YELP DATA CHALLENGE

ANALYZING AND PREDICTING BUSINESS SUCCESS USING BUSINESS ATTRIBUTES AND CUSTOMER REVIEWS

December 2nd, 2018
Sahil Arora, Sanjana Rosario, Riddhi Kanoi, Huy Nguyen

Abstract: Restaurant review platforms have become our default way of searching for restaurants for every occasion. However, when looking for specific attributes of a restaurant or less generic contexts, this search isn't as straightforward. This gives rise to a need to analyze 'hidden' features that factor into the success of a food business such as customer's perception through reviews and a restaurant's ambience. Through this project, we hope to achieve two purposes. First, we want to enable future users to search for restaurants based on more distinct categories with topic modelling techniques to help them select the best dining options. Second, we want to employ machine learning models and natural language processing to gain insights into the success factors of a business, and predict whether a business is going to be rated highly based on those factors.

1. DATASET DESCRIPTION

The dataset used for this project is the Yelp Dataset (https://www.yelp.com/dataset). This includes:

- "business.json": includes information directly related to the business such as latitude, longitude, business name, attributes, and categories
- "review.json": includes the reviews received by the businesses above referenced by business ID
- "user.json": includes the information about the users who wrote the review
- "checkin.json": includes information about check-ins for the businesses
- "tip.json": includes short pieces of suggestion and advice written by the users

Due to time constraint of this project, we will focus mainly on the first two files.

2. DATA PREPROCESSING

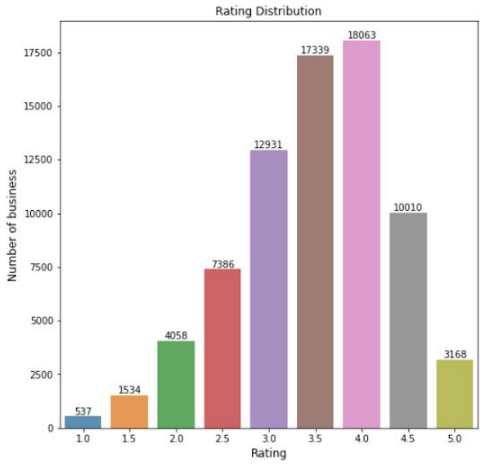
We perform these preprocessing steps:

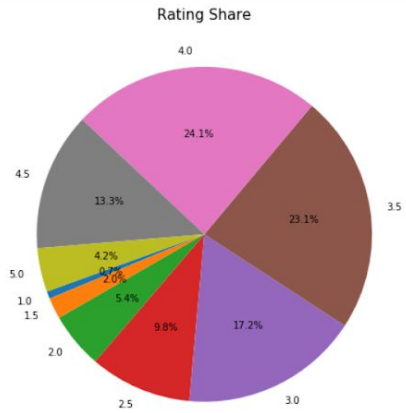
1. Convert json to csv: For some analyses, we use json_to_csv.py converter provided by Yelp to convert the above files into CSV. For some other analyses, we converted it directly for memory efficiency purpose.
2. Slice the data location-wise and category-wise: Location-wise, for the scope of this project, we will only be looking at business and reviews coming from North American business and users (USA and Canada). In addition, we will only look at the food, drink and restaurant sector.
3. Merge the business, review and user data sets (using left join) to perform exploratory data analysis and building models
4. For topic modelling, we exclude businesses in Canada because of potential cultural, semantic, language differences. In future works we plan to extend this to reviews in Canada as well.
5. For machine learning purpose:

a. Label the attributes with either one-hot encoding (creating dummy variables) or label encoding.
b. Transform the date features in the reviews dataset to perform datetime operations.

3. DATA EXPLORATORY ANALYSIS

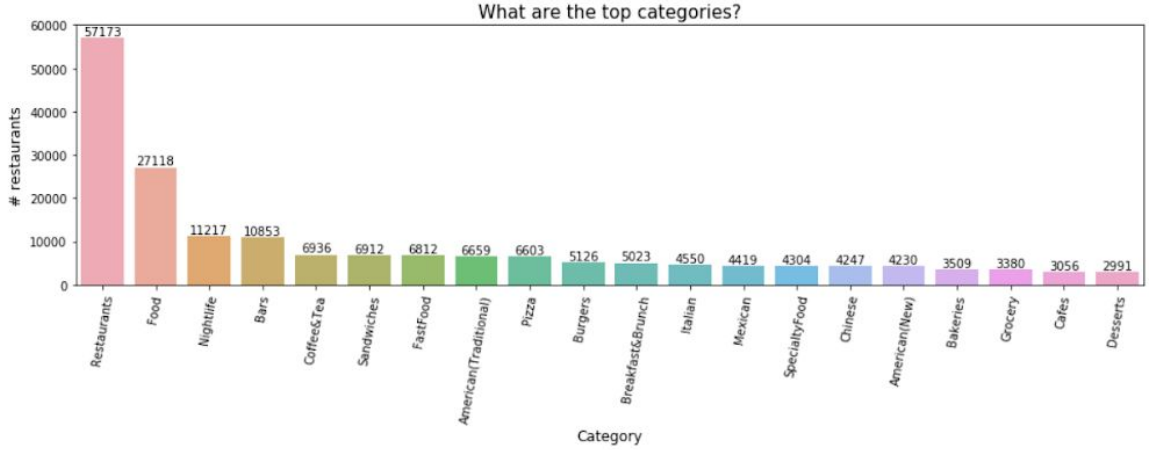
a. RATING SHARE AND DISTRIBUTION

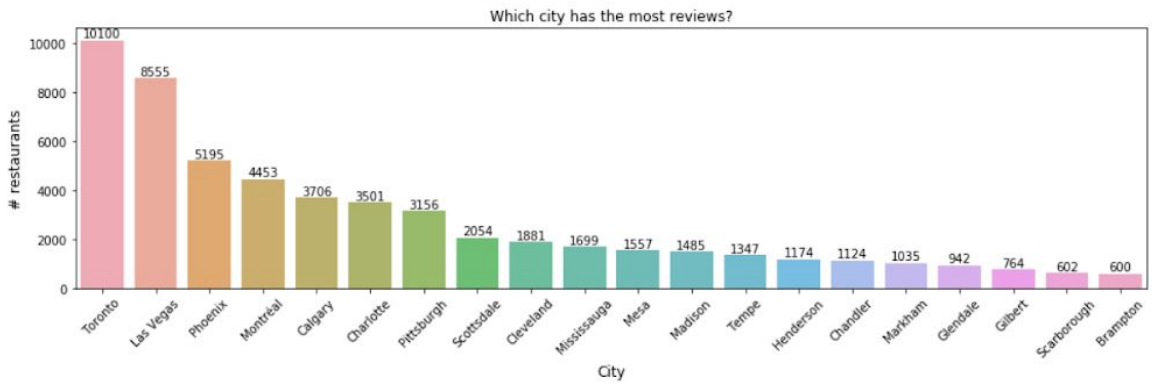




We observe that the ratings are mostly in the range of 3.5 (23.1% of total ratings) to 4 stars (24.1% of total ratings).

b. BUSINESS DISTRIBUTION ACROSS CITIES AND STATES

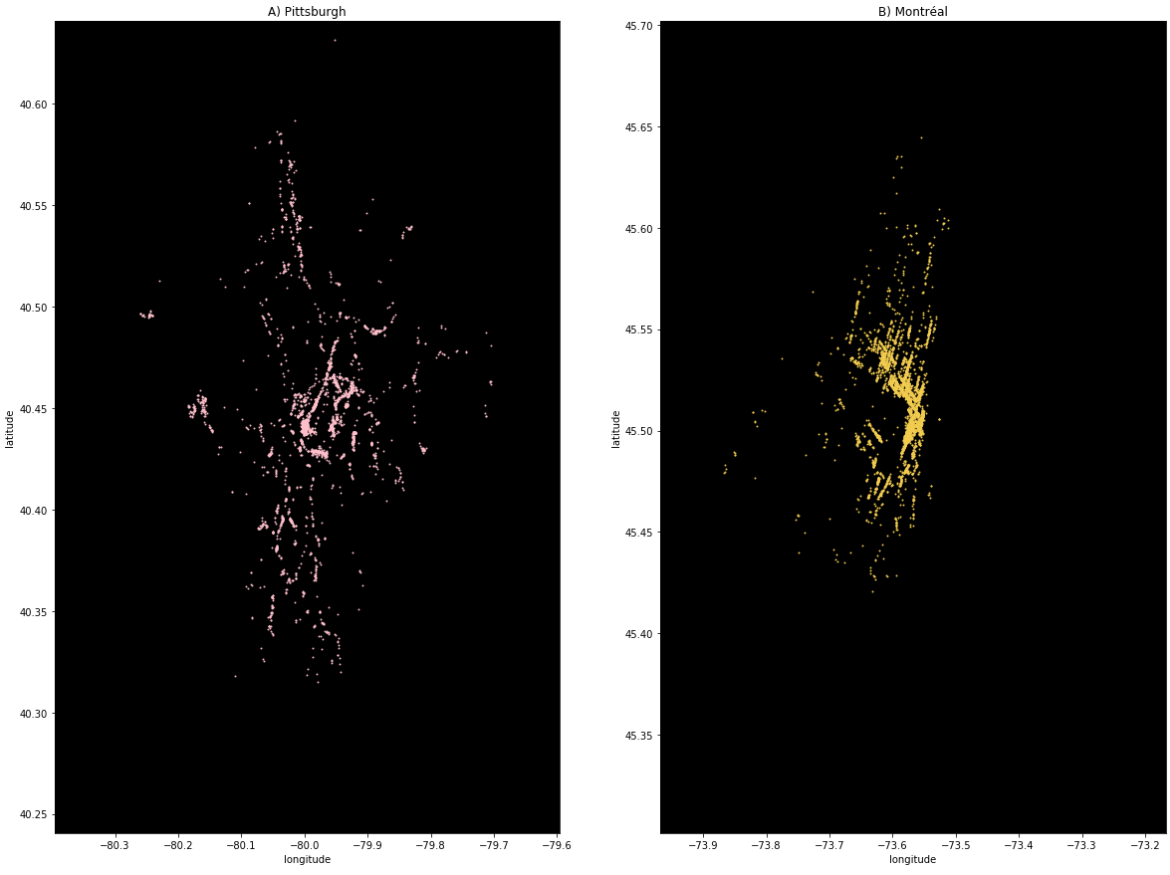




c. COMPARISON BETWEEN MONTREAL AND PITTSBURGH

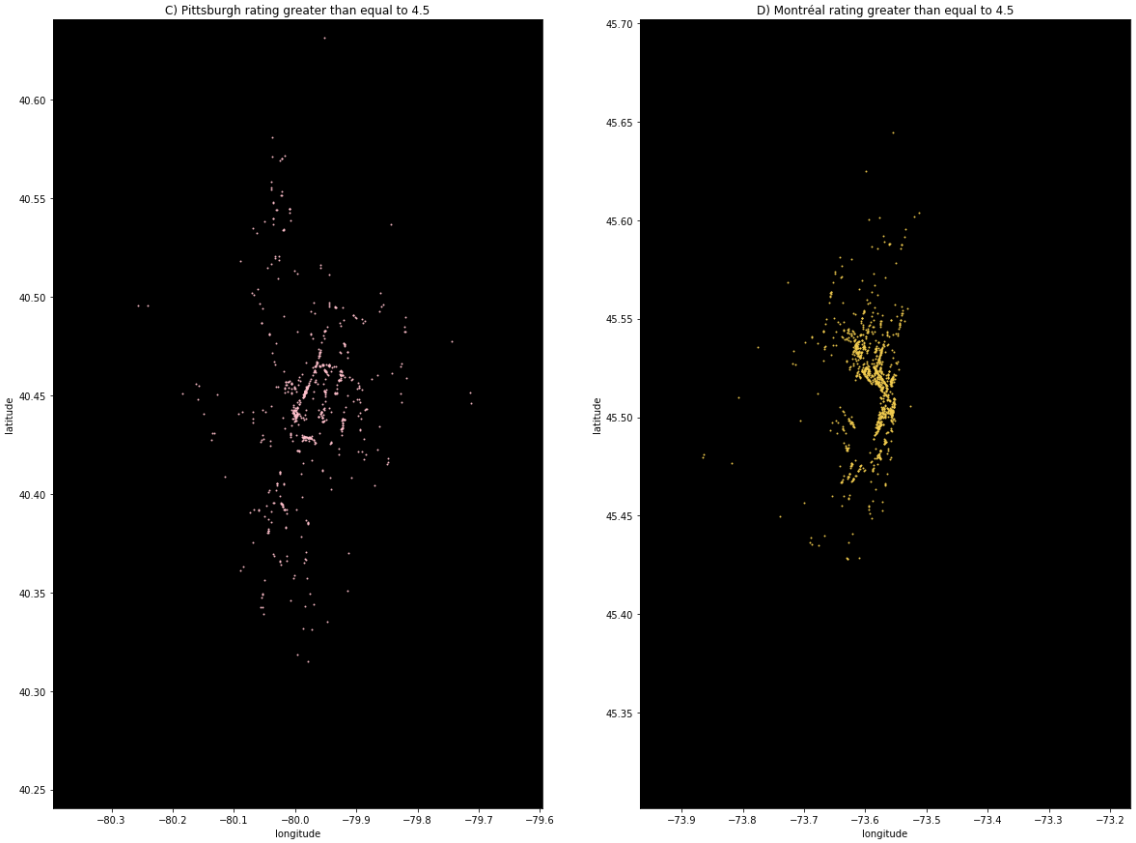
The graph above depicts the number of reviews received by each city. We used this information to consider two cities situated in two different countries to understand the difference in the trends of restaurants and reviews. We considered Montréal (located in Canada) and Pittsburgh (located in the USA) since the number of reviews received were comparable and they fell within the Top 10 cities with the highest reviews. Thus, keeping these cities in mind we computed our data in the following way:

- We looked at the 'latitude', 'longitude', 'stars', and 'review_count' of the restaurants in both Montréal and Pittsburgh
- We calculated the max and min values of the latitude and longitude for both cities
- We imported the library "matplotlib.pyplot" and plotted two graphs in order to visualize the spread out of the restaurants based on the reviews received
- We selected "Scatter plot" as the graph type and defined the square marker 's', alpha values, and color of the graph.
- We then printed out both the graphs to get the following result



Looking at the above graph, we instantly notice a difference in the two cities. The data points in Montréal (graph B) is more concentrated when compared to that of Pittsburgh (graph A). Even though Montréal is approximately three times in size (size of Montréal 166.6 mi^2 and size of Pittsburgh 58.35 mi^2) the area covered by its restaurants isn't as spread out. Additionally, you notice that the data points in both the graphs are more concentrated in the center and dwindles as we move away.

We next repeated the same steps in order to filter out restaurants with ratings greater than equal to 4.5 stars. The resultant graphs are as follows:



The above graph depicts restaurants in Montréal and Pittsburgh with ratings greater than equal to 4.5 stars. It is interesting to see how the number of restaurants in Pittsburgh (graph C) reduces far more rapidly than those in Montréal (graph D) when the filter is applied. The surface area covered by restaurants with ratings greater than equal to 4.5 stars is once again more concentrated in Montréal. It isn't as spread out as Pittsburgh. However, the restaurants are still more dense towards the center as compared to the exteriors.

4. ANALYSIS AND METHODS

a. TOPIC MODELLING

In order to build a tool to search through user reviews, we use the "reviews" and "businesses" datasets. We start off by pre processing the datasets and filtering for only restaurants in the US. We exclude Canada in this topic modelling because we recognize how a geographical difference can have negative effect on natural language processing (and we recognize some of the reviews from Canada may not be entirely in English). This is also done to reduce the size of the dataset and also to zoom in on restaurant reviews.

The "businesses" dataset contains categories that we split up and manually search to identify tags related to restaurants like "restaurants", "nightlife", "bars", "cafes", etc. We then filter for business that contain these tags using a user defined function.

We aggregate all of the reviews for each restaurant and process them. We remove stop words, tokenize the remaining words and also filter out infrequent words in order to get a collection of words that we think "describe" the restaurant. We treat the aggregated review tokens for each restaurant as a document and input this list of documents to an **Latent Semantic Indexing** (LSI) model to identify categories. LSI is a model commonly used to identify categories within a document and based on that matches a new "example" document to the identified categories. In this context, the collection of all reviews are the documents that the LSI identifies categories from. The user input search query is the "example" document that need to be matched to the identified categories. Based on the categories that user's search query matches best with, it is clustered with restaurants with similar categories. We output the top 50 best matches of the model but also control for restaurants with poor ratings (<3.5).

b. WORD EMBEDDINGS

It is essential to develop a sophisticated word representation model to analyze similarities and establish connections among customer reviews and tips. We compute continuous vector representations of words from Yelp's customer reviews using Word2Vec. The Word2Vec model produces a vocabulary, with each word being represented by an n-dimensional numpy array.

These vectors prove to perform much more better on our test set with regards to measuring syntactic and semantic word similarities. Being able to measure similarity means being able to quickly analyze and measure sentiments and topics in customer reviews. The result of this embedding is a spatial space displaying scatter points. The points with the same colors representing words that are highly correlated.

After converting the reviews to vectors we use TSNE to reduce the dimensionality and visualize the vectors in space. TSNE is pretty useful when it comes to visualizing similarity between objects. It works by taking a group of high-dimensional (100 dimensions via Word2Vec) vocabulary word feature vectors, then compresses them down to 2-dimensional (x,y) coordinate pairs. The idea is to keep similar words close together on the plane while maximizing the distance between dissimilar words.

c. BUSINESS SUCCESS PREDICTIONS

The machine learning models we use for this project include linear regression, logistic regression, decision tree, random forest and bootstrapping. We decided to choose many models and compare the results among them to figure out which model performs the best on the set of features that we have as well as the importance of features with respect to each model. For regression tasks, we use the star ratings (on the continuous scale of 0.0 to 5.0) as the label. For classification tasks, we also 1 and 0 to indicate success (a restaurant achieving 3.5 stars and above on average will receive a 1 and a 0 otherwise).

In both regression and classifications tasks, we ran OLS regression beforehand to narrow down the features that explain the variances in our labels. In addition, for the classification task, we also ran recursive feature elimination to pick the most important features. These features, as well as the features appearing in feature importance as a result of the random

forest algorithm, are important to restaurant owners because they can infer future business performance.

5. RESULTS

a. TOPIC MODELLING

Upon running the "Review Topic Modelling.ipynb" code, a prompt will appear to input the user's preferences. Examples of contextual searches could be "informal lunch with family" or "bright light ambiance and crowded place". Once we enter our preferences, upto 50 restaurant suggestions are output, along with their reviews and ratings. We imagine this tool would be handy to have along with existing filters on Yelp's website.

b. WORD EMBEDDINGS

The result for word similarity using Word2Vec is displayed below. Using this tool will help us quickly analyze and understand a customer review and gain insights into the sentiments shared among words. Here we use Python library Genism's implementation of word2vec model to train our word vectors. In order to better capture the corpus statistics, such as the word-word co-occurrence matrix, we train word vectors using the reviews.

We utilized previous insights from previous word embedding research (Mikolov et al.). Here Xij is the (i,j) entry of the co-occurrence matrix denoting the number of co-occurrence of word i and word j; w, w˜, b and ˜b are input and output word vector and intercept terms as in Word2Vec. The weight function f(Xij) is defined as:

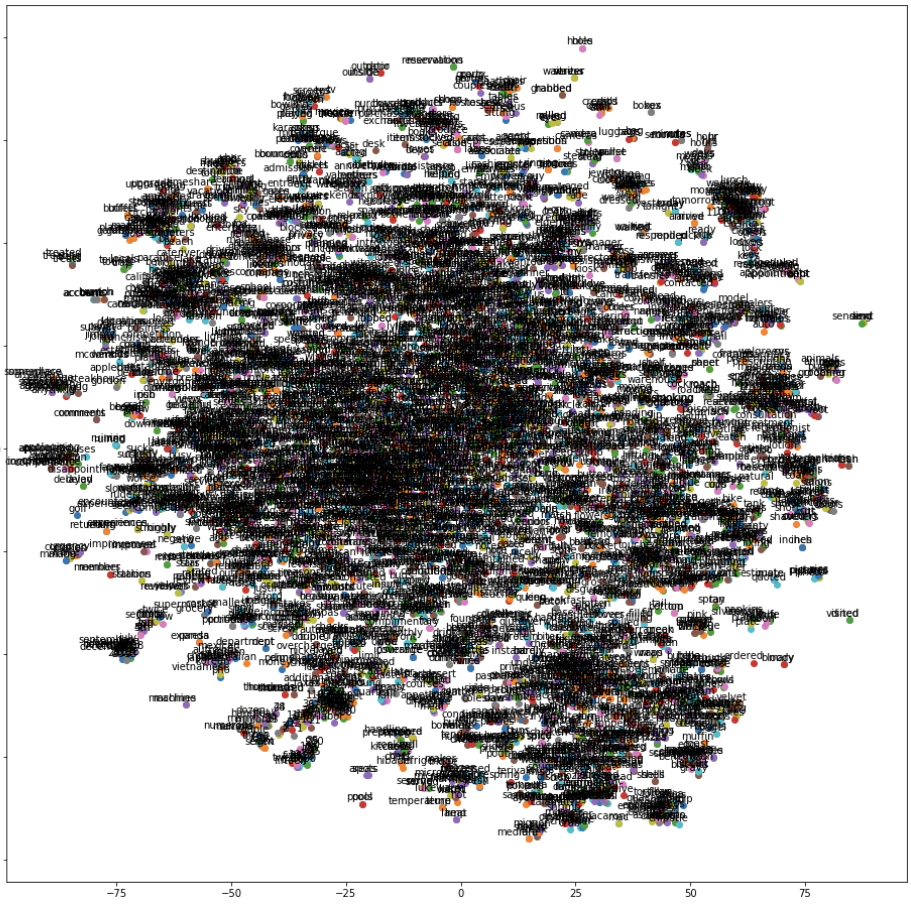
$$f(x) = (x/x\ max)^{\alpha} \text{ if } x < x\ max$$
$$f(x) = 1 \text{ otherwise}$$

VISUALIZATION

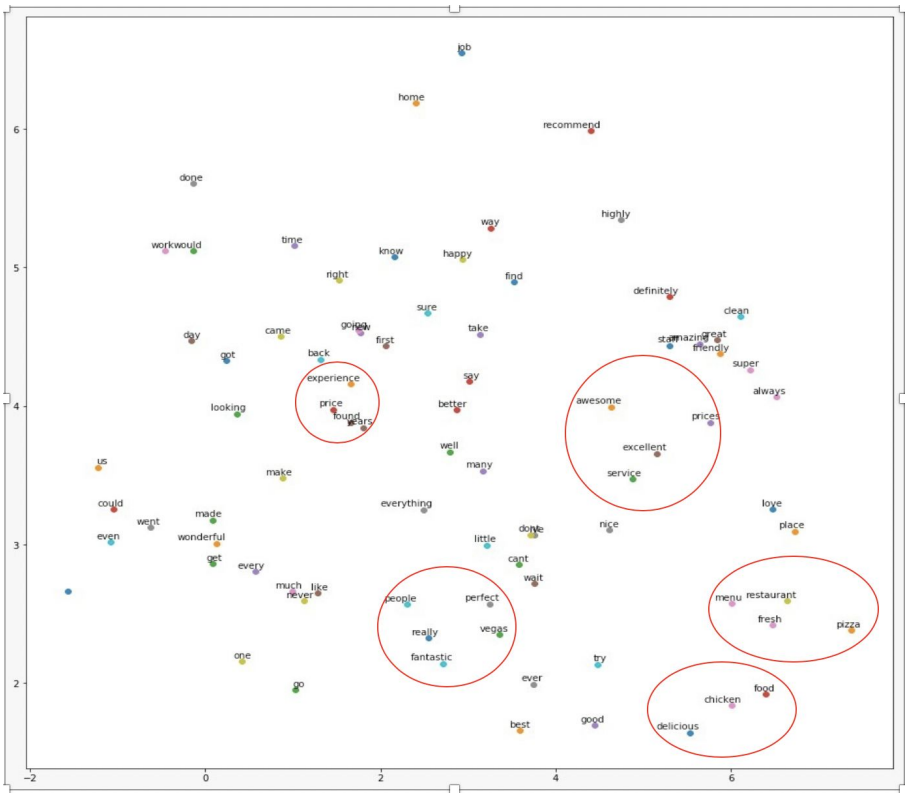
(t-SNE) t-Distributed Stochastic Neighbor Embedding is a non-linear dimensionality reduction algorithm used for exploring high-dimensional data. It maps multi-dimensional data to two or more dimensions suitable for human observation. With help of the t-SNE algorithms, you may have to plot fewer exploratory data analysis plots next time you work with high dimensional data.

Initial Word2Vec embeddings with a low minimum count of occurences:

One of the key challenges was to effectively find the minimum count of occurrences from the entire corpus that plots the t-SNE as well as retains the inherent meaning of the sentences in the words. The picture above depicts one of the initial word2Vec representations that didn't help us draw good inferences.



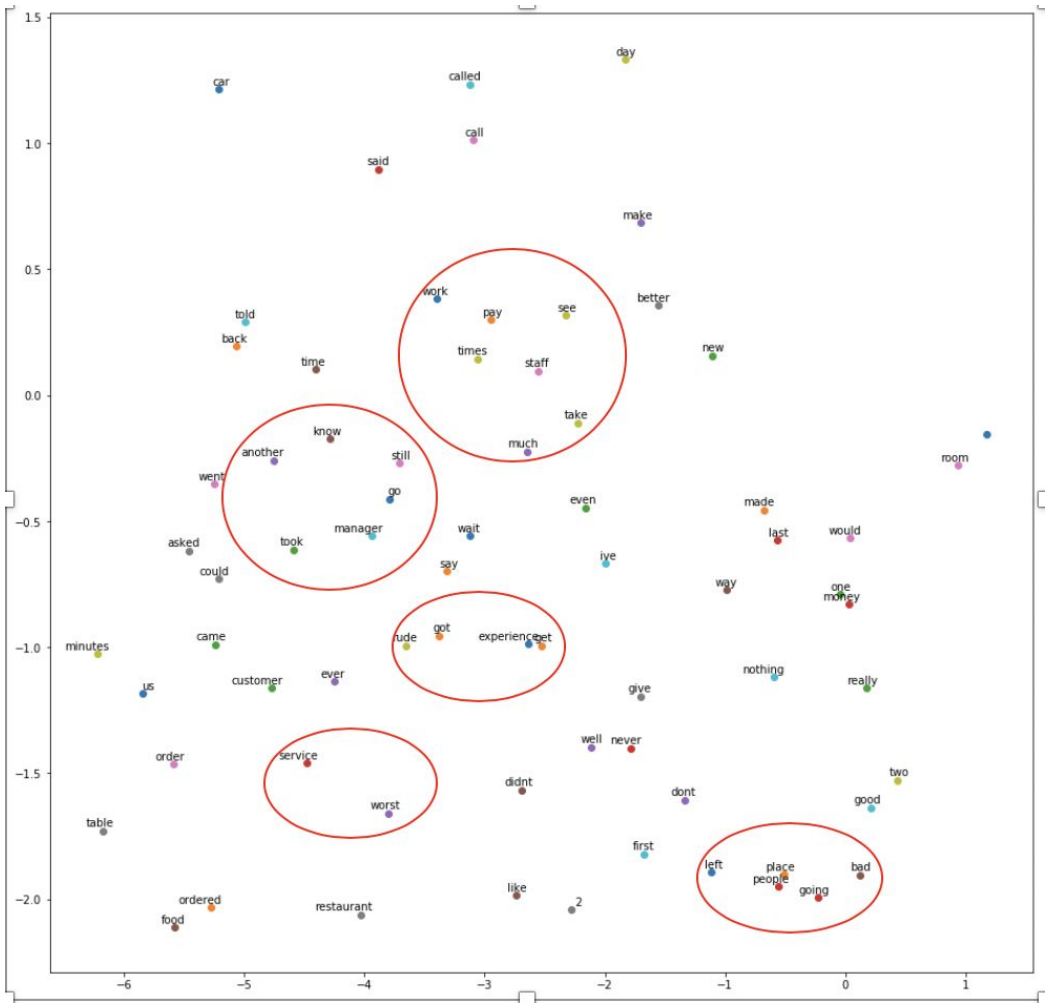
Vector Representation of Entities scoring Well in reviews (> 3.5) :



Inferences for Positive Sentiment reviews:

1. The word “**experience**” of a customer is highly correlated to the word “**back**” and “**bars**”. This can help us investigate whether customers are likely to come back to the entity if their experience is good whether be it a bar or a Hair Salon
2. The word “**menu**”, “**food**” and “**chicken”** is closely related to the word “**fresh**”. This can help us validate whether the restaurants that serve fresh food are more likely to be rated higher.
3. As expected “**excellent**” and “**service**” are correlated due to the fact that people are highly likely to rate an entity higher if the service is good
4. Also, the words”**good**”, “**delicious**” and “**chicken”** are spatially correlated leading to the premise that a restaurant with a great chicken dish is more likely to succeed than a restaurant with other specialties

Vector Representation of Entities scoring Poorly in reviews (<3) :



Inferences for Positive Sentiment reviews:

1. The word “**experience**” of a customer is highly correlated to the word “**rude**”. This can help us investigate whether an entity is poorly rated solely because of rude behavior of the staff.
2. The word “**manager**” and “**service**” are closely related. This can help us validate whether the entities that have unsatisfactory managers are likely to be rated lower
3. As expected “**service**” and “**worst**” are correlated due to the fact that people are likely to rate an entity poorly if the service is bad
4. Also the words ”**time**”, “**wait**”, “**minutes**” and “**took**” are spatially correlated leading to the premise that an entity that takes time irrespective of what the product is highly likely to be rated lower

c. BUSINESS SUCCESS PREDICTIONS

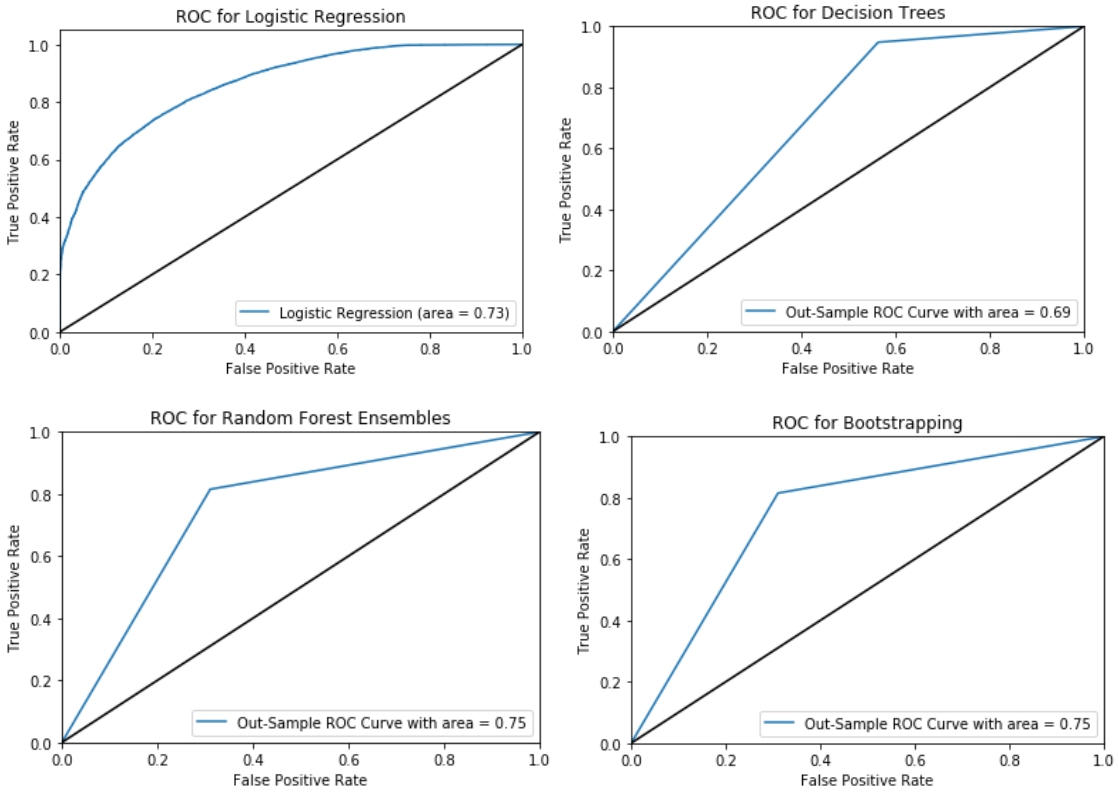
We preselect a number of features to use on our models based on previous research on Yelp (Hood et al.)

We achieve a relatively positive result for all of our models. The results are as follows:

	Mean-squared Error	**Variance Score**

Linear Regression	0.29	59.00%
	Accuracy	**Area under the ROC**
Logistic Regression	**78.00%**	**73.00%**
Decision Tree	76.75%	69.00%
Random Forest	75.97%	76.00%
Bootstrapping	**77.09%**	**75.00%**

We suggest using either logistic regression or bootstrapping because they are relatively simple and yield good results while not too computationally expensive. Below is the ROC for our models.



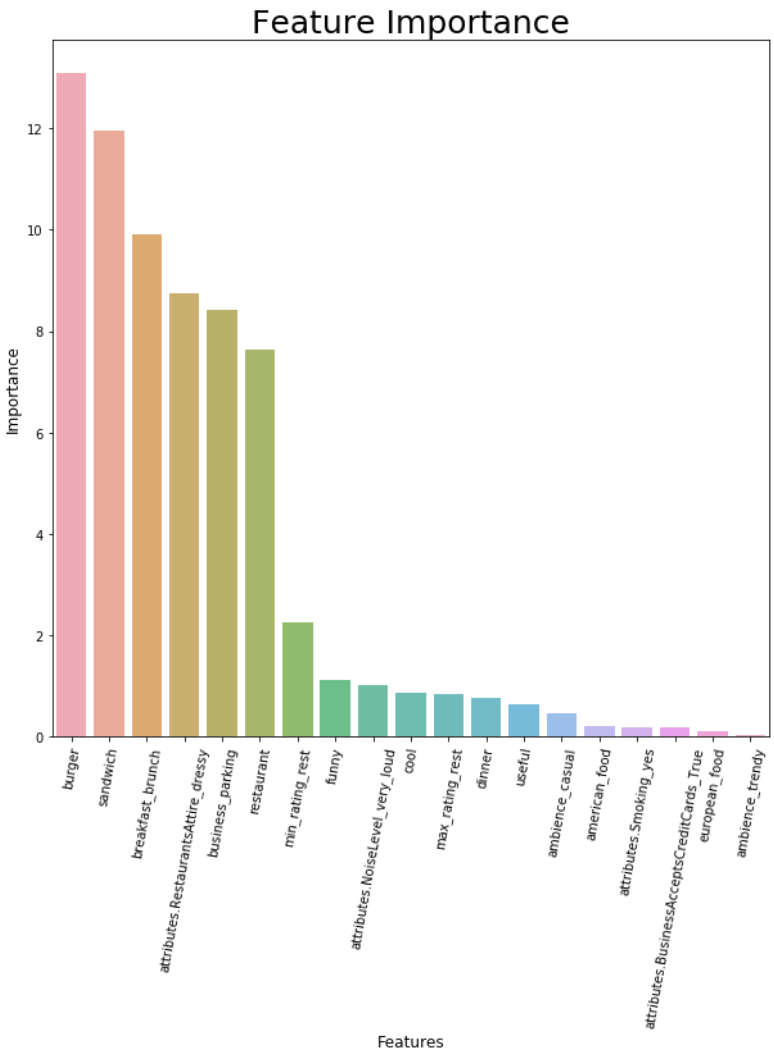
The features that are used as initial input of the above classification algorithms are:

	Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
review_count	0.0005	0.0000	35.0330	0.0000	0.0005	0.0006
attributes.RestaurantsAttire_casual	-0.0026	0.0082	-0.3243	0.7457	-0.0186	0.0133
attributes.RestaurantsAttire_dressy	0.1417	0.0173	8.1859	0.0000	0.1078	0.1757
attributes.RestaurantsAttire_formal	0.0365	0.0488	0.7489	0.4539	-0.0590	0.1321
attributes.RestaurantsReservations_True	-0.0014	0.0056	-0.2442	0.8071	-0.0123	0.0095
attributes.GoodForKids_True	-0.0209	0.0064	-3.2878	0.0010	-0.0334	-0.0084
attributes.Smoking_yes	0.0951	0.0181	5.2537	0.0000	0.0596	0.1305
attributes.NoiseLevel_very_loud	-0.2393	0.0141	-16.9132	0.0000	-0.2670	-0.2115
attributes.Open24Hours_True	-0.1832	0.0998	-1.8351	0.0665	-0.3788	0.0125
attributes.RestaurantsGoodForGroups_True	-0.0067	0.0060	-1.1087	0.2676	-0.0186	0.0051
attributes.WiFi_free	0.0027	0.0048	0.5551	0.5789	-0.0067	0.0121
attributes.BusinessAcceptsCreditCards_True	-0.0370	0.0058	-6.4294	0.0000	-0.0483	-0.0257
attributes.RestaurantsPriceRange2	0.0072	0.0031	2.3694	0.0178	0.0012	0.0132
attributes.HappyHour_True	-0.0201	0.0083	-2.4064	0.0161	-0.0365	-0.0037
attributes.OutdoorSeating_True	0.0346	0.0049	7.0369	0.0000	0.0250	0.0442
ambience_casual	0.1033	0.0059	17.4762	0.0000	0.0917	0.1149
ambience_trendy	0.1321	0.0118	11.2129	0.0000	0.1090	0.1552
attributes.AgesAllowed	-0.0774	0.0129	-5.9977	0.0000	-0.1027	-0.0521
max_rating_rest	0.6134	0.0019	324.1459	0.0000	0.6097	0.6171
min_rating_rest	0.4232	0.0022	190.3736	0.0000	0.4188	0.4275
useful	-0.0273	0.0029	-9.3960	0.0000	-0.0329	-0.0216
funny	-0.1075	0.0040	-27.0204	0.0000	-0.1153	-0.0997
cool	0.1655	0.0041	40.0235	0.0000	0.1574	0.1736
american_food	-0.0444	0.0065	-6.7971	0.0000	-0.0572	-0.0316
asian_food	0.0225	0.0063	3.5634	0.0004	0.0101	0.0348
european_food	0.0844	0.0068	12.3901	0.0000	0.0711	0.0978
sandwich	0.0527	0.0072	7.3742	0.0000	0.0387	0.0667
burger	-0.2651	0.0082	-32.4981	0.0000	-0.2811	-0.2491
restaurant	-0.0988	0.0078	-12.5897	0.0000	-0.1142	-0.0834
date_difference	-0.0000	0.0000	-16.7505	0.0000	-0.0000	-0.0000
earliest_till_now	-0.0001	0.0000	-39.7746	0.0000	-0.0001	-0.0001
latest_till_now	-0.0000	0.0000	-15.5543	0.0000	-0.0000	-0.0000
no_music	-0.0000	0.0000	-0.4496	0.6530	-0.0000	0.0000
background_music	-0.0124	0.0139	-0.8891	0.3739	-0.0396	0.0149
business_parking	0.1968	0.0049	39.8234	0.0000	0.1871	0.2065
breakfast_brunch	0.0531	0.0083	6.4295	0.0000	0.0369	0.0693
dinner	0.1045	0.0066	15.9116	0.0000	0.0916	0.1173

The important feature suggested by our Random Forest Ensembles are as follows. Because the performance of our models are relatively similar, we can use this to judge which features are most explanatory of the variances in our prediction.

From the features below, we have the following suggestions for the restaurant owners:

1. Burger and sandwich seem to be good features explaining the labelling. While sandwich is viewed favorably, offering burgers seems to be negatively correlated with a food business' success. This should be investigated further.
2. Maintaining a good facility is essential for a successful restaurant. Customers seem to favor those with good business parking and avoid those with loud noises. Food businesses labeled as 'restaurant' also seem to perform better, which shows customers' emphasis on having good facility.
3. Breakfast and brunch places is correlated positively with high ratings.
4. Other features are harder to interpret and built into a strategy. However, they should be kept in consideration for further analysis.



6. CONCLUSIONS

In this project, we employ many techniques to analyze users' reviews as well as machine learning algorithms to classify business to predict their successes. We use Latent Semantic Indexing (LSI) model to identify categories and enable users to search through reviews to identify restaurant/food business according to their needs. We use word embeddings and Word2Vec to identify similarities among words in the review in order to make inference about the business.

For machine learning, logistic regression and bootstrapping are shown to perform sufficiently better compared to other models and can be used to gain insights into how their business will perform based on a number of input features. We also collect the features that are highly indicative of a business success or failure. Using this information, food business owners can make informed decision for the operations of their business.

In future works, we plan to extend the Word2Vec model to infer hidden messages in the user reviews and use more sophisticated machine learning algorithms such as neural networks and XGBoost to improve our success predictions even more.

REFERENCES:

1. Yelp Dataset and Data Challenge Website: https://www.yelp.com/dataset/challenge
2. Mikolov et al., "Efficient Estimation of Word Representations in Vector Space" (2013)

3. Hood et al., ”Inferring Future Business Attention” (2014)