Project: Restaurant Review Rating

Long Ly (1000866434)

Motivation:

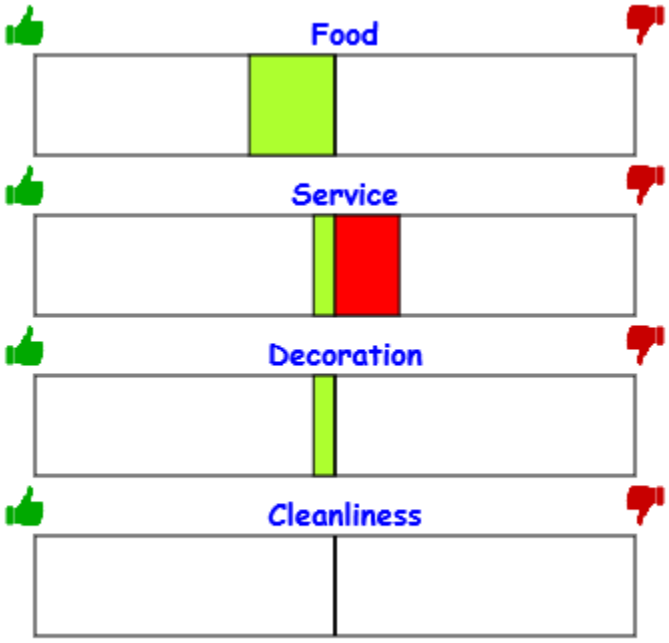
Yelp rating on reviews based on the number of stars are sometimes very ambiguous. A 4-star review does not necessarily mean all of the aspects of the restaurant are positive from customers' experience. For example, the food may be great and delicious, but the service is really slow and unsatisfied. On the other hand, a 2-star review does not also reflect all negative information about food or service at all; users may discuss about other aspects of the restaurant such as the parking lot, the decoration, or the cleanliness of the restaurant that they did not like about. Thus, simple rating restaurants based on the average number of stars they received does not reflect the real opinion of the customers about various aspects of the restaurant themselves. In addition, users have to go through a huge amount of reviews just to get basic ideas of the restaurant; some of the reviews are not even relevant to the restaurant. Therefore, the objective of this project is to save time for users by providing a more accurate rating system based on users' sentiment in their reviews.

Analysis Task:

There are two data sets will be using from Yelp: business data, and review data. The business data from Yelp contains many different business categories, but only 'Restaurant' category will be used for this project ('restaurant_data' file). For review data, any irrelevant reviews to the "Restaurant" category as well as does not have any usefulness vote will be omitted from the data set ('review_data' file). Each review will be cleaned by removing all special characters and all text will be changed into lowercase. Nevertheless, stop words will not be removed from the reviews and the text will not be stemmed since they may contain important information about users' sentiment. The clean data will be used to train a classifier to identify the category of every sentences in the reviews: Food, Service, Decoration, and Cleanliness. (This idea is adapted from "Classifying Yelp reviews into relevant categories" of Vaibhav Saini, 2014) A new rating system will be used based on the positive or negative experience based on each sentence of the reviews.

Deliverable:

The project will be delivered as a website to demonstrate the usefulness of this project. The user will be able to search for the restaurant by name, or keyword. The website will look up the closest match or exact match of the restaurant. The list of restaurants will be shown and users need to choose the restaurant that they wish to view the rating by choosing the hyperlink to redirect to the rating page. In the rating page, the bar chart look like the figure below will show on the screen.



Graph 1: Example of a restaurant rating

(adapted from "Classifying Yelp reviews into relevant categories" of Vaibhav Saini, 2014)

In the example above, there are four different rating categories: Food, Service, Decoration, and Cleanliness. The green color on the left of the bar represents the percentage of positive rating, and the red color on the right side represents the percentage of negative rating for each category based on users' sentiment. The sample chart above shows that the restaurant has very good food but its service is terrible. Few users also had good impressions on the decoration of this restaurant. By using this chart, Yelp users can easily have a better idea of what their experiences with this restaurant will be.

Change of Plan:

There are a lot of changes in the project since the project proposal. Initially, the proposal was to design a method to extract negative information about the restaurant. The method will be able to tell which aspects that customers did not like about the restaurant so that new customers will know what their expectations will be when visiting the restaurant. However, after carefully researched, the first project proposal is not feasible. In order to extract negative information, I need to go through a lot of text analysis combining with sentiment analysis. Thus, the time I need to complete the method would be beyond the required time of this project. With the same motivation, to help users not to go through a huge numbers of reviews, I decided to provide a better rating system using all the information provided from Yelp users. Negative information is important but new customers also need to see positive information about the restaurant as well. For example, some people only want to enjoy delicious food; they do not even care about the service as well as the decoration of the restaurant. Furthermore, if the restaurant has 90% of positive reviews and about 5% of negative reviews, it will not be helpful to all users since all they can see is the negative information. Thus, by providing both positive and negative information, new customers will have a better idea of what they need to expect for the restaurant.

Design of Method:

First, I will tokenize all the words in the review data to find all unigrams, and calculate the frequency of each token in the entire corpus. Next, using the frequency found, I was able to determine the most common words and put them into a list. Each word will be categorized manually into Food, Service, Decoration, and Cleanliness. In addition, I will determine the sentiment value of each word (if possible). Using the Naïve Bayesian Classification, I will train the data into 4 different classifiers for each category: Food, Service, Decoration, and Cleanliness to classify each sentence in every review. If a sentence contains multiple sentiment value for the same category, the average value will be calculated and rounded to the closest integer. For example: -1.666667 will be rounded to -2. After calculating the total sentiment value of each sentence, each review will have the positive and negative rating for each different category.

The rating (both positive and negative) of the restaurant will be using the below formula and represent as bar chart.

$$possitive\ or\ negative\ rating = \frac{|\sum sentiment\ values|}{2\ x\ number\ of\ ratings}$$

Implementation of Method:

GitHub repository contains the source code to clean the business data as well as the review data from Yelp. Since the output of these data are large, they will be hosted in different location (Google Drive). Among 452413 reviews, I will randomly choose 3000 reviews to build the classifiers for this project and save in 'train_data' file. As mentioned earlier, the frequency of each word in the reviews will be calculated and only words with frequency of 50 or higher will be added into the list of most common words ('common_words' file). Among all the unigrams, I was able to collect 1000 most common words using the frequency method. Since some words are generic, they sometimes do not belong to any of categories. The rest of the others will be manually assigned into Food, Service, Decoration, and Cleanliness. The table below demonstrates how I manually assigned the word:

Amazing	food	service	decoration	cleanliness
Appetizers	food			
Authentic	food	decoration		
Back	(generic words)			

Table 1: Example of categories assignment

(The full list including the categories will be in the CSV file 'common_words')

There are many available dictionary to determine the sentiment value of a single word, but "Subjectivity Lexicon" dictionary has the largest amount of words as well as good accuracy. (Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. Proc. of HLT-EMNLP-2005). It contains 8222 words; each word is divided into different type of subjectivity as well as different prior polarity. There are also many information in the dictionary, but I will only use "Type" and "Prior Polarity" to determine the sentiment value for this project.

Type	Prior Polarity	Sentiment Value
Strong Subjectivity	Positive	+2
Weak Subjectivity	Positive	+1
Strong/Weak Subjectivity	Neutral	0
Weak Subjectivity	Negative	-1
Strong Subjectivity	Negative	-2

Table 2: Sentiment value corresponding to the subjectivity type and prior polarity of each word

Each review will be tokenized into sentences. For each sentence, words will be collected and compare with the most common words to classify into different categories. One sentence may belong to multiple categories but only the category that has maximum number of words will be chosen. In addition, each word will be checked with the "Subjectivity Lexicon" dictionary in order to find the sentiment value of the sentence. In case of many sentiment value found in one sentence, the average will be calculated and the value will be rounded to the convention below:

Range	Rounded Value
value <= -1.5	-2
-1.5 < value <= -0.5	-1
-0.5 < value < 0.5	0
0.5 <= value < 1.5	+1
value >= 1.5	+2

Table 3: Rounding number convention

The next example will demonstrate how sentence classification works.

Word	Food	Service	Decoration	Cleanliness	Sentiment
the	0	0	0	0	0
waitress	0	1	0	0	0
is	0	0	0	0	0
outstanding	1	1	0	1	+2
Total	1	2	0	1	+2

Table 4: Example of sentence classification for "The waitress is outstanding"

Since “Service” has the maximum number of occurrences, the sentence will be classified in “Service” category. The word “outstanding” has the sentiment value of ‘+2’ based on “Subjectivity Lexicon” dictionary. Thus, this sentence will belong to “Service” category with sentiment value ‘+2’. In addition, if there are multiple categories that have the same amount of occurrences, the first category encountered will be chosen since all the categories are already ranked based on their popularity on the reviews.

Notice that if the sentence contains negative words such as “not”, the sentiment value will be reversed. For example, the sentence “The food isn’t bad at all”. Using the original method would result that the sentence is in “Food” category with sentiment value ‘-1’ (the word “bad” has sentiment value ‘-1’). However, since there is the appearance of the word “isn’t”, the sentiment of this sentence will be corrected to ‘1’.

After evaluating each sentence, the review will have both positive and negative score for each category. These score will be used in calculating the average positive and negative experience for the particular restaurant to display as a graph.

Food		**Service**		**Decoration**		**Cleanliness**	
Positive	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
+2	-1	0	-2	0	0	+2	0

Table 5: Example of positive and negative rating for all categories for a single review

Result and Evaluation:

Once done with the training data, the result is evaluated manually by randomly checking the reviews with the calculated rating. Most of the training data have the correct results. However, further review rating may contain the inaccuracy due to the incomplete list of the most common words. The list does not reflect the true common words in the entire corpus. In addition, there may be some inaccuracy in classifying manually each words in the list.

After finishing the evaluation of the training data, I went back to process all the review dataset and the program was completed within minutes with 452413 reviews. The output will be mapped to the corresponding restaurants based on the ‘business_id’. The result will be similar to the business dataset. Along with all business information, each restaurant has been added with two extra information: the number of review and the list of all positive and negative rating in four categories: Food, Service, Decoration, and Cleanliness in the format below:

{

'num_rating': [(number of reviews given)],

'rating': [(list of 8 numbers: positive and negative rating for 4 categories)],

'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)

},

...

}

(The file is stored as ‘restaurant_review’)

Outcome:

The project is demonstrated by the website. Due to lack of knowledge in building web pages and limited time, the website only contains necessary information to demonstrate the usefulness of this project. On homepage, user can type in the name, or keyword of the restaurant; a list will pop up as users enter in more than 3 characters. After selecting the name from the list, a list of all matching restaurants will be showed along with their locations.

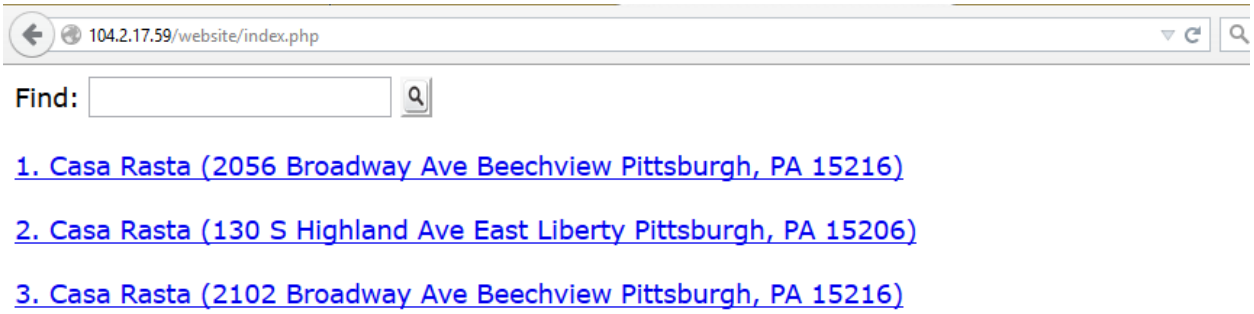


Figure 1: Example when users choose "Casa Rasta" in the search box

Once users choose any of the hyperlink, they will be redirected to another web page that has the positive and negative rating of each category: Food, Service, Decoration, and Cleanliness

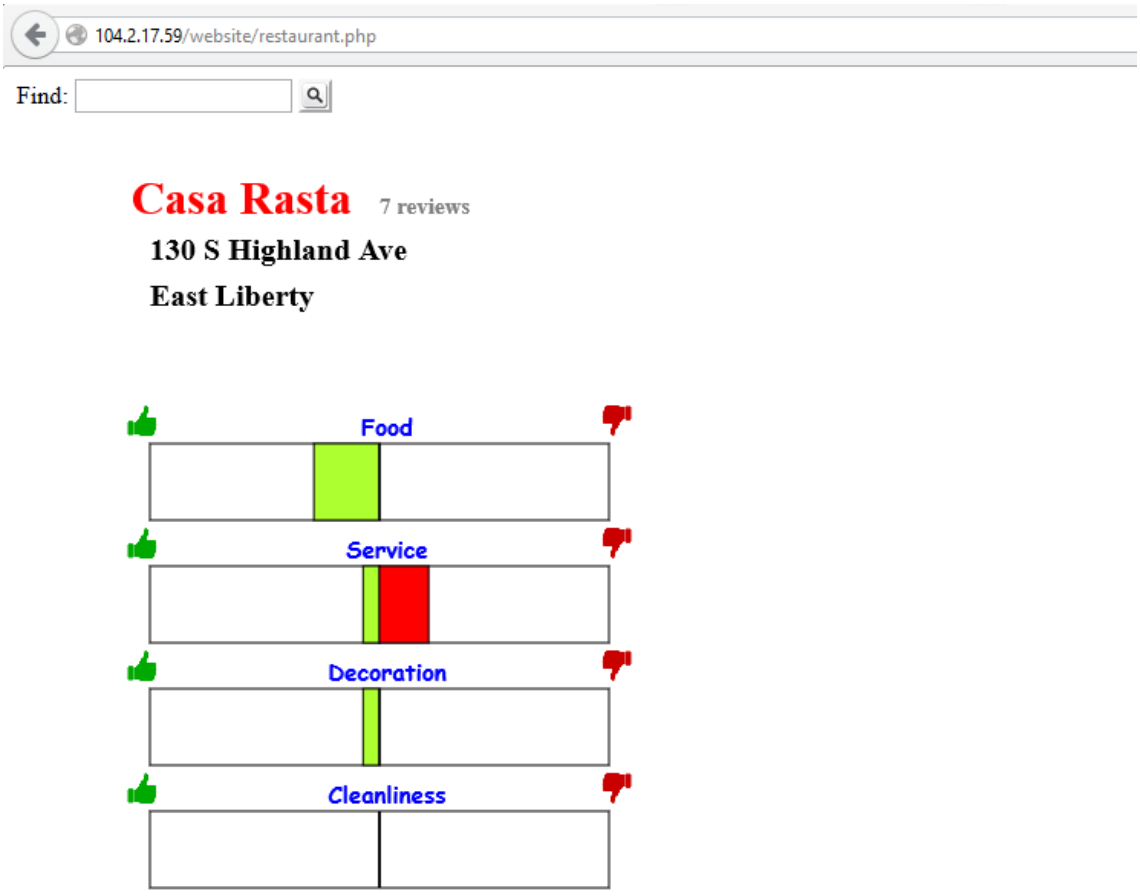


Figure 2: Rating review of "Casa Rasta" restaurant

The graph above showed the rating review of "Casa Rasta" restaurant. There are a total of 7 reviews from Yelp users. The food of this restaurant seems good while there are many people dislike the service here. Few users also like the decoration of this restaurant. None of the reviews mention about the cleanliness of this restaurant, so there is no information about the positive and negative rating.

Difficulties Encountered:

I spent a lot of time finding more information about sentiment analysis for each sentence of the review. Since users sometimes discuss various topics about the restaurant, it seems impossible to find a method to extract the correct negative information about the restaurant. Therefore, I decided to slightly change the objective of the project in order to make it deliverable at the end of the semester. In addition, since the project falls into multi-label classification, I also spent a lot of times finding the best method to classify the reviews into multiple categories. However, after discovering that there is no correlation between each category, I'm able to build multiple classifiers for all categories.

Another difficulty I encountered after getting all necessary data was to build a website to demonstrate the project. Due to lack of knowledge in building web pages, I had to spend a lot of time figuring out how to integrate all the components. Since the rating is different for all restaurant, it is even harder to build a dynamic graph for the website. I believe building the website only took me 50% of the time I spent for the project. However, since the required time for this project is limited, I was only able to build enough features to demonstrate the project. If I had more time for this project, I would have put more graphics as well as integrated all the reviews into the website. For now, users can only search for name and location, and they can only see the rating chart. Moreover, since I'm the only one working on this project, all tasks are done by myself. Thus, it would have been better if I had worked in team.

Finally, I had troubled using GitHub as well as hosting website. Since I'm not familiar with using GitHub, I accidentally deleted some commits that result in removing all code in the GitHub repository. It took me almost a day to figure out the command to use to recover all those files since I could not locate them anywhere including the Recycle Bin. In addition, the website I built somehow only works in local host. Once I put into Omega, it no longer works. The GitHub

pages do not provide PHP execution. Therefore, I hosted the website on my personal computer using home network.

URL of the website:

http://104.2.17.59/website/index.php

I host the website in my personal computer using the home network, so the website may not be available when grading. To avoid attack, I'll only turn on the server when the website needs to be graded.

References

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann (2005). Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. *Proc. of HLT-EMNLP-2005.*

Saini, Vaibhav, et al. Classifying Yelp reviews into relevant categories. University of California at Irvine, 2014.

Data:

Review Data (clean):
https://drive.google.com/open?id=0Bwry0GwTJUEzT3ZtMTd0cnhSMzA&authuser=0

Restaurant Data (clean):

https://drive.google.com/open?id=0Bwry0GwTJUEzT1E5eXlXYnVWMDg&authuser=0

Business Review: only contains the business id with the rating list

https://drive.google.com/open?id=0Bwry0GwTJUEzT0UwWmZGSzVIMm8&authuser=0

List of common words: contains all the common words with their categories

https://drive.google.com/open?id=0Bwry0GwTJUEzLTVScUxmVUdaQWs&authuser=0

Restaurant reviews: contain all the review with extra information about rating (JSON format)

https://drive.google.com/open?id=0Bwry0GwTJUEzOThiRWdzdzg0czA&authuser=0

Train Data: the data used to get the most common words list

https://drive.google.com/open?id=0Bwry0GwTJUEzenJZemZjb0d0aUU&authuser=0