Prediction of Star Ratings from Yelp User Reviews

Sanket Mhaiskar
Indiana University
Bloomington, Indiana
smhaiska@iu.edu

Akshay Kamath
Indiana University
Bloomington, Indiana
akkamath@iu.edu

Pratish Merchant
Indiana University
Bloomington, Indiana
pmmercha@iu.edu

Abstract

Yelp is a business review website that helps its customers to search for local businesses as per their requirement. Among other things, users mainly rely on the ratings that are displayed for a business as they potentially provide a quick summary of verbose reviews. Hence, we posit that summarizing reviews as ratings becomes an important task for automatic recommendation system. Additionally, the star ratings don't always reflect the actual content and complexity of user sentiment, which can be misleading for other users. To address these problems, we have used Machine Learning classification methods like Support Vector Machines and Naive Bayes using unigram, bigram, trigrams and a combination of unigram and bigram features to predict the overall star rating based on user review. Our results have shown that SVM performs better than Naïve Bayes on Yelp Reviews. Additionally, we perform sentiment analysis to find most positive and most negative words.

1 Introduction

Nowadays, all e-commerce websites and applications offer its users a review section and an overall star rating for their products and services. This is true for a product oriented site like Amazon, a social platform like YouTube or business oriented website like Yelp, all of which have deep integration of user reviews on their website. These reviews help other users in making decisions related to the product or service.

One such website is Yelp which provides its users a platform to express their personal opinions based on experiences and provide an overall star rating based on their experiences. Yelp provides this wide range of information in form of a dataset which contains 1.6M reviews divided across different business categories like Restaurants, Hotels, Shopping, Bars, Beauty & Spas, Nightlife, Health & Medical, Automotive, Home Services and Fashion.

A typical user review consists of opinion rich information typically containing an overall star rating, a text review which can either contain a user experience or a critical analysis of the product or service or both, some pictures that give an idea of the experience. Of these we are mainly interested in the user review text and its star rating.

For making an informed decision of selecting a business, users may want to check all the reviews before making a choice. Many a times, reviews can be verbose and discursive. In such cases, users rely on the star rating for making a quick decision. Hence predicting a star rating from a given review text becomes an important task in building product suggestion tools for users of such sites.

There are also some challenges that arise in such a task, for example, the user reviews don't always correspond to the overall star rating. A user may praise a product, but won't give a full star rating or berate a product and still give an average rating. Sometimes, the star rating can be completely missing. In such cases, it becomes important to determine the true nature of user review and map it to an overall star rating. The magnitude specified in star ratings helps in computing and engineering statistics and also helps users in filtering products.

For tackling the above problem, we have employed two Machine Learning Algorithms, Naïve Bayes and SVM. Initially ratings were classified into 2 categories. The results for Naive Bayes based on this was very low and that of SVM was 10% better than the majority class.

Then we extended our work to 5 star rating system, by considering unigrams, bigrams, trigrams and a combination of unigram and bigram. Even in this case, SVM performed better by 20% of the majority class and Naïve Bayes performed better by 10% compared to the majority class.

We also perform sentiment analysis on the review text, using NLTK library and Senti Words which perform tokenization and categorizing data into most positive and negative words based on frequency of words and strength of emotion. Also, parse the sentence and provide an overall score to the review based on individual sentence score.

2 Literature Review

Sentiment analysis using machine learning methods have been successfully explored in research space. In the scope of most of the works, sentiment analysis is treated as a classification task, usually using binary classifiers i.e. positive or negative. Zainuddin et al [6] follow this approach. They conclude with experimental analysis that by using Chi-Square feature selection may provide significant improvement on classification accuracy.

Among other algorithms, SVM and Naive Bayes have shown good performance across various domains. Fuchun Peng et al [4] and Jong, Jason et al [7]. Naive Bayes has proven to be relatively successful for simple classification tasks across wide domains (Fuchun Peng et al) [4]. These algorithms have also been successfully applied to research in opinion and review mining. Chen Li et al [8] have applied machine learning techniques for suggesting restaurants to a user.

Fangtao Li et al [1] have created a learning framework which includes reviewer and product information to generate a multi-dimension tensor to predict reviews.

Related research also suggests that Machine Learning algorithms outperform human produced baselines as suggested by Bo Pang et al [2].

Various techniques of creating custom features have been applied to these algorithms with favorable results for the task of classifying text data.

Apoorv et al. [3] explore the use of POS specific polarity to in tandem with a tree kernel to examine sentiments in twitter data.

Similarly, Fuchun Peng et al [4] explore the use of bag of words features as unigrams, bigrams and their combinations for Naive Bayes to achieve high performance.

Some other techniques include combining methods like effective negation handling, word n-grams and feature selection by mutual information to improve accuracy have been tried by Vivek Narayan et al [5][15].

For our project, we have explored machine learning algorithms and used a variety of techniques and found that combining unigrams and bigrams along with Support vector machines have produced good results.

3 Data

For our experiments, we have used a dataset provided by Yelp for its Dataset challenge, which can be found online. All the data is provided in JSON format and is classified into Business, Check-in, Reviews, Tips and User information. The data consists of 1569264 reviews, approximately 500,000 tips by 366000 users for 61000 businesses.

The distribution of review data compared to star ratings in the data set as described in Figure 1. We prepared a custom dataset by extracting 25000 reviews and their corresponding star ratings. The distribution of stars for 25k records is shown in Figure 2.

Our initial approach involved predicting the review as positive or negative (Thumbs up-Thumbs down model) .We labelled star ratings in 2 categories: 1 and 2 as negative (Thumbs down) and star ratings 3, 4 and 5 as positive (Thumbs Up).

This approach was then extended to 5 category model, where we considered the star ratings as labels for a review. For e.g., a review with 4 stars would be classified in 4th category.

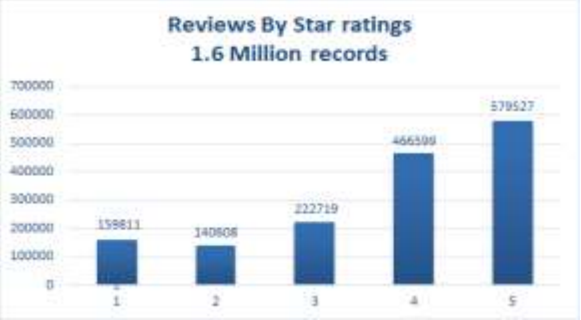

Figure 1

Additionally, we cleaned the data by removing stop words and other fields that were not relevant to sentiment analysis and rating prediction.

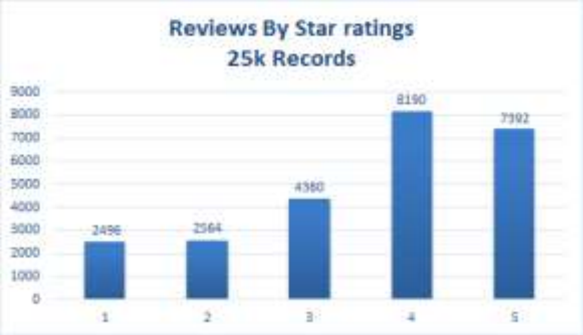

Figure 2

4 Method and Data Splitting

For the purpose of star rating prediction and review classification into positive (Thumbs Up) and negative (Thumbs Down) categories, we have used supervised machine learning approach. For the purpose of training on data, we have made the use of Support Vector Machine and Naive Bayes classifiers using Scikit Learn Library. The TfidfTransformer function first computes the term frequency and inverse document frequency for each word in the corpus.

tfidf_dataSVM=TfidfTransformer (use_idf=True, smooth_idf=True).fit_transform (dataSVM)

We split the data into 80% Training and 20% Test, of the total data using the count vectorizer function of Scikit Learn.

data_train, data_test, target_train, target_test = cross_validation.train_test_split (dataNB, target, test_size=0.2, random_state=37)

The ngram_range parameter specifies the lower and upper boundary of the range of n-values for different n-grams to be extracted. All values of n such that min_n <= n <= max_n will be used.

count_vectorizerSVM = CountVectorizer (binary='false', ngram_range= (1, 2))

We tested and compared the accuracy of both the Classifiers on unigram, bigrams, and trigrams as features by playing with the ngram_range parameter. But it was observed that the classifiers performed better when a combination of unigrams and bigrams as features was used.
As advised by the Professor, we also used 10 fold cross validation for data splitting and training using the cross_val_predict function to calculate the mean accuracy for 10 folds.

Predicted = cross_validation.cross_val_predict (svm, reviews, stars, CV=10, n_jobs=8)

Support Vector Machine classifier performed better than the Naive Bayes classifier irrespective of the feature vector. The percentage of the Majority Class was used as the baseline.
For mining positive and negative words, we employed sentiment analysis techniques using methods from NLTK library and comparing it with a dictionary called Senti-Words. The tokenized words were then classified into positive and negative words by giving it a score and classified based on frequency and strength. These words were plotted on a word cloud using Kumo Library.

5 Evaluations

As discussed above, we have used two approaches of Naïve Bayes and SVM approaches for both the prediction categories. The baseline for all results was the majority class which was 79% for 2 categories and 32% for 5 categories.

5.1 Results for Prediction on Thumbs Up\Down.

- Bernoulli Naive Bayes Result

Accuracy	Majority Classifier
80.10%	**79%**

	Precision	Recall	F1Score	Support
Thumbs Down	0.50	0.43	0.46	1487
Thumbs Up	0.86	0.89	0.88	6014
Avg./Total	0.79	0.80	0.80	7501

- Support Vector Machine Result

Accuracy	Majority Classifier
90.79%	**79%**

	Precision	Recall	F1Score	Support
Thumbs Down	0.89	0.61	0.72	1487
Thumbs Up	0.91	0.98	0.94	6014
Avg./Total	0.91	0.91	0.90	7501

5.2 Results for Prediction on 5 categories section

- Bernoulli Naive Bayes Result

Accuracy	Majority Classifier
43.93%	32%

	Precision	Recall	F1Score	Support
1	0.55	0.32	0.40	706
2	0.28	0.13	0.17	718
3	0.34	0.17	0.22	1343
4	0.43	0.49	0.46	2465
5	0.47	0.70	0.56	2206
Avg./Total	0.42	0.44	0.41	7501

- Support Vector Machine Result

Accuracy	Majority Classifier
56.86%	32%

	Precision	Recall	F1Score	Support
1	0.75	0.66	0.70	250
2	0.50	0.27	0.35	249
3	0.53	0.35	0.42	469
4	0.49	0.69	0.57	786
5	0.67	0.65	0.66	747
Avg./Total	0.58	0.57	0.56	2501

- SVM using K-Fold Cross Validation

Accuracy	Majority Classifier
52.56%	32%

	Precision	Recall	F1Score	Support
1	0.64	0.77	0.70	2495
2	0.49	0.22	0.30	2564
3	0.48	0.35	0.40	4360
4	0.48	0.52	0.50	8190
5	0.56	0.66	0.60	7392
Avg./Total	0.52	0.53	0.51	25001

- N- Gram Summary

	Naïve Bayes Accuracy	SVM Accuracy
Unigram	42.61%	55.31%
Bigram	41.35%	53.57%
Trigram	33.73%	41.75%
Unigram + Bigram	43.93%	56.85%

- Comparison between Naïve Bayes and Support Vector Machines for star rating prediction.

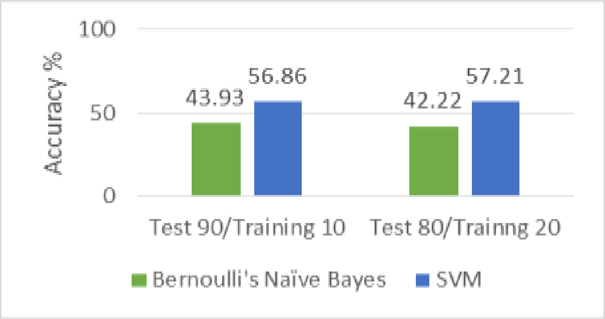

Figure 3

5.3 Word cloud based on polarity of words from text.

To further visualize polarity in the review text, we have generated the word clouds for most positive and most negative words generated by assigning scores to each word occurring in a review text using SentiWord as denoted in figure 4. Additionally, we also generate a word cloud for most frequently occurring positive\negative word in yelp review texts.

In both figure 4 and figure 5, text in red denote words with a negative polarity and the text in green denote words with a positive polarity.

- Most positive-negative words used in reviews.

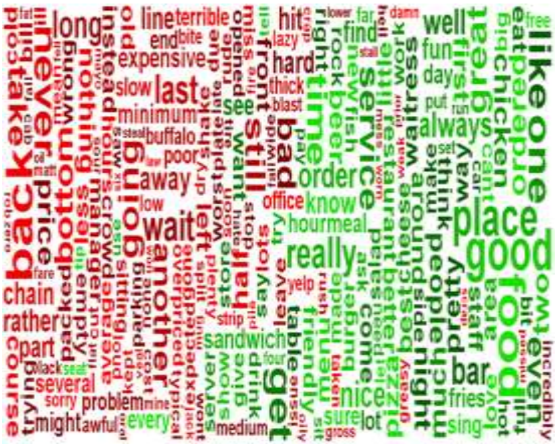

Figure 4

- Most frequently occurring positive-negative words used in reviews

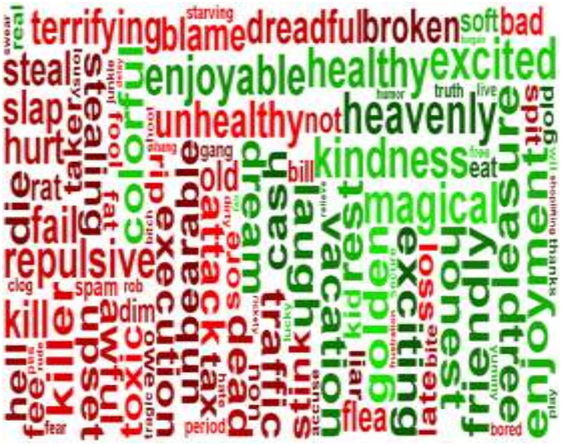

Figure 5

6 Conclusion

In this paper we have presented various approaches to predict star ratings based on the review text. In doing so, we experimented with multiple classification algorithms like Support Vector Machines and Naïve Bayes and Tf-idf weighting scheme. First we trained and tested the classifiers only for two labels positive and negative and achieved a good accuracy of prediction of about 90% for around 25000 reviews. We then extended the number of classes to 5, where each star rating was a label. It was observed that Support Vector Machine classifier does a better job of prediction as compared to Naïve Bayes when using the same parameters and feature vectors, in both the cases. We also experimented with different feature extraction models like unigrams, bigrams and trigrams. But a combination of unigrams and bigrams as features produced a better accuracy in prediction.

The focus of this paper was to analyze the sentiment in the Yelp business reviews but it can also be extended to any other domain which has similar model. For future work, one can experiment with preprocessing and feature extraction to further improve the accuracy of the prediction .Prediction of star ratings can in future help in designing a more accurate recommendation system. Analysis can further help in reducing and curbing fake reviews and also verifying their validity. The user data can also be used to predict the authenticity of the review and can help in the validity of the rating and thus help in building a better recommendation system.

7 Reference

[1] Li, Fangtao, Nathan Liu, Hongwei Jin, Kai Zhao, Qiang Yang, and Xiaoyan Zhu. "Incorporating Reviewer and Product Information for Review Rating Prediction." *Twenty-Second International Joint Conference on Artificial Intelligence*.

[2] Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs Up? Sentiment Classification Using Machine Learning Techniques." *Thumbs Up? Sentiment Classification Using Machine Learning Techniques.*

[3] Agarwal, Apoorv, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. "Sentiment Analysis of Twitter Data." *Sentiment Analysis of Twitter Data*

[4] Peng, Fuchun, and Dale Schuurmans. "Combining Naive Bayes and N-Gram Language Models for Text Classification."

[5] Narayanan, Vivek, Ishan Arora, and Arjun Bhatia. "Cross-domain Sentiment Classification Using an Adapted Naïve Bayes Approach and Features Derived from Syntax Trees." *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval and the International Conference on Knowledge Management and Information Sharing* (2013)

[6] Zainuddin, Nurulhuda, and Ali Selamat. "Sentiment Analysis Using Support Vector Machine." *2014 International Conference on Computer, Communications, and Control Technology (I4CT)* (2014)

[7] Jong, Jason. "Predicting Rating with Sentiment Analysis." *Springer Reference* (2011)

[8] Li, Chen. Prediction of Yelp Review Star Rating Using Sentiment Analysis

[9] Fan, Mingming, and Maryam Khademi. *Predicting a Business 'Star in Yelp from Its Reviews'*

[10] Gamallo, Pablo, and Marcos Garcia. "Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets.

[11] Rohini S Rahate and Emmanuel M. Article: Feature Selection for Sentiment Analysis by using SVM. International Journal of Computer Applications 84(5):24-32, December 2013.

[12] Mullen, Tony, and Nigel Collier. "Sentiment Analysis using Support Vector Machines with Diverse Information Sources."

[13] Vovsha, Apoorv Agarwal Boyi Xie Ilia, and Owen Rambow Rebecca Passonneau. "Sentiment Analysis of Twitter Data."

[14] Zhang, Yinshi. "Semantic Feature Analysis and Mining for Yelp Rating Prediction."

[15] Narayanan, Vivek, Ishan Arora, and Arjun Bhatia. "Fast and accurate sentiment classification using an enhanced Naive Bayes model." arXiv preprint arXiv: 1305.6143 (2013).

8 Acknowledgements

We would like to extend our thanks to Professor Muhammad M. Abdul-Mageed for providing us with guidance and motivation for pursuing this project to completion.

9 Bio

Akshay Kamath

- **M.S Computer Science**
- **Interests:**

1. Academic: Design of Algorithms, Machine Learning, Natural language Processing and Software Engineering.
2. Personal Interests: Listening to music, Reading fantasy fiction novels, playing table tennis.

- **Contributions:**

1. Implemented Naïve Bayes classification using NLTK for classifying Thumbs Up\Thumbs Down and documented results.
2. Modified code provided in course work for SVM to work with the Yelp Data Set and identified that performance suffers for review set of more than 5000.
3. Implemented K-Fold cross validation using Sci-Kit learn and documented results.
4. Performed POC to do additional tasks on the Yelp data set to identify state wise trends for a particular business and visualize it on map using Fusion Charts. This was not included in our final work, however, it can be considered for extended work in Yelp data set challenge.
5. Tried a POC with word cloud in Python, however this was not used in the end as better alternatives were identified.

Sanket Mhaiskar

- **M.S Computer Science**
- **Interests:**

1. Academic: Data Mining, Cloud Computing and Mobile Computing.
2. Personal Interests: Playing Soccer, Participating in Hackathons and E-Sports

- **Contributions**:

1. Mined and cleaned the data by removing stop words and irrelevant fields using NLTK.
2. Implemented Support Vector Machine and Naïve Bayes Classifiers from the Scikit Learn Library for rating prediction into 5 categories.
3. Tried experimenting with other classifiers like Multinomial Naïve Bayes and K-Neighbors Classifier for better accuracy.
4. Tried to infer patterns in the dataset by trying to identify the Time and day for the most check-ins in a Thai Restaurant from the Check-ins data in the Yelp Dataset.
5. Also implemented code to auto-label star ratings into positive or negative.

Pratish Merchant

- **M.S Computer Science**
- **Interests** :

1. Academics: Data Mining, Artificial Intelligence, Machine Learning.

2. Personal: Music enthusiast, like to explore new destinations and enjoy any good sport that can revitalize the mind and body.

- **Contributions**:

1. Breakage of reviews into sentences and cleaned the data by removing stop words and irrelevant words using NLTK library
2. Implement dictionary of words using SentiWord based on SentiWordNet library.
3. Sentiment analysis and scoring of words in a sentence using NLTK library and SentiWord to find most positive and negative words based on frequency and also on strength of words.
4. Plot the polarity of words using Kumo library which is open source library in JAVA.
5. Research, data gathering and validation of results.
6. Also, implemented an overall sentence scoring algorithm based on score from words which outputs a score out of 5. However, this has not been implemented in project as complete statistical analysis could not be performed due to time constrains.

10 Suggestions by Professor

- Verify results using K-fold cross validation.

  We have implemented the same and documented the results for predicting star ratings using SVM as this was the better performer out of the two.