Predict Yelp Star Ratings From Text Reviews

Leyao Li

Objective

Review websites, such as Yelp and Google Reviews, are the major sources of information where users turn to when choosing between similar services and products. Among various functions present in these review websites, star rating is the most powerful factor in determining a user's first impression on a business. This project aims to shed light on the business's ratings on Yelp using user's textual reviews and built classifiers to predict the star ratings. Furthermore, this project will investigate the ability for each model developed to distinguish between 1-star and 5-star ratings and below 2-star and above 4-star ratings.

1 Dataset

The dataset used in this project was collected from Yelp which is available at Yelp's website. It is a subset of Yelp's businesses, reviews and user data, which covers over 8,000,000 reviews for 160,585 businesses. For the purpose of this project, 5000 sample reviews from the *review.json* file were randomly chosen and only columns containing text review and stars were used in this project. The sampled dataset used is available in the repository. A plot that shows the distribution of star ratings is shown in *Figure 1*. As *Figure 1* shows, 5-star ratings are the major ratings, and 5-star and 4-star ratings make up two-thirds of the 5000 samples.
To evaluate the results the data was divided into 75% training and 25% testing. For the training dataset, both text reviews and business's star ratings were included and for the testing dataset. Only text reviews were provided to the testing dataset and models were trained to predict the star ratings on the testing dataset, which were then compared to the actual ratings.

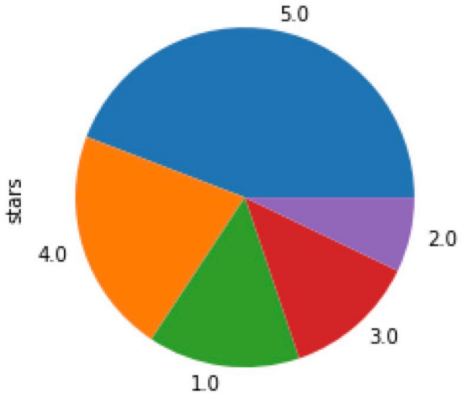

Figure 1: Distribution of Star Ratings in the Dataset. 4-star and 5-star ratings make up two-thirds of the 5000 samples.

2 Background

There has been some researches on this set of data and results from this project can be compared to those from the already published ones.
In the research carried out by Fan et al. (Fan & Khademi, 2014), they developed a support vector regression model, a linear regression model and a decision tree to predict business's ratings. They used RMSE as the evaluation metric and the linear regression model had the best performance among all. From there they concluded that there might exist a linear relationship between the features in the text reviews and the stars a business received. Therefore, this project would implement a linear SVM, which also assumes linear relationships between predictors and the label, to model the Yelp data.
Asghar (Asghar, 2016) modeled this dataset with several classifiers including a logistic regression model and a linear SVM. The labels used in Asghar's research are integers of {1,2,3,4,5}. The input of the logistic regression Asghar used is also *TFIDF* vectors and it achieved highest accuracy in predicting the labels among all models, which had a 64% accuracy. The linear SVM performed well

too, with overall accuracy of 63%. Given the close performance of the two models, this project aims to divide the tasks into 2 parts, one is to predict the ratings with true label as {1,5} and the other is to predict the ratings with true label as {below 2, above 4} to compare these two models' ability to distinguish between good and bad ratings.

A study carried out by Lakosona et al. (Laksono et al., 2019), they built a Naive Bayes classifier for classifying sentiments based on reviews on Tripadvisor. The Naive Bayes classifier from their research had a slightly better performance than Textblob, a Python based sentiment analyzer. Given the wild application and successes of Naive Bayes in text classification, this project will implement a Naive Bayes classifier as the baseline model.

3 Methodology

In this paper, 10 different prediction models were built by combining each of 2 feature extraction methods with each of 3 distinct supervised learning algorithms.

3.1 Word Representation

The feature vectors fed into the classifiers were composed using 2 methods: TF-IDF and Word Embeddings. The hypothesis proposed is that models with Word2vec word embeddings as input feature vector will achieve better performance as it would be able to capture semantic similarities.

3.1.1 Term Frequency-Inverse Document Frequency(TF-IDF)

TF-IDF measures how many times a word appears in a document and the inverse document frequency of the word across a set of documents. The TF-IDF representation of words is intended to reflect single word's importance to a document in a corpus. TF-IDF vectors are widely applied and are computationally effective. Using TF-IDF vectors allows for more importance given to less common words.

3.1.2 Word Embeddings

Word2vec uses neural network to learn word associations from a large corpus. Word2vec represent each word in a list of numbers. Once trained, the model will be able to detect semantic similarities between words represented by the vectors.

3.2 Classifiers

Three supervised learning algorithms would be implemented. The h

3.2.1 Naïve Bayes Classifier

A Naïve Bayes Classifier makes the assumption that there exists conditional independence between any pair of features. It makes predictions based on joined probability calculated. Naïve bayes model was fed with TF-IDF vectors only, and was treated as a baseline model for this project.

3.2.2 Logistic Regression

In logistic regression models, the conditional probability that a feature vector for review ***r*** belongs to star rating labeled with {1-star,5-star} or {<=2-star, >=4-star}.

3.2.3 Linear Support Vector Machine(SVM)

A linear SVM learns from the training dataset and creates a hyperplane that best separates the data into different classes. Linear SVMs are helpful in text classifications since they reduce the need for labeled training instances (Joachims, 1998). Linear SVMs are also applied in shallow semantic parsing. (Pradhan et al., 2004)

3.3 Preprocessing

2 experiments were performed in this project. In the first experiment only 1-star and 5-star ratings from the 5000 samples were used to train and test the 3 classifiers. In the second experiment, reviews with <=2-star ratings and >=4-star ratings were used to train and test the 3 classifiers. The purpose of these 2 comparisons is that the 1-star and 5-star reviews are of the more extreme ratings and a hypothesis proposed is that <=2-star and >=4-star ratings would be more difficult for classifiers to differentiate than the 1-star and 5-star ratings. This comparison allows for another view to compare classifiers' ability to differentiate classes.

4 Results

The results for classification on the 1-star and 5-star reviews for Naïve Bayes with TF-IDF, Logistic Regression with TF-IDF, Linear SVM with TF-IDF, Logistic Regression with Word2vec

and Linear SVM with Word2vecare shown in *table 1*. Based on the results shown, for the classification of 1-star and 5-star reviews, Linear SVM with TF-IDF vectors had the best performance, with precision, recall and F-1 scores of 0.95. The performance of Naïve Bayes classifier with TF-IDF vectors and the performance of Logistic regression classifier with TF-IDF vectors are very close to that of the Linear SVM with TF-IDF vectors, while the 2 models with Word2vec both had less ideal performances in terms of precision, recall and F-1score.

Model	Precision	Recall	F-1
Naïve Bayes + TF-IDF(baseline)	0.92	0.92	0.92
Logistic Regression + TF-IDF	0.93	0.93	0.93
Linear SVM + TF-IDF	0.95	0.95	0.95
Logistic Regression + Word2vec	0.79	0.8	0.75
Linear SVM + Word2vec	0.79	0.77	0.7

Table 1: Classification Results on 1-star and 5-star reviews for Naïve Bayes with TF-IDF, Logistic Regression with TF-IDF, Linear SVM with TF-IDF, Logistic Regression with Word2vec and Linear SVM with Word2vec.

For the classification of <=2-star and >=4-star reviews, Linear SVM with TF-IDF vectors also had the best performance, with precision, recall and F-1 scores of 0.93. The performance of Logistic regression classifier with TF-IDF vectors had very close scores to those of the Linear SVM with TF-IDF vectors, while the 2 models with Word2vec both had less ideal performances in terms of precision, recall and F-1score.

Model	Precision	Recall	F-1
Naïve Bayes + TF-IDF(baseline)	0.89	0.89	0.89
Logistic Regression + TF-IDF	0.91	0.91	0.91
Linear SVM + TF-IDF	0.93	0.93	0.93
Logistic Regression + Word2vec	0.76	0.77	0.72
Linear SVM + Word2vec	0.76	0.75	0.65

Table 2: Classification Results on <= 2-star and >=4-star reviews for Naïve Bayes with TF-IDF, Logistic Regression with TF-IDF, Linear SVM with TF-IDF, Logistic Regression with Word2vec and Linear SVM with Word2vec.

5 Discussions

The results in this project show that the Logistic Regression model with TF-IDF vectors as input features for classifying 1-star and 5-star reviews achieved the best performance, with precision, recall and F-1 scores of 0.93. For the classification of <= 2-star reviews and >= 4-star reviews, the Logistic Regression with TF-IDF vectors still achieved best performances among all 5 models. Models with Word2vec vectors as input failed to exceed the performances of models with TF-IDF vectors as input in both classification tasks, and they also failed to exceed the performance of the baseline model. The hypothesis proposed in this project that linear SVM would had the best performance regardless of type of input vectors was confirmed correct. However, the hypothesis that models with Word2vec as input feature vector would perform better to the models with TF-IDF vectors was incorrect. This potentially suggests that in the classification of review's star rating, the appearance of certain single words might be the determinant factor of the final star-rating associated with a review.

It is also found that although linear SVM outperformed logistic regression models in both classification tasks, the difference in the performance is small. Considering the time to train both models, it would be computationally preferable to choose logistic regression models with TF-IDF vectors on especially large dataset.

Since the dataset used in this project is highly imbalanced, with over 2/3 of the data labeled with >= 4-star ratings, one further direction of this study is to collect, train and test on more balanced data.

References

Asghar, N. (2016). Yelp Dataset Challenge: Review Rating Prediction. *ArXiv:1605.05362 [Cs]*. http://arxiv.org/abs/1605.05362

Fan, M., & Khademi, M. (2014). Predicting a Business Star in Yelp from Its Reviews Text Alone. *ArXiv:1401.0864 [Cs]*. http://arxiv.org/abs/1401.0864

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In C. Nédellec & C. Rouveirol

(Eds.), *Machine Learning: ECML-98* (pp. 137–142). Springer. https://doi.org/10.1007/BFb0026683

Laksono, R. A., Sungkono, K. R., Sarno, R., & Wahyuni, C. S. (2019). Sentiment Analysis of Restaurant Customer Reviews on TripAdvisor using Naïve Bayes. *2019 12th International Conference on Information Communication Technology and System (ICTS)*, 49–54. https://doi.org/10.1109/ICTS.2019.8850982

Pradhan, S. S., Ward, W. H., Hacioglu, K., Martin, J. H., & Jurafsky, D. (2004). Shallow Semantic Parsing using Support Vector Machines. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, 233–240. https://www.aclweb.org/anthology/N04-1030