Prediction of rating based on review text of Yelp reviews

Sasank Channapragada

Ruchika Shivaswamy

ABSTRACT

In this project, we aim to predict rating of businesses listed in the Yelp dataset based on review text. We also intend to classify the reviews as funny, useful or cool, the metric used by Yelp to evaluate user reviews. Linear regression and different classification techniques such as Naïve Bayes' Classification and Support Vector Machines were used for different features.

1. INTRODUCTION

The dataset used for this task was obtained from the Yelp dataset challenge, which consists of 1.6M reviews and 500K tips by 366K users for 61K businesses. It has 481K business attributes such as hours, parking availability and ambience. It contains a social network of 366K users for a total 2.9M social edges.

The dataset consists of five files – business, review, user, check-in and tip. Business and review files are primarily used for this predictive task. The business data file is a json file that comprises of attributes of each business listed on Yelp, and the review file consists of those of a review.

The text of a review is often overlooked in such predictive tasks in favour of features such as the user's and business' previous rating history. However, if the sentiment of the text of a user's review can be estimated suitably, it would be the best indicator of the rating. Ultimately, a review is what the opinion of the user is about the business in his own words and not a mathematical predictive task. Thus, it is essential to be able to predict what the user feels about a business from the review text and this is the task of rating prediction from review text was chosen.

2. EXPLORATORY ANALYSIS

The primary features of a business being used in our data analysis are business category and location (state and city). The preliminary analysis of the dataset includes study of distribution of reviews with respect to category of the business and its location. We also look at the funny, useful and cool rating given to user reviews by other users based on category of business and location. A distribution of number of reviews based on length of review was analyzed, and the impact of the latter on rating was deduced.

From the plot of categories vs the number of reviews (Figure 1) for each category, it was observed that the number of reviews for the category 'Restaurants' with 1083622 (69.05% of the total data) reviews, was by far the highest and that for category 'Firearms' was the least with a total of 3 reviews. Since impact of a particular word on rating will vary drastically across categories, considering all categories in the same text mining model would lead to unsuitable results. It was therefore concluded that considering reviews of only Restaurants would result in a more accurate model.

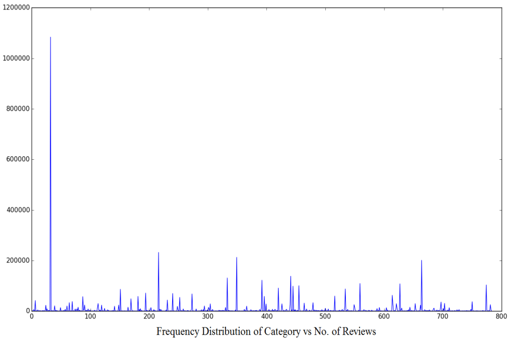

Figure 1: Frequency Distribution of Category vs Number of Reviews

Figure 2 shows a plot of states vs the number of reviews per state. It was evident from the plot that the two peaks are at NV (Nevada) and AZ (Arizona) with 752904 and 636779 user reviews respectively, which constitute 88.56% of all reviews. A minimum was observed at HAM with 3 reviews in total. It was also observed that some of these places were outside the United States and thus, do not have recognized abbreviations for their states. Reviews specific to America were obtained by using the latitude and longitude information given, and it was found that there were 1685580 American reviews and 44246 non-American reviews. Since some of the user reviews were in a different language (like German), only American reviews were considered for the text based predictions as it is almost certain to be in English and will have some uniformity about them.

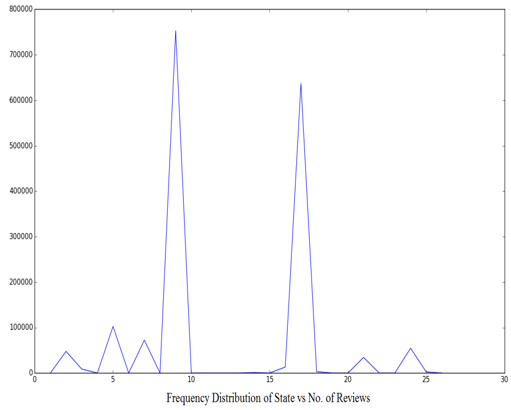

Figure 2: Frequency Distribution of State vs Number of Reviews

Figure 3 shows the plot of the cities vs the number of reviews for each city. It was apparent from this graph that Las Vegas with the highest number of reviews (685090 which constitutes 43.66% of the total) ranked the highest and Fort Kinnaird ranked the least with 3 user reviews. During the analysis, it was observed that there were many spelling errors in the list of cities. There were cities such as 'Last Vegas' and duplicate entries of other cities with alternate spellings, such as 'Pittsburgh' and 'Pittsburg'. It was thus concluded that the city was probably manually entered and that this count is prone to error.

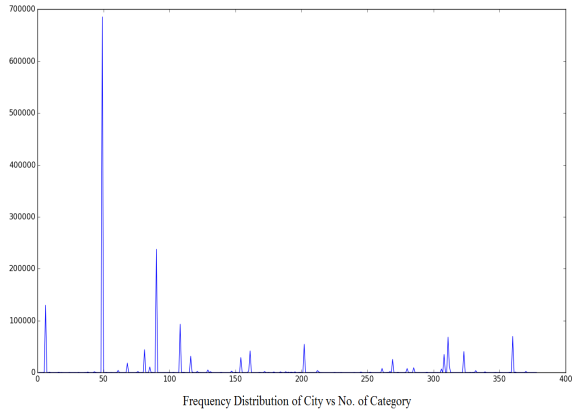

Figure 3: Frequency Distribution of Cities vs Number of Reviews

Table1 and Table2 show a list of the highest and lowest average funny, useful and cool ratings given by the users based on category and location.

Figure 4 shows the distribution of length of review text in words vs total number of reviews. We observe that this word frequency is increasing till around review length 100 after which there is a steady decrease in the number of reviews till 200. Following review length of 200 words, there is a sharp decline in the number of reviews. We have considered number of words instead of number of characters as the parameter as the number of words is a better indicator of length as written by humans. That is to say that the length of the words is not of the same importance. In order to ensure that outliers that have extremely short or long reviews are not considered in training the model, we consider the range of data between 100 and 200 words in length.

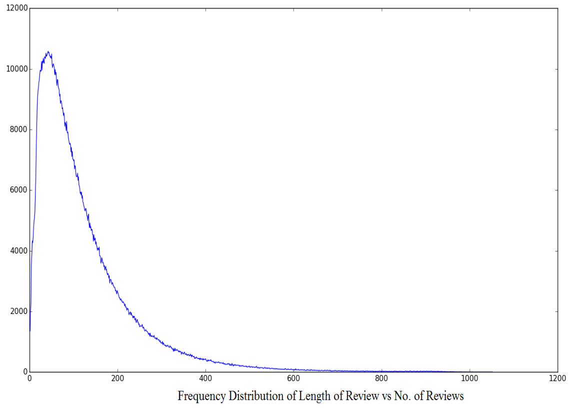

Figure 4: Frequency Distribution of Length of review text vs Number of Reviews

Figure 5 plots the relationship length of review text in words against the average rating for reviews of that particular length. The trend that is observed indicates that there is a steady decrease in the average rating as the number of words in the review increases. Moreover, the variation in average rating for lengths that are approximately the same is seen to be extremely high as the length increases, especially beyond 400, when compared to lower to middle lengths. Once again, it is observed that a range of 100 to 200 is a stable range of length of words in a review to consider for rating prediction as the impact of the length on the rating is lesser than other cases.

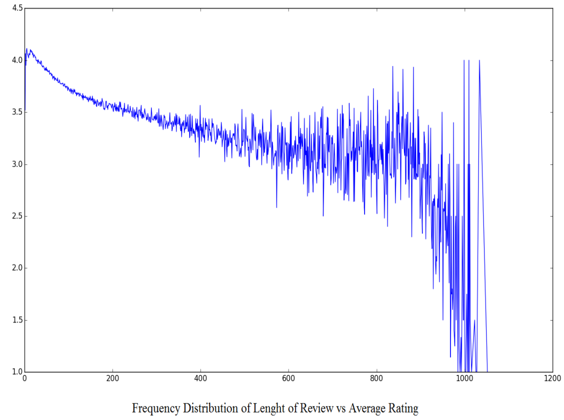

Figure 5: Distribution of Length of review text vs Average Rating

Figure 6 shows us a combination of the data in Figure 4 and 5. It is a plot of the review length in words against the average rating, with the number of reviews directly proportional to the size of the circles. We see the high concentration of reviews in the range below 200 and the random scatter of the reviews at higher word lengths. The three graphs help us prune the dataset to only those that have review lengths of 100 to 200 words.

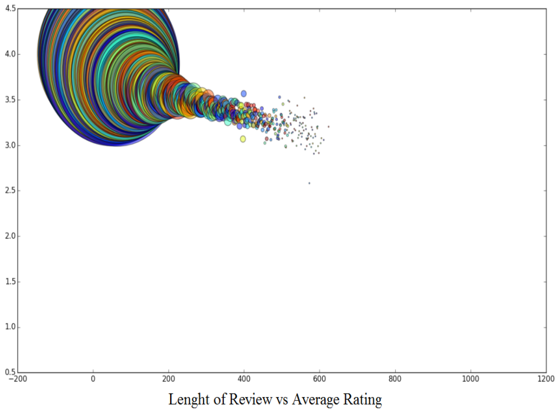

Figure 6: Plot of Review Length vs Average Rating

In addition, a study on the highest and lowest average funny, useful and cool ratings given by the users based on category and location was performed. The results are shown in table1 and table 2.

	Funny	**Useful**	**Cool**
Category	Hang Gliding (2.6)	Bartending Schools (3.6)	Food Banks (3.3158)
State	NTH (0.588)	CA (2.86)	KHL (0.875)
City	Firth of Forth (4.5)	Firth of Forth (4.75)	Firth of Forth (4.75)

Table 1: Table showing funniest, most useful and coolest category, state and city

	Funny	**Useful**	**Cool**
Category	Beer Garden (0.004)	Beer Garden (0.037)	Curry Sausage (0.013)
State	BW (0.009)	BW (0.056)	BW (0.017)
City	Karlsruhe (0.008)	Stutensee (0.029)	Karlsruhe (0.162)

Table 2: Table showing lowest values of average rating for funny, useful and cool metrics

The last preliminary analysis that was carried out was finding those words that impacted rating the most in these reviews. We restricted ourselves to those reviews in English for this purpose. The words in both the negative and the positive lists are listed along with the weight of their impact.

List of Negative Words
-0.574534775036,worst
-0.448868792982,horrible
-0.378600813243,terrible
-0.350349682212,mediocre
-0.331361093369,rude
-0.317155781901,overpriced
-0.306593243034,disappointing
-0.286445026595,bland
-0.281415745615,poor
-0.235995224561,dirty
-0.226727988344,soggy
-0.221516235209,overcooked
-0.218313610646,sorry
-0.215726863487,sad
-0.205020971351,barely
-0.204804929747,dry
-0.19960366805,money
-0.19945440401,unfortunately
-0.171656464399,frozen
-0.162521519714,ok
List of Positive Words
0.22969424517,incredible
0.22021030178,perfection
0.210545100774,amazing
0.20127478621,outstanding
0.198872591801,die
0.195025375968,awesome
0.187158124791,fantastic
0.180104838892,excellent
0.177589923364,glad
0.168881220894,delicious
0.149910846186,highly
0.148122274009,wonderful
0.141797332107,best
0.140686302994,perfect
0.138874261024,reasonable
0.130849910412,favorites
0.125970905508,favorite
0.123738624028,complaint
0.11979537329,thank
0.119616722182,loved

3. PREDICTIVE TASK

As discussed, the predictive task that was chosen is the prediction of rating of a review given only the review text. We also attempt to classify this review as funny, useful or cool, which are the same metrics that Yelp uses to evaluate a review based on the votes of other users. The data has been pruned to include just those reviews of Category 'Restaurants', reviews that were written about places in the United States and reviews that have a review text length of 100 to 200 words. This reduced the dataset to a size of 162K.

There are two ways to approach this problem. The first method is to approach it as a classification problem, classifying the review's rating in one of the five clusters (of rating 1-5). The second method is to predict an exact value of the rating, which can be a decimal value. The first method is more realistic as the rating of a business can only be an integer. However, the second method is also followed to evaluate its performance. In the case of predicting whether a review is funny, useful or cool; only the first method can be followed as it is strictly a classification problem.

If the problem is seen as a classification problem, then the accuracy of the classification and mean squared error of the rating prediction are the two possible evaluation metrics. In the case of predicting the decimal value of the rating, the metric can only be mean squared error. The baseline chosen for evaluating performance of the model is that of predicting average rating of the training data as the rating of each review in the test data irrespective of review text.

The model was trained over 80% of the data and then tested over the rest 20% of the data. The training and test error is reported in each case and the various models are compared using the performance metrics that are chosen and the best model is recommended.

A review was considered as funny if the number of 'funny' votes it received are greater than the number of 'cool' and 'useul' votes that it received. The same applied to reviews that were classified as 'cool' and 'useful.' While training the classifier, we ignored those reviews that could not be clearly classified as one of these three categories.

4. FEATURES

The first feature set that was used was a frequency distribution of the 1000 most common words that occur in the review text. This feature gave a list of the words that are most commonly used in restaurant reviews. During the exploratory anaysis, we studied how each word impacted the reviews. After pruning the data, a new list of 1000 words was generated that were most likely to impact the reviews of the data and used as our feature in classification and prediction.

The next feature set that was used was a frequency distribution of the 1000 most common words with stemming that occur in the review text and the last feature that was used was a distribution of the 1000 most common adjective words, after tagging them based on the parts of speech.

The training data was converted into a form of {feature,label} dictionary. Since our predictive task required us to only use review text, it was converted into the different features. This was accomplished by checking if each word in the review text already existed in the word set that was formed by checking for the 1000 most commonly occurring words, stemmed words or adjectives as needed by the particular feautre. Following this, the count of each of these words was computed and used as the feature. The 1000 word list would be used during the test process as well to compute the feature of the review text.

5. MODEL

A total of two classification models and one prediction model were considered. The two classification models are Naive Bayes' Classifier and the Support Vector Machine Classifier. The prediction method that was used was linear regression. Each of these three models was run with the three feature sets that have been discussed above (1000 frequent words, stemmed words, adjective words).

In the case of rating prediction, linear regression was allowed to predict ratings between two integers and thus, was expected to have a better MSE than the two classification methods. The classification methods have the disadvantage of having to predict only a whole number and the minimum absolute error when a wrong prediction is made is 1.

The SVM Classifier was expected to do better than the Naive Bayes' model as the latter assumes conditional indepence between features and that is a presumptuous assumption to make in the reviews of restaurants. Among the features, we assumed that the feature with adjective words being taken into account would perform the best as these are the words that tend to reflect the opinion of a user most in a review text.

Thus, the model that is expected to perform the best in terms of classification is the SVM classifier with feature as the frequency distribution of the 1000 most common adjectives in the review texts.

Since the data to be processed was pretty large and the operations that needed to be performed would take up a lot of time, improvement in run time was achieved by the use of appropriate dictionaries as needed. The lists review_data and business_data were reduced to a dictionary usa_data that contained only the pruned data along with just the features that we needed (review text, rating, user id, funny votes, useful votes, cool votes). Another dictionary was used to extract the 1000 most common words (or stemmed/adjective words, as the feature demanded) and while building the features for the review text, it was compared with the list that was generated to check if it is a part of the feature. This reduced run time almost exponentially when compared to earlier attempts.

6. LITERATURE

The dataset used for this task is the one available as a part of the Yelp dataset challenge (http://www.yelp.com/dataset _challenge). Most of the predictive tasks previously performed on this dataset have rating predictions primarily based on user and business attributes. However. research has been carried out, not just in the general area of text mining and sentiment analysis, but in text mining for predictive tasks in review and rating systems. The impact of text derived information has been previously studied at the sentence level, with the help of the topic information on various datasets. Various methods have been adopted in the past, including regression[1], bag of opinions method [2] and classification [3].In movie reviews, it has been observed that Naive Bayes' had a slightly better accuracy than the SVM method. However, this was in combination with other features of the dataset. Hence, the results differ from the ones chosen here.

The Yelp dataset has been extensively studied as well. Attempts have been made to gauge information from the review text by predicting what the user felt about various aspects of the business, such as service, quality of food and ambiance. [4] [5] If the user experience can be divided into various aspects, then a function of these can be used to predict the overall rating. Another approach that has been taken is to classify 1 and 2 stars together, 4 and 5 stars together, in order to gauge the general opinion of the user. However, this falsely increases the accuracy of rating prediction. It is an analysis of the user's sentiment but should not be used for rating prediction tasks.

Work has been done to take this a step further as well. If a feature for some customization of a user is included, we can treat the reviews of each user as separate entities. There is expected to be a uniformity in the reviews that a user writes and different users have different ways of expressing the same emotion. [6] Other measures of evaluation such as precision and recall have been used for baseline comparison were used in combination with feature extraction methods such as the term frequency-inverse document frequency, the Latent Dirichlet Allocation and non-negative matrix factorization. These methods were studied, compared and evaluated using the Yelp data set. NMF with an added sentiment layer and tf-idf model produced better results [7]

7. RESULTS AND CONCLUSION

The results can be seen in two parts. The first part is a study of which feature out of the ones we have chosen leads to the highest accuracy in classification and least error. The second part is a study of which of the classification models that we have chosen is the right one to choose.

The data of 162K was divided into 80% training data and 20% test data. In evaluating which feature is the best, the first model that was studied was the Naive Bayes' classification. We observed that selecting the most common adjectives as the feature performed better than the other two models. This was in line with the expectation at the start of the process. This trend is mirrored in the SVM classification and linear regression as well, with accuracies computed as higher and the MSE computed as lower. This further made sense when we looked at the most informative features, as shown in listing 1 . These were the results obtained when the feature was selecting the most common words, without any restriction on part of speech. It was noticed that most of these words were adjectives. Since the other words do play a role in the feature, it is logical that if all the words in the bag of words that are selected as a feature were to be adjectives, the performance would improve. All errors reported are summarised in Table 3

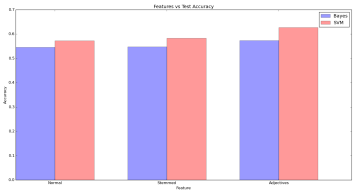

Figure 7: Accuracy vs features

Listing 1: Most informative features.

```
horrible = 2           1 : 5       =      322.3 : 1.0
terrible = 2           1 : 5       =      234.5 : 1.0
worst = 2              1 : 5       =      153.6 : 1.0
rude = 2               1 : 5       =      113.4 : 1.0
poor = 2               1 : 5       =      110.3 : 1.0
minutes = 4            1 : 5       =       84.6 : 1.0
great = 4              5 : 1       =       76.7 : 1.0
bland = 2              2 : 5       =       74.8 : 1.0
delicious = 2          5 : 1       =       74.1 : 1.0
awful = 2              1 : 4       =       72.6 : 1.0
worst = 1              1 : 5       =       66.8 : 1.0
told = 3               1 : 5       =       62.3 : 1.0
disappointing = 2      2 : 4       =       59.5 : 1.0
mediocre = 2           2 : 5       =       58.5 : 1.0
horrible = 3           1 : 5       =       52.5 : 1.0
rude = 3               1 : 4       =       50.7 : 1.0
unfortunately = 2      2 : 5       =       48.0 : 1.0
manager = 3            1 : 4       =       47.2 : 1.0
awful = 1              1 : 5       =       45.0 : 1.0
horrible = 1           1 : 5       =       39.7 : 1.0
```

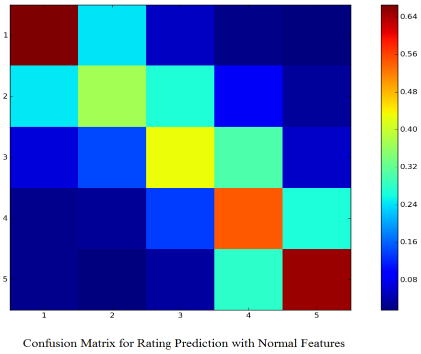

Figure 8: Confusion matrix for Naive Bayes model with normal feature representation

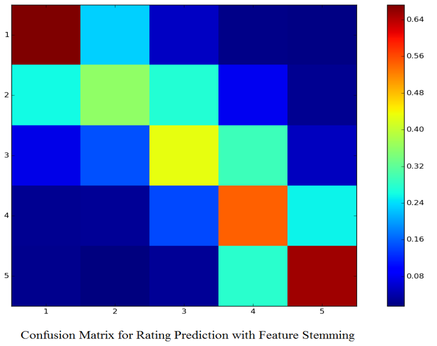

Figure 9: Confusion matrix for Naive Bayes model with stemming

While evaluating which of the two models is the best, we must compare the models while keeping the feature a constant. We see a common trend of SVM being better than Naive Bayes Classifier for all the features. This is because Naive Bayes falsely assumes that there is conditional independence between the features when it is not necessarily true. From Figure 7, we can see that SVM has slightly better accuracy for all the three sets of features. This is true for the MSE as well as seen in Table 3. We also looked at the confusion matrices of the various combinations of models and features and Figure 8 shows the confusion matrix plot of the Naive Bayes Classifier with the normal feature representation. We are not including regression in our comparison here as linear regression is not a method of classification. It predicts a value between two integers and was used as a means of comparison as to which feature was better as was seen in the previous part. Thus, even though the MSE of linear regression is lesser than that of Bayes and SVM, it is not the right way to solve this problem of rating prediction as we have assumed that the prediction must be a valid rating, that is, an integer.

In conclusion, our initial assumption and prediction that SVM Classifier with adjectives as the best feature holds true in terms of accuracy when compared to the other models and features that we have looked at.

The next part of our result was that of the classification of the reviews as funny, useful or cool. This time, a slightly different result was observed. The normal feature selection

Method	Feature	Training Accuracy	Test Accuracy	Training MSE	Test MSE
Naive Bayes	Normal	0.570822281167	0.545622289879	0.858901318571	0.891441399883
Naive Bayes	Stemmed	0.5702544496982	0.547590491128	0.848790989121	0.887166712796
Naive Bayes	Adjective	0.604289502045	0.57282859294	0.8294920583929	0.8628959202048
SVM	Normal	0.602858292944	0.57228482034	0.824917495832	0.852948592024
SVM	Stemmed	0.618492749293	0.582959294921	0.792847582824	0.829592015832
SVM	Adjective	0.668283935532	0.627329942532	0.783592052329	0.801341973298
Regression	Normal	-	-	0.680479136465	0.696500323356
Regression	Stemmed	-	-	0.664022418602	0.679323930374
Regression	Adjective	-	-	0.64729385292	0.65294742923
Baseline	-			1.73246546763	1.903547565

Table 3: Comparison of Testing Accuracy, Training Accuracy and MSE for Different Models and Features

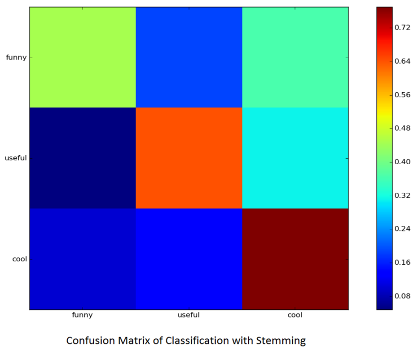

Figure 10: Confusion Matrix for Classification into Funny, Useful and Cool with normal feature respresentation

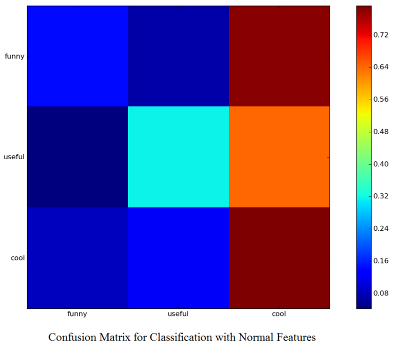

Figure 11: Confusion Matrix for Naive Bayes' Classification into Funny, Useful and Cool with stemmed feature respresentation

turned out to do better than the stemmed one. We plot the confusion matrix to inspect this further.

	Training Accuracy	Testing Accuracy
Normal	0.694145803339	0.655448358388
Stemmed	0.684740945721	0.641612444843

Table 4: Performance of Classification into Funny, Useful and Cool

The confusion matrices that were obtained for the above two cases (as seen in Figure 10 and Figure 11) was not of the nature that is normally expected. This, added to the unexpected trend of the normal feature set doing better than the stemmed one leads us to conclude that the features that are used for rating prediction cannot be used for the classification into funny, useful and cool. A more complicated set of features that does not just use word frequency distribution would be needed for this purpose. The main difference between rating prediction using review text and the funny, useful, cool classification using the text is that the text is a good indicator of what the user feels about the business. However, the text alone may not be good enough to predict what other users will feel about the review. We will have to model features such as the length of the review, the user's helpfulness history and features that do not involve just text mining. Thus, we conclude that text mining is a great way to predict the rating of the particular review. However, it is not enough to predict funny, useful and cool scores of the review as seen by other users.

8. REFERENCES

[1] Ganu, Gayatree, Noemie Elhadad, and Amélie Marian. ”Beyond the Stars: Improving Rating Predictions using Review Text Content.” WebDB. Vol. 9. 2009.

[2] Qu, Lizhen, Georgiana Ifrim, and Gerhard Weikum. ”The bag-of-opinions method for review rating prediction from sparse text patterns.” Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, 2010.

[3] Scaria, Aju Thalappillil, Rose Marie Philip, and Sagar V. Mehta. ”Predicting Star Ratings of Movie Review Comments.”

[4] Technical Report: http://www.ics.uci.edu/˜vpsaini/files/technical_report.pdf

[5] Yelp Challenge Presentation: http://www.ics.uci.edu/˜vpsaini/

[6] Li, Chen, and Jin Zhang. ”Prediction of Yelp Review Star Rating using Sentiment Analysis.”

[7] Chada, Rakesh, and Chetan Naik. ”Data Mining Yelp Data-Predicting rating stars from review text.”