Yelp Review Prediction

John S. Burke, Sherman Y.L. Sze, Hien D. Nguyen
jsburke@bu.edu, ysze@bu.edu, heinous@bu.edu
Boston University

Abstract

In the era of digital information, one efficient approach that businesses utilize to attract recognition is to maximize their overall online review ratings. Yelp provides an extensive open-source dataset that contains users' reviews of many restaurants. By applying Machine Learning techniques to Yelp's data, we aim to develop a model that predicts the rating of an input review based on its provided text.

1 Introduction

Many organizations and companies collect user feedback and reviews to inform others of product and service quality; these reviews can also be of value to the organizations themselves. However, review text poses the general problem of being widely varying in terms of length, content, and quality, which presents a multitude of problems for computational evaluation. Possible applications of learning over such reviews and feedback can range from targeting advertisements to fraudulent user detection.

The first major challenge present in textual analysis of this nature is finding a scheme to classify the reviews under. Many companies such as Amazon and Yelp allow users to rate a product or businesses like restaurants. Yelp in particular rates services between one and five stars in tandem with the review text written by the user. Intuitively, given a particular language, certain elements of a given review will correlate with the rating accompanying it such that certain features of a one star review will be peculiar to one star reviews and likewise a five star review and its associated text.

In this paper we leverage machine learning algorithms to classify reviews into their associated rating. The problem is approached from two angles. The first utilizes a Support Vector Machine and the second an Artificial Neural Network. Under both methods, machine learning prediction vastly exceeds that of random guessing or simply always guessing one of the five ratings.

2 Setting and Algorithms

2.1 Yelp's Data and Preprocessing

Yelp provided five datasets for their dataset challenge in the ninth round. Of particular interest for this project were the `yelp_academic_dataset_review.json` and the `yelp_academic_dataset_business.json` datasets. We focused this study only on restaurants from Phoenix, AZ, so first the business dataset was filtered for establishments that met those criteria. The restaurants' identification field was then used to find associated reviews from the review dataset. Next, the review text and rating (the `text` and `stars` fields in the json, respectively) were extracted from those objects. The review text itself was then cleansed by removing escape sequences, punctuation, numbers, and other symbols that do not carry germane lexical information. Contractions, however, were not expanded.

2.2 Text Conversion

In order for the data to be usable, it has to be presented in a fashion that machine learning algorithms can handle effectively. Yelp's review data skews toward higher ratings strongly; five star reviews alone constitute forty percent of reviews. In order to ensure the algorithms will be trained properly, the reviews are filtered so that each of the five ratings has roughly equal representation. Furthermore, the ordering of these reviews is then shuffled so that certain ratings or businesses which may have clustered in Yelp's files do not bias the predictive systems in unforeseen ways.

Next, the corpus of the review text is transformed into a Document Term Matrix (DTM). In a DTM, each subvector represents a review as a feature vector, with each feature being a given word. This value is calculated by means of Term Frequency - Inverse Document Frequency (TF-IDF) as follows [2]:

$$tfidf(t, r, C) = tf(t, r) * idf(t, C)$$

where `t` is a given word, `r` is a given review (a *document* in normal technical terms), and `C` is a *Corpus* which is the whole of the reviews being predicted over. The `tf()` is the term frequency which represents the importance of a word in a review by means of how often it appeared. `idf()` is the inverse document frequency, which scales down the impact of words which occur frequently across the whole corpus. The resulting feature vectors are then restricted to 1,000 elements in addition to the tool also removing English 'stop-words' such as articles, prepositions, and other things that carry little lexical weight. The resultant DTM is associated with thee vector of labels, which together are conducive for use in machine learning algorithms.

2.3 Learning Models

As discussed earlier, the goal is to predict the rating of a review given only the review text. Therefore, the task at hand can be approached as an ordinary classification problem, where each review is to be classified into one of five categories (1-5 stars). In dealing with text categorization, it is of paramount importance that a learning model is able to acknowledge and generalize in high dimensional feature spaces. After conducting extensive literature review, we decided that the Support Vector Machine (SVM) classifier and the Multilayer Perceptron (MLP) classifier are appropriate models for text classification and analysis.

The data that was used for the two aforementioned learning methods consisted of 50,000 user reviews from Phoenix, AZ. Both models were trained over 80% of the data and then tested over the remaining 20% of the data. The training and test accuracy as well as mean squared error is reported in each case and the best model is recommended.

2.3.1 Support Vector Machine

SVMs were chosen as one of two learning models to classify text because they have proven to be both theoretically and empirically well suited for text categorization. While theoretical analysis concludes that SVMs have the ability to generalize high dimensional feature spaces well, experiments have further shown that SVMs consistently achieve good performance in text classification tasks, outperforming many other existing classification models [3].

Using the `LinearSVC` and `SVC` packages available in `scikit-learn` in Python, we classified Yelp review texts using the linear SVM kernel as well as polynomial kernels of different orders. The regularization parameter C was also varied in order to come up with the best SVM model. Due to time and computational constraints, the Gaussian kernel was not attempted in this project.

2.3.2 Artificial Neural Network

For the purpose of comparisons and optimizations, we developed an Artificial Neural Network (ANN) as a second learning model for text classification. While SVMs are considered to be more suitable for this task, researchers have shown that ANN models can statistically have equivalent performance compared to SVMs [4].

Our ANN model uses the same train-test data split as the SVM. We implemented our model using Multi-Layer Perceptron from the widely used `scikit-learn` Python package. Our neural network

uses the sigmoid activation function, a single hidden layer of 100 neurons, and a quasi-Newtonian solving method. Optimization tolerance was set at `1e-4` for a constant learning rate.

3 Results

Given that the input datasets are balanced across all five classes, random guessing would result in predictions of approximately 20% accuracy. Similarly, guessing one label every time would produce a similar result. This sets a baseline to surpass. With 50,000 input vectors split along the aforementioned training-testing ratio, the SVM with linear kernel and regularization parameter equal to 1 outperformed all of the attempted polynomial kernels, averaging 55.58% accuracy with a mean squared error of 0.9624 "stars". Likewise, the Neural Network achieves a mean accuracy of 55.81% with a mean squared error of 0.8662 "stars". The observed maximum accuracy was 58.2% for the linear SVM and 60.2% with the Neural Network. Thus, both models far exceed the 20% minimum established by random guessing.

Furthermore, it is observed that, unsurprisingly, that as the training set grows, so does the accuracy; however, this growth ought to be noted as a situation of diminishing returns. The accuracy achieved at 50,000 inputs is also very close to previously established work, which will be discussed later. The table below illustrates the rough accuracies for various input sizes with the same 80:20 training to testing split:

Table 1: Accuracy of methods over growing input sizes

Input Size	SVM	NN
1,000	.4523	.4251
5,000	.4891	.4655
10,000	.5270	.5218
50,000	.5558	.5581

Both the accuracy and mean squared error place these result roughly amongst what has been achieved in previous work of a similar nature that filtered review words in slightly different ways [1]. Furthermore, this adds the Multi-Layer Perceptron approach as another valid means for approaching this type of problem.

4 Future Work

Several design options were not pursued in this project that might produce better predictions. It can be asserted that filtering for certain lexical categories, such as adjectives, can improve accuracy [1]. This along with the removal of concatenative morphological affixes through word stemming may allow TF-IDF to produce a matrix that is easier to train with. Extending the feature vector, using different SVM kernels, and more elaborate hidden networks in the MLP, which were not done here due to computational and time constraints, also arguably hold potential for more accurate predictions. Other items, such as reviews being marked 'funny' among other parameters, may have information that these models can exploit.

It also stands to reason that this should be extensible to reviews of places that are not restaurants, such as places like hardware stores or car dealerships. Since these are in a different sector than food service, the models trained here specifically would probably do less favorably, but retraining them either for specific business types or for any business in general would be an easy next step for direct utilization of this work.zation of this work.

5 Conclusion

We addressed the problem of analyzing a review on Yelp to predict a rating through the two step method of transforming a given text into a numeric structure and using that to train machine learning algorithms. Though this treatment only classifies the reviews into the associated review system, it shows that the text reviews on Yelp can be classified, which may have applications in detecting misuse of such utilities by users or other entities. The methods presented perform significantly better than random or obvious simplistic approaches such as always guessing one rating, and, if extended with

more robust computational systems could produce further benefits with minimal effort; investigation and extension of the models presented here hold potential for immediate usage in industry.

6 References

[1] *Prediction of rating based on review text of Yelp reviews*, Channapragada and Shivaswamy

[2] *TF-IDF*, Wikipedia: TF-IDF

[3] T. Joachims. *Text categorization with support vector machines: Learning with many relevant features. Technical Report 23*, Universitat Dortmund, LS VIII, 1997.

[4] Zaghloul, Waleed, Sang M. Lee, and Silvana Trimi. *Text classification: neural networks vs support vector machines.* Industrial Management & Data Systems 109.5 (2009): 708-717.