# *Classification-Based Destination Recommendation System*

## IS 688 – Web Mining

## Group: F

**Professor**
**Christopher Markson**

**Submitted By:**
**Priyanka Patil**
**Pranjali Telavane**
**Gauravkumar Pawar**
**Sharmili Nag**
**Agrim Sachdev**

# Table of Contents

## Objective

To design a recommendation system that helps users to discover new places of interest in different categories based on similar users' habits.

## Overview

The emergence of information technology and its broad adoption within the tourism industry has led to an explosion in the availability of destination-related information, which greatly helps travelers in planning trips and/or formulating expectations about tourism experiences. We decided to use these available resources and to design a destination recommendation system. The whole and sole purpose of the system is to provide destination recommendations to the users as per his/her mood or interest. For this, we have used data from Foursquare, which contains check-ins of the users in New York City.

To build this recommendation system we have combined different machine learning techniques. Initially we have used K-Means clustering to divide users in seven clusters based upon there visiting frequency. Then we have used decision tree classification to assign test user to respective clusters. Then on that cluster, we have used cosine distance metric to find users who have similar interests as test user. Finally, we will suggest test user three destinations, which he has not visited previously.

## Introduction

Searching for travel-related information and services is one of the top web activities and there is a fast-growing number of websites that support a traveler in the selection of a travel destination or a travel service.

Trip planning is a complex problem-solving activity; it includes many other activities in it like destination selection, hotel booking, car booking and many more. Normally before selecting any travel destination, we check with our friends if they have any suggestion or recommendations and then we select best possible option from it. Now days because of digital word we have digital footprints of the users in the form of their check-ins.

Currently there are very few websites, which are good at destination recommendations, so we decided to design a destination recommendation system, which will help user the in selection of new destinations. For this, we have used data from Foursquare, which contains check-ins of the users in New York City.

## All About Recommendation

**Recommendation System**: A recommendation system is a software program, which attempts to narrow down selections for users based on their expressed preferences, past behavior, or other data, which can be mined about the user or other users with similar interests.

General Requirements for Recommendation Systems: To make a viable recommendation, three things are needed as follows-

- **Background Information** - the information that the system has before the recommendation process begins
- **Input Information** - the information that a user must enter to the system in order to trigger a recommendation
- **An Algorithm** - this will combine background and input information to arrive at the recommendation

## All About Data

- **Dataset source**: https://archive.org/details/201309_foursquare_dataset_umn

Initial Foursquare dataset contained 227,428 check-ins from 1,083 users in the New York City area. Each of these check-in rows contained information about the user's ID, the venue ID, the venue category ID, the venue category name, the location of the venue (latitude and longitude) time of the check-in. To improve performance of recommendation system, we have removed irrelevant columns from dataset. After all the preprocessing, we left with the below files.

## Main Dataset

| File Name | Column Name | Defination |
|---|---|---|
| processed_data.csv | UserID | Unique user identifier |
| | VenueID | Unique venue identifier |
| | Venue sub Category ID | Unique venue category identifier |
| | Venue sub category name | higher-level categories provided by the Foursquare API |
| | Venue main category | Subcategories provided by the Foursquare API |

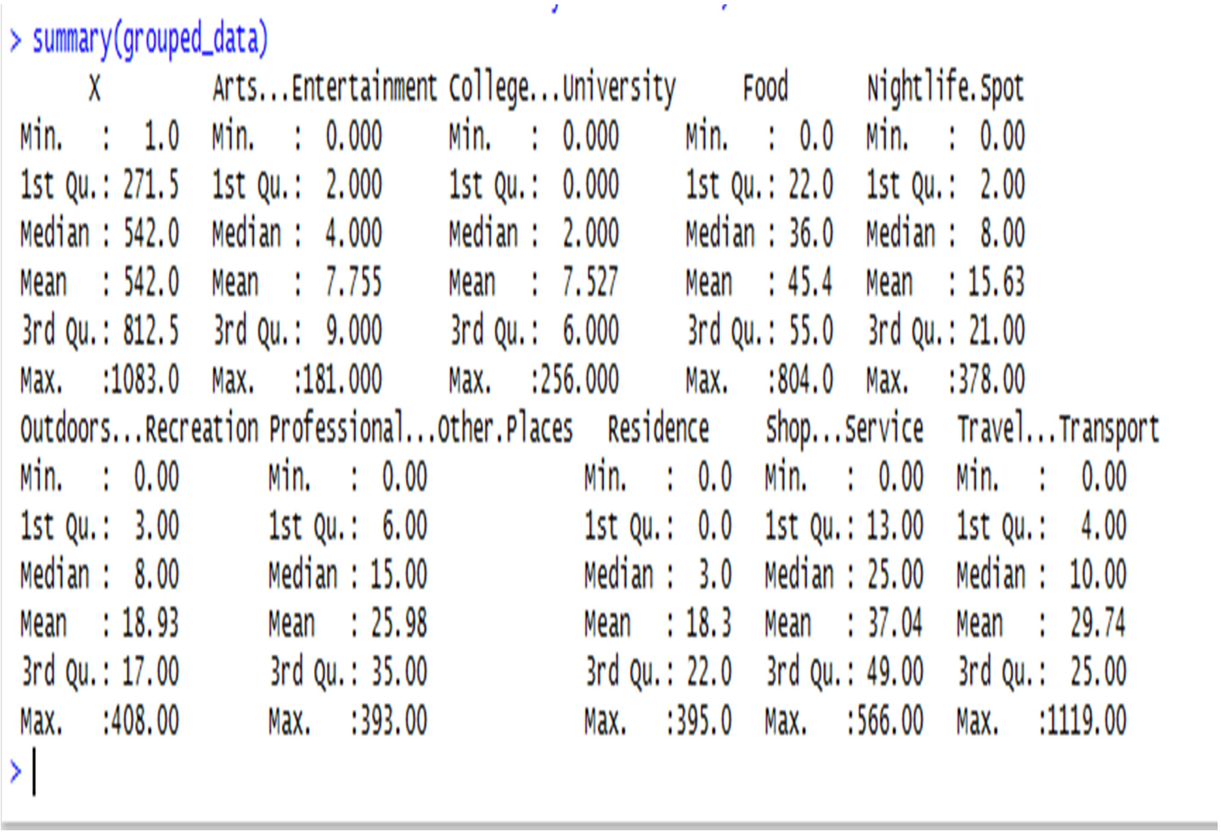
- **Supporting Files**

| File Name | Details |
|---|---|
| processed_data_subcategories.txt | List of check-ins for each of the 1083 users in the nine main categories of venues. Used for computing similarity measures |
| processed_categories_frequency.csv | List of check-ins for each of the 1083 users in the nine main categories of venues. Used for computing clusters for data. |
| categories.csv | A list of the nine main categories provided by foursquare, used while asking user what category of place he would like to visit. |

- **Data Analysis:**

**Distribution of data across nine main categories**

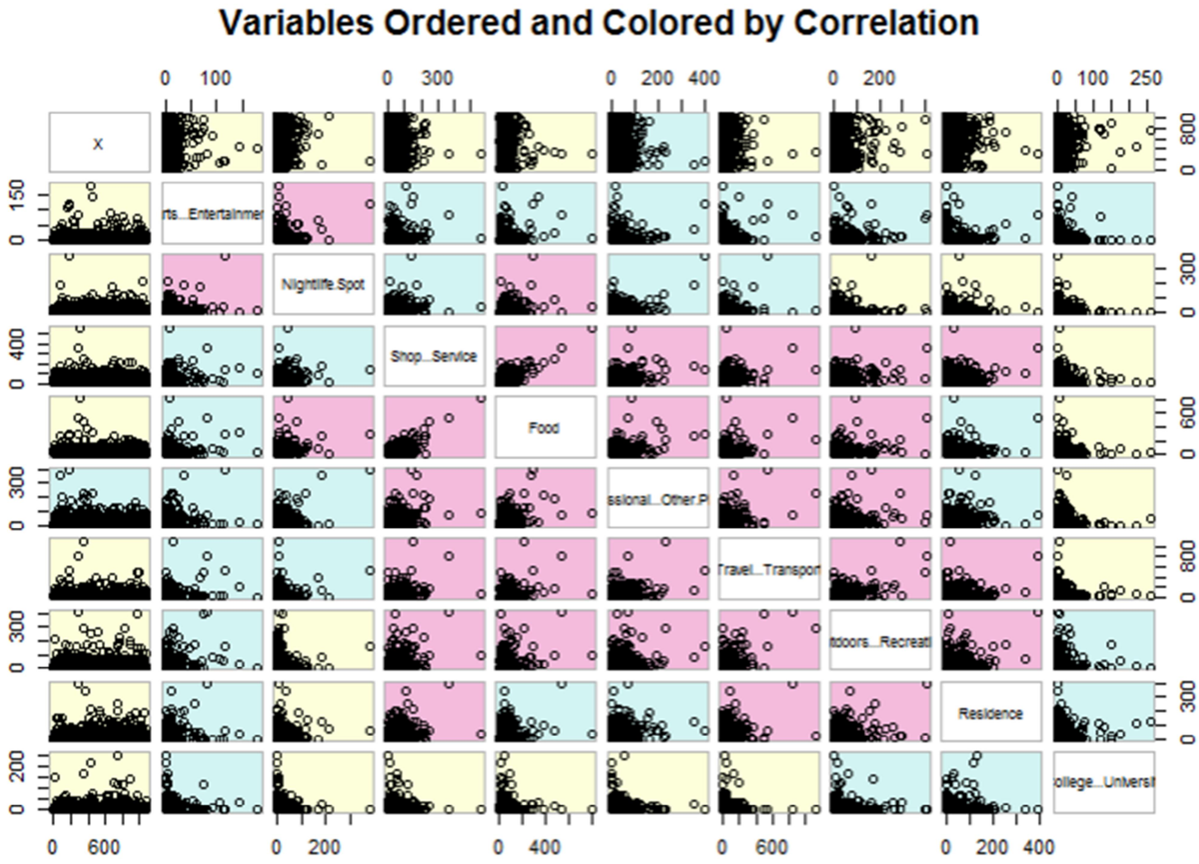
**1) Summary**



```
> summary(grouped_data)
      X          Arts...Entertainment College...University      Food      Nightlife.Spot
 Min.   :   1.0  Min.   :  0.000      Min.   :  0.000      Min.   :  0.0  Min.   :  0.00
 1st Qu.: 271.5  1st Qu.:  2.000      1st Qu.:  0.000      1st Qu.: 22.0  1st Qu.:  2.00
 Median : 542.0  Median :  4.000      Median :  2.000      Median : 36.0  Median :  8.00
 Mean   : 542.0  Mean   :  7.755      Mean   :  7.527      Mean   : 45.4  Mean   : 15.63
 3rd Qu.: 812.5  3rd Qu.:  9.000      3rd Qu.:  6.000      3rd Qu.: 55.0  3rd Qu.: 21.00
 Max.   :1083.0  Max.   :181.000      Max.   :256.000      Max.   :804.0  Max.   :378.00
 Outdoors...Recreation Professional...Other.Places   Residence     Shop...Service  Travel...Transport
 Min.   :  0.00        Min.   :  0.00               Min.   :  0.0  Min.   :  0.00  Min.   :   0.00
 1st Qu.:  3.00        1st Qu.:  6.00               1st Qu.:  0.0  1st Qu.: 13.00  1st Qu.:   4.00
 Median :  8.00        Median : 15.00               Median :  3.0  Median : 25.00  Median :  10.00
 Mean   : 18.93        Mean   : 25.98               Mean   : 18.3  Mean   : 37.04  Mean   :  29.74
 3rd Qu.: 17.00        3rd Qu.: 35.00               3rd Qu.: 22.0  3rd Qu.: 49.00  3rd Qu.:  25.00
 Max.   :408.00        Max.   :393.00               Max.   :395.0  Max.   :566.00  Max.   :1119.00
> |
```

**Analysis: -** It shows that for these nine main categories, measures such as mean will be biased since there are few users with very large number of check-ins and the rest are concentrated close to the origin**.**

**2) Scatterplots**



**Analysis: -** From above plot, we can infer that, the spread of the data was not optimal for clustering algorithms as most of the data is clustered around just one point.

**Our Solution:-**

Because of issues with mean value, we used trimmed mean to solve this issue.

Trimmed mean- A trimmed mean is a method of averaging that removes a small-designated percentage of the largest and smallest values before calculating the mean. It is calculated including only 10 percentile to 90 percentile of the original data, and for all users whose check in count was below the mean, we set the value to 0, and if the value was greater than the mean, we set the value to 1.

- ## System Design

- ## System Elements

| System Elements | Details |
|---|---|
| Designing Tool | Rstudio Version 1.0.44 |
| Programming Language | R |
| Dataset Format | CSV files |

- ## High Level Design

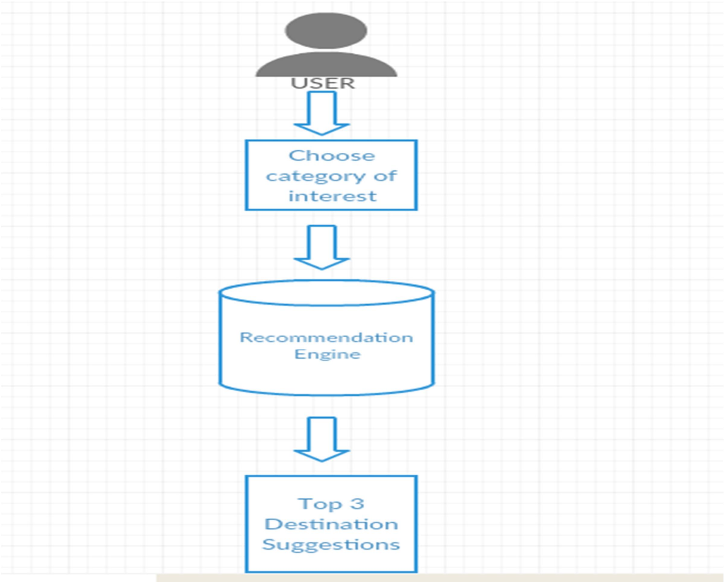


**Fig. 1 Destination Recommendation System**

- ## Algorithms

- ## Clustering

### I. SSE- Sum of Squared Error

We were not sure about the number of clusters to divide the users when forming user clusters. In order to determine the proper number of clusters, we used sum squared error (SSE) in relation to the number of clusters. We discovered best relationship between these variables with 7 clusters after modeling this relationship.
The squared error is defined as the sum of the squared Euclidean distances between each element in the cluster and the cluster centroid $C_k$.

The squared error is defined as

$$se_{K_i} = \sum_{j=1}^{m} \| t_{ij} - C_k \|^2$$

Given a set of clusters K= $\{K_1, K_2, \ldots K_n\}$, the squared error for K is defined as

$$se_K = \sum_{j=1}^{k} se_{K_j}$$

### II. K-means:

We clustered the users using different algorithms. We discovered that, **K-Means** was the most effective algorithm. K-means is an iterative clustering algorithm. Items are moved among set of clusters until the desired set is reached. We decided to form our clusters using K-means algorithm for the final recommendation system. K means algorithm was suitable to our needs and gave a number of advantages over other algorithms like less time complexity, well defined non overlapping clusters.

The cluster mean of $K_i = \{t_{i1}, t_{i2}, \ldots\ldots t_{im}\}$ is defined as

$$m_i = \frac{1}{m} \sum_{j=1}^{m} t_{ij}$$

- ## Classification

After the users were assigned to a cluster, the cluster number was added as an additional attribute to the user data. The decision tree is most utile algorithm in classification problems.

### Decision Tree

With this approach, a tree is constructed to model the classification process. The 2 basic steps of this algorithm are: building the tree and applying the tree to the dataset. A decision tree classifier was built using the cluster number as the class labels for classification. This model assigns a class label to a test user which is the same as the class label for the most similar users. This will be helpful in reducing the search space when searching for similar users.

- ## Recommendation System

The last phase of the system is recommendation system. Recommendation System helps to enhance our decision making capabilities. These decisions could be towards anything varying from choosing from selection of books to movies to any products.

There are two inputs to this system: the user number and the broad category of venues in which the user wants a recommendation.

A list of all the venues in the dataset is checked sequentially and a frequency value is associated with each venue. This frequency value is defined as the count of visits to that venue in the original check in dataset. With the help of the decision tree created from the training dataset, the test user is first classified to identify the user group that the test user belongs to. Cosine similarity measure is used to identify three users that are most similar to the test user.

The list of venues in the category that has been requested by test user is then checked sequentially in order to recommend the user. This list contains the venues visited by the similar users. Based on the frequencies of visits, the top three venues from this list are then extracted for recommendation.

It is possible that the list of recommendations has less than 3 venues that is if the test user went to all the venues in that category that the similar users went to or if any of the similar users have not been to any venue in that category. In such case, first, search visits the original list of venues and then the top venue which is not in the list of recommendation as well as not visited by the test user are then added to the list of the recommendations.

At the end of this process, the list of recommendations has 3 venue IDs which represent venues in the category chosen by the user. Each of the 3 venue IDs are used to make three calls to the Foursquare API. The API resolves the venue IDs and returns the details which are then displayed to the user.

- **Output**

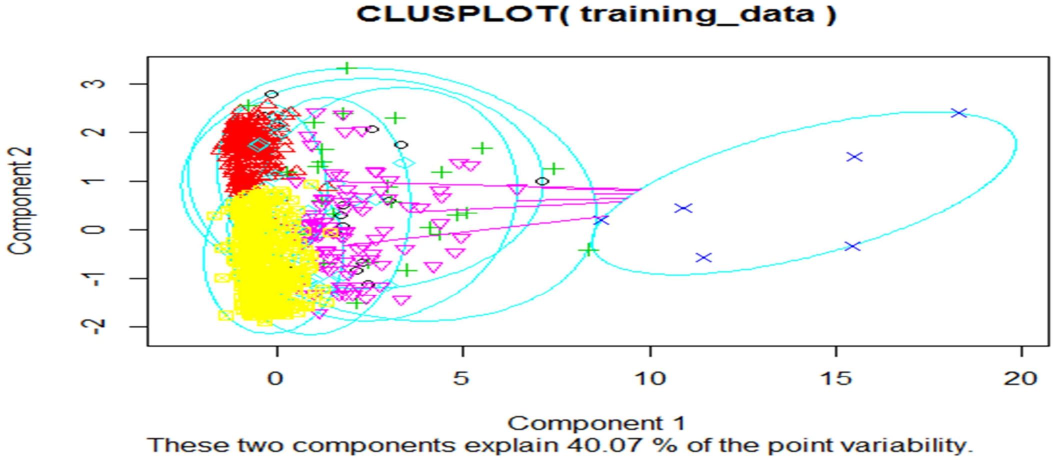
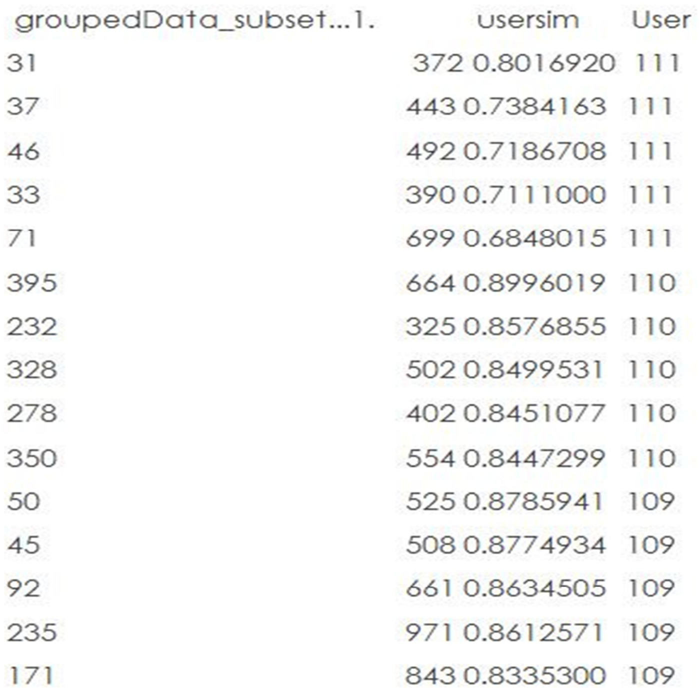
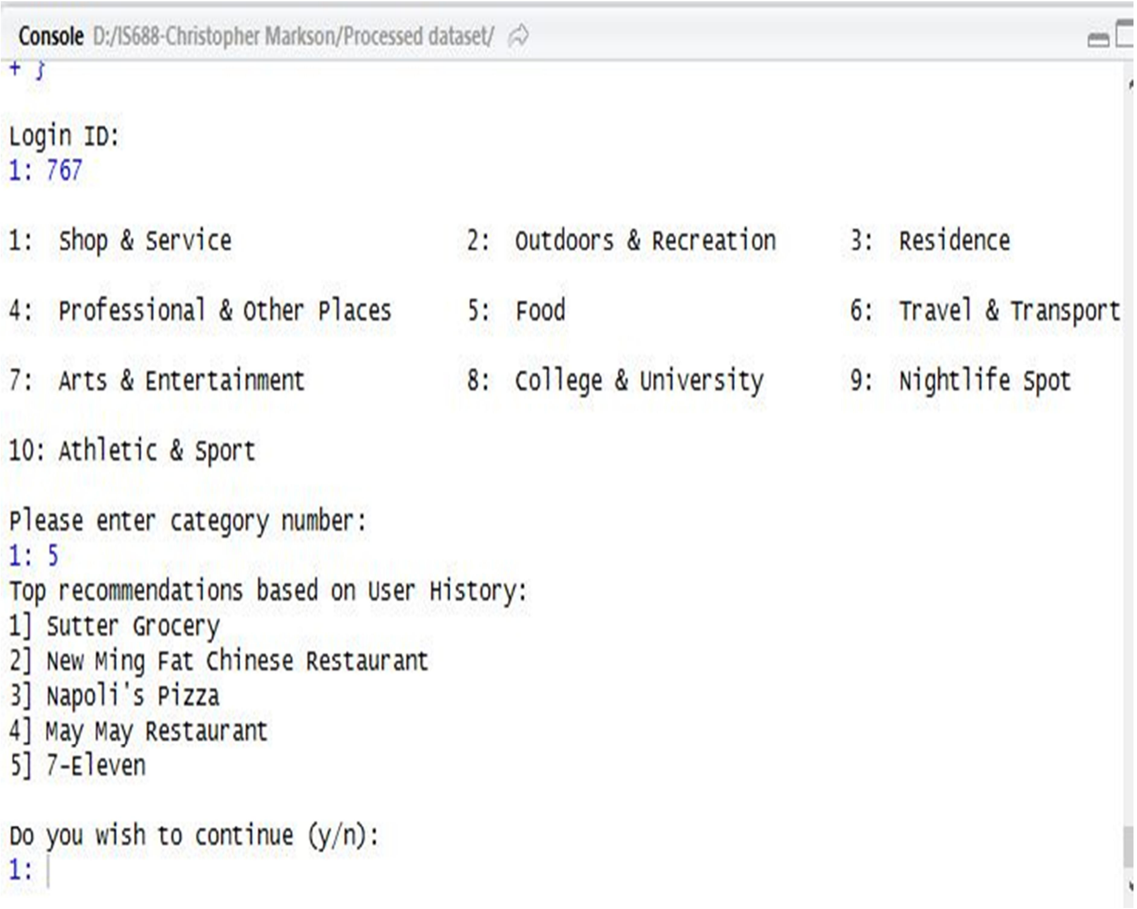


**Fig. 2 Cluster Plot for training data**



```
groupedData_subset...1.        usersim     User
31                          372 0.8016920  111
37                          443 0.7384163  111
46                          492 0.7186708  111
33                          390 0.7111000  111
71                          699 0.6848015  111
395                         664 0.8996019  110
232                         325 0.8576855  110
328                         502 0.8499531  110
278                         402 0.8451077  110
350                         554 0.8447299  110
50                          525 0.8785941  109
45                          508 0.8774934  109
92                          661 0.8634505  109
235                         971 0.8612571  109
171                         843 0.8335300  109
```

**Fig. 3 Cosine Similarity of the Top Users**



```
+ }

Login ID:
1: 767

1:  Shop & Service              2:  Outdoors & Recreation      3:  Residence

4:  Professional & Other Places       5:  Food                   6:  Travel & Transport

7:  Arts & Entertainment          8:  College & University       9:  Nightlife Spot

10: Athletic & Sport

Please enter category number:
1: 5
Top recommendations based on User History:
1] Sutter Grocery
2] New Ming Fat Chinese Restaurant
3] Napoli's Pizza
4] May May Restaurant
5] 7-Eleven

Do you wish to continue (y/n):
1: 
```

**Fig. 4 Recommendation of the Venues**

## Conclusion

Recommender systems provide users with personalized content and services by filtering, prioritizing and efficiently delivering relevant information in order to alleviate the problem of information overload.

In spite of significant progress in the research community, and industry efforts to bring the benefits of new techniques to end-users, there are still important gaps that make personalization and adaptation difficult for users.

Here, a recommendation system has been implemented based on clustering and classification for helping users discover new places of interest in different categories based on similar users' habits. We are essentially predicting the user's likeability of a place the user has never visited.

The clustering algorithm employed was k means since it practically works well, the clustering results are easily interpretable, and it is fast and efficient in terms of computational cost, typically O(K*n*d).

For classification using the decision tree, cosine similarity measure is used to identify three users that are most similar to the test user and a list containing the venues visited by the similar users is generated, out of which the top three venues are then extracted for recommendation. Advantages of the decision tree model lie is its transparent nature, specificity, and comprehensiveness. The decision tree makes explicit all possible alternatives and traces each alternative to its conclusion, allowing for easy comparison. Its ability to assign specific values to problem, decisions, and outcomes of each decision reduces ambiguity in decision-making. A decision tree also allows for partitioning data in a much deeper level, not as easily achieved with other decision-making classifiers such as logistic regression or support of vector machines.

Finally, a call to the Foursquare API referencing the venue IDs is made to retrieve the details displayed to the user.

## Future Work

Storing the past history of results for each users allows us to build context for future predictions, thereby improving and dynamically updating the result set for every subsequent prediction. This is one of the enhancements we wish to incorporate.

We also plan to improve the effectiveness and performance by implementing a different algorithm on different user segments – based on their newness to the system, and previously available data.

We propose to use social relationship between users to identify their neighborhoods.

Another extension is to assign users to several clusters (overlapping clusters) by using an algorithm other than k means for clustering and use the opinion of these clusters to generate recommendations. Overlapping clusters could depict the real-world situations where users participate in different communities. That would further improve our recommendation system.

## References

1. Data Mining Introductory and Advanced Topics, Margaret H. Dunham
2. https://archive.org/details/201309_foursquare_dataset_umn
3. Mohamed Sarwat, Justin J. Levandoski, Ahmed Eldawy, and Mohamed F. Mokbel. LARS*: A Scalable and Efficient Location-Aware Recommender System, IEEE Transactions on Knowledge and Data Engineering TKDE
4. https://en.wikipedia.org/wiki/Recommender_system
5. https://web.stanford.edu/class/cs345a/slides/12-clustering.pdf
6. https://en.wikipedia.org/wiki/Decision_tree_learning
7. http://infolab.stanford.edu/~ullman/mmds/ch9.pdf