahparves62@gmail.com

September 25, 2021

1 Abstract

A recommendation system is a system that provides suggestions to users for certain resources like books, movies, songs, etc., based on some data set. Restaurant recommendation systems usually predict what restaurant a user will like based on the attributes present in previously liked restaurant. Such recommendation systems are beneficial for organizations that collect data from large amounts of customers, and wish to effectively provide the best suggestions possible. The project analyzes the data of rating provided by the end users and use the data to recommend foods and restaurants to the users. Through this project, I have collected the necessary details of some of the most popular restaurant The approach adopted to do so is content-based filtering using similarity matrix. This data set consists of restaurants of Hyderabad/India collected from Zomato.

2 Introduction

The recommendation system plays an essential role in the modern era and used by many prestigious applications. The recommendation system has made the collection of apps, creating a global village, and growth for abundant information. Not all the data available on the Internet is of use or provides satisfactory results to the users. Data in such huge volumes often turns out to be inconsistent and without proper processing of this information, it gets wasted. In such cases, users have to run their search multiple times before they finally obtain what they were originally looking for. To solve this problem, researchers have come up with recommendation systems.

A recommendation system provides relevant information to the users by taking into account their past preferences. Data is filtered and personally customized as per the user requirements. With more and more data available on

the Internet, recommendation systems have become really popular, due to their effectiveness in providing information in a short time-span. Recommender systems have been developed in various areas such as music, movies, news, and products in general. In today's age, a majority of organizations implement recommendation systems for fulfilling customer requirements. LinkedIn, Amazon, and Netflix are just a few to name. By the method in which recommendation systems work, they can be broadly classified into three categories—Content-based, Collaborative and Hybrid approach.

1. Content Based recommendation system :Content Based recommendation system considers the user's past behavior and identifies patterns in them to recommend to recommend items that are similar to them.

2. Collaborative filtering analyses : Collaborative filtering analyses the user's previous experiences and ratings and correlates it with other users. Based on the ones that have the most similarity, recommendations are made.

3. Hybrid Approach Both content-based- and collaborative-based filtering : Hybrid Approach Both content-based- and collaborative-based filtering have their own limitations. To overcome this, researchers suggested this approach which would combine the advantages of both the methods.

4. TD-IDF: TD-IDF are word frequency scores that try to highlight words that are more interesting, that's frequent in a document but not across documents. The TF-IDF Vectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new documents.

5. Vectorizer: Vectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new documents.

6. Cosine similarity: Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis.

In this paper, we focus on identifying Recommender system will look at the reviews of other restaurants.System will recommend us other restaurants with similar reviews and sort them from the highest rated.

2.1 Motivation

During early years of recommendation systems, content-based approaches; where an item similar to the items highly rated by a user, were provided as a recommendation to the user. One limitation to this approach was that the result would most likely be over specialized and features pertaining to a particular

class of items had to be initially provided. Also, users with less previous rating data could not be provided with well-tailored recommendations. Another widely used method, collaborative filtering, also has its own limitation e.g. if a restaurant is reviewed by only a small number of users, it will rarely show up in the recommendation results even if it receives very high ratings from reviewers. Even after making slight tweaks to the collaborative approach algorithm, at times it is possible that highly rated ones do not align with the individual's tastes. Thus, to provide targeted and good recommendation, this problem inspires the making of a model-based recommendation system.

2.2 Problem Statement

When searching for restaurant's information and making decisions on where to eat people rely on review sites. This is a difficult job for an average internet user because of information overload factor of modern day Internet and online critic reviews. Even if there are sites which recommend restaurants using content-based approach or collaborative approach, efficient prediction from small amount of data of a particular restaurant is very difficult due to sparsity of the data. The way this problem is modelled is to be able to predict 'Yes' or 'No' for any given restaurant and user.

2.3 Scope and Limitations

2.3.1 Scope:

The restaurants and hotels within the valley will be listed in the application.

2.3.2 limitations:

1. Only the registered restaurants and hotels will be listed in the application.
2. Only the registered users will be allowed to rate the foods.
3. Users will be allowed to rate a particular food on the restaurant only once.

2.4 Organization of Project Report

1. introduction
2. Related Works
3. System Design
4. Experiment
5. Compare with other works
6. Conclusion
7. Future Work

8. Recommendations

3 Related Works

Recommender system has been widely used in recent days, especially in the field of e-commerce. Listed below are some of the popular application based which uses recommendation algorithm

1. Amazon.com Amazon.com is the largest internet-based retailer of US. It uses recommendations as a targeted marketing tool in manyemail campaigns and on most of its websites' pages. Clicking on "Your Recommendations" link clients are directed to a page where they can filter their recommendations by product line and subject area, rate recommended products and rate their previous purchase. Our shopping cart recommendations offer product suggestions to the clients based on the items in their shopping cart(Linden, 2003)

2. Netflix Netflix is the world's leading internet television network with over 81 million members in 190 countries. It has a massive database of TV shows, movies, documentaries, etc. It recommends videos to the user by the shows they watch, their ratings on previously watched shows, etc (Melville, 2010).

3. TripAdvisor.com TripAdvisor is one of the world's largest travel site, which enables travelers to plan and book their trip to almost every part of the world. It recommends places to the users by whether, their past travel patterns, type of trip, etc (Melville, 2010).

4. Moviefinder.com Moviefinder.com allows customers to locate movies with a similar "mood, theme, genre or cast" to a given film. From the information page of the film in question, customers click on the Match Maker icon and are provided with the list of recommended movies, as well as links to other films by the original film's director and the main actors (Melville, 2010)

5. We Predict

   We Predict uses item-based collaborative filtering method. It recommends movies to customers based on their previously indicated interests. Customers enter a rating on a 5-point scale – from A to F – for movies they have viewed (Melville, 2010).

6. YellowNepal

   YellowNepal is an application available in both web and mobile platform. It provides the list of restaurants within Kathmandu valley. It only performs location based search. The user preference and ratings are not taken into consideration in the application.

4 System Design

Our recommendation system is based on Collaborative filtering . It works by searching a large group of people and finding a smaller set of users with tastes similar to a particular user.

4.1 Dataset Implemented

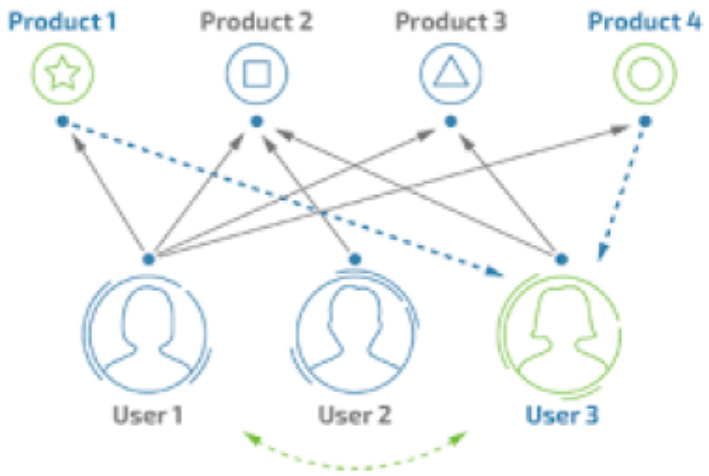
For restaurant recommendation system, ZomatoRestaurants Hyderabad datase has been taken from Kaggle, an online platform for data science enthusiasts.

4.2 Algorithm

1. Data collection
2. Data preprocessing
3. Algorithms used

4.3 Algorithm used

1. User-Based Collaborative Filtering: This filtering methodology is used to recommend items to the user based on the rating of other users having similar preferences. The algorithm works perfectly fine when the number of users and items are less. But, when the number of items starts increasing problems of data sparsity occurs.



Algorithm

(a) Collect preference of all the users listing the ratings of users on different food items.

(b) Calculate similarity between the users using Pearson Correlation Coefficient.

$$r = \frac{n(\sum xy) - (\sum x).(\sum y)}{\sqrt{(n\sum x^2 - (\sum x)^2)(n\sum\ y^2 - (\sum y)^2)}}$$

where,

n = number of pairs of scores

$\sum xy$ = sum of products of paired scores

$\sum x$ = sum of x scores

$\sum y$ = sum of y scores

$\sum x^2$ = sum of squared of x scores

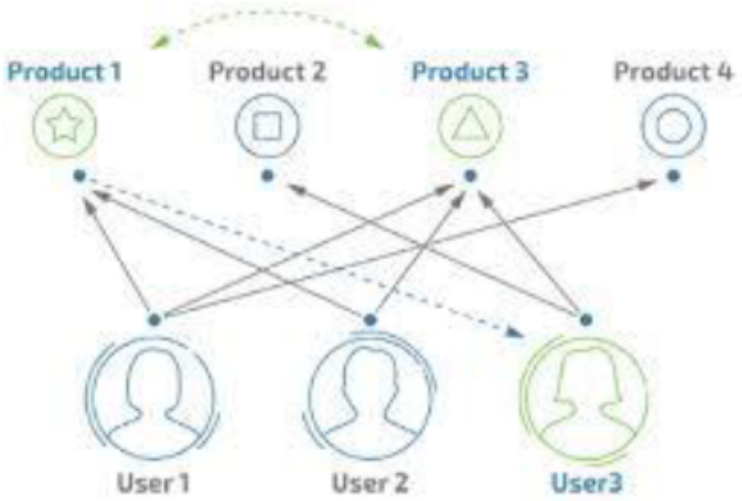
$\sum y^2$ = sum of squared of y scores

(c) Sort the similarity scores between the users such that the users in descending order.

(d) Produce weighted score that ranks the users by taking the multiplying their ratings of different foods with the similarity score.

4.4 Algorithm used

1. This filtering methodology is used to recommend items to the users based on their previous ratings. This method is similar to user-based collaborative filtering, except that the similarity between items is calculated instead of similarity between users.



Algorithm

2. List all the items with the given to them by different users.
3. Calculate similarity between the items using Pearson Correlation Coefficient.

$$r = \frac{n(\sum xy) - (\sum x).(\sum y)}{\sqrt{(n\sum x^2 - (\sum x)^2)(n\sum y^2 - (\sum y)^2)}}$$

where,

n = number of pairs of scores

$\sum xy$ = sum of products of paired scores

$\sum x$ = sum of x scores

$\sum y$ = sum of y scores

$\sum x^2$ = sum of squared of x scores

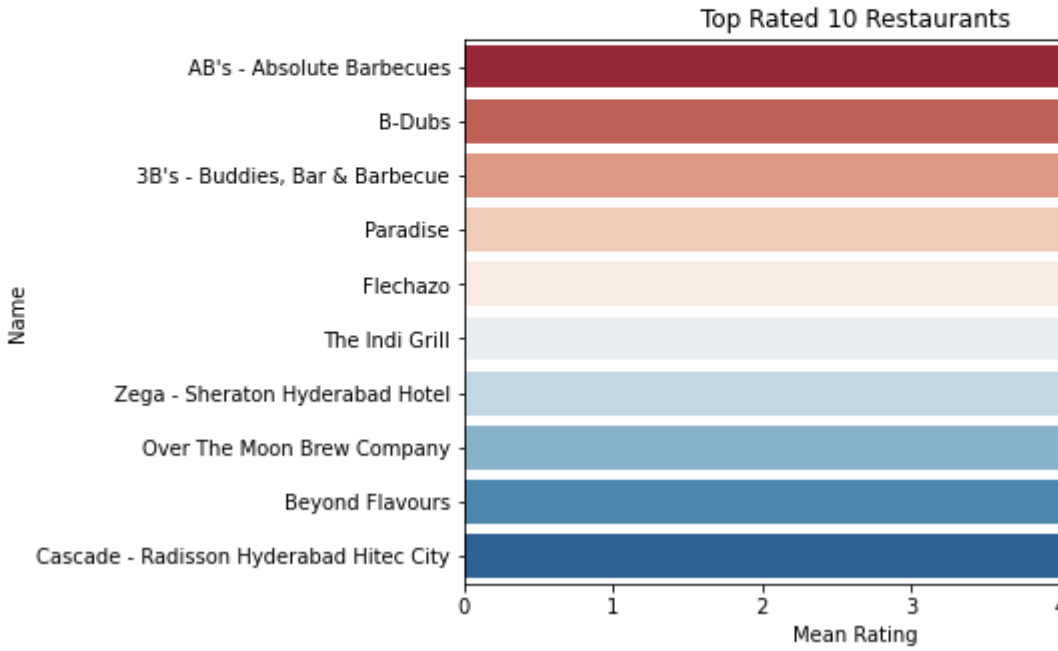
$\sum y^2$ = sum of squared of y scores

4. Sort the similarity between the items such that the items in descending order.
5. Produce weighted scores that rank the items by multiplying the ratings by different users similarity score.

5 Experiment

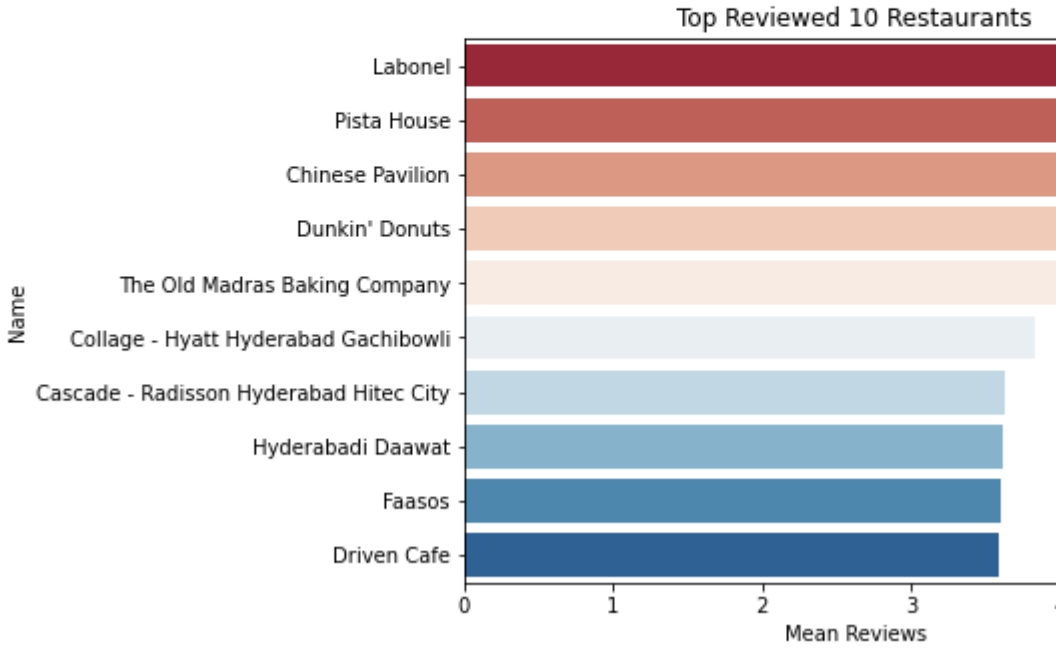
Top rated 10 restaurant

```
In [26]: df_rating = df.drop_duplicates(subset='Name')
         df_rating = df_rating.sort_values(by='Mean Rating', ascending=False).

         plt.figure(figsize=(7,5))
         sns.barplot(data=df_rating, x='Mean Rating', y='Name', palette='RdBu'
         plt.title('Top Rated 10 Restaurants');
```



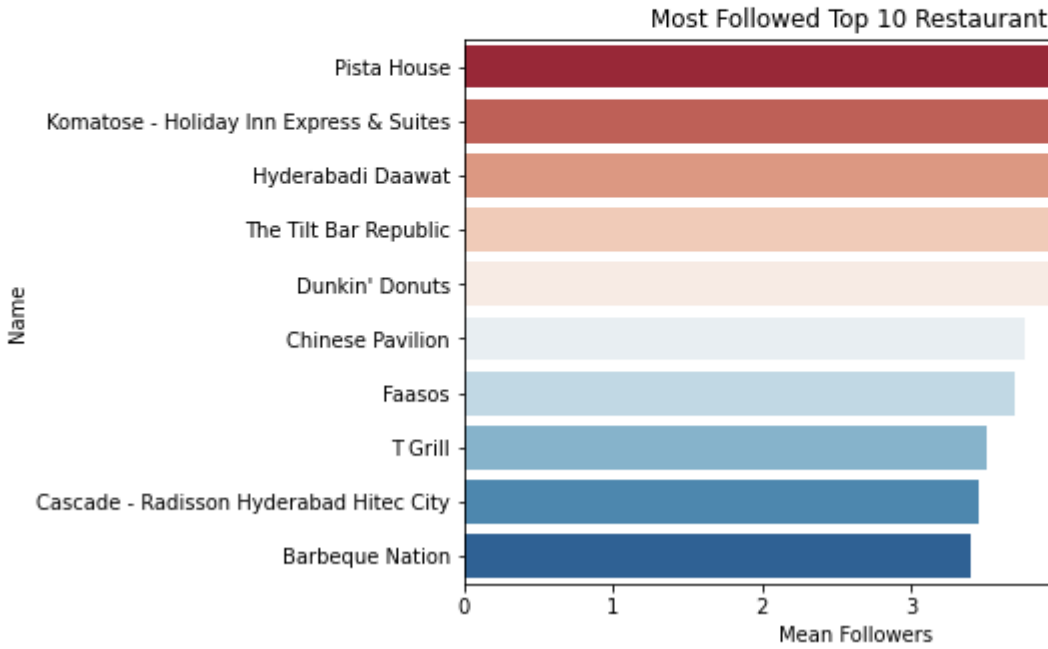
Top reviwed 10 restaurant

In [27]:
```
df_reviews = df.drop_duplicates(subset='Name')
df_reviews = df_reviews.sort_values(by='Mean Reviews', ascending=Fals

plt.figure(figsize=(7,5))
sns.barplot(data=df_reviews, x='Mean Reviews', y='Name', palette='RdB
plt.title('Top Reviewed 10 Restaurants');
```



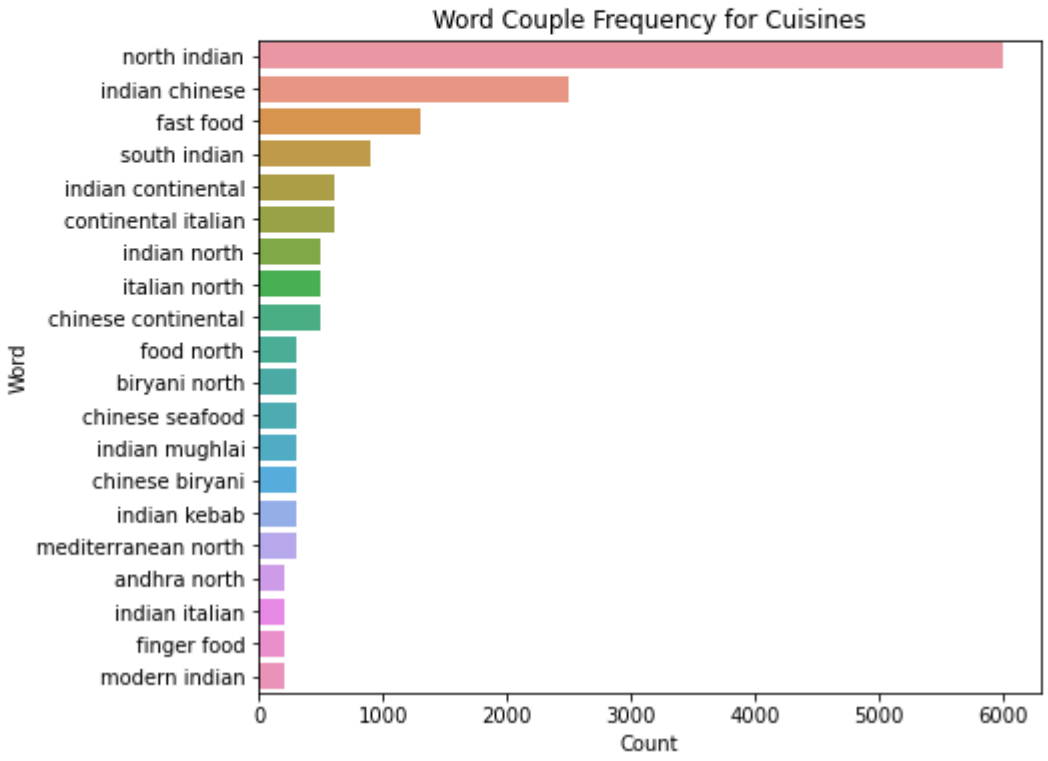
Most Followed Top 10 Restaurants

In [28]:
```
df_followers = df.drop_duplicates(subset='Name')
df_followers = df_followers.sort_values(by='Mean Followers', ascendi

plt.figure(figsize=(7,5))
sns.barplot(data=df_followers, x='Mean Followers', y='Name', palette=
plt.title('Most Followed Top 10 Restaurants');
```



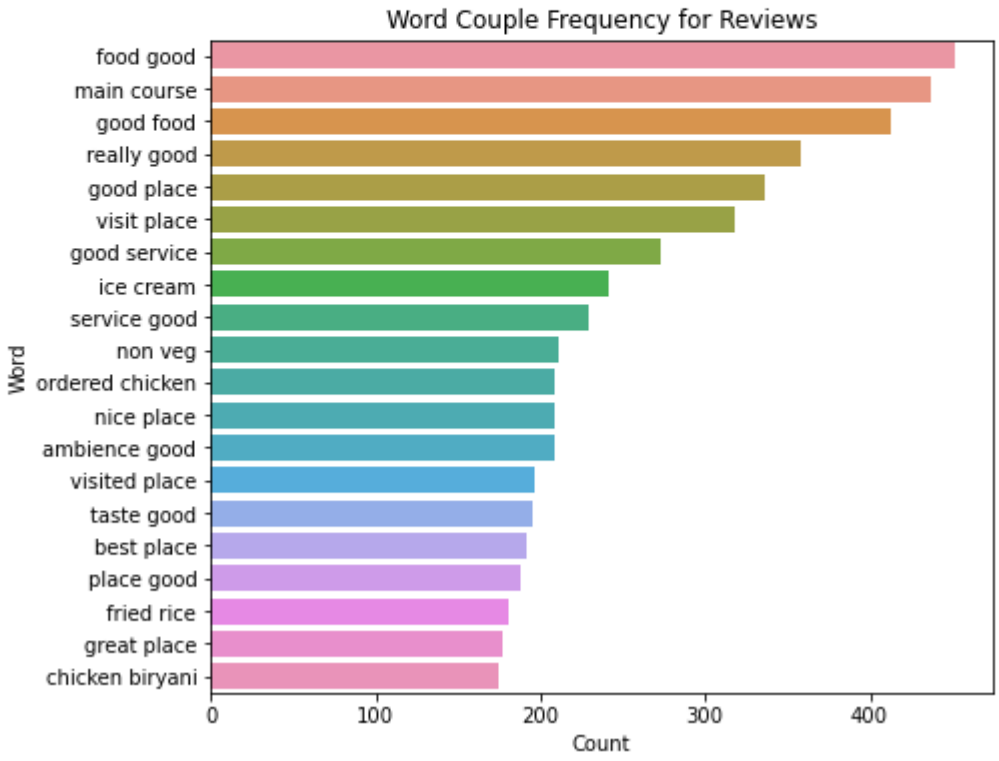
Here we use the frequency for cuisines
Top 20 two word frequencies for Cuisines

In [30]:
```
# Top 20 two word frequencies for Cuisines
list1 = get_top_words(df['Cuisines'], 20, (2,2))

df_words1 = pd.DataFrame(list1, columns=['Word', 'Count'])

plt.figure(figsize=(7,6))
sns.barplot(data=df_words1, x='Count', y='Word')
plt.title('Word Couple Frequency for Cuisines');
```



Top 20 two word frequencies for Reviews

In [31]:
```
# Top 20 two word frequencies for Reviews
list2 = get_top_words(df['Review'], 20, (2,2))

df_words2 = pd.DataFrame(list2, columns=['Word', 'Count'])

plt.figure(figsize=(7,6))
sns.barplot(data=df_words2, x='Count', y='Word')
plt.title('Word Couple Frequency for Reviews');
```



Apply Cuisines similarity

In [33]:
```
def recommend(name, cosine_similarities = cosine_similarities):

    # Create a list to put top 10 restaurants
    recommend_restaurant = []

    # Find the index of the hotel entered
    idx = indices[indices == name].index[0]

    # Find the restaurants with a similar cosine-sim value and order
    score_series = pd.Series(cosine_similarities[idx]).sort_values(as

    # Extract top 30 restaurant indexes with a similar cosine-sim val
    top30_indexes = list(score_series.iloc[0:31].index)

    # Names of the top 30 restaurants
    for each in top30_indexes:
        recommend_restaurant.append(list(df.index)[each])

    # Creating the new data set to show similar restaurants
    df_new = pd.DataFrame(columns=['Cuisines', 'Mean Rating', 'Cost',

    # Create the top 30 similar restaurants with some of their column
    for each in recommend_restaurant:
        df_new = df_new.append(pd.DataFrame(df[['Cuisines','Mean Rati

    # Drop the same named restaurants and sort only the top 10 by the
    df_new = df_new.drop_duplicates(subset=['Cuisines','Mean Rating',
    df_new = df_new.sort_values(by='Mean Rating', ascending=False).he

    print('TOP %s RESTAURANTS LIKE %s WITH SIMILAR REVIEWS: ' % (str(

    return df_new
```

6 Compare with other works

Though Both used Collaborative filtering but in their proposed system they made the system in such a way that users can also get recommendation on new restaurant. In our case we only get recommendation on the restaurant which the dataset contains only. Also we used cosine similarity to fine the similarity between users.

7 Conclusion

The project Restaurant Recommendation System was successfully completed by using User based and Item-based collaborative filtering. The data set were collected from survey which was preprocessed on the basis attributes. The data were then used to model the system. In this paper, a list of restaurant is used for calculating recommendation points. We developed a Collaborative filtering recommender system. We evaluated the proposed method with Zomato Restaurants Hyderabad dataset. It is able to recommend restaurant based on user preferences.

8 Future Work

While the application in its current state satisfy the original design requirements for this project, there are room for improvements. With more time, I would like to implement the following features.

1. Improved recommendation engine using other machine learning algorithms such as NLP, Clustering based and combining the result for better prediction accuracy.
2. Better and more efficient data pipeline to handle larger datasets while maintaining fast response time. Expands the application to include more metropolitan areas.
3. Implements more sophisticated user data collections to collect implicit feedbacks (time stayed on page, number of time user views a business, etc.).
4. Allow users to personalize their experience on web UI.

9 Recommendations

The data used on the application is solely based on the data extracted web scraping. In order to commercially use the product, it is important to collect data of all the restaurants in the valley. Furthermore, since the use of mobile phones is huge, the application will be more effective if built in mobile platform.

10 References

1. Xiaoyuan Su, T. M. (2009). A Survey of Collaborative Filtering Techniques. Florida: Department of Computer Science and Engineering, Florida Atlantic University
2. Data-Source: https://www.kaggle.com/akdagmelih/restaurant-recommender-system/data

3. Code Link GitHub:https://github.com/abdullahparves62/Data-Mining/blob/main/Final