STAT 628 Module 3 Summary

Group members: YICEN LIU, HUA TONG, ENZE WANG

Introduction and Background

Nowadays, customer reviews play a pivotal role in doing business. More and more business owners are trying to get ideas from reviews to improve the quality of their products and services. It is easy for humans to understand the sentiment behind the reviews, while this work is difficult for machines. It would be helpful if there exists an algorithm which could extract useful information from the comments.

Yelp is an Internet company which provides a platform for users to post reviews of businesses. For this project, we will analyze a large number of reviews for **pizza** business on Yelp and provide helpful suggestions to business owners in order to improve their ratings on Yelp. Welcome to our main code on Github: `https://github.com/sai-liu/628Yelp`

Data Cleaning

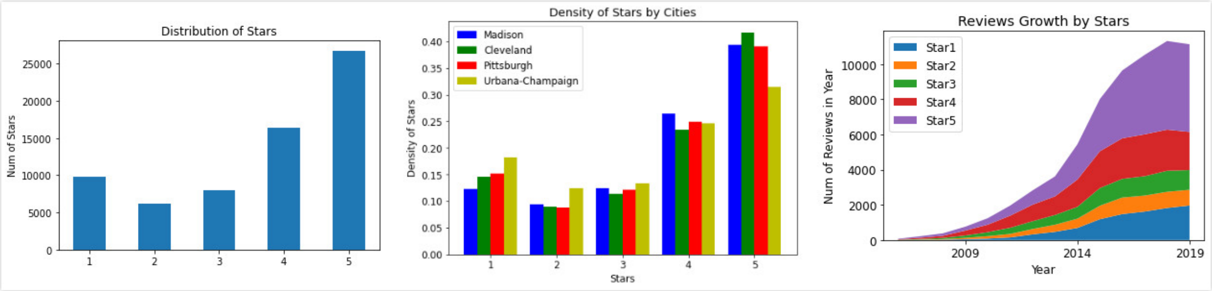
The original data are four JSON files, so we first write code to transform them into CSV files, then clean the four datasets successively.

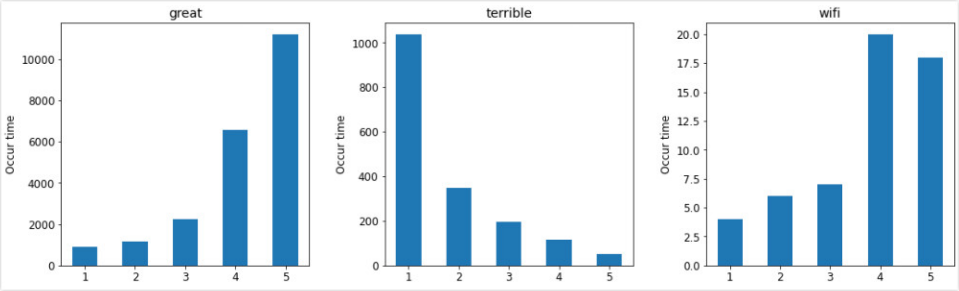
We notice that our data should be from four places: Madison, Cleveland, Pittsburgh and Urbana-Champaign. Therefore, we remove the observations which are too far away from these cities (more than 55 km). Then we select pizza businesses which are open, and remove some useless columns. For all remaining attributes, we check their values and do some transformation to make sure that each of them has appropriate factor levels for analysis. For the reviews and tips data, we try to detect some unreadable items (not English or meaningless expressions) and remove them.

Data Analysis

Exploratory data analysis:

Before doing complicated modeling and tests, we explore the data and make some straightforward graphs to get familiar with the datasets. We generate a histogram to find out the distribution of stars. Then we make graphs of stars together with cities and time to look at the differences and trends. We also generate tables and graphs to find out the frequency and star distribution of some important words from the reviews. This step is really helpful for our later analysis.





Word importance calculation:

In order to give constructive suggestions to business owners, we need to find out some key factors that affect the ratings. Our first step is extracting pivotal words from the reviews, so we should construct an index to measure the importance level of each word. We try two different methods, TF-IDF and regression, to calculate the word importance matrix.

TF-IDF is a popular method for text mining. The TF-IDF index is calculated as:

TF(WORD) = #term WORD appears in a document / #terms in the document
IDF(WORD) = ln(#documents / #documents with term WORD in it)
*TF-IDF=TF*IDF*

This index tends to increase if a word appears more often in a document. However, for some words that are too common, like 'and', 'is', 'are', it will be quite small. Therefore, TF-IDF should be an appropriate method for this problem.

Regression is a traditional way to deal with this problem. Here we try to fit linear and tree regression model for the occurence of words and star ratings. The absolute value of regression coefficients are simply the importance of the words. The top 5 results of two methods are listed below:

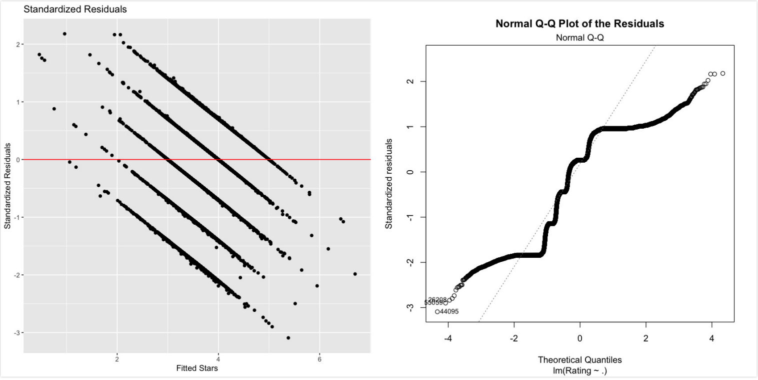
TF-IDF			Tree Reg		
Word	Frequency	TF-IDF	Word	Frequency	Coefficient
pizza	97915	0.014674	delicious	0.023	0.0731
good	40169	0.011126	great	0.024	0.0726
food	34945	0.010568	not	0.015	0.0660
place	36894	0.010550	worst	0.007	0.0450
time	27346	0.010237	best	0.010	0.0436

Key words selection and analysis:

Now we already have the word importance matrix. However, we notice that many words are irrelevant to our analysis. For example, we all know that the word 'delicious' tends to be in a high-star review, and we cannot make any inference from it. In fact, we are supposed to focus on nouns, especially food and service, since they are useful for us to raise suggestions by judging positive and negative reviews. Here we list important noun words related to food and service, together with some adjectives and adverbs which show strong sentiment, in order to prepare for our future steps.

From the important words list, we generate a word matrix which contains the occurrences of each key word in each review. In this way, it is straightforward that we could do regression to find the connection between occurrences of words in reviews and the star ratings. Here we try both linear and lasso regression models. Some results of linear regression are presented below, including part of coefficients and p-values, standardized residuals and Normal QQ plot. Since the star ratings take only integers 1-5, the residuals are not continuous. We can see that the assumptions do not meet well here and the R^2 is quite low. Therefore, it seems that linear regression model performs poorly.

Variable	Coefficient	Std.error	T-value	$Pr(>\|T\|)$
beer	0.0543	0.0107	5.069	4.01e-07
wine	0.0956	0.0162	5.886	3.98e-09
potato	-0.0109	0.0279	-0.390	0.6962
shrimp	-0.1260	0.0307	-4.102	4.11e-05
lettuce	-0.6189	0.0397	-15.573	$< 2e-16$
Multiple R^2:	0.02058		Adjusted R^2:	0.01877



Next we fit the lasso regression model. It is shown that there is no significant difference between results of linear and lasso regression. Considering that our regression problem is on a huge sparse matrix, we decide to use the lasso results.

In the light of the outcomes, we can test the difference of the star ratings of reviews which contains a specific word or not. For example, ”beer” and ”wine” have positive regression coefficients, which means that the occurrence of ”beer” and ”wine” in a review may increase its rating. This result coincides with our later analysis of business, and we will give such suggestions in our Shiny app.

Attitude analysis of reviews and nouns:

This step is the core procedure of our algorithm. If we are going to raise suggestions to a restaurant from its reviews, we have to tell whether the reviews are positive or negative. What's more, if a kind of food or service appears in a review, we should try to find the customer's attitude about it. For instance, if someone writes “the ice cream here is terrible”, we should tell that it is a negative review, and the customer dislikes the ice cream. If there are many similar reviews, we will remind the business owner to pay attention to the flavor of ice cream.

Previously we have generated two lists containing all positive and negative adjectives (adverbs). In order to judge a review, we try calculating the number of positive and negative words. We scan each word, and if it is positive with no negative adverb (‘no’, ‘not’, ‘never’, etc.) in front of it, we will count positive +1; if there is negative adverb in front of it, we will count negative +1. As for negative words, we carry out the same work. In this way, we could judge a review by comparing the number of positive and negative expressions in it. After testing, we could tell reviews like ‘it is good’, ‘it is not bad’, ‘wow pizza here’ as **positive** ones **(+1)**, and reviews like ‘it is not good’, ‘it is bad’, ‘it is not a good apple’ as **negative** ones **(-1)**. For reviews that do not show their sentiment, we denote them as **neutral** ones **(0)**.

Then we judge the attitude of the nouns in a review. Since the attitude information can only be extracted from the words near that noun, we need to focus on several parts of the review which contains our target noun. Our idea is cutting a whole review into pieces by some important symbols, which are the key nouns we listed before, and then find out the pieces that contain our target noun so that we can do similar judgement as above. For example, if a review contains the word ‘cheese’ and we want to find out the customer's attitude about it, we will split the review into pieces and keep the ones that contain ‘cheese’. Then we merge the selected pieces to form a new review for ‘cheese’ so that we can count the number of positive and negative words in it and draw conclusions. Finally, we can generate a matrix which contains the attitude of each key noun in each review.

We plan to test the accuracy of our previous judgement. First, we use ANOVA to test if the mean star ratings of positive, neutral and negative reviews of a noun are the same. The result shows that for most key words, there are significant differences among the three kinds of reviews. Then we use T-test to make sure that the mean ratings of positive reviews are greater than negative ones. All test results indicate that our judgements are quite accurate.

Word	ANOVA p-value	Pos-Neu	Neu-Neg	Pos-Neg	T-test p-value
pizza	< 2e-16	0.767	0.906	1.673	< 2e-16
cheese	< 2e-16	0.567	0.625	1.192	< 2e-16
sauce	< 2e-16	0.625	0.588	1.213	< 2e-16
service	< 2e-16	1.030	1.312	2.342	< 2e-16
salad	< 2e-16	0.634	0.923	1.558	< 2e-16

Analysis of business:

Then we deal with the business data. Will we be able to improve our star ratings by adding some food or service? To answer this question, we plan to do ANOVA and T-tests for business attributes to see if there are significant differences.

Before analyzing the data, we remove some unnecessary variables and make a summary of the data to get familiar with factors and their levels. For some variables that have too unbalanced values, making inference seems unpersuasive. For example, only 1.2% of pizza restaurants do not offer takeout, so the percentage is too low for us to draw conclusions. After we make tests for the attributes successively, we find that cities, alcohol, bike-parking, noise level, reservations, WiFi and car-parking are significant factors of star ratings. And restaurants which are good for kids and groups also receive higher ratings. We will give appropriate suggestions based on the results of tests.

Suggestions

Finally, we can raise suggestions for each restaurant according to the analysis above. The recommendations are based on both business attributes and customer reviews.

It is quite simple to give proposals given the results of business analysis. We propose that restaurants should keep beneficial services if they already offer them. Otherwise, we suggest that business owners should try to add them. For example, if one business offers alcohol, we would suggest keeping it, or else we would say that they should offer alcohol if possible. For some special services, like delivery, we will ask business owners to raise their delivery quality instead of cancelling the service.

As for suggestions from reviews,

Summary, Advantages and Disadvantages

In this project, we apply Python and R to extract useful information from the business and reviews data, and then try to give appropriate suggestions to business owners in order to help them improve their ratings. Generally speaking, our algorithm works well in most cases and we believe that if the owners follow our proposals on Shiny app, their ratings will tend to increase. However, machine is not human being, so it is possible that the algorithm makes mistakes when judging reviews. We still need to look for more advanced method to improve the accuracy of results.

YICEN LIU, edits what. **HUA TONG**, edits what. **ENZE WANG**, main editor of what.