INST737

Digging Into Data

Project Report:

Yelp Academic Dataset

Submitted By:

Srikanth Jaikumar(sri14@umd.edu)

Utkarsha Devkar(utkarshadevkar@yahoo.in)

Instructor:

Dr. Patrice Seyed

INTRODUCTION

Yelp is an organization that publishes reviews on a variety of business establishments. Recently, they decided to make the data they have accumulated through their platform open to the public in the form of an challenge to come up with new insights. The reason behind choosing this dataset lies in the size of the data being so large so as to represent an idea data mining problem. The dataset contains over 2.5M reviews and close to 1M tips. The data has been collected from 686K user and represents close to 171K businesses that range from restaurants, gas stations, medical facilities etc. A dataset of this size can be effectively made use of to learn the skills vital for deriving insights out of the data. Geographically, the dataset is inclusive of information collected from business establishments located in UK, Germany, Canada and U.S.. There are a number of categorical attributes that are present in the dataset ranging from good for attributes to hours open and parking availability.

A number of academic papers have also been written using this dataset. The same can be downloaded using the below link.

https://www.yelp.com/dataset_challenge/dataset

OBJECTIVES

The main objectives of this project are:-

1) Exploration of data for pre-processing and identification of research questions
2) Calculation of accuracy and identification of features that would act as significant predictors to predict the rating of a restaurant as “Satisfied” or “Unsatisfied”
3) Application of User Based Collaborative Filtering Algorithm on a filtered dataset in order to predict the rating that would be provided by a user for a restaurant

DATA PREPROCESSING

The original datasets were downloaded from the ‘Yelp Dataset Challenge round 8’ available at https://www.yelp.com/dataset_challenge/dataset. It consisted of five files in JSON format which contained data with respect to business, user, review, tip and check-in. First, the JSON files were converted into CSV format using Python. Since the business dataset consisted of various types of establishments including restaurants, we only considered the ‘Restaurants’ by extracting the “restaurant” keyword from all the tags given to a specific business.

Since the dataset size was extremely large, we also created small chunks of the ‘Review’ dataset to facilitate our initial analysis and then combined them all for final analysis. Also, certain columns from various datasets were removed while preprocessing, which would help increase the speed of execution in R.

DESCRIPTIVE STATISTICS AND VISUALIZATIONS

Before our analysis to answer the research questions, we explored the datasets for find any patterns and visualized the same. Following are the visualizations implemented in R programing:

1. Frequency Distribution of Review Count

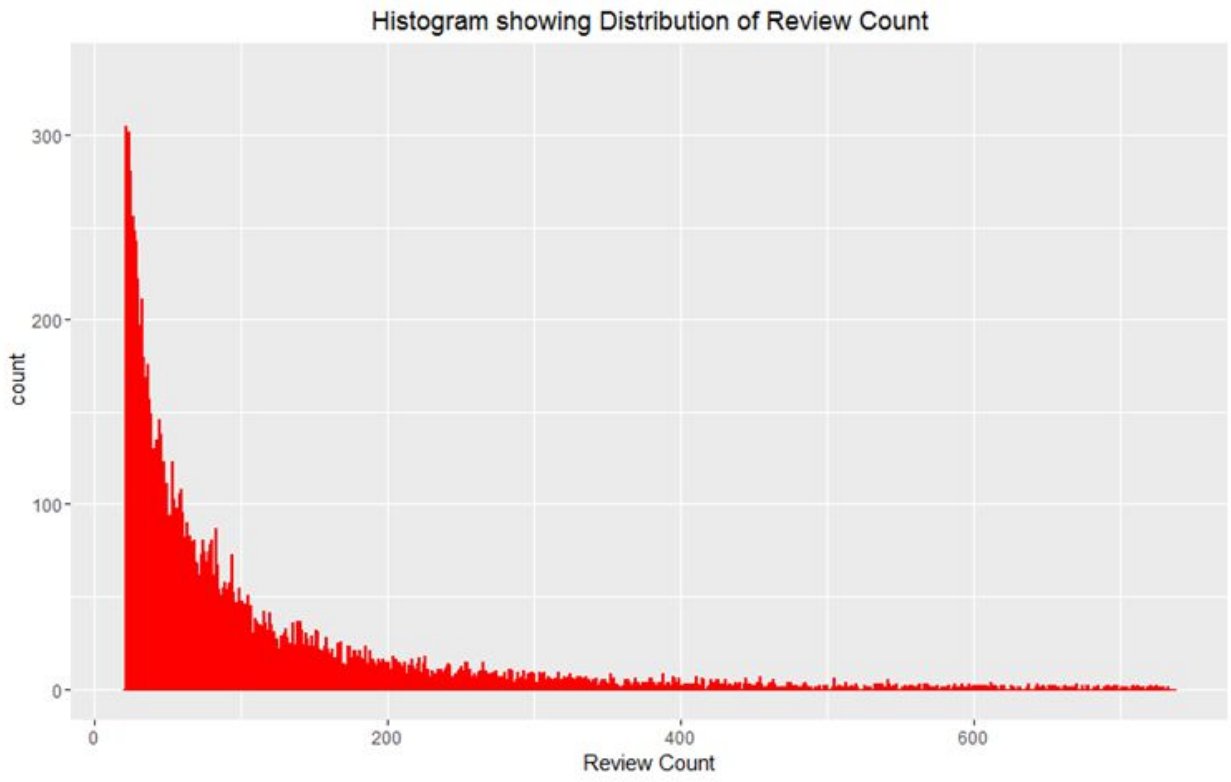
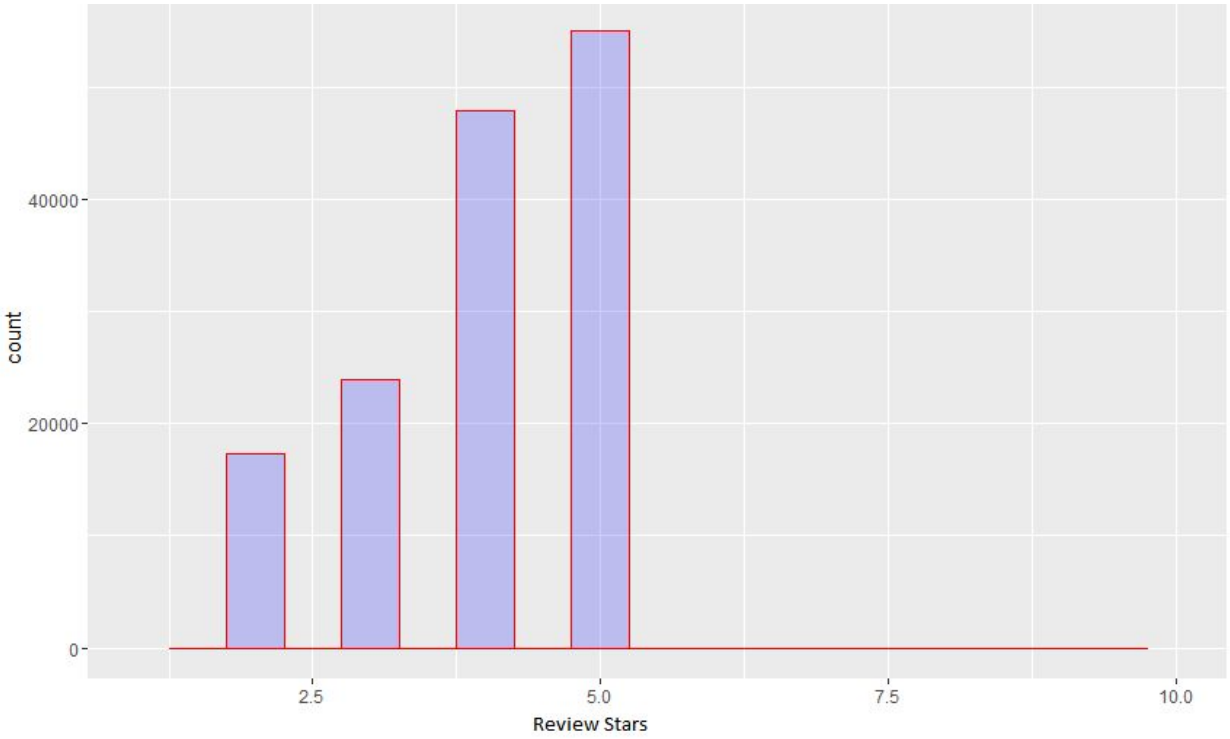


Fig. 1

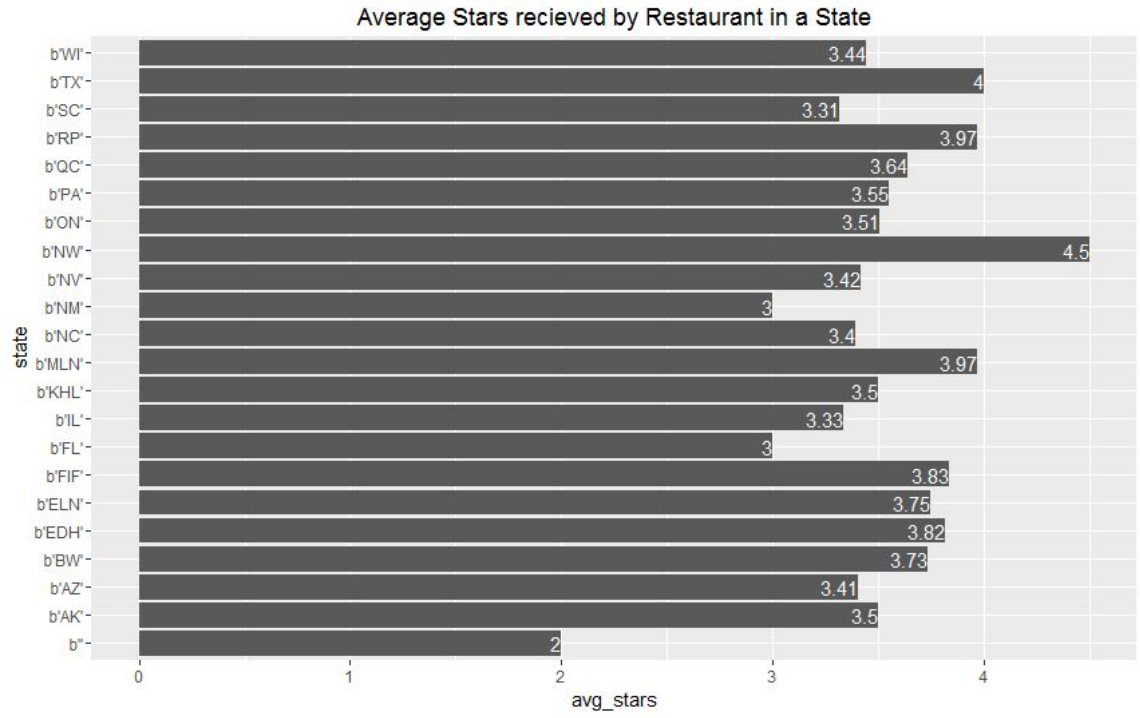
This is the histogram of the number of 'review_count' received by a restaurant. It can be observed that most restaurants have the total review count between 0 to 150 whereas very few restaurants have a total of review_counts above 400.

2. Distribution of 'Review Stars'



The above graph was implemented to view the distribution of the general rating/stars achieved by restaurants. It can be observed that a very high number of restaurants have received stars above 3.5.

3. State-wise comparison of 'average_stars' received



This visualization was implemented to explore if there was any association with the geographic location of the restaurant with respect to its State and the average stars/rating received by the restaurants in that particular State.

PREDICTING USER-SATISFACTION

One of the objective of our project was to answer the question: *"What factors would help predict the user-rating received by the restaurant?"*

This helps in predicting whether a given user is going to give a satisfied or unsatisfied rating to a specific business. The target variable used for this is : review_stars.

This variable was first modified into a dichotomous dependant variable as : 'Satisfied' and 'Unsatisfied', where all the values(stars) above 3.5 were categorized as **'Satisfied'** and all below 3.5 stars as **'Unsatisfied'**.

In case of the datasets, three datasets were mainly used. They were the 'Business', 'User' and the 'Review' dataset.

The three objects as as below:

business

```
{
   'type': 'business',
   'business_id': (encrypted business id),
   'name': (business name),
   'neighborhoods': [(hood names)],
   'full_address': (localized address),
   'city': (city),
   'state': (state),
   'latitude': latitude,
   'longitude': longitude,
   'stars': (star rating, rounded to half-stars),
   'review_count': review count,
   'categories': [(localized category names)]
   'open': True / False (corresponds to closed, not business hours),
   'hours': {
      (day_of_week): {
         'open': (HH:MM),
         'close': (HH:MM)
      },
      ...
   },
```

```
    'attributes': {
        (attribute_name): (attribute_value),
        ...
    },
}
```

review

```
{
    'type': 'review',
    'business_id': (encrypted business id),
    'user_id': (encrypted user id),
    'stars': (star rating, rounded to half-stars),
    'text': (review text),
    'date': (date, formatted like '2012-03-14'),
    'votes': {(vote type): (count)},
}
```

user

```
{
    'type': 'user',
    'user_id': (encrypted user id),
    'name': (first name),
    'review_count': (review count),
    'average_stars': (floating point average, like 4.31),
    'votes': {(vote type): (count)},
    'friends': [(friend user_ids)],
    'elite': [(years_elite)],
    'yelping_since': (date, formatted like '2012-03'),
    'compliments': {
        (compliment_type): (num_compliments_of_this_type),
        ...
    },
    'fans': (num_fans),
}
```

First, in order to focus on the restaurants, only the ‘Restaurant’ category of all the business establishments was subsetted.

We also implemented some feature engineering before developing the final models. This included computing additional features namely:

- Number of friends: This refers to the number of friends the user has on Yelp obtained from the User dataset
- The compliments score: The combined score of all complement related attributes from User dataset
- Votes_score: The combined score of all votes related attributes from the User dataset namely votes.useful, votes.cool and votes.funny
- Good_for_score: This variable is aimed towards summarizing all the values given for Good for attributes such as good for lunch, good for kids, good for group obtained from the business dataset
- Park_score: This variable is aimed towards summarizing all the values related to parking attributes for a business

The three datasets were then merged by joining each review with its corresponding business based on business_id attribute and joining each review with its corresponding user based on user_id attribute.

In order to run the classification models, the merged dataset was split into 75% train set and 25% test set while stratified sampling of Caret is adopted to preserve the distribution of the outcome in the training and test sets, based on the distribution of the target variable on different classes. The three classification models which are implemented include logistic regression, decision trees and SVM.

The average accuracy obtained by running approximately 10 tests with different combination of independent variables is tabulated as below:

Model	**Average Accuracy**
Logistic regression	0.73
Decision Trees	0.72
SVM	0.72

Thus, the 'Logistic Regression' model was finalized since it gave better accuracy. The best model we have is the feature combination of business stars + review count of user + review count of business + average stars of user + number of fans + number of friends + WiFi availability with an accuracy of 74.2%.

A description of the Independent variables that provided the best prediction are :-

Business stars: This is the rating that the business has already been provided found in the business dataset. Please note that the rating we make use of as our DV is derived from the reviews dataset

Review count of user: The number of times the business has been reviewed by the user

Review count of business: The total number of times the business itself has been reviewed

Review count of user: The total number of reviews that have been provided by the user

Average stars of user: The average rating that has been provided by a user found in the User dataset

Number of fans: The number of fans the user has found in the User dataset

Number of friends: The number of friends the user has found in the User dataset

WiFi: Provides whether a business establishment is equipped with Wi-Fi connectivity or not

USER-BASED COLLABORATIVE FILTERING

The main idea behind User Based Collaborative Filtering(UBCF) is to predict the rating that would be provided by a user to a restaurant that he/she has not visited based on the ratings that have been provided by other users who have visited the restaurant. The similarity between users is defined by the differences of the ratings they've given to the same businesses.

Assume that there are 'n' users and 'm' businesses in a dataset. We built a n*m matrix 'M' which would use the row index *i* to indicate *ith* user, column index *j* to indicate *jth* business and the value M[i][j] for the rating as shown below. For each record, there will be some cells in the matrix that would be filled with a rating indicating that it has been given a rating and some cells are filled with blanks or NA's indicating the absence of rating.

	b1	b2	b3	b4	b5
u1	?	4.0	2.0	1.0	2.0
u2	3.0	?	?	5.0	4.0
u3	?	4.0	3.5	3.0	?
u4	5.0	?	4.0	?	2.0
u5	2.0	3.0	3.0	4.0	?
u6	?	?	4.0	4.5	3.5

The questions marks that you see in the above matrix are the fields that we would like to predict using cosine similarity. In order to run this algorithm, we made use of "Recommenderlab" library to conduct UBCF on the Yelp dataset.

Due to the large volume of users and businesses in our dataset, the initial matrix built comprised of a larger number of rows and columns corresponding to users and businesses. Running the UBCF on such a large matrix was computationally complex and we also incurred the risk of affecting the accuracy of the prediction due to a large number of NA's.

In lieu of these problems, we filtered the data(number of users and businesses) that would be provided as input for UBCF. We set a threshold on the number of reviews given by users in the dataset and also on the number of reviews that have been provided to a business. Moreover, we also filtered the business dataset comprising only of restaurants to only select those businesses that had a rating above or equal to 4. In this manner we increased our odds of predicting the rating for a restaurant containing higher number of review count in both business and user datasets.

TOOLS USED

Python and iPython in Anacondas

R: Recommenderlab, carat, cwhmisc, e1071, plyr, rpart, metrics, stringr, ggplot2

WORK ALLOCATION

Data preprocess: Srikanth Jaikumar & Utkarsha Devkar
Descriptive statistics and Visualizations: Srikanth Jaikumar & Utkarsha Devkar
Factors to predict the rating of a restaurant: Srikanth Jaikumar & Utkarsha Devkar
User Based Collaborative Filtering: Srikanth Jaikumar & Utkarsha Devkar
Report: Srikanth Jaikumar & Utkarsha Devkar
Presentation: Srikanth Jaikumar & Utkarsha Devkar