Predicting Yelp Restaurant Reviews

Wael Farhan
UCSD: A53070918
9500 Gilman Drive
La Jolla, CA, 92093
wfarhan@eng.ucsd.edu

ABSTRACT

Starting a restaurant is a tricky business. Restaurant owners have to keep up with the customer demands, and should continuously update their restaurants according to the trends. Otherwise, they could fall behind and get out of business.

In such a complex dynamic environment, we will need a lot of restaurant data in order to make logical reasoning for future predictions. Luckily, there is an abundant data that is available for the public to analyse and make some inferences that will positively affect businesses performance. There are two major data sets that could be used for this task. The Yelp data set[1] and Google data set [2].

In this paper, we introduce a linear regression predictor that will give restaurant owners some insight on how well they are performing and slightly improve their customer satisfaction based on their restaurant attributes.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval

General Terms

Data Mining

Keywords

data mining, recommender systems, topic models

1. INTRODUCTION

Although there are many data sets that could be used to study reviews and ratings of restaurants, in this paper we have selected the Yelp data set, since it has a lot of restaurant features that could be analysed in order to predict success. Those features include:

- Type of alcohol served
- Noise level
- Price range
- Happy hour
- Smoking options
- Has TV
- Wi Fi
- Good For Groups
- Good For Dancing
- ...etc

This dataset includes 1,569,264 review (990,627 of which are on restaurants), spanning from May 2005 to January 2015. These reviews are made on 61,184 businesses (39,292 of which are on restaurants). Each review contains a 1-5 star rating, which indicates customer satisfaction level.

Before delving deep into how to build a model, we first need to analyse the data. Data analysis will take 3 different approaches: The first is how each feature affect the overall star rating. Secondly, detecting any annual recurrences. The third and the final approach is figuring out any long term trends.

1.1 Feature Inspection

In this section we will investigate if there is any direct correlation between a feature and the average star rating. Table 1 lists all the available features with average rating for each attribute.

Attribute	True	False
Good for Kids	3.754038	3.743862
Good For Groups	3.749967	3.776866
Good For Dancing	3.790404	3.756751
Happy Hour	3.770330	3.679684
Has TV	3.747325	3.757372

Table 1: True or False Attributes

From table 1 we can see that people give higher ratings for restaurants that serve happy hour and for restaurants that are good for kids. We can also see that there is a slight preference for restaurants that do not have TV over the ones that do.

[1]Yelp data set https://www.yelp.com/academic_dataset
[2]Google http://jmcauley.ucsd.edu/data/googlelocal.tar.gz

Attribute	Value	Average
Alcohol	beer and wine	3.752021
	none	3.743458
	full bar	3.754723
Noise Level	very loud	3.686702
	average	3.753040
	loud	3.742398
	quiet	3.754341
Smoking	yes	3.829582
	outdoor	3.771631
	no	3.732646
Wi Fi	paid	3.774057
	free	3.741309
	no	3.754618
Price Range	1	3.750137
	2	3.748467
	3	3.768744
	4	3.766327

Table 2: Other attributes

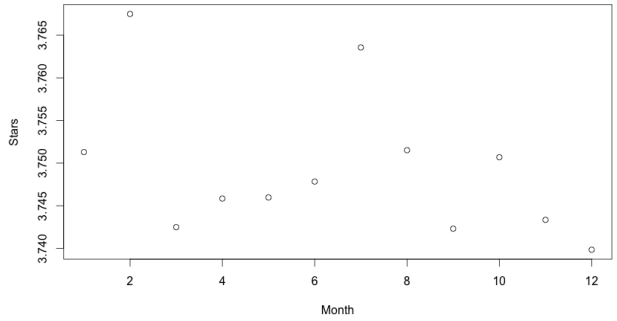

Figure 1: Average ratings per month

From table 2 we notice that reviewers tend to like restaurants that serve alcohol (wine and beer or full bar) over the restaurants that don't. Also, they prefer restaurants that allow smoking (Which are common since the data has been collected since 2005).

All the above inferences are not sufficient for making any solid assumptions about how reviews are affected. So we have analyzed the data to search for more insights.

1.2 Annual Recurrences

In this section we will study annual trends, or people repetitive behaviour throughout the year. For example, figure 1 shows how people rate restaurants per month. It is obvious that people are tending to give restaurants higher ratings in February (lot of holidays in this month, also Birthdays and Valentine's Day) and July (Beginning of summer) more than any other months.

Figure 2 shows how various unique values of 4 different features (Alcohol, Noise Level, Price, and Smoking) were rated across the months of the year. In general the differences are not significant, however, we were able to make some interesting observations. For example, people tend to like restaurants that serve alcohol in winter more than summer. And they seem to dislike very loud restaurants in April more than any other month.

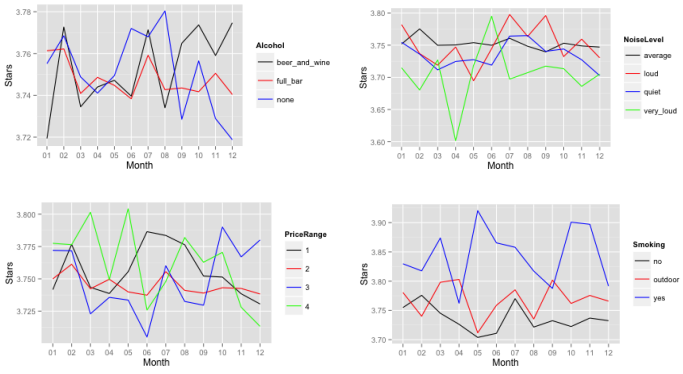

Figure 2: How various features are rated annually

1.3 Long Term Trends

Now we will try to understand how people are changing on the long term, starting from 2005 until January 2015. Our first guess would be to check if there is any change in preference for certain feature along time. Figure 3 shows how the average review for each unique value of noise level per month.

Unfortunately, we cannot inference a lot from these figures. We have also tried to make the same graph for all the other features, apparently people are not changing. But this figure pinpointed the fact that our data is highly fluctuated in the early years. It is probably because of the fact that in the early days of Yelp there was not enough reviews, so one outlier review could make a great influence on the overall mean.

An interesting observation to point out is that people are tending to review restaurants that serves alcohol more over time. Figure 4 shows the number of reviews per unique restaurant. So on average each "full_bar" restaurant gets 2.2 review per month and this number seems to be growing over time. "beer_and_wine" restaurants have very similar trend. On the other hand non-alcoholic restaurants does not seem

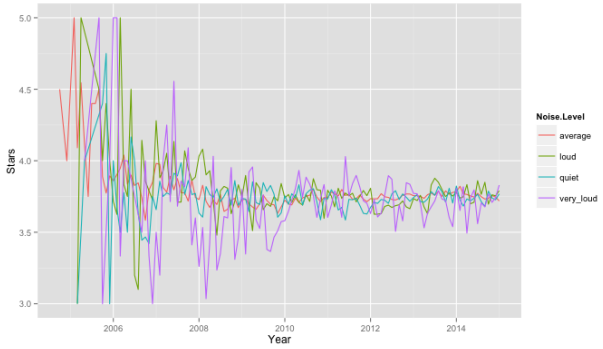

Figure 3: Average ratings per month

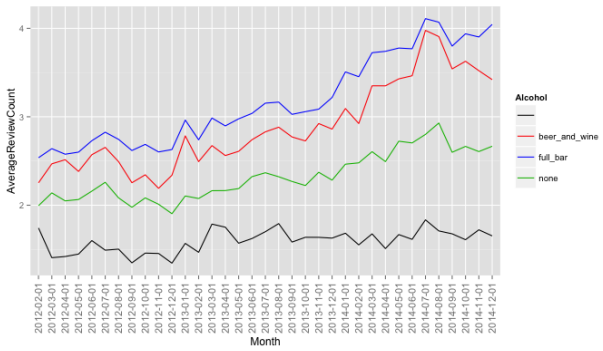

Figure 4: Average ratings per month

to be reviewed as much.

2. PREDICTIVE TASK

After our intensive data analysis that is done in the previous section, we would like to predict the star rating for a given visit based on the time and restaurant attributes.

$$f(date, restaurant features) -> rating[1-5]$$

The data analysed so far does not give a substantial effect on the overall rating, but combining those features could give us some useful insights. In the upcoming section we will discuss various prediction models and we will finally pick the one that makes closest prediction to our validation set.

We will measure the correctness of our prediction model using Mean Squared Error (MSE) for regression models:

$$\sum_{i=1}^{n} [y_i - f(date_i, features_i)]^2$$

We will be looking to minimize MSE as much as possible on training set while making sure it does not over-fit on our validation set. The error rate for each prediction model will be compared with two baselines. The first is random guessing, and the second is predicting the average rating (3.7493) every time.

3. LITERATURE REVIEW

Vasa, Vaidya, Kamani, Upadhyay and Thomas (2014) examined the yelp dataset[3]. They were investigating whether a restaurant will succeed or not. They assigned value of 1 for successful business and 0 for a failed one. They came out with three hypotheses that could be a major contributor for the success of a restaurant:

- Whether food category determine success of a restaurant.
- If the location of the business has any influence.
- Having large amount of features and amenities.

In their prediction, they only took into consideration businesses in and around Phoenix AZ. The features they used are:

- Zip Code
- Mexican
- American
- European
- Latin American
- Fast Food
- Amenities
- Meal Options

They considered various prediction models ranging from decision trees to logistic regression to neural networks. Their best model using neural networks with R squared of 0.58 on training and 0.51 on test set.

By the end of the research they found out that Mexican and Fast Food restaurants are not likely to succeed in Phoenix. They also denoted that people tend to like cheap restaurants.

> The findings were clear and concise: A restaurant is highly likely to be a success if it caters to certain popular categories and provides a large number of services, regardless of its location. This particular finding is in contrast to the preliminary belief of location being the most crucial factor to success, and can probably be attributed to the fact that a better location also equals greater competition. [4]

Our prediction is different in couple of ways. First, we are trying to make our predictor work in multiple cities across the United States. Second, our prediction is meant to output [1-5] star rating per review instead of Boolean value for each restaurant.

4. FEATURE SELECTION

First of all we will present raw data available from Yelp and how we pre-process this data to generate eligible features.

From the Yelp dataset we are only concerned with two types of objects. **Business Object:**

{
'type': 'business',
'business_id': (encrypted business id),
'name': (business name),
'neighborhoods': [(hood names)],
'full_address': (localized address),
'city': (city),
'state': (state),
'latitude': latitude,
'longitude': longitude,
'stars': (star rating, rounded to half-stars),
'review_count': review count,

[3]http://www-scf.usc.edu/~adityaav/Yelp%20-%20Final.pdf

[4]Page 9 "Yelp Predicting Restaurant Success" http://www-scf.usc.edu/~adityaav/Yelp%20-%20Final.pdf

'categories': [(localized category names)]
'open': True / False,
'hours'
{ (day_of_week): { 'open': (HH:MM),
'close': (HH:MM) }, ... },
'attributes': { (Noise Level): (value),
(Alcohol): (value)... }, }
}

And the **Review Object:** {
'type': 'review',
'business_id': (the identifier of the reviewed business),
'user_id': (the identifier of the authoring user),
'review_id': (the identifier of the authoring user),
'stars': (star rating, integer 1-5),
'text': (review text),
'type': (type of the object)
'date': (date, formatted like '2011-04-19'),
'votes': {
'useful': (count of useful votes),
'funny': (count of funny votes),
'cool': (count of cool votes)
}
}

The first step we did was to join the two objects together. We will need all the restaurant attributes be available for each review. So we built a script that joins each review with its corresponding business based on **business_id** attribute. While doing that, we also filtered out all the reviews that are not targeted for a restaurant. Eventually the pre-processing step converted 2 JSON files; 1.3GB reviews.json and 53MB business.json into 3 CSV files; 97MB training.csv, 32MB validation.csv and 32MB test.csv

During this process we also included a new attribute called "Month" which was extracted from the date information, this field will be used to predict any annual recurrences that might occur as we discussed in our statistical analysis.

In the pre-processing step we also flatten out some attributes removing them from their parent field. These attributes are: Alcohol, Noise Level, Smoking, Has TV, Good For Groups, Good For Kids, Good For Dancing, Price Range, Happy Hour and Wi Fi. These 10 features along with the month and state will be the pool of features that we will select from while building our model. Finally, we trimmed down some of the unnecessary attributes, since it will make our files much smaller and easier to read and manipulate.

After fetching up 990,627 reviews and joining them with the 39,292 restaurant, we shuffled them and split them into three groups: training set (60%), validation set (20%), test set (20%) and finally a sample training set which is 1/3 of our actual training set (This is used for fast model inspection).

5. MODEL SELECTION

In this section we will discuss various models that we will investigate in order to predict the star rating review. We will consider both classification methods (Naive Bayes and Neural Network) then we will tackle down some regression models (Random Forest and Linear Regression).

5.1 Naive Bayes

In the first attempt will be using Naive Bayes model to classify reviews in 5 categories (from 1 to 5). Naive Bayes maybe not the best tool to make inferences about complicated dataset like restaurant reviews. But, it will give us some useful insight for building upcoming models.

Package 'e1071'[5] from Cran is used to build the Naive Bayes model.

Classification using Naive Bayes has 37.18% accuracy on the training set, while it is 37.017% accurate on the validation set, which is better than random guess (about 20%). This also means that we are not over-fitting since training and validation errors are very similar.

Looking at the confusion matrix below, it is pretty clear that Naive Bayes is classifying all reviews to be 5 star reviews.

Pred \Label	**1**	**2**	**3**	**4**	**5**
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	20119	17726	28011	58930	73339

The reason behind this behaviour is that the prior probability is much more dominating than the posterior one.

Prior probability distribution:

	1	**2**	**3**	**4**	**5**
Probability	0.1014	0.0884	0.1415	0.2970	0.3718

Sample of the posterior probability:

Happy Hour	**1**	**2**	**3**	**4**	**5**
True	0.3334	0.3334	0.3334	0.3336	0.3337
False	0.3332	0.3332	0.3332	0.3331	0.3331
NA	0.3333	0.3333	0.3333	0.3332	0.3331

The posterior probability is very low, which means that there is a very weak correlation between the star rating and the features when they are being handled independently. In the next model (Neural Networks) we will be looking into combining some features.

5.2 Neural Networks

Since Yelp dataset is really complex and there might be some hidden relationships that we missed or will not be able to comprehend. So in this section we will apply non-linear statistical data modelling tool, the Neural Networks.

The dataset is ran against the R function "nnet" [6]. Output network is really complicated, a simplified version is displayed in figure5. There are many parameters to learn using this model, like the number of nodes in the hidden layer and number of iterations. So the model is ran against validation several times to reach the optimized parameters.

[5]http://cran.r-project.org/web/packages/e1071/e1071.pdf
[6]http://cran.r-project.org/web/packages/nnet/nnet.pdf

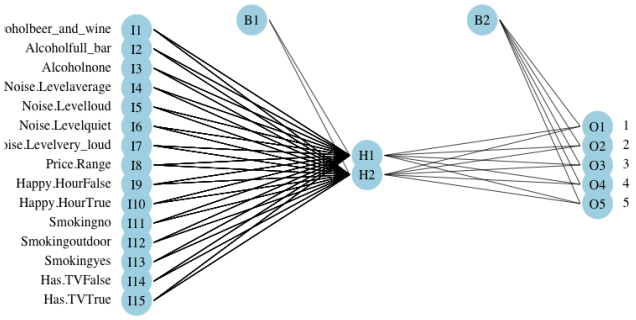

Figure 5: Simplified version of the Neural Network

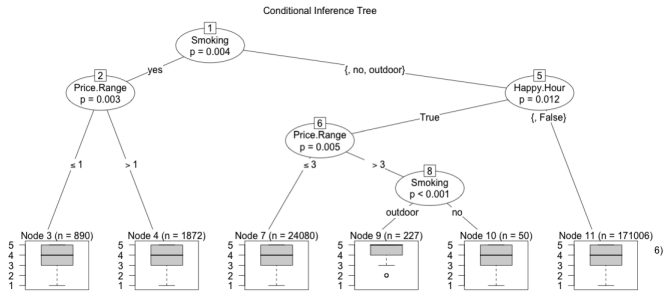

Figure 6: Decision Tree on sample training set using only 3 features

After learning the model we ran the model against the validation set and the resultant accuracy was 30.24% (training set accuracy is 30.56%) which is better than random guess (20%). But it is still worse than Naive Bayes.

Confusion matrix is as follows:

Pred \Label	1	2	3	4	5
1	118	113	187	322	350
2	1698	1449	2302	4715	5945
3	1018	952	1572	3044	3837
4	7834	6773	10750	11358	28972
5	9339	8329	13038	31696	33792

The reason behind the low accuracy for this model is that our underlying assumption of combining multiple features together would make a better predictor is not correct.

Since business owners are not concerned about one particular review, but will be more interested about the average reviews they get for a particular month. So, we decided to shift gears and study regression models that would provide a more insightful view for a specific restaurant.

5.3 Random Forests (Decision Trees)

For model is built using "party"[7] package from cran project[8]. Running the library against our sample training set using only 3 features gave us the tree in figure6. It is easy to notice that the distributions at the leaves are very similar except for one case where Smoking=outdoor, Happy.Hour=True, and Price.Range=4. In that particular case people tend to review the place between 4 and 5 while other possible combinations range between 3 and 5.

Next we train our decision tree against the whole training set and calculated the training error and validation error:

- Training MSE : 1.71338
- Validation MSE : 1.71606

This model barely beat our baselines (Recall our baselines are random guessing and taking the average every single time).

[7]http://cran.r-project.org/web/packages/party/party.pdf
[8]http://cran.r-project.org/

Attribute	Value	Attribute	Value
(Intercept)	3.732e+00	Month02	1.603e-02
Month03	-7.884e-03	Month04	-5.379e-03
Month05	-4.991e-03	Month06	-3.629e-03
Month07	1.247e-02	Month08	4.271e-05
Month09	-8.463e-03	Month10	5.746e-04
Month11	-7.119e-03	Month12	1.099e-02
Alcoholbeerwine	3.256e-02	Alcoholbar	2.339e-02
Alcoholnone	3.056e-02	Noise.average	1.099e-02
Noise.loud	8.802e-03	Noise.quiet	-6.871e-03
Noise.vloud	-5.054e-02	Price.Range	5.408e-04
Happy.HourF	-4.243e-02	Happy.HourT	2.084e-02
Smokingno	-1.308e-02	Smokingout	1.739e-02
Smokingyes	9.913e-02	Has.TVF	-1.107e-02

Table 3: Linear Regression Coefficient Sample

- Random Guessing Baseline MSE : 4.28530
- Average Baseline MSE : 1.71844

5.4 Linear Regression

We will now try to fit our training data into a linear model. We use "lm"[9] function which is built inside the R programming langauge. Table3 shows a sample of coefficients. As expected Month02 and Month07 has positive influence on the reviews just as analysed previously in figure1. The coefficients also indicate that people like average noise places and hate very loud places.

Running this model against training set and validation set produces the following errors:

- Training MSE : 1.70439
- Validation MSE : 1.70814

So this model beat the baselines and Decision Trees too. It is the best model so far.

6. RESULTS AND CONCLUSION

[9]https://stat.ethz.ch/R-manual/R-devel/library/stats/html/lm.html

Attribute	Value	Attribute	Value
Noise Level	0.14147	Alcohol	0.14185
Smoking	0.14430	Good for Kids	0.14188
Wi-Fi	0.14092	Good For Groups	0.14201
Good For Dancing	0.14454	Happy Hour	0.14412
Price Range	0.14200	Has TV	0.14128

Table 4: Coefficients of determination R^2

In summary, after trying out multiple models ranging from non-parametric models to parametric ones. The best option that we ran into was the Linear Regression.

After running this model on our test set the results was as expected:

- Linear Regression Test MSE : 1.701775
- Average Baseline MSE : 1.711977
- Random Guessing Baseline MSE : 4.16550

So our model is slightly better than taking the average every single time. This model is better but it is not good enough to run it in a production or industrial environment.

In conclusion, we realize that it is not an easy task to build a predictor to using these features. We will check coefficient of determination (R^2), recall the equation of R^2:

$$\begin{aligned} R^2 &= 1 - (MSE(f)/Variance(y)) \\ 0 &\rightarrow TrivialPredictor \\ 1 &\rightarrow PerfectPredictor \end{aligned}$$

Table 4 shows the coefficients for all of our predictors. All of our coefficients have low values which justifies the unpredictability of the data. Eventually, these numbers are indicating that it is really hard to make a predictor that could do well using these features.

The reason of this phenomena is that our predictors does not affect review consistently for each customer. That means people are highly unpredictable when it comes to review restaurants, or personal preferences varies a lot from one customer to another.

Another reason behind the low predictability is that restaurant data is updated while reviews stay the same. So for instance, if some restaurant changed the "Smoking" attribute from "True" to "False" then all the previous reviews for this restaurant would be assigned to a restaurant with the updated features. Yelp dataset does not provide any way for us to detect these updates. Due to the new law that prohibits indoor smoking, many restaurants have been shifting from smoking to non-smoking, and this will have a huge impact on R^2 coefficients.