Capstone Project 2

Milestone Report

1 Background

This project explores the possibilities to predict restaurant ratings using Yelp reviews. The research is based on Yelp's publicly available data set for education purposes.

The data set includes information on various types of businesses from a number of cities and regions in the U.S. and other countries. In order to perform meaningful research that can be readily interpreted by non-experts, the project examines data specifically about restaurants in Phoenix, Arizona. This filtering was made simply due to the availability of data on Phoenix (3500 restaurants and 300,000 reviews about them).

By interpreting the influence of features a restaurant has on Yelp (items such as price level, location, availability of alcohol, parking and ambiance, among others), provides an objective set of criteria to predict what rating the restaurant is likely to receive. This part of the analysis uses regression models to determine 'learn' the relationship between features and ratings on a training set, and then predict ratings for a test data set.

However, ratings given by customers reflect their subjective experience of the restaurant they are reviewing – experiences such as whether they liked the food, how well they were served and how they felt about the price paid for what they got. This part of the analysis uses natural language processing (NLP) methods to process the reviews text and determine the relationship between words/phrases contained in a review and the reviewer's score.

2 Problem and Client

Opening a restaurant is a risky investment even for experienced restaurateurs and its success rests to a very large extent on its customers' perception and loyalty. Moreover, Yelp has become a leading source of information for potential customers and businesses that have a high Yelp rating are more likely to be chosen by potential customers. Hence understanding what constitutes a high rating, or what actions may be taken to improve Yelp ratings, is valuable information for restaurant owners and managers.

As the Yelp data set is strictly intended for educational purposes, there client in this project is a hypothetical one. Indeed, the main goals of the project are:

- To build an understanding of the relationship between restaurant features and ratings
- To develop tools for processing written reviews and inferring information from them, which includes both a general sentiment (positive/negative) and an actual Yelp rating (from 1 to 5

stars). Once developed, this tool is highly scaleable and reusable for any written information about any restaurant.

- To draw conclusions about what aspects contribute to having good overall ratings, by putting together the features and reviews analyses and trying to identify elements that are common to best-rated restaurants.

The answers obtained from the exercise should ultimately help to define an investment strategy.

3 Data Source and Data Processing

The Yelp Dataset is published by the business review service Yelp for academic research and educational purposes.

After filtering for restaurants, there are approximately 52,000 restaurants with approximately 2.9 million user reviews related to them. Filtering for Phoenix, Arizona only provides a set of approximately 3,500 restaurants and 300,000 reviews.

The raw data is available in six of files in *.json* format, of which two are relevant for the project:

- **business.json -** the records for individual businesses
- **review.json -** the records for reviews users wrote about businesses

The files are text files (UTF-8) with one json object per line, each one corresponding to an individual data record.

Extracting relevant data and processing for analysis was performed through the following steps:

Features data for regression analysis. The *business.json* file includes features about each individual restaurant, which can be directly converted to a pandas data frame using the *json_normalize* function from the pandas library. This yields 90+ columns that describe the restaurant's price range, meals served, availability of alcohol, ambiance, opening hours, location, parking, number of reviews and average rating rounded to the nearest half-star. The attributes were processed so that all categories are described as numerical values, forming the *X-matrix* of features. There is a large proportion of NaN values, which are replaced by column-wise averages. The averaged ratings are separated into the *y-vector* of target values. The output for this exercise are the *X-matrix* and *y-vector*, split into training and test sets prepared for regression analysis using a variety of methods.

NLP processing of reviews text. The *reviews.json* file holds the information about each review, including the business_id it refers to, text of the review written by the user and star rating given by the user. Each review from the set of businesses pertaining to restaurants located in Phoenix is loaded into a data frame for analysis.

The analysis itself is performed using a previously trained Word2Vec model, which was obtained through the following steps[1]:

- The spaCy library was used to perform sentence detection, text tokenization and normalization, converting tokens to lemmas, part-of-speech recognition and identification of punctuation, stop-words and white space.
- Next, the gensim library's Phrases function was used to detect bi-gram and tri-gram phrases. A dictionary of over 80,000 words was built from the reviews text. The final corpus including bi-gram and tri-gram phrases was parsed to be stored in lemmatized form, excluding punctuation, white space and stop words but including the words and phrases that bring meaning to the text.
- The gensim Latent Dirichlet Allocation (LDA) model was used to group the lemmatized text data into 50 word clusters, modeling 50 topics most commonly described in the reviews. This gives the possibility to extract topics discussed in any review and thus represent the review text in a much simpler format.
- Finally the gensim Word2Vec function was used to convert the entire corpus into a vector representation of the entire dictionary of 80,000+ words. Each word is represented as a vector in a 100-dimensional feature space. This will allow us to model each review into this 100-feature vector space.

All the training was performed using the full set of 2.9 million restaurant reviews available (regardless of location i.e. including the Phoenix reviews) in order to obtain the best possible model. The key outputs for this exercise are

- LDA topic model that allows us to extract the key topics of a review
- Word2Vec model for approximately 85,000 words, that allows us to represent a word/sentence/review in the 100-D vector space

4 Predicting Restaurant Ratings

4.1 Predicting Average Ratings from Features

The features matrix after data cleaning includes 69 features for 3,515 restaurants, and the target ratings for the 3,515 restaurants are averaged over the reviews written about them and rounded to the nearest half-star.

A number of prediction algorithms were tested, and their parameters tuned for optimization:

- Linear Regression
- Ridge Regression

1 The NLP processing to obtain the Word2Vec model and LDA model were performed following the PyData 2016 tutorial by Patrick Harrison "Modern NLP in Python".

- Lasso
- Random Forest Regression
- Gradient Boosting Regression
- MLP Regression

Each model was tested for the R^2 on the training and test sets, and the mean squared error (MSE) on the test set predictions. The summary of results is presented below:

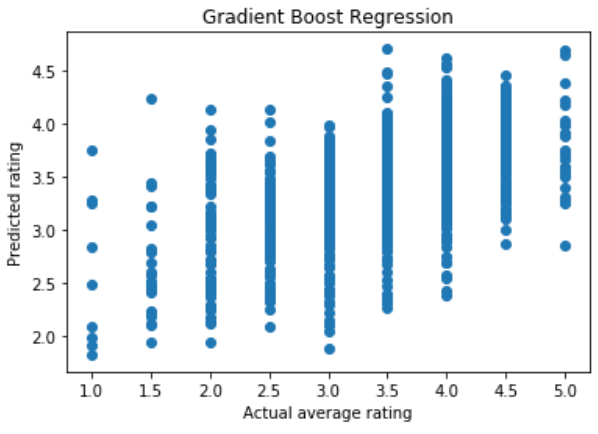
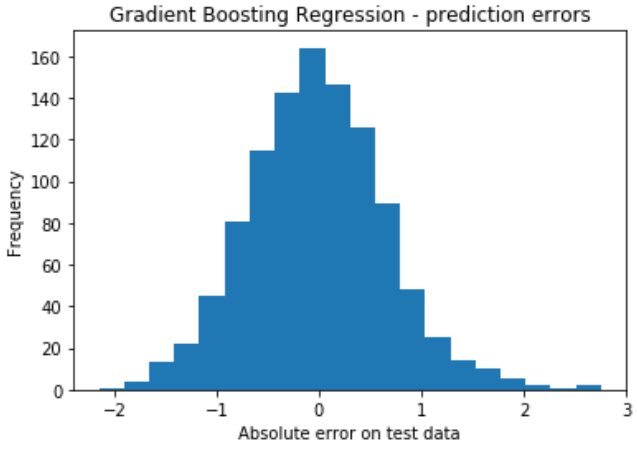
Model	Linear	Ridge	Lasso	Random Forest	Gradient Boosting	MLP
R^2 training data	0.21	0.21	0.21	0.86	0.68	0.22
R^2 test data	0.20	0.20	0.20	0.26	0.34	0.22
MSE test data	0.54	0.54	0.54	0.50	0.45	0.53

The linear model, even with more advanced methods such as Ridge Regression and Lasso, do not give a convincing result. The neural network MLP model does not perform any better, while the random forest model overfits the training data for a minor improvement in the test data score.

The Gradient Boosting model yields the best results, with parameters set to:

- (n_estimators=200, loss= 'ls', max_depth=5, learning_rate=0.1, max_features='log2')

Notably, limiting the max_features parameter has a positive effect on performance. Also, the model is sensitive to the max_depth parameter: a higher tree depth of e.g. 10 tends to overfit the model to the training data and is inaccurate on the test data, while a lower depth of e.g. 3 does not provide sufficient complexity to distinguish between relatively similar data points.

Scatter Plot of Predictions vs. Actual Data	**Histogram of Prediction Errors**
	

The standard deviation of the raw training data is 0.82 and the prediction errors have a standard deviation of 0.67. Hence, considering this result, the scatter plot of actual vs. predicted test data and histogram of prediction errors, we can conclude that the gradient boosting model does have some predictive power and picks up a general trend of lower vs. higher scores, but it does not provide an accurate and reliable way of predicting average ratings based only on features.

The next section will look at the more subjective aspects of the review in order to improve the predictions.

4.2 Sentiment Analysis and Prediction

This section considers the text data from reviews in order to learn and predict the sentiment of a review, as a positive (marked with a 1) or negative (marked with a 0). The training and test data for this was taken from the 300k Phoenix restaurant reviews. The data was split in the following way:

- Reviews rated 4 and 5 stars were labeled as ‘positive’
- Reviews rated 1 and 2 stars were labeled as ‘negative’
- Reviews rated 3 stars were excluded from the data as they are indecisive

The data now included ca. 275K reviews and these were split into training and test sets. The following classification algorithms were used to train and predict reviews for positive/negative sentiment:

- Random Forest classifier
- MLP classifier

Each review was represented as a 100-feature vector using the trained Word2Vec model. In order to obtain a single vector for each review (given that reviews vary in length), an average vector was derived for each review. This was performed through the following steps:

1. Using spaCy to parse the revview by converting words to lemmatized tokens and remove punctuation, white space and stop words
2. For each remaining token in the review, the Word2Vec model was used to represent the token in vector form
3. For all tokens in the review, the vectors were summed and divided by the number of tokens in order to obtain a mean vector that represents the ‘meaning’ of the review in 100-D vector space
4. The vector representation of the reviews is therefore converted to an X-matrix of 100-D vectors for each review, and the y-vector is a 1/0 representation of the review sentiment.

The training data was used to learn the relationship between the vector representation and review sentiment, and used on the test data to predict sentiment.

Results of the analysis are presented below:

Model	Proportion of accurate predictions
Random Forest Classifier (500 estimators)	91.6%
MLP Classifier (solver='adam', alpha=1e-3)	94.2%

The neural network based MLP classifier gave the best result, after some tuning of the *alpha* parameter, solver used and the learning rate. The default parameters give an accuracy score of 94.0%, and this was increased to 94.2% by increasing the value of *alpha* and using the 'adam' solver.

The purpose of this exercise was to build a tool that can distinguish between positive and negative reviews, as a (potentially more robust) alternative to predicting actual ratings. In order to enforce a decision between positive and negative sentiment, it is necessary to remove the indecisive 3-star ratings from the data. However the model can be used to test any review for sentiment, and in case of reviews that are not conclusively positive or negative, the model will make a decision between these two classes.

It is also worth noting hat the relatively high scores are somewhat helped by the fact that the Word2Vec model was trained on data that also includes the test reviews (it was trained on the whole set of 2.9 million restaurant reviews in the Yelp data set). Hence a set of completely new, unseen reviews may well include words or phrases that are not included in our existing Word2Vec dictionary that would not be convertible by the Word2Vec model, resulting in a loss of meaning and a potentially lower prediction accuracy.

4.3 Predicting Ratings from Reviews

In order to predict actual rating reviews, the same Word2Vec representation of review meaning was used, this time including the 3-star reviews as well. The y-matrix of target results includes the rating given by the user as an integer between 1 and 5. A classification model was used again as a way of predicting the actual rating and classify reviews into 5 distinct groups. However, a regression model may also be appropriate here and would provide more precise information about a review (e.g. a 3.6 and 4.4 would both be classified as a 4-star rating in the classification model).

The classification models tested are the following:

- Random Forest Classifier
- Ada Boost Classifier
- MLP Classifier

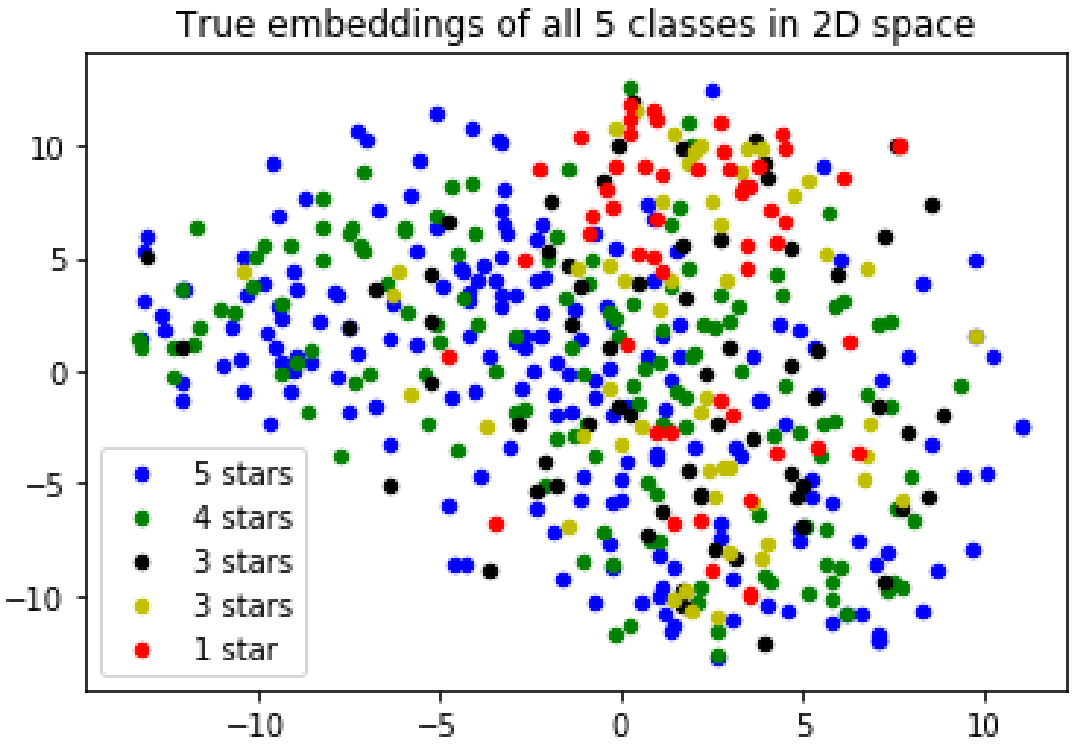
The results obtained are presented below:

Classifier model	Accurate predictions	Bad predictions	MSE
Random Forest	54.0%	9.90%	1.226
Ada Boost	53.4%	6.72%	1.141
MLP	57.3%	5.54%	0.876

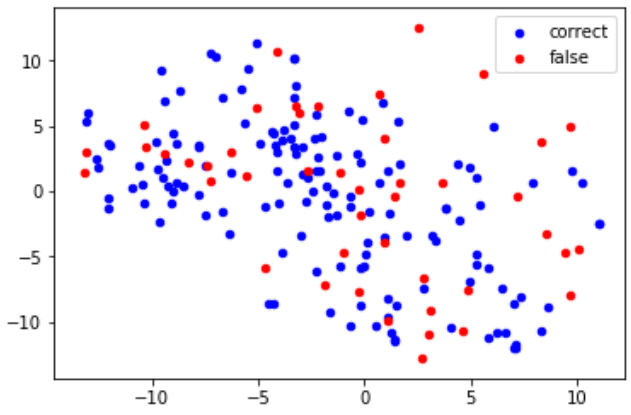
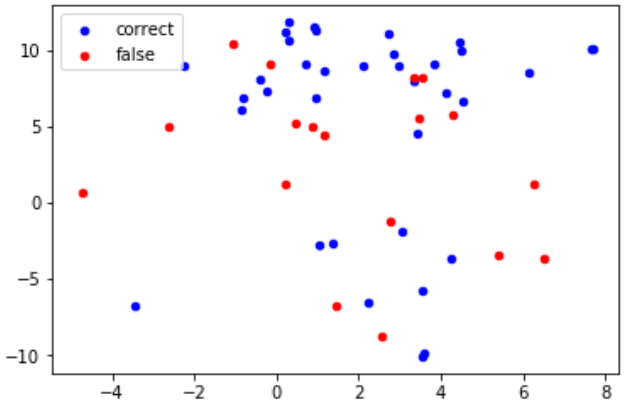
Accuracy on test data was measured using three metrics: the percentage of accurate predictions, percentage of 'bad' predictions (where the prediction is off by two or more classes) and the mean squared error (MSE).

Here, the MLP classifier gave the best results, with the highest proportion of accurate prediction, lowest proportion of 'bad' predictions and the lowest MSE. In this case the neural network method outperformed decision-tree based ensemble methods

Further, results of the 100-D vector representation were reduced to 2-D using t-SNE dimensionality reduction to reveal the embedding of classes.



The embeddings are difficult to distinguish in 2D space, so we can consider the correct and false classifications of individual classes. Results shown below are for the MLP classifier::

Correct and false classification of 5-star ratings	Correct and false classification of 1-star ratings
	

5 Next Steps

The completion of the project will require the following next steps:

1. Combining the predictions from features and reviews into a unified model
2. Drawing key conclusions about what elements might contribute to a very good star rating (or reasons why a restaurant may have a very low score)