Restaurant Strategy Selection Based on Yelp

Hongyang Yu
hy268@rutgers.edu

Yuzhuo Li
yl976@rutgers.edu

Abstract

Nowadays, lots of people want to start their own business. Even though they did the market investigation before start a business, there are still many risks that they cannot make sure if the customer like their product or not. What we did is finding which kind of business has more possibility to get success. By providing several strategies, we can find a better strategy for those people to help them avoid the risks.

1 Introduction

For a new opened restaurant, a bar or a dessert shop, the most significant element is absolutely the quality of food and service, but other elements including location, product type, and accessory facility, also make important effect on the profit. So we try to analyze which kind of element will make the most significant effect.

In this project, we provide various elements as features, such as location, accessory facility, cuisine and dish name. Based on these elements, after people choose the elements they will have in their new restaurant, we will give you a final evaluation presented by a number. To make this goal, we will extract top 300 noun phrases from the review which means these noun phrases are the most popular. Also, we will consider the attributes and categories every shop has. For all these factors, using deep learning and random forest regression model to train the data and then get the evaluation. Because the data is so large and to handle a much bigger dataset, we use Spark [6] to extract data, analyze the data and train the model.

Our dataset includes different fields like restaurant, dessert shop, bar etc. Based on this data, our training model can fits for different fields.

2 Data

We used three datasets in our work. One is the business dataset including elements: business id, name, address, city, state, latitude, longitude, attributes, stars, categories. In our work, we processed data state by state.

The second dataset we will use is the review dataset. It includes the elements: review id, business id, stars, text, useful, funny, and cool. We will extract the useful information from the text for each business id.

The third dataset is tip dataset including text and business id. Tips also gives us very important information for the business, especially sometimes, tips are the simplicity version of the reviews. The data can be downloaded from Yelp [1].

3 Feature Extraction

3.1 Category Features

A category is a list of string, which is a succinct description of a restaurant. We transformed the category to a unique fixed size numeric vector for each business on spark. In our project, the dimension of transformed numeric vector is chosen 20.

3.2 Attribute Features

First, we extracted all attributes. In our dataset, there are 39 different attributes. The type of attribute is a wrapped string array. Some attributes contains multiple subattributes. For example, parking condition contains parking lot condition, validated condition, valet condition, etc. The feature extraction idea is the same as bag of words. We count each attributes and make a fixed size vector to represent the features.

3.3 Review Text Parsing

We need to find all the noun phrases in the review. Why we want to find only the noun phrase is because noun phrase usually represents the featured item in this shop like the dish of the restaurant menu or the specific wine of the bar. We use the parse function in openNLP [7].

For example, we have a sentence “The quick brown fox jumps over the lazy dog”. After tagging we can get “Top(NP(NP(DT The)(JJ quick)(JJ brown)(NN fox)(NNS jumps))(PP(IN over)(NP(DT the)(JJ lazy)(NN dog)))(..)))” You can find that we only need to set the condition to choose the word with attribute NN, then we can get the noun phrases. In our work, the input is all the sentences and the output is the noun phrases.
After getting the noun phrases, in case, we also do the tokenization. Then we count and sort the word based on their frequency by using map and reduce. If we get too many features, we can only limit our features to any number we want. In our work, we limit it to 300.

With the noun phrase, we handle our training data again that for each business id and the corresponding review and tip text, we delete the words not included in the 300 noun phrases.

4 Model and algorithm

4.1 Word2Vec

Word2vec [5] is a group of related models that are used to produce word embedding. These models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, typically of several hundred dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are located in close proximity to one another in the space.

With this algorithm, we can transfer the extracted noun phrases, attributes and categories to vectors. The input is the data frame with two columns, one is the business id and the other is the list of words. The output is the data frame with two columns, one is the business id and the other is the list of vectors.

4.2 Random Forest Regression

Firstly, we tried random forest to do regression. The random forest regression algorithm [2] is as follows:

1. Draw n_tree bootstrap samples from the original data.
2. For each of the bootstrap samples, grow an unpruned classification or regression tree, with the following modification: at each node, rather than choosing the best split among all predictors, randomly sample m_try of the predictors and choose the best split from among those variables. (Bagging can be thought of as the special case of random forests obtained when m_try = p, the number of predictors.)
3. Predict new data by aggregating the predictions of the n_tree trees (i.e., majority votes for classification, average for regression).

An estimate of the error rate can be obtained, based on the training data, by the following:

1. At each bootstrap iteration, predict the data not in the bootstrap sample (what Breiman calls “out-of-bag”, or OOB, data) using the tree grown with the bootstrap sample.
2. Aggregate the OOB predictions. (On the average, each data point would be out-of-bag around 36% of the times, so aggregate these predictions.) Calculate the error rate, and call it the OOB estimate of error rate.

We use spark built-in random forest regression function. The input is the last output data frame we talked above. And the output is the predicted stars. Since this is a supervised learning algorithm. The true star is the corresponding star for every business id.

4.3 Deep Learning

In order to improve the accuracy, we used deep learning to do regression. Our deep learning library is provided by Deeplearning4j [3]. The deep learning model is a Multi-layer Neural Network with back-propagation. The diagram of our network is shown in Figure 3.1.

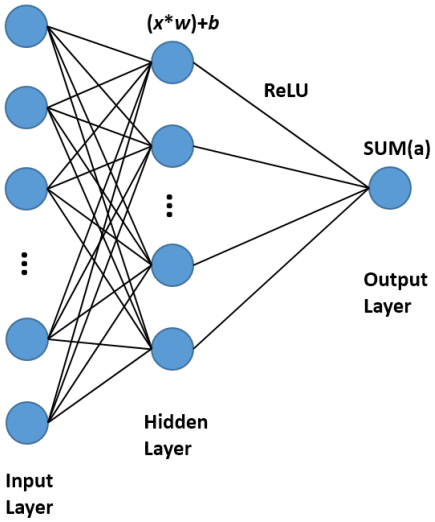
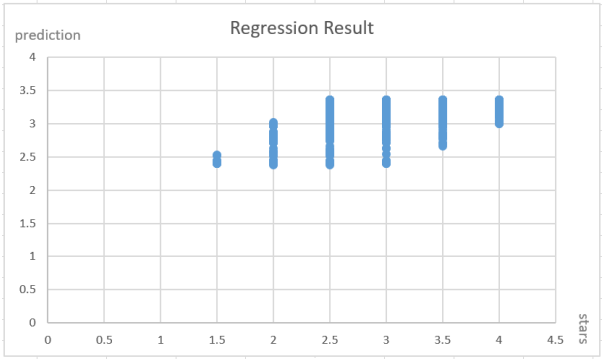


Figure 3.1 Regression Multilayer Neural Network

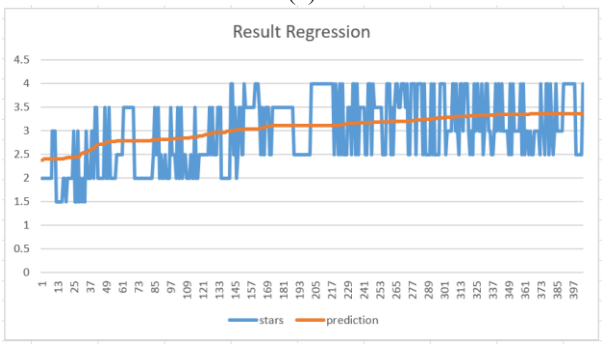
In our project, the model has a 3 layers, with a input layer, a hidden layer, and a output layer. The hidden layer has 10 neurons. The activation function is a rectified linear unit (ReLU), because it doesn't saturate on shallow gradients as sigmoid activation functions do [4]. We use backpropagation to make the network learn. The loss function is Mean Squared Error (MSE).

5 Results and Evaluation

In this part, we chose 400 examples as our validation set. For random forest regression, the Root Mean Squared Error (RMSE) on is 0.863. For deep learning regression the RMSE is 0.614 the random forest regression result is show in Figure 5.1. The deep learning regression result is show in Figure 5.2. We can see the deep learning regression is better than the random forest one. We think the reason why the range is kind of large is that though some restaurants chose a good strategy at first, the quality of food and service can make them perform really different.

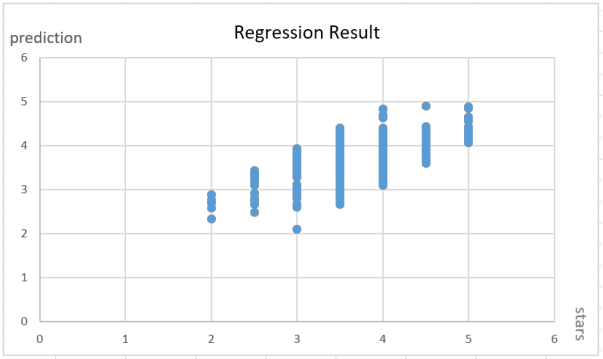


(a)

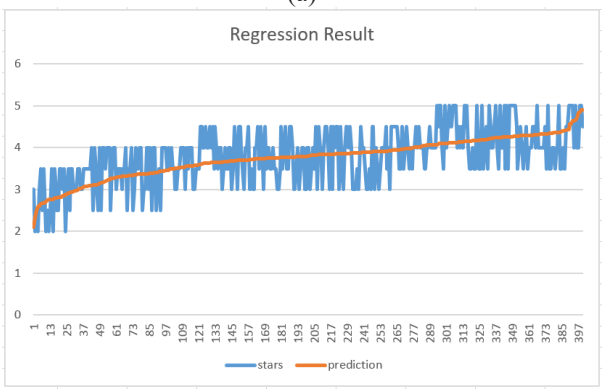


(b)

Figure 5.1 Random Forest Regression Result
(a) Scatter plot of validation set: the x axis is the ground truth of stars, the y axis is the predicted score.
(b) Line chart of validation set: the x axis is sample, while the y axis is the score



(a)



(b)

Figure 5.2 Deep Learning Regression Result
(a) Scatter plot of validation set: the x axis is the ground truth of stars, the y axis is the predicted score.
(b) Line chart of validation set: the x axis is sample, while the y axis is the score

6 Conclusion and Further discussion

Basically, this is just a support method to help people. Based on our results, we can help people find in a specific area, which element can help people to build a more popular restaurant or bar and get more benefit.

Still, we have space to improve. We can add more layers or change the algorithm for each layer in the deep learning algorithm to increase the accuracy or we can use sentence sentiment detector to exclude the noun words with the negative adjective word followed. So we hope we can get a better result in the future.

Acknowledgments

The work is conducted on the dataset from Yelp (https://www.yelp.com/dataset_challenge/dataset). And the libraries we used during this project include Spark, openNLP, and Deeplearning4j. The project was implemented on Spark notebook and Eclipse. The authors would like to thank Dr. Gerard de Melo and Mr. Fangda Han for their helpful suggestions.

References

[1] https://www.yelp.com/dataset_challenge/dataset
[2] Andy Liaw and Matthew Wiener Dec.2002 Classification and regression by randomForest
[3] https://deeplearning4j.org/
[4] https://deeplearning4j.org/linear-regression
[5] Goldberg, Yoav, and Omer Levy. "word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method." arXiv preprint arXiv:1402.3722 (2014).
[6] http://spark.apache.org/
[7] https://opennlp.apache.org/