Yelp Data Analysis

Yuhang Lan: ylan27@wisc.edu

Xici Luan: xluan5@wisc.edu

Bi Qing Teng: bteng2@wisc.edu

Hongwei Pan: hpan55@wisc.edu

Introduction

The main focus of this Yelp Data Analysis is to propose data-driven, actionable decisions to all business owner in the fast food industry to improve their ratings on Yelp. The whole project mainly consist of the following parts:

1. Provide recommendations on specific business attributes:
    - Preprocessing on the business data.
    - Applying ANOVA for testing different attributes' statistical difference.
    - Using histogram to staightforwardly compare the scaled ratings.
1. Provide recommendations based on the reviews:
    - Preprocessing the text data.
    - Drawing WordCloud to intuitively show the frequently mentioned useful words.
    - Establishing Doc2Vec model for finding similar important words.
    - Setting reasonable rules and verifying the significance of these words in determining the ratings.
    - Using histogram to staightforwardly compare the scaled ratings.
2. Provide personalized suggesions for all fast food resturants owners:
    - Provide recommendation based on the business attributes they provided.
    - Extract important features from their reviews and provide suggestions on it.

Business Attributes Analysis

Pre-processing Business Data

In this section, the business attributes for each business and the corresponding star rating from *business.json* are used for analyses. There are 16,541 business IDs and the 10 business attributes studied include **Business Parking**, **Restaurants Delivery**, **Restaurants Reservations**, **Outdoor Seating**, **Noise Level**, **Restaurants Take Out**, **Restaurants Price Range**, **WiFi**, **Bike Parking** and **Restaurants Good For Groups**. The proportions of missing values in this subset of variables are calculated and there is no missing value for star rating. Overall, there is approximately 21% of missing values in the 10 business attributes and the following table shows the proportion of missing values for each attribute. Each attribute has < 35% of missing values.

Business Parking	Restaurants Delivery	Restaurants Reservations	Outdoor Seating	Noise Level	Restaurants Take Out	Price Range	WiFi	Bike Parking	Good For Groups
0.249	0.121	0.120	0.205	0.330	0.098	0.141	0.335	0.344	0.127

Since some businesses have missing attributes, the star ratings between businesses with no missing attribute and businesses with at least n (n = 1,..,10) missing attributes are compared using two-sample t-test. The significant p-values from the corresponding test suggest that there is a difference in star ratings.

The average star rating for businesses with less than n missing values in business attributes and at least n missing values in business attributes are also compared using Welch two-sample t-test since Levene's Test shows that the two samples have unequal variances (p-value < 2.2×10^{-16}). The significant p-values from the corresponding test suggest that there is a difference in star ratings. An example of Welch two-sample t-test results for bussiness with less than 5 missing values in business attributes and at least 5 missing values in business attributes is as shown below.

Thus, business owners are advised to fill in as much information as possible.

```
        Welch Two Sample t-test

data:  biz_fast_food$stars[which(missing_each_business < 5)] and biz_fast_food$stars[which(missing_each_business >= 5)]
t = 13.344, df = 3956.7, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.2293808 0.3083965
sample estimates:
mean of x mean of y
 3.162266  2.893377
```

Before carrying out the analysis, the missing attributes are imputed using *rpart* with option *method* = *class* since the business attributes are factor levels. The codes for imputation can be found in **biz_fastfood_final.R**.

ANOVA Test

An ANOVA model is fitted for star ratings on the business attributes and the results are as follows. The p-values of the two attributes **RestaurantsTakeOut** and **WiFi** are not significant. Removing **WiFi** from the model gives a non-significant p-value for **RestaurantsTakeOut**. However, a comparison between full model and reduced model shows that the full model is a better fit (p-value for the test is 0.000137).

```
Analysis of Variance Table

Response: stars
                          Df  Sum Sq Mean Sq   F value    Pr(>F)
BusinessParking            1  1159.6 1159.65 1734.2671 < 2.2e-16 ***
OutdoorSeating             1    95.7   95.74  143.1787 < 2.2e-16 ***
NoiseLevel                 3   135.7   45.23   67.6471 < 2.2e-16 ***
RestaurantsTakeOut         1     0.6    0.65    0.9667    0.3255
RestaurantsPriceRange2     3   129.2   43.08   64.4212 < 2.2e-16 ***
WiFi                       2     0.8    0.41    0.6119    0.5423
BikeParking                1    20.7   20.72   30.9823 2.644e-08 ***
RestaurantsGoodForGroups   1    17.7   17.74   26.5349 2.618e-07 ***
RestaurantsDelivery        1    69.8   69.81  104.4020 < 2.2e-16 ***
RestaurantsReservations    1   271.9  271.87  406.5838 < 2.2e-16 ***
Residuals              16525 11049.7    0.67
```

Histograms Comparing Scaled Ratings

The proportion of star ratings for each business attribute is plotted. The codes for these plots and the tests done on business attributes can be found in **biz_fastfood_plots_imputed.R**, the corresponding plots can be found in the file **Figures/BusinessAttributes**. One example of the plot is shown below. Proportion is used as y-axis because the business attributes data is highly imbalance and proportion can more accurately reflect the relationship between star ratings and whether to include this attribute or not. i.e. Proportion of "False" for 1-star is calculated as dividing *number of False in 1-star businesses* by *number of 1-star businesses*.

From the following plot for **Business Parking**, the "True" proportion for star ratings 4 and 5 are higher than the "False" proportion while the "False" proportion for star ratings 1 and 2 are much higher than the "True" proportion. For star rating 3, the "True" and "False proportions are similar.

The next table shows the average star ratings for "True" and "False". The average ratings for "True" and "False" differ by about 0.7. Levene's Test shows that the two samples have unequal variances (p-value < 2.2×10^{-16}). A Welch two-sample t-test is done on the star ratings for "True" and "False". The results show that there is a difference in the average star ratings between "True" and "False" (p-value < 2.2×10^{-16}). This suggests that businesses which provide business parkings do have a higher star rating than businesses without business parkings.

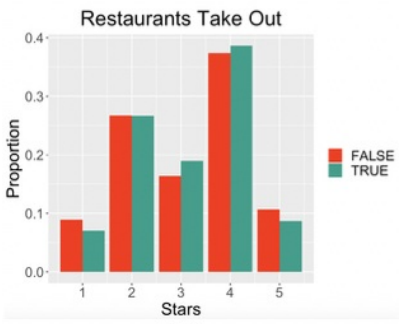
The plots and two-sample t-test results are similar for the attributes **Restaurants Delivery**, **Restaurants Reservations**, **Outdoor Seating**, **Noise Level**, **Restaurants Price Range**, **Bike Parking** and **Restaurants Good For Groups**.

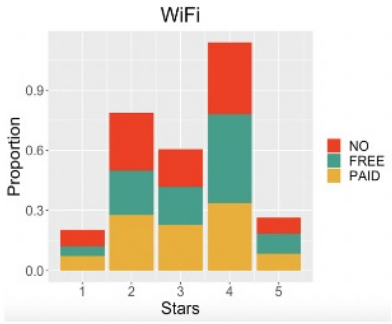
As mentioned earlier, **Restaurants Take Out** and **WiFi** do not have a significant effect on star rating. The following plot shows the "True" and "False" proportions for **Restaurants Take Out**. As shown, the "True" and "False" proportions are similar across all ratings. The average star ratings for "True" and "False" in the next table are similar. Levene's Test shows that the two samples have equal variances (p-value = 0.07023). A two-sample t-test is done on the star ratings for "True" and "False". The results show that there is no difference in the average star ratings between "True" and "False" (p-value = 0.8811).

There are three levels in **WiFi** - "No", "Free" and "Paid". The following plot shows the corresponsing proportions for **WiFi**. As shown, the "No" and "Paid" proportions are similar across all ratings. The proportion of "Free" is higher in ratings 4 and 5 than in ratings 1 and 2 while the "No" and "Paid" proportions are higher in ratings 1 and 2 than in ratings 4 and 5.

The average star ratings for "No" and "Paid" in the next table are similar and are a little different from the average star rating for "Free". Tukey HSD test below shows that there is a significant difference in average rating between"No" and "Free".

These results show that all business attributes except **WiFi** and **Restaurants Take Out** have an effect on star rating, which tally with the results from the previous section.





Business Parking	Average Star Rating
False	2.865
True	3.531

```
        Welch Two Sample t-test

data:  biz_fastfood$stars[BP_false_index] and biz_fastfood$stars[BP_true_index]
t = -40.76, df = 16510, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.6979758 -0.6339257
sample estimates:
mean of x mean of y
 2.864787  3.530738
```

Restaurants Take Out	Average Star Rating
False	3.142
True	3.152

```
        Two Sample t-test

data:  biz_fastfood$stars[TO_false_index] and biz_fastfood$stars[TO_true_index]
t = -0.1496, df = 16539, p-value = 0.8811
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1425967  0.1223741
sample estimates:
mean of x mean of y
 3.142349  3.152460
```

WiFi	Average Star Rating
No	3.072
Free	3.319
Paid	3.084

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = biz_fastfood$stars ~ biz_fastfood$WiFi)

$`biz_fastfood$WiFi`
                diff        lwr         upr     p adj
NO-FREE   -0.24727688 -0.2908234 -0.20373035 0.0000000
PAID-FREE -0.23490273 -0.5246037  0.05479828 0.1385240
PAID-NO    0.01237414 -0.2761871  0.30093543 0.9944457
```

Review Analysis

Preprocessing Text Data

Text Data Preprocessing

This involves the following procedures:

1. Change all letters to lowercase
2. Use regular expression to modify some words (e.g. change *mustn't* to *must not*)
3. Remove all punctuations
4. Remove all stopping words (e.g. *I*, *you*, *food*, *etc*)
5. Normalize the verb and noun (e.g. change *eating* to *eat* and *birds* to *bird*)

Vectorizing Text Data - *TF-IDF*

The words needed to be encoded as vectors of integers to use as input to an algorithm. We used TF-IDF to achieve this goal. TF-IDF stands for “Term Frequency – Inverse Document Frequency”, which are the word frequency scores that try to highlight words that are more important.

Term Frequency: This summarizes how often a given word appears within a document.

Inverse Document Frequency: This downscales words that appear a lot across documents.

Basically this method reduced values of common word that are used in different documents. And we used TfidfVectorizer to finish vectorizing texts.

Feature extraction

Most machine learning algorithms can't take in straight text, so we will create a matrix of numerical values to represent our text. As specific word has meaning but when we combine group of words the meaning might get changed, we should also decide on which n-gram to use. Finally, unigram should be the choice by taking into consideration of computational efficiency of algorithm and interpretability of the model.

Tackling Missing Data

For the review data set, there are two kinds of varibales, the star ratings of the restruants and the vectorized reviews.

1. Missing on ratings
    - There is no way we can imputate on people's reviews. In this case we delete this data in the review analysis part.
2. Missing on response varible
    - We establish a simple Naive Bayes model based on the data with no missing value. The response variable is the star ratings and the predictors are the vectorized reviews. The training accuracy and test accuracy of the model is 61% and 63%, indicating that our that our model does not suffer from overfitting. The relatively low accuracy of this model is due to the fact that the number of our target catogory is 5, increasing the difficulty of our model to precisely put one review to the respective rating category. Apart from this, this model makes sense and could be used to fill out the missing values with missing value on star ratings.

Doc2Vec Model

The principle of Doc2Vec Model can be summarized according to *Le and Mikolov(2014)*, "every paragraph is mapped to a unique vector, represented by a column in matrix D and every word is also mapped to a unique vector, represented by a column in matrix W. The paragraph vector and word vectors are averaged or concatenated to predict the next word in a context… The paragraph token can be thought of as another word. It acts as a memory that remembers what is missing from the current context — or the topic of the paragraph."

More specifically, we apply the DMC(Distributed Memory Concatenation) model, where the paragraph vectors are obtained by training a neural network on the task of inferring a center word based on context words and a context paragraph. More importantly, the DMC model has the inner function **most_similar**. That is, after training over a DMC model (*model_dmc*), if we query for *model_dmc.is_similar('good')* it will return words including perfect, terrific, incredible, etc. This is an extremely useful feature because for instance if we find the word 'avocado' important for getting high reviews, by finding its similar words (according to the result: guacamole, pinapple, etc.), we can further suggests the owners also includes these similar gradients to their menus.

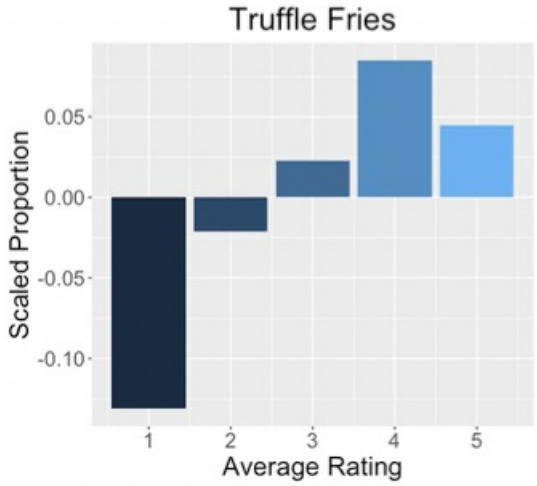
Rules for Determining Words' Significance

After find some key words, the next thing we should do is to set up rules to determine whether this word is significant in terms of improving/decreasing resturants' ratings. Here, we set up two rules:

1. The proportion of the positive ratings is 0.1 larger than the proportion of positive ratings in the review data set.
2. The proportion of the positive words nearby compared with the negative words nearby is 80% larger.

Histograms Comparing Scaled Ratings

Here we use the example of "truffle fries" to verify that it is important in positive ratings and therefore resturants are recommended to include it in their menu. The y-axis the plot is the proportion of the rating with the words "truffle fries" minus the proportion of the rating given the whole review data set. According to the plot, the proportion of the 3, 4 and 5 ratings are all larger while that of 1 and 2 ratings are smaller, indicating the positivity of "truffle fries". What's more, after finding both the positive and negative words nearby, the proportion of positive words is 95%, further indicating that business owners are adviced to include this in their menu for higher ratings.



# of positive words nearby (good, great etc.)	# of negative words nearby (bad, terrible etc.)
2149	101

Provide recommendation based on the business attributes they provided

We checked if there were any missing values in the attributes. Based on our findings, reducing missing values would significantly improve the ratings of fast food restaurant. If so, we encourage business owners to fill out their attributes on Yelp. And within the attributes we researched, we found to improve the quality of attributes would also result in better ratings. For example, the ratings of stores with free WiFi were significantly higher than that of stores with paid WiFi. Therefore, if a resturant does not provide free WiFi, we would suggest it to include free Wifi in their servie. Based on these results, We encourage business owners to refine their attributes as much as possible.

Extract important features from their reviews and provide suggestions on it

This process involves extracting important information from the reviews of each fast food restaurant and gave them specific advice.

To achieve this goal, we first decided on a list of keywords. This list contained words from several aspects, including service (e.g. waiting time, staff friendlinss, etc.), environment (e.g. cleanness of the restroom, table, etc), flavor of food (e.g. salty, oily, etc). Next, we calculated the proportion of the appearance of these words in the positive and negative reviews. Basically, we calculated the frequency scores of these specific words. For example, when calculating the score of "filthy", which indicating sanitation problem of a restaurant, we counted how many times this word appear in a negative review and adjusted the score by the length of the review. Adding up all the scores in a negative reviews would result in our final score. Sometimes a word appears in the positive reviews, and we would subtract some scores from the total score of a specific word (e.g. "clean"). As a matter of fact, the higher the proportion score was, the more serious the problem was with that specific restaurant. By setting some threshold on the proportion scores, we confirmed the severity of the problems and gave specific suggestions in our APP.

Advantages and Drawbacks

Advantages

1. The recommendations we give are feasible and actionable.
2. For verifying the significance of certain attributes or words, we introduce different methods, guaranteeing the accuracy of our measurement.
3. Taking into consideration different people's rating preference and data imbalance when comparing ratings.
4. The results are clearly plotted and highly readable.
5. The Shiny APP we create guarantees great user experience.

Drawbacks and Future Improvement

1. For predicting ratings, the prediction accuracy could increase. This can be solved by extracting or including more important features (e.g. length of the text) and using more advanced models.
2. The Doc2Vec model can only find similar uni-gram word of the given word(s).

Contributions

Yuhang Lan: Drawing barplots and calculation of more attributes

Xici Luan: Preprocessing data and extracting features

Bi Qing Teng: Imputation and analyses on business attributes and visualizations

Hongwei Pan: Drawing wordcould plots and creating shiny app