Sentiment Analysis of Yelp Review Text for the Prediction of Yelp Star Ratings

Kuan Siew Weng

11 November 2015

1. Introduction

The explosive growth and influence of social media, blogs and online review sites such as Facebook, Twitter, Yelp, TripAdvisor, has stirred up much interest in businesses to mine and quantify user opinions and sentiments expressed in tweets, posts and user reviews.

For restaurants in particular, sentiments expressed on such review sites may have a significant impact on their customer volume and revenues.

The aim of this study is to investigate the predictive accuracy of applying sentiment analysis and machine learning methods on Yelp restaurant reviews to predict the 5-star ratings of each review from user sentiments extracted from the review text.

Due to the complex nature of natural language processing and sentiment analysis, the study is confined to review texts written in the English language only, to keep the scope of the project manageable.

2. Methods and Data

Data

The data used for this study is the academic dataset provided by the Yelp Dataset Challenge Round 6 website, which included 1.6 million reviews by 366 thousand users for 61 thousand businesses.

As the focus of this study is on restaurant reviews, I will take a 10% random sample of the provided review dataset, and do an inner join of the sample review dataset to the business dataset by **business_id**, and select reviews that are categorized as "Restaurants" only. The resulting cleaned dataset for analysis comprised of 98841 reviews for 15464 restaurants.

Methods and Tools

In this study, I used a methodology, which comprised of the following steps:

1. Explore the cleaned reviews dataset, looking for language patterns in both negative and positive reviews, according to the 5-star ratings.
2. Select a sentiment polarity model and customize for the restaurant review context.
3. Select and extract additional features from the text such as total sentence count, total word count, positive words count, negative words count.
4. Define and refine the stars prediction model (response and predictor variables)
5. Experiment with various classification methods (RandomForest, Naive-Bayes, SVM, Multinomial Regression, Gradient Boosted Machines, ..etc)
6. Repeat steps 1 to 5 iteratively until training model cross-validation results are optimal.
7. Select three classification methods and predict using the test dataset
8. Compare and tabulate the accuracy results of the three methods.

At the end, the selected sentiment polarity model is the model offered by the qdap R package by Tyler Rinker, and the three selected classification methods were RandomForest (rf), Naive-Bayes (nb), and Penalized Multinomial Regression (multinom) which would be trained and tested using the caret R package.

Figure 1: A Comparison of 1-Star and 5-Star Reviews Trigram Word Clouds

Exploratory Data Analysis

Using the **NLP** and **qdap** packages, I explored the review text field in search of the most commonly occuring unigrams, bigrams and trigrams in the 1-star and 5-stars rated reviews to understand the language used by restaurant patrons in the most negatively and positively rated reviews.

Negative reviews frequently contain n-grams such as "horrible experience", "terrible service", "mediocre food" as well as warnings such as "don't waste money", "go somewhere else", "never be back", whereas positive reviews frequently contain n-grams such as "great food", "great service" and also on repeat visits such as "can't wait to be back", "go back again", "cant go wrong", . . . etc. This is illustrated in the Figure 1, which is a comparison of the Trigram Word Clouds generated for 1-Star and 5-Stars Reviews.

Exploration of the review sample data showed that nearly 2 our of 3 restaurant reviews are rated 4 or 5 stars. This suggests that there is a strong bias of reviewers to post positive reviews rather than negative reviews.

Another observation is that most reviews are less than 100 words long although a few reviews even exceeded 400 words. However, the length of the reviews does not appear to be correlated to their star ratings.

These initial observations are summarized in the figure 2.

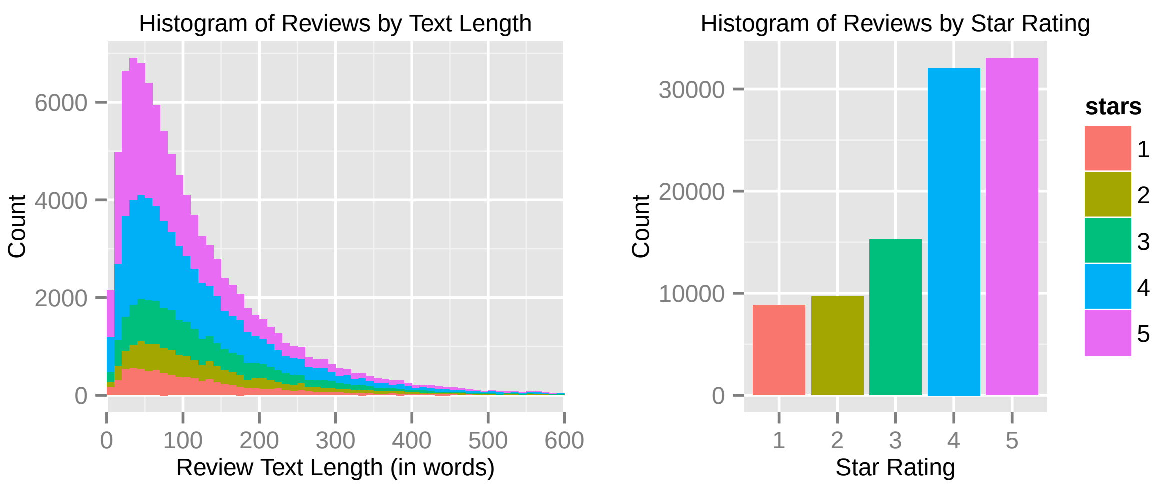

Figure 2: Histogram Plots of Reviews by Word Count and Star Ratings

Sentiment Polarity Analysis

The **qdap** package provides a function called **polarity**, which can be used to analyze and quantify the sentiment of a text, and returns a sentiment polarity score between -1 (for most negative sentiments) and 1 (for most positive sentiments).

The **polarity** function is dependent on the polarity dictionary used, which it defaults to the word polarity dictionary used by Hu & Liu (2004).

The inital Review Sentiment Polarity Model (Model 1) used the default polarity dictionary, and the review text was analyzed as a whole. The results revealed that overly positive scores were assigned to many of the negative reviews. The variances of the polarity scores for reviews with the same star rating were quite high as well.

To enable the **polarity** function to consider the type of language used by the reviewers, I created a restaurant-reviews context dictionary that extends the default dictionary with positive and negative n-grams occurring often in reviews, such as "never be back", "can't wait to be back", and also tune the model by adjusting the weightages. The following is an example output of the **polarity** function.

```
polarity(sent_detect("This is worst place ever. Crap Food. Horrible service."),
         polarity.frame = REVIEWS_DICT, constrain=T)
##   all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
## 1 all               3           9        -0.83       0.101             -8.225
```

For the final Review Sentiment Polarity Model (Model 2), this custom polarity dictionary was used, and the polarity of each sentence of the review text was analyzed, and their average value was taken as the review's sentiment polarity score.

The distribution of sentiment polarity scores by star ratings of the two models is shown in the two box plots of Figure 3.

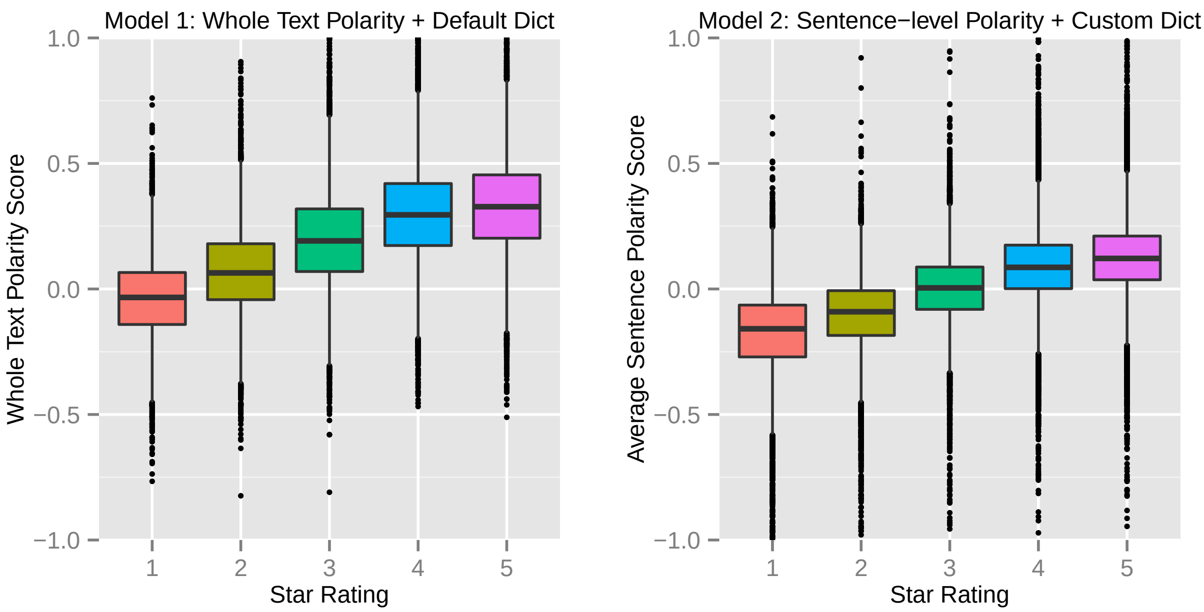

Figure 3: Polarity Box Plots by Star Ratings

These plots show that by using a context-specific custom dictionary and analyzing polarity at the sentence level, the review polarity values for Model 2 were not positively skewed like Model 1; and the variance of the Model 2 polarity scores within each star rating were narrowed as well, though outliers are still present.

Star Ratings Prediction Modeling

After several iterations, the selected review sentiment model based on sentence-level polarity, and the final review stars prediction model formula contains the following predictor variables:
- average polarity score returned by the **polarity()** function
- total sentence count, total word count, positive word count, negative word count.

The three classification methods selected for the final prediction testing for this study were Random Forest, Naive-Bayes and Penalized Multinomial Regression.

For training, the three methods were fitted on 70% of the restaurant review sample dataset using 3-fold cross-validation, and thereafter tested on the other 30% of the data.

3. Results

Model testing for all three methods returned consistent accuracy results to those achieved during training, suggesting that overfitting may not be a concern. The accuracy attained during testing for each of these methods are summarized in the following set of tables:

Table 1: Final Model - RandomForest Prediction Accuracy By Class

	Class: 1	Class: 2	Class: 3	Class: 4	Class: 5
Pos Pred Value	0.4086	0.2507	0.2123	0.3751	0.4551
Sensitivity	0.4279	0.1783	0.1171	0.3822	0.5743
Specificity	0.9391	0.9420	0.9204	0.6950	0.6555

Table 2: Final Model - Naive-Bayes Prediction Accuracy By Class

	Class: 1	Class: 2	Class: 3	Class: 4	Class: 5
Pos Pred Value	0.3833	0.2736	0.2186	0.3731	0.4528
Sensitivity	0.3892	0.2303	0.1414	0.3589	0.5630
Specificity	0.9385	0.9335	0.9074	0.7112	0.6592

Table 3: Final Model - Multinomial Prediction Accuracy By Class

	Class: 1	Class: 2	Class: 3	Class: 4	Class: 5
Pos Pred Value	0.4144	0.3049	0.2359	0.3525	0.4871
Sensitivity	0.5438	0.0886	0.0425	0.4228	0.6388
Specificity	0.9245	0.9780	0.9748	0.6280	0.6631

Table 4: Final Model - Overall Accuracy by Prediction Method

	RandomForest	NaiveBayes	Multinomial
Accuracy	0.3893	0.3834	0.4141

The results showed that the overall accuracy attained for all three methods are rather low, with the highest being 41%, by the Penalized Multinomial Regression method

Examining the prediction statistics by class, it is observed that the prediction sensitivity for star ratings 2 and 3 are quite low for all three methods, indicating that reviews rated at these levels were mostly misclassified, but prediction measures were much better for ratings 1 and 5.

4. Discussion

The Complexity of User Review Language

These results of this simple study show how complex natural language processing is, and therefore how difficult for an algorithm to gauge the user sentiment accurately from informal English text alone.

Interestingly, it seems much harder to predict sentiment polarity correctly for negative reviews as compared to positive reviews. There may be a number of possible reasons for this:

1. Star ratings given by reviewers is subjective and vary depending on a reviewer's personal disposition, cultural bias and context. Therefore, different reviewers may give different ratings for similar sentiments expressed. One man's 4-stars is another man's 5-stars.
2. Reviewers may be positive with some aspects of his experience, e.g food, but very negative on others, e.g. the attitude of waiters
3. Lowly-rated reviews often contain as many as positive words as negative words, when reviewers expressed unmet expectations in positive terms, e.g. "Planned a happy celebration; but . . . "
4. The use of localized slang and swear words in reviews, e.g. "The food is da bomb !!", where a normally negative word "bomb" is used to express positive sentiment. Our dictionary is not sufficiently populated with such slang words.
5. The use of sarcastic language, especially in lowly-rated reviews, was not detected by our simple review sentiment model.
6. Spelling mistakes, e.g. "Graet food!!", were missed by our model.

It is therefore not surprising that the accuracy is so low, that it would not be very useful to predict the 5-star rating of the review from its text with these methods.

The Alternative Thumbs-Up/Thumbs-Down Rating System

In this final subsection, I would like to explore an alternative to the 5-star rating system. In 2010, Youtube replaced their 5-star rating system with a Thumbs-Up/Thumbs-Down rating system, touting it as a simpler and less ambiguous way for viewers to express their sentiment for a video.

With over 60% of Yelp restaurant reviews in our sample positively rated at 4 or 5 stars, it would be reasonable to infer that reviewers who rated a restaurant at 3 stars or less, to have some negative sentiment. Thus, we can reasonably consider 4-5 stars as a "Thumbs-Up" and 1-3 stars as a "Thumbs-Down" for a restaurant review under the Thumbs-Up/Thumbs-Down rating system.

If we apply this heuristic mapping to our reviews sample, and train our models to predict the Thumbs-Up/Thumbs-Down rating instead of the 5-star rating, we indeed find that the prediction accuracy measures attained for all methods to be much higher and more useful as a result. The results are summarized in the table 5 below.

Table 5: Thumbs-Up/Thumbs-Down Model - Prediction Accuracy

Method	Overall.Accuracy	Pos.Pred.Value	Sensitivity	Specificity
RandomForest	0.7643	0.7894	0.8750	0.5517
NaiveBayes	0.7391	0.7978	0.8080	0.6069
Multinomial	0.7691	0.7851	0.8934	0.5306

To reproduce and review the results of this study, the R Markdown source for this paper can be found at : DSS_Capstone.Rmd - Kuan (2015).