Sentiment analysis using Yelp reviews

Li Du, Xiaochi (George) Li

Introduction

Natural language processing techniques can be applied to many aspects. Especially when associated with machine learning algorithms, these techniques can be particularly powerful in tackling many problems. In the project, we analyzed reviews from Yelp using a series of natural language processing techniques and also tried to predict the sentiments from these reviews represented as the star level using the review texts. Based on the findings from the processing and predictive modeling, we also tried to answer a few research questions regarding the performance of the models as well as how different/similar the reviews are for each category.

Dataset

Yelp has compiled a dataset that provided well-structured information[1]. For example, this dataset includes 5,996,996 reviews, 188,593 businesses, 280,992 pictures for 10 metropolitan areas. The information is available as JSON files and categorized as “business”, “review”, “user”, “checkin”, “tip” and “photo”. For the purpose of this project, we used only “business” and “review” as the review subset that has Yelp users’ reviews for the businesses as texts which are excellent raw data for the application and evaluation of natural language processing techniques.

Date preprocessing

The raw data were first screened using their designated categories, and businesses with multiple categories were excluded from the further analysis. In particular, we only kept restaurants for this project. The reviews for these qualified businesses were also investigated and the only the ones with at least two “useful” tags were retained for the final analysis. This will reduce the complexity introduced by

[1] The Yelp dataset is made publicly available at: https://www.yelp.com/dataset

having multiple categories for one business and to some extent ensure the reviews are the reflection of the truth by considering the level of acceptance by the yelp users. The preprocessing dramatically reduced the size of the dataset from over 3 millions to 0.9 million. In order to allow analysis that can be performed by reasonable amount of computational resources, the 0.9 million records were randomly sampled and final dataset for analysis contains about 144000 records. The dataset includes: reviews, categorization by cuisine, categorization by restaurant type, and star levels.

The reviews were first tokenized and made free of English stop words (as implemented by the nltk library in Python). The punctuations were subsequently removed from each review, and the tokenized words were concatenated back to form a string.

Figure 1 shows the proportion of the reviews/restaurants of each category in the entire dataset. Reviews of 4 and 5 stars together consists of more than 50% of the entire dataset and the ones with the other star levels each takes up approximately 15%. In terms of the types of the restaurants, American type cuisine consists of almost 70% of all the restaurants and about 57% of all the restaurants were categorized as “nightlife”.

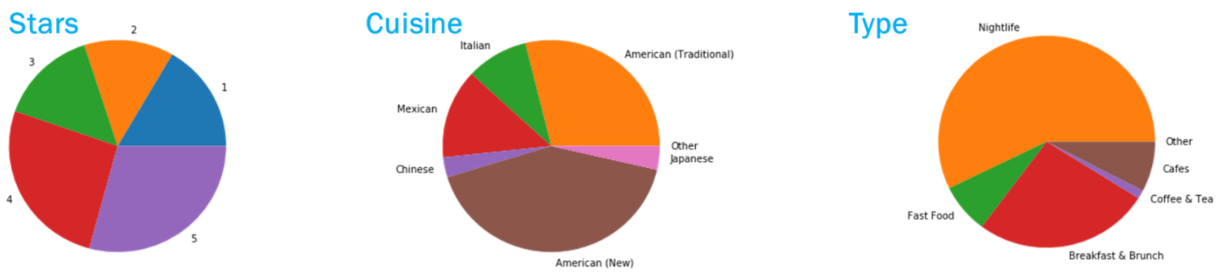

Figure 1. Break-down of the reviews by stars, cuisine and restaurant type

Analysis and discussions

Tools

The analysis was performed in Python with nltk and sklearn as the core libraries. In particular, the tokenization and word vectorization functions were the primary tools for this project.

Model selection

In order for any machine learning algorithms to be able to ingest the words, the texts should first be vectorized. In this project, we applied and evaluated the count vectorizer, hashing vectorizer and tf-idf vectorizer that are implemented by the sklearn library. Count vectorization simply counts the number of each words in each review whereas tf-idf vectorization is equivalent to calculating the tf-idf values after the count vectorization. The implementation of latent semantic analysis was evaluated after the count and tf-idf vectorization process to further reduce the dimensionality of the dataset.

A random forest classifier follows the vectorization processes to predict the star level of each review record. While this problem can be considered as a classification problem as the case in this project, it is also possible to apply regression algorithms for this problem, which has been demonstrated in other similar studies[2]. An advantage of considering this project as a classification problem is the relative straight-forward measure of model performance. The fact that the target variable is discrete and numeric, and the predicted stars are continuous may cause some complication in the model evaluation process. Table 1 shows the comparison of various vectorization approaches and their resulting model performance measured as classification accuracy.

All of the combinations showed similar performances. ***Tf-idf vectorization without no further processing was selected as the modeling approach for this project due to its weak winning margin***.

[2] Fan, M., Khademi, M.. 2014. Predicting a business star in Yelp from its reviews text alone. Available as arXiv:1401.0864.

Table 1. Classification of various vectorization approaches[3]

Model structure	**Classification accuracy**
Count vectorizer + Random forest classifier	75.99%
Tf-idf vectorizer + Random forest classifier	76.03%
Count vectorizer + Truncated SVD + Random forest classifier	72.53%
Tf-idf vectorizer + Truncated SVD + Random forest classifier	74.34%

A closer look at the confusion matrix (Figure 2) generated based on modeling result suggests that the classifier does well on predicting the reviews with strong sentiment, both positive and negative. However, reviews with moderate level of sentiments are relative difficult to model, for example, reviews with 2 and 3 stars.

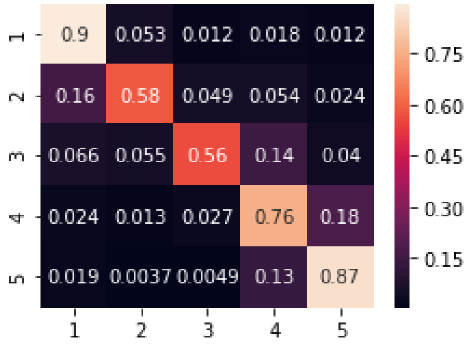

Figure 2. Confusion matrix for the modeling results (using tf-idf vectorization)

[3] The words that appear in very high (> 95% of the records) and very low (< 1% of the records) frequency were excluded from the vectorization process as they may contribute limited amount of information to the analysis. Both single words and bi-grams were considered in the vectorization process. The random forest classifier was defined to have 100 trees.

Analysis of modeling results

The difference in the performances among various star levels are very likely due to the presence of certain "sentimental" words. Figure 3 shows the word cloud for the words/bi-grams that are most important and least important as indicated by the feature importance of the trained random forest classifier. Note that this word cloud is not based on the frequency of appearance of these words, but rather the ranking of importance. For example, the larger sizes mean either "most important" or "least important" depending on the specific plot. The results were not surprising: words that are the most important to successful prediction of star levels are the words with strong emotions, such as great, good, amazing. In contrast, the least important words to the classification include mostly the nouns and verbs with very neural sentiments.

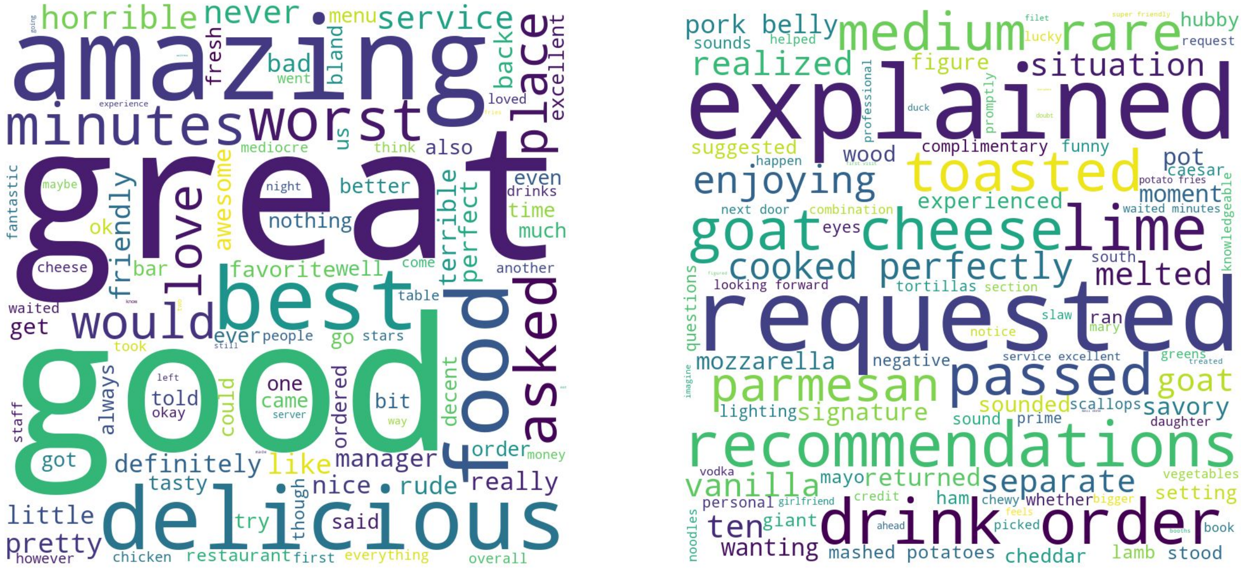

Figure 3. Words and bi-grams that are identified as most important (left) and least important (right) by the random forest classifier

The confusion matrix indicated that the model can, in some cases, make extremely biased prediction, such as predict 1 star reviews as 5 star or the way. We would like to gain some insights on these scenarios. By comparing the average length of the reviews of these "totally wrong" predictions, it was found that these reviews tend to be shorter (411 characters for 1 star reviews but predicted as 5 star and 464 characters for 5 star review but predicted as 1 star) than the correctly predicted ones (545 characters). In addition, the words that are most used[4] in these reviews as demonstrated by Figure 4 are generally vague and neural. In the 1-star reviews which were classified as 5 star reviews, there are even some words showing positive sentiments and they may very likely contribute to the incorrect classification. This might be an issue caused by removing stop words in the tokenization process as "NOT" was in nltk's stop words vocabulary. Removing such words can change the meaning of some reviews, potentially resulting in the misclassification.

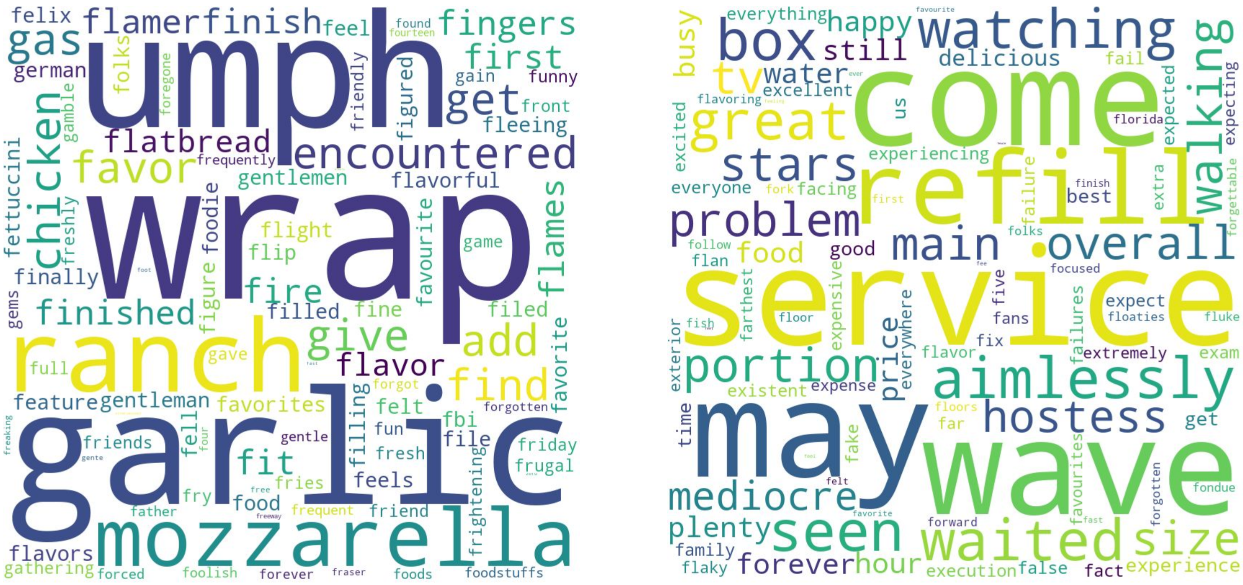

Figure 4. Words frequently used in highly biased in predictions (left: 5 star predicted as 1 star, right: 1 star predicted as 5 star)

Similarity of reviews across categories

In this project, we were also interested in whether the reviews on different restaurants with the same star rating are similar or not. Figure 5 shows the cosine similarity calculated based on the tf-idf values of the words used in the reviews. It suggests that for all the 1-star reviews that are posted to the six types of cuisines, people may talk about the "un-satisfactory" aspects different for Asian (i.e. Chinese and Japanese) cuisines compared to other cuisines, and the two types of Asian restaurants are not alike either. By examining the words with high tf-idf values for American restaurants and Chinese restaurants (See Figure 5), it appears that people may talk about food a bit more for American cuisine and service a bit more for Chinese restaurants. However, these figures did not provide strong evidence showing

[4] Note that "most used words" are defined here as having the highest tf-idf values instead of raw counts.

unique features for the reviews to each cuisine and more analysis needs to be conducted to fully understand these underlying differences.

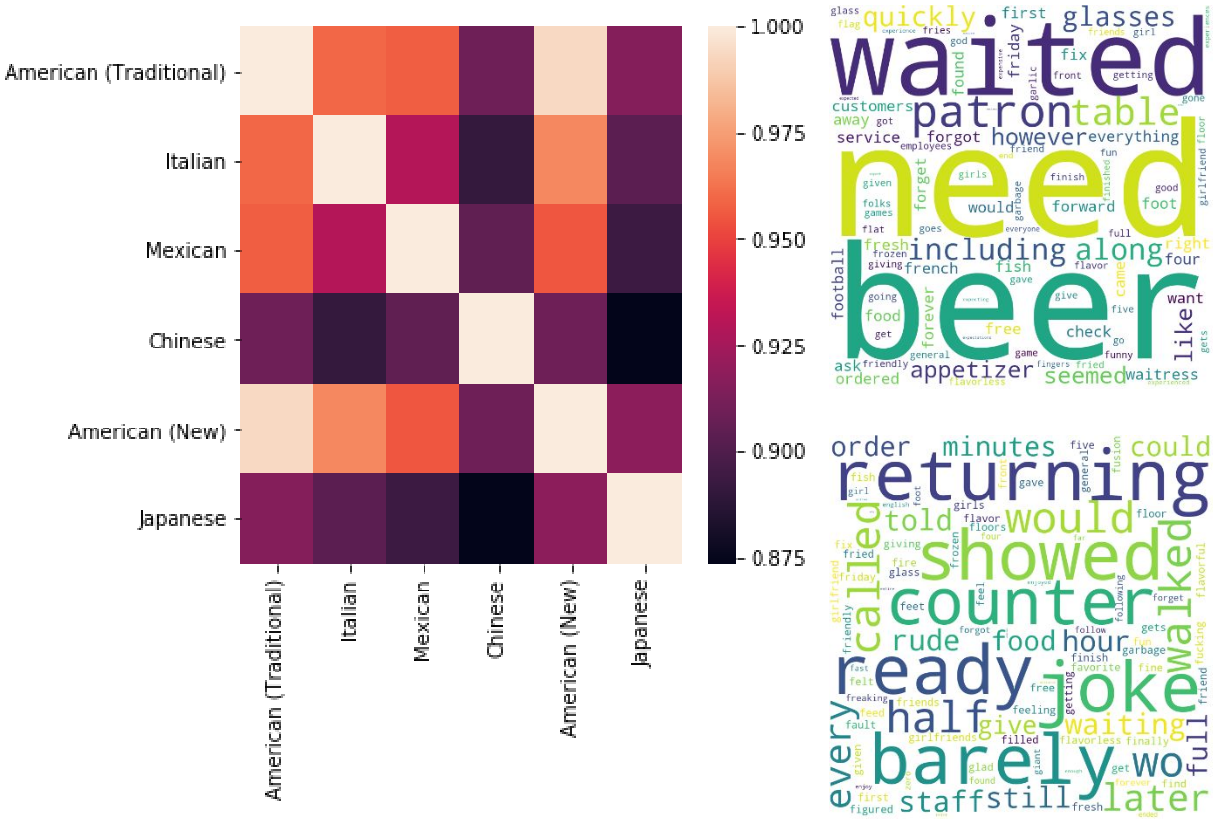

Figure 5. Cosine similarity calculated based on the tf-idf values of the words in 1-star reviews for the six cuisines (left), and frequently used words in American (traditional) restaurants (upper right) and Chinese restaurants (lower right)

To further evaluate the similarity across various restaurant categories, a model was developed and trained using reviews posted to restaurants categorized as “nightlife” and used to predict the sentiment of other categories. The model performed decently well on “cafes” with classification accuracy of over 78%, but far from ideal for any other categories (accuracy around 50%). More analysis such as the similarity between each type pairs should be done in order to further understand the results.

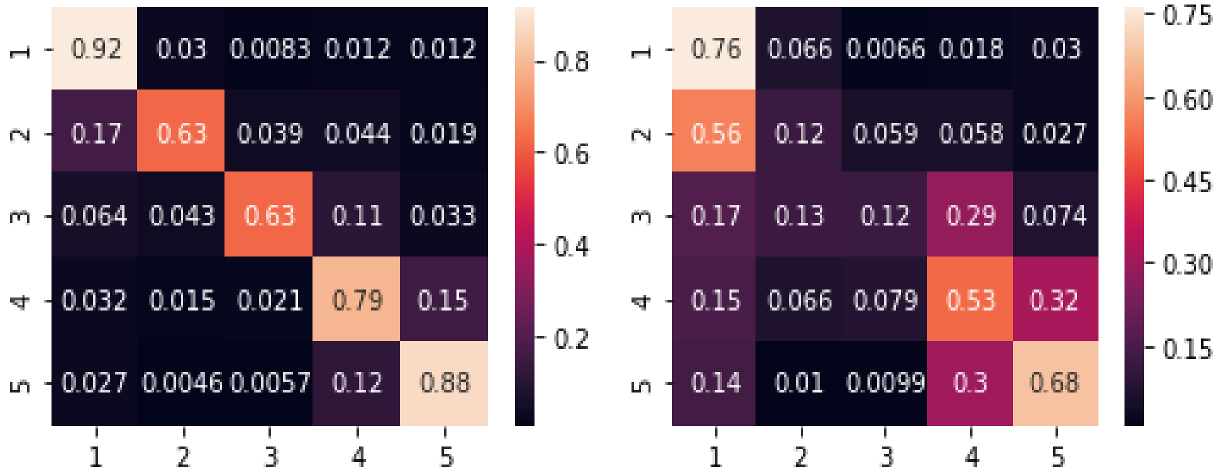

Figure 6. Performance of a model trained by reviews for the nightlife restaurants on cafes (left) and coffee shops (right)

Conclusion and lessons learnt

The analysis and modeling performed in this project suggests that that tf-idf vectorization yielded the best results in term of the modeling accuracy for the star levels associated with the reviews. However, other vectorizations with or without dimension reduction techniques were able to yield similar performance. Further model tuning such as number of dimensions to retain after the dimension reduction and the hyperparameters for the random forest classifiers may be needed for in-depth evaluation of the various models.

The outcome from this project reveals that reviews with strong sentiments are relatively easier to predict, such 1 star reviews and 4,5 star reviews. Consistently, words and bigrams with strong sentiments (such as awesome, great , horrible) are the most predictive to the classification of star levels associated with each review, whereas words and bigrams with neural sentiments (which are mostly nouns and verbs) are the least important to the classification.

It also appears that people talked about good and bad aspects of the different types of restaurants differently. Cosine similarity of the tf-idf scores of the 1-star reviews on the 6 cuisines indicates that the reviews on Asian restaurants are different compared to other cuisines. A further extended analysis by training models on one cuisine and predicting the star levels on other cuisines also suggests that the words used in the reviews are different since the performances of these models are generally only moderate (~50% accuracy).

Given the analysis performed in this project, we realized a few caveats, which should be addressed in future work: (1) tokenization with the removal of stopwords implemented by the nltk packages excluded certain words that were actually helpful. For example, "not" was excluded in the tokenization process, which likely resulted in certain level of information loss. (2) there are confounding factors in the similarity analysis and they were not considered thoroughly. For example, people leaving negative reviews may be using different sets of words. To make it more complicated, due to lack of spelling checks in the Yelp review systems, misspelled words were treated as distinct appearance, and the misspelling can also be potentially associated with certain group of users who may be more likely to leave certain types of reviews (e.g. longer vs. short, neural vs. strong etc.). Carefully designed analysis can partially address certain aspects of these caveats and more data may be needed to understand the full story.

Contributions

All the member of the group contribute equally to the project, with each one focusing more on certain components. Xiaochi (George) Li took the lead on the data collection, data preprocessing and exploratory analysis. Li Du was the lead on the modeling and results interpretation and discussions. However, both group members are deeply involved in all the processes.