Text Analytics Project

predict rating from yelp reviews

Presented By:

Deepika Agrawal

Connor Jones

Lin Zhang

Jin Long

Table of Contents

[Introduction 2](#_Toc531032598)

[Business Applications 2](#_Toc531032599)

[Literature 2](#_Toc531032600)

[Objective 3](#_Toc531032601)

[Dataset 3](#_Toc531032602)

[Data Pre – processing 3](#_Toc531032603)

[Exploratory Data Analysis 4](#_Toc531032604)

[Methodologies 9](#_Toc531032605)

[N-Gram 9](#_Toc531032606)

[LSTM 10](#_Toc531032607)

[First Experiment 10](#_Toc531032608)

[Second Experiment 12](#_Toc531032609)

[LDA 14](#_Toc531032610)

[LDA model for restaurant reviews 16](#_Toc531032611)

[LDA model for different ratings 17](#_Toc531032612)

[Insights of LDA Model 18](#_Toc531032613)

[Appendix 19](#_Toc531032614)

[Works Cited 19](#_Toc531032615)

# Introduction

Text analysis is the study of text and find useful insights from it. Text mining, also known as text data mining involves algorithms of data mining, machine learning, statistics, and natural language processing, attempts to extract high quality, useful information from unstructured formats. Techniques like categorization, entity extraction, and sentiment analysis are used to identify insights, patterns, and trends in large volumes of unstructured data. Some of the applications of text mining or text analytics include spam filtering, social media data analysis, fraud detection, customer care service etc.

# Business Applications

According to Inc. Magazine, 84 percent of people trust online reviews as much as friends when it comes to making shopping decisions. Almost every consumer-facing business needs to manage its online reputation in order to maintain customers’ trust.

Our project will demonstrate a system for analyzing review text data in order to inform strategic decisions regarding online reputation management. By building models that predict the review rating and sentiment score of text data, businesses will be better able to manage customer complaints by intelligently allocating customer service resources. This process lets a business optimize its customer service spend to help the customers who are most likely to leave a negative review, negatively impacting the business for years. This, along with improving the business’s understanding of what drives customers to complain, will provide a long-term value to all businesses that market consumer products.

# Literature

Before starting to look into the data, we looked for some works that have been done previously and resemble with the objective of our project. Researchers in

this field have tried to find efficient algorithms for predicting rating, helpfulness,

and sentiment of the reviews [5].

On famous websites like Amazon and Yelp, many products and businesses receive tens or hundreds of reviews, making it impossible for readers to read all of them. Generally, readers prefer to look at the star ratings only and ignore the text. However, the relationship between the text and the rating is not obvious 1. In particular, several questions may be asked: why exactly did this reviewer give the restaurant 3/5 stars? In addition to the quality of food, variety, size and service time, what other features of the restaurant did the user implicitly consider, and what was the relative importance given to each of them? How does this relationship change if we consider a different user's rating and text review? The process of predicting this relationship for a generic user (but for a specific product/business) is called Review Rating Prediction [2].

Identifying both topical and sentiment information in the text of a review is an open research question. Review processing has focused on identifying sentiment, product features or a combination of both at once [3].

# Objective

For this project, our objective is to use different deep learning methods to predict the ratings from Yelp reviews solely based on the reviews text.

# Dataset

The dataset for this project was taken from <https://www.kaggle.com/yelp-dataset/yelp-dataset>. The dataset was too large so for this project we combined all the tables and took 2.5 million records from the whole dataset after removing the duplicate records. But the time taken to run even a simple query was too long, so we decided to go with 1 million records only.

# Data Pre – processing

As discussed in the previous section, we have various columns in the dataset but for modeling purpose, we need two of those namely, text and review rating. Initially, we started with removing the duplicate values and any missing values. Since it is all text based, we could not impute the data points and hence removed any missing values. We also converted the date column to the desired data type of datetime. After all these steps, we moved to text (reviews).

Text column in the dataset is raw text which means it has punctuation marks, some letters are in upper case while others are in lower case. Before starting with EDA or modeling, text needs to be cleaned. For that purpose, we started with converting all the rows in the text column to lower case letters so that we can have some similarity throughout the reviews.

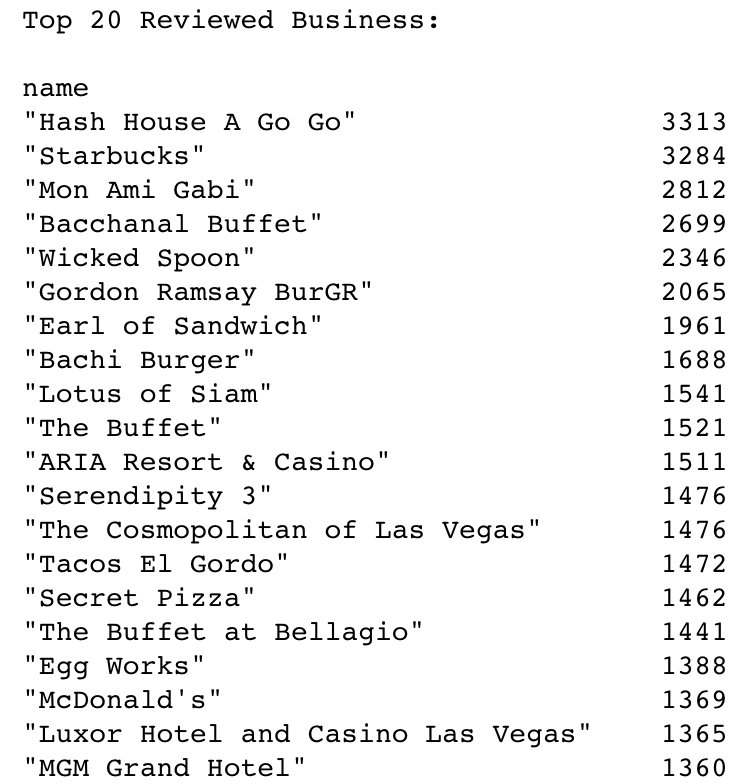
Punctuation marks can create hinderance while modeling because there are lots of them in a single review and if they are not removed, punctuation marks will also become part of the sparse matrix (discussed in modeling section) while modeling and thus affect the results. Thus, after converting the reviews to lower case, punctuation marks were removed from all the records.

Although the stop words like and, the, but plays an important role in reviews, they had to be removed because they are used by the user repeatedly which can again lead to errors while modeling. After all these three steps, we got clean text containing all words with no punctuation and stop words.

# Exploratory Data Analysis

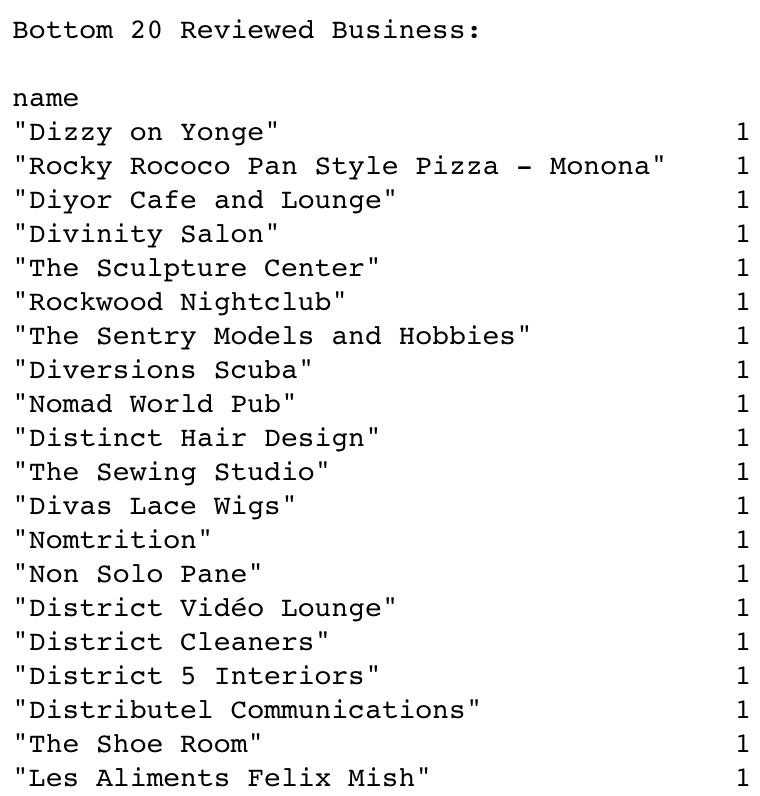
After pre – processing the data and before moving into modeling, some of the basic analysis was done like what are the top reviewed businesses or what is the distribution of different ratings.

1. **Top 20 reviewed businesses**: The dataset contains reviews from customers for 52119 different businesses. Out of all businesses, we looked into the top 20. It can be seen from Figure 1 that “Hash House A Go Go” with 3313 reviews acquires first place and “MGM Grand Hotel” with 1360 reviews is at the bottom among top 20.



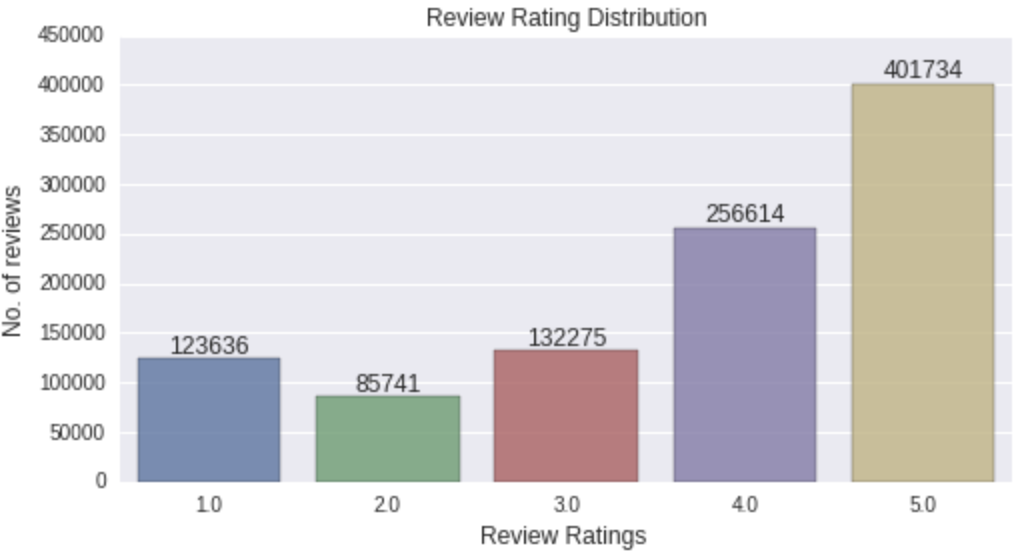
**Figure 1**

1. **Bottom 20 reviewed businesses**: Similar to the top 20, we also looked into the businesses which had least number of reviews. From Figure 2, we see that “Les Aliments Felix Mish” with only 1 review has least number of reviews in the entire dataset of 1 million records.



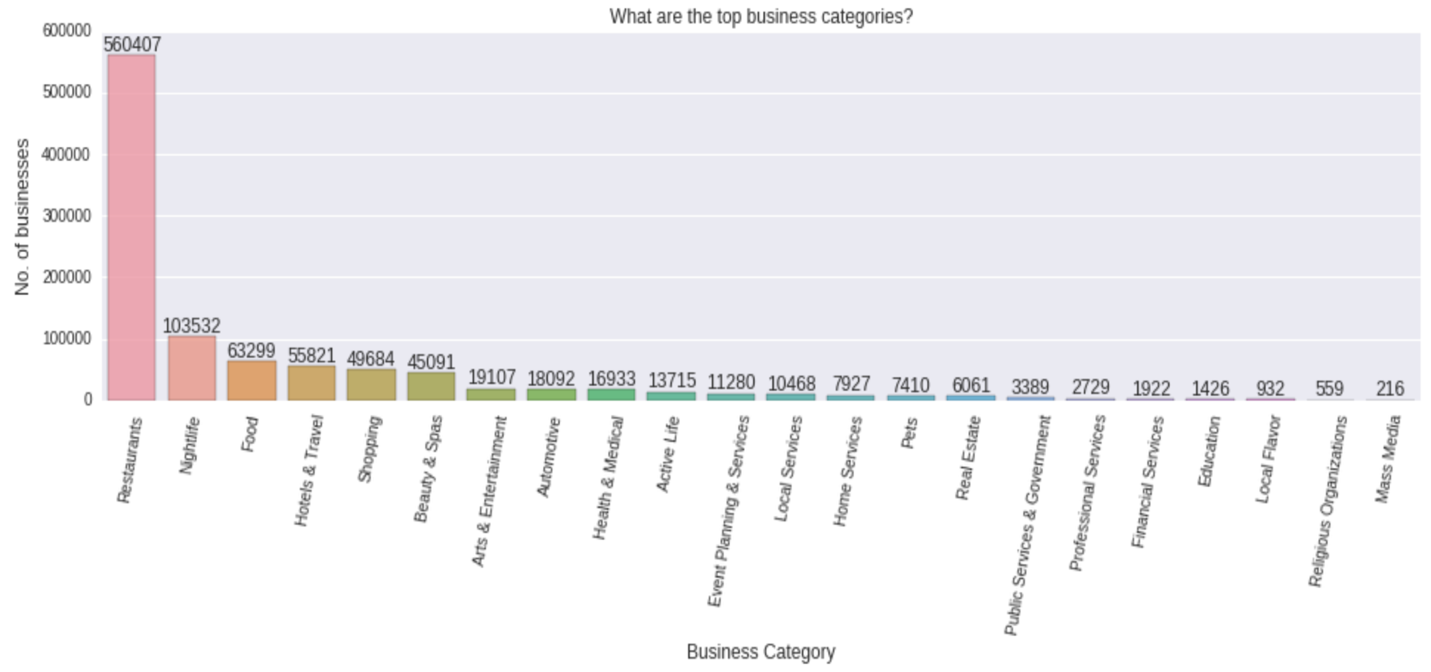
**Figure 2**

1. **No. of reviews by review rating:** After looking at the top 20 and bottom 20 businesses, we looked into the distribution of review ratings. Figure 3 shows that rating 5 dominates others with almost 400k records while rating 2 has the least number of reviews with only 85k records out of 1million.

****

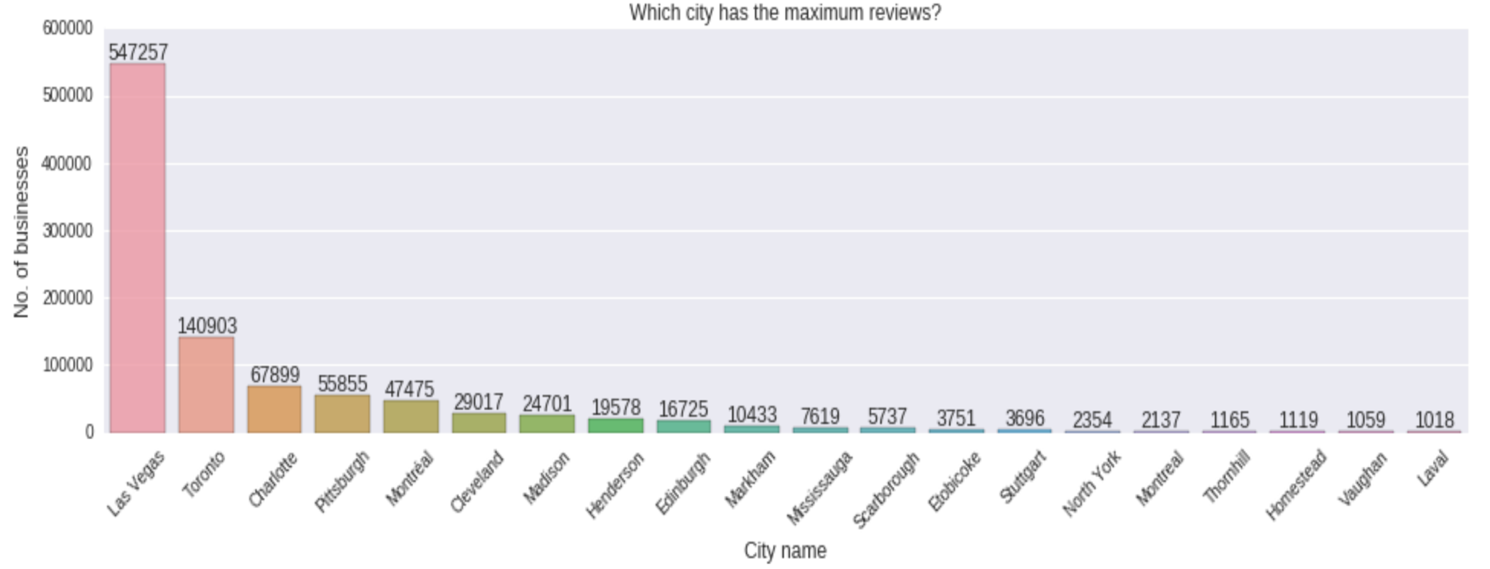
**Figure 3**

1. **Popular business categories:** From Figure 1 and 2, we saw top 20 and bottom 20 businesses. But we don’t know their business categories. So, we looked into some of the most popular business categories as shown in Figure 4. Restaurants is the most popular among customers for posting reviews while Mass Media being the least popular.



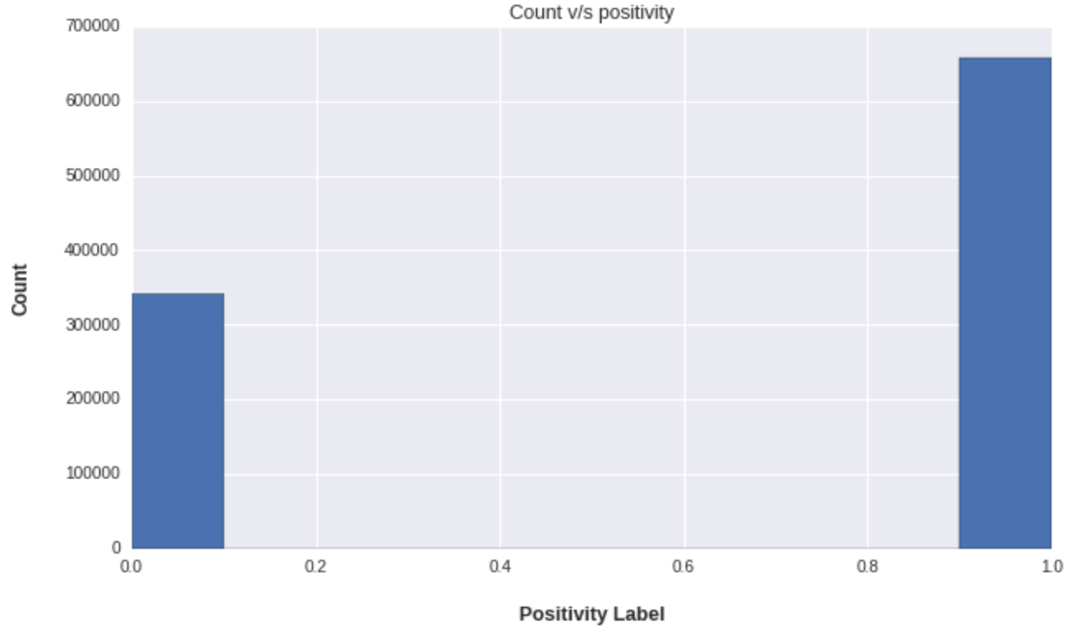
**Figure 4**

1. **Number of reviews by cities:** Customers from different cities post different number of reviews. People from certain places post more reviews as compared to another. As seen in Figure 5, Las Vegas has got the maximum number of reviews. The reason can be that it is a popular place among tourists and so people from all around the place come here for vacation and hence post reviews for the places they visit.

****

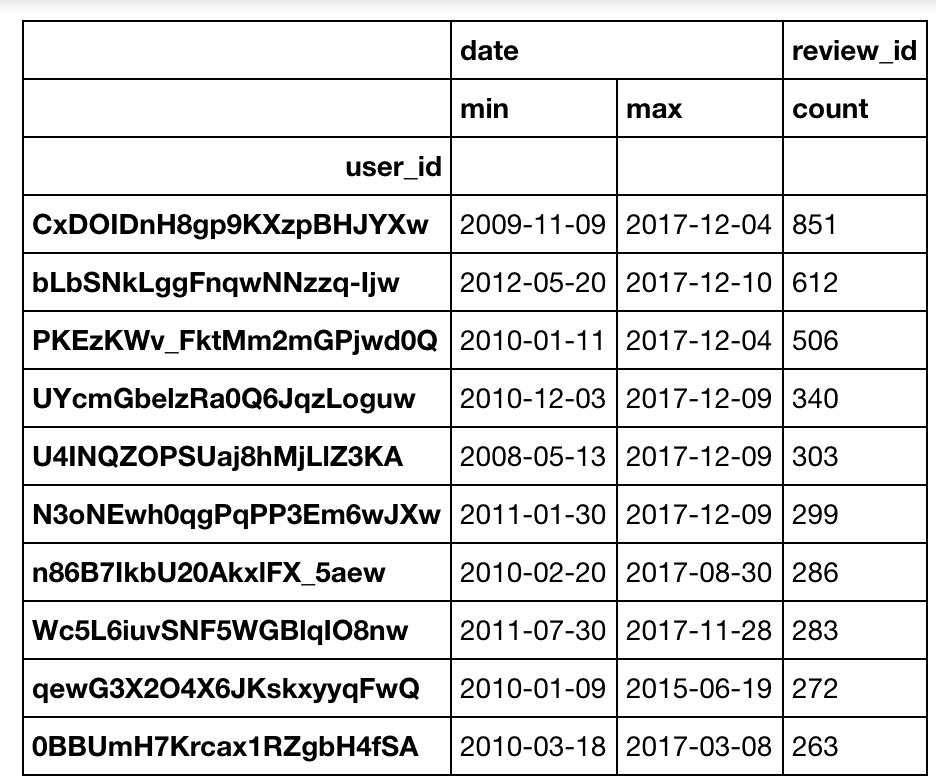
**Figure 5**

1. **Number of positive and negative reviews:** From Figure 3, we saw how different review ratings vary and which rating is more dominant than others. For the sentiment analysis, we have to see how the number of positive reviews varies from negative reviews. Positive reviews are more as compared to the negative ones.

****

**Figure 6**

1. **Top 10 users:** As seen in previous figures, few cities’ business has more reviews, so there is also a possibility that few users post more reviews are compared to others. The user who is most active posted 851 reviews in 8 years from 2009 to 2017.

****

**Figure 7**

1. **Word cloud from reviews:** Since we are working on text analytics, we thought of looking into the word cloud showing most frequently used words in bigger fonts as compared to the words that are not used frequently in the reviews. From Figure 8 it can be seen that place, first, experience, food are some the words that have been used most frequently by users in their reviews.

****

**Figure 8**

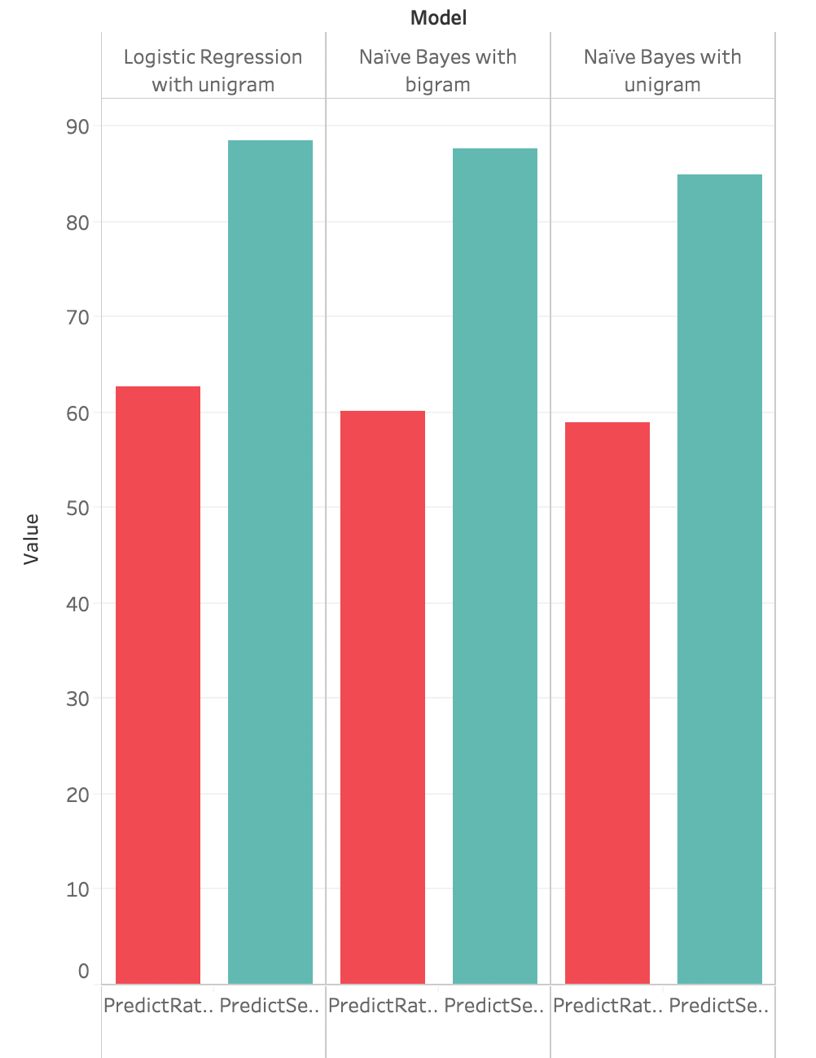
# Methodologies

## N-Gram

This is a method used frequently in text mining or text analysis in order to understand the polarity (positive or negative) of text. For this project, we have reviews from the users and our aim is to predict the sentiment or rating from the text of reviews. Sentiment can be 0 (negative) or 1 (positive) but review rating varies from 1 to 5. We were given rating in the dataset and we first found sentiment based on rating and then used that sentiment as prediction variable. If the rating is 3, 4 or 5, sentiment score was given as 1 and if it is 1 or 2, sentiment score was given as 0.

Now that we have the sentiment score which is our prediction variable, we tried to do the modeling. For that we used models like Naïve Bayes Classifier and Logistic Regression. Since these models are number based, we had to convert text to numbers which can be done using CountVectorizer module from sklearn. This will convert the text to a sparse matrix. With this we have x and y variables which can be used in model.

From the following figure, it can be seen that Logistic Regression proves out to be giving maximum accuracy of 62% to predict rating and 88% to predict sentiment when applied to unigram text followed by Naïve Bayes Classifier on bigram text giving an accuracy of 58% to predict rating and 84% to predict sentiment.



**Figure 9**

## LSTM

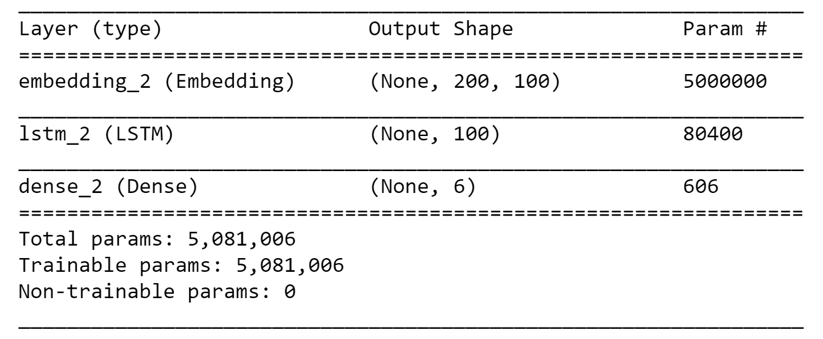
LSTM stands for “Long Short-Term Memory.” It is a method used for many regression and time series prediction, but it can be used for text data as well. Utilizing neural networks, LSTM is a variant of Recurrent Neural Networks (RNNS) but have several advantages over them. They solve the problem of exploding and vanishing gradients that often occur with RNNs. This makes them ideal for working with text data. For our project, we sought to categorize reviews into separate review rating categories with LSTMs.

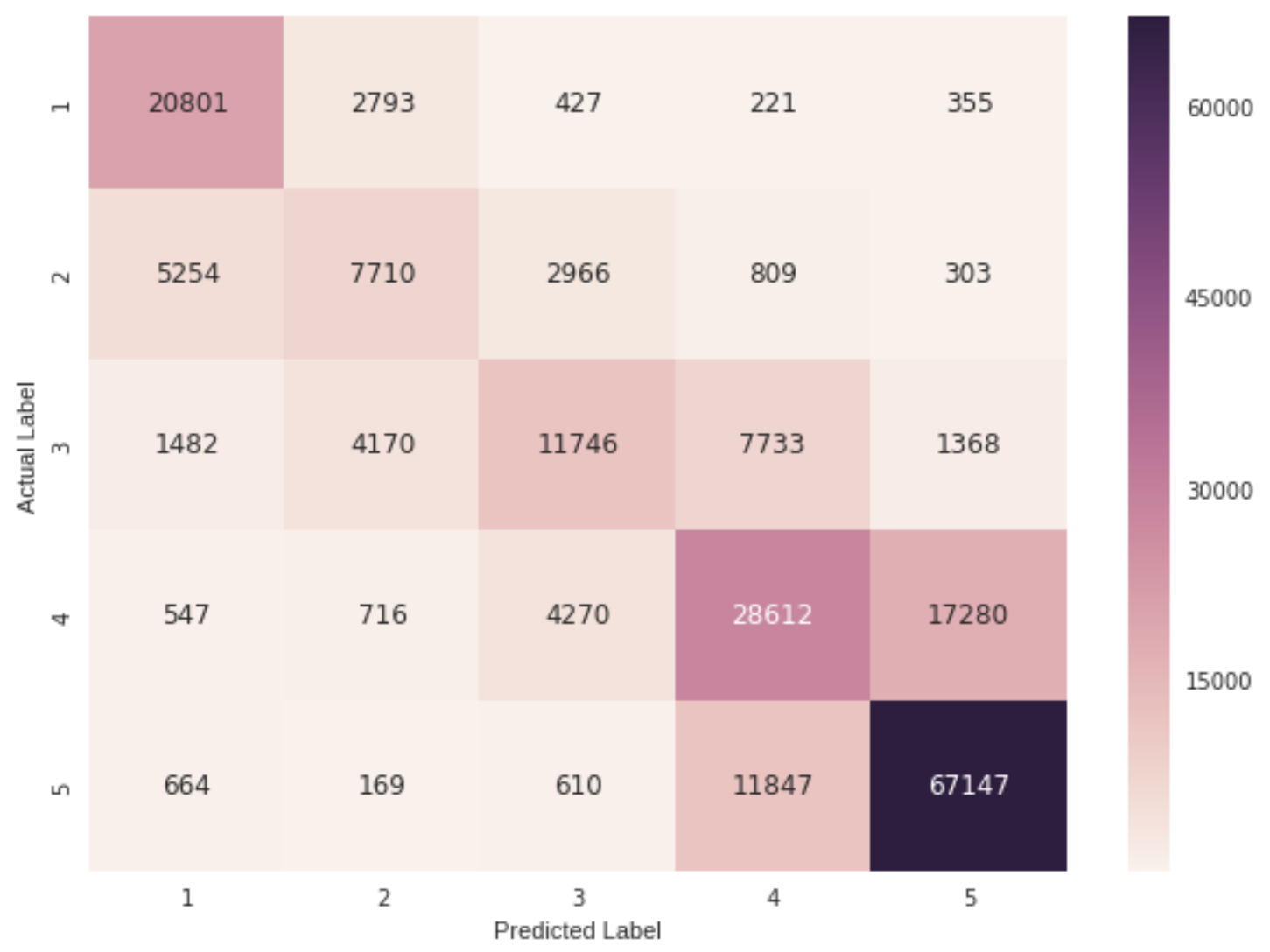
### First Experiment

As an experiment, we began our modeling with a relatively less complex neural network. On one hand, we set up three limitations in the first experiment. First, we limited the size of the overall vocabulary at 50,000 most-commonly-used words, meaning that we were only utilizing the most frequent words in the training dataset and skipped the ones that are less frequently used. Second, the size of training data was also limited and only 25,000 out of more than 2 million reviews were used. Finally, we also constrained the number count of words for each review and the cap was 200 words for very review.

On the other hand, we used sigmoid as our activation function for the final dense layer and “binary\_crossentropy” as the loss function, and the below table is the actual setting of our LSTM network layers. In the embedding layer, we had the vocabulary size of 50,000 words, output dimensionality of 100 and input length of 200 for each training data point. In the LSTM layer, we had the output dimensionality of 100 and output dimensionality of 6 in the final dense layer. The main reason why the output dimensionality in the final dense layer is 6 is because we have 5 different labels (rating from 1 to 5) and we converted this target variable to a one-hot encoding matrix.

As a result, the below graph shows a very promising result. After training for only 3 epochs, we were gradually reducing the loss and increasing the predictive accuracy for both training dataset and validation dataset. This experiment was a huge success as we considered all those limitations in this initial experiment. As we observed a simple LSTM network performed well on the small dataset, we decided to train our model on bigger dataset and the accuracies were 90%, 89%, and 68% for training dataset, validation dataset and test dataset respectively. The below graph was the confusion matrix for the test dataset.



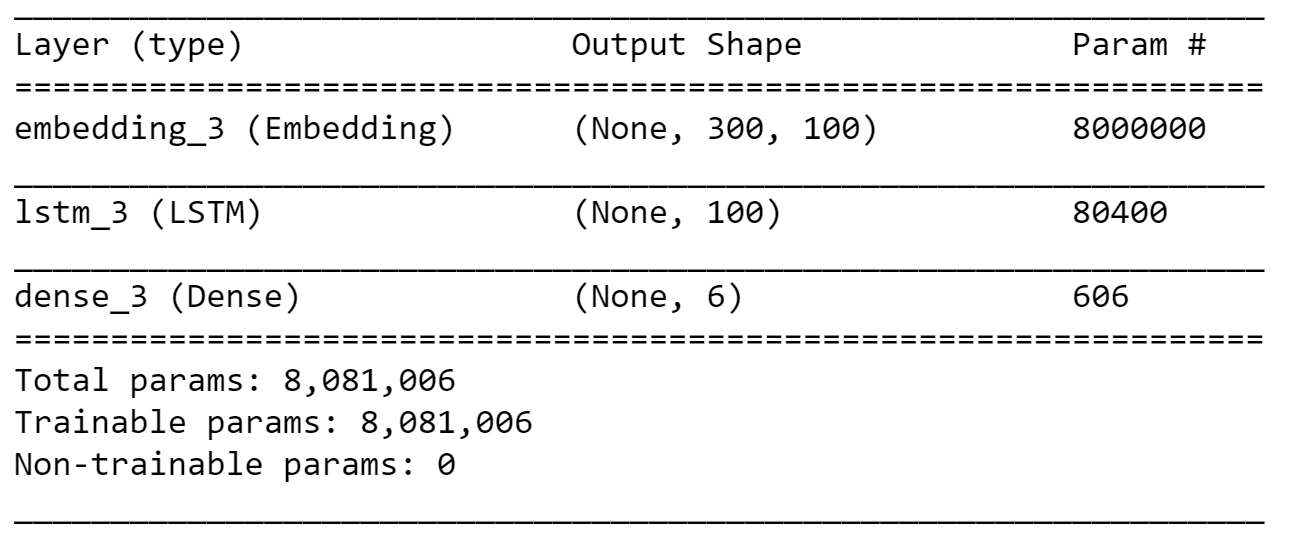
[[1]](#footnote-2)

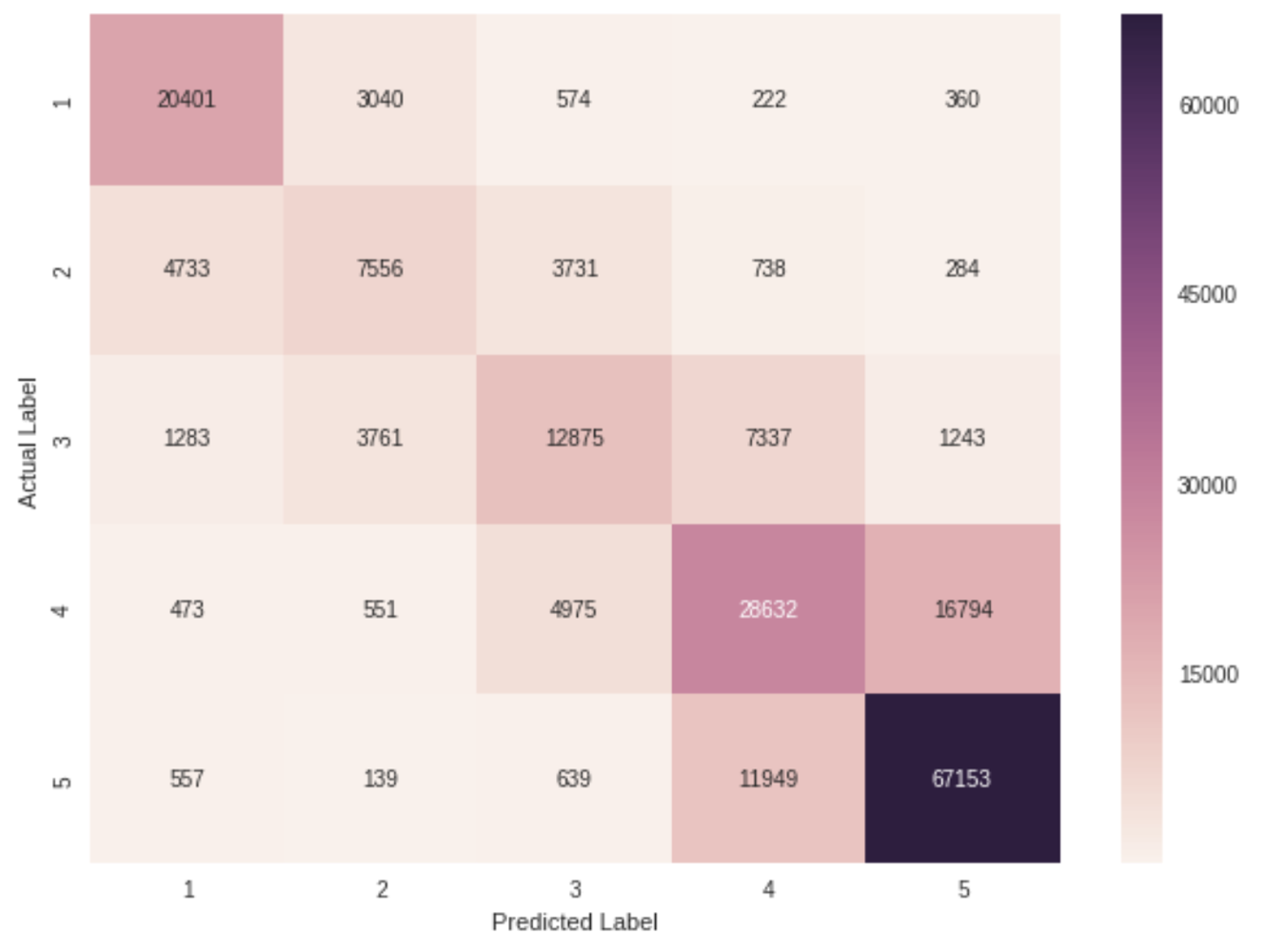
**Figure 10**

### Second Experiment

Upon gaining initial success using a relatively simple LSTM network, we decided to construct a more complex model to see if the predictive accuracy could be improved. On one hand, we increased the size of the overall vocabulary to 80,000 and the size of training data to 1 million reviews, and finally the number count of words for each review was also increased to 300 words for very review.

On the other hand, the activation function for the final dense layer and the loss function were not changed, and the below table is the actual setting of our LSTM network layers. Overall, the total number of parameters (nodes) in the LSTM network was increased by approximately 3 million and we also trained for 10 epochs as compared with previous setting, and we hope this change could improve the predictive power. However, the accuracy for training dataset was gradually increasing from 88% to 92% while validation dataset’s accuracy was consistently around 89% and no any indication for major improvement was observed. In addition, similar outcome was found in the test dataset and the accuracy was also 68%. As shown in the below confusion matrix for the test dataset, there were no obvious changes, except that the number of predictions for rating 3 was slightly increased as the number of predictions for other categories decreased such as rating 1 and 2.





**Figure 11**

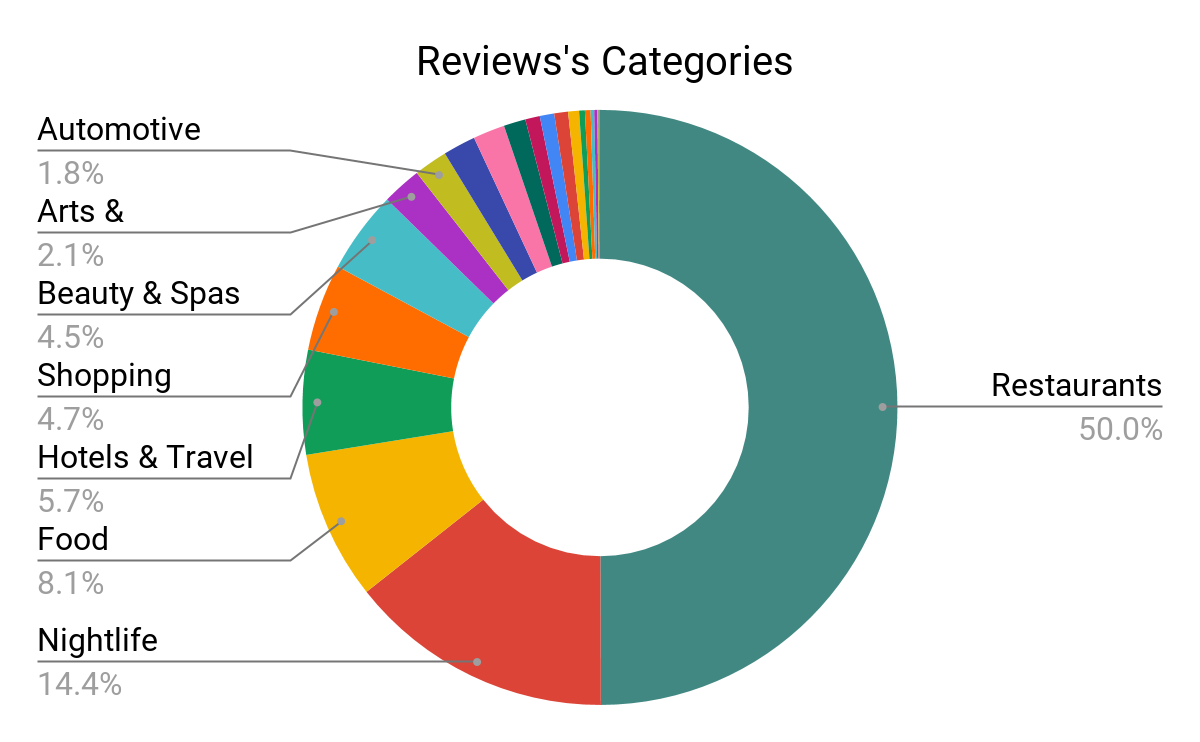
## LDA

LDA is short for latent Dirichlet allocation. It’s a way of automatically discovering topics that a bunch of sentences contain. In more detail, LDA represents documents as mixtures of topics that spit out words with certain probabilities. It assumes that documents are produced in the following fashion: when writing each document, we can find the following discoveries:

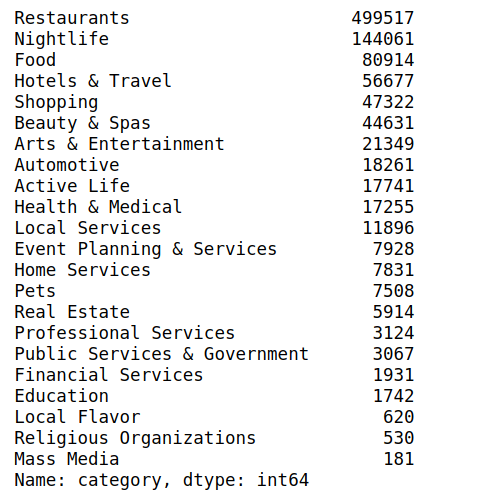
* Decide on the number of words N the document will have (say, according to a Poisson distribution).
* Choose a topic mixture for the document (according to a Dirichlet distribution over a fixed set of K topics). For example, assuming that we have the two food and cute animal topics in a bunch of sentences in a paragraph, we might be able to decide the document to consist of 1/3 food and 2/3 cute animals.
* Generate each word w\_i in the document by:
* First picking a topic (according to the multinomial distribution that you sampled above; for example, you might pick the food topic with 1/3 probability and the cute animal's topic with 2/3 probability).
* Using the topic to generate the word itself (according to the topic’s multinomial distribution). For example, if we selected the food topic, we might generate the word “broccoli” with 30% probability, “bananas” with 15% probability, and so on.

With these founds, we will get scores- the probability weight of different topics of a passage and these scores can be used as features in the machine learning model for predictions. For instance, for all the reviews of yelp dataset, we found 5 topics have been talked in general, in each review, we can get scores for the individual topics, 0.4,0.1, 0.2, 0.25, 0.05 for topic 1, 2, 3, 4 and 5. Th these scores can be used as features for yelp ratings predictions.

Before running LDA for the whole dataset, we found a yelp review category list and classified all the yelp reviews based on the list, here’s the total categories for all the reviews, which we can see that the majority of yelp reviews in the dataset are about restaurants, nightlife and Food. There are 499519 out of 1 million reviews about restaurants and 144061 about nightlife, and 80914 reviews about food.



**Figure 12**



**Figure 13**

We think the it might make more sense to run the LDA model in each category so that we can know that the topics being talked about in specific category.

### LDA model for restaurant reviews

After obtaining the clean text from text cleaning pipeline, we ran LDA on the model, here are the five topics we found eventually in all the reviews on restaurant. In topic 1, the top 10 popular words are place, good, food, great, service, love, really, price, best, delicious are the top 10 popular words being talked about in the review for restaurants.

**Topic 1:**   
 '0.028\*"place" + 0.028\*"good" + 0.028\*"food" + 0.021\*"great" + 0.013\*"servic" + 0.012\*"love" + 0.010\*"realli" + 0.010\*"price" + 0.010\*"best" + 0.009\*"delici"'

**Topic 2:**

'0.034\*"chicken" + 0.028\*"fri" + 0.019\*"burger" + 0.017\*"sauc" + 0.016\*"sushi" + 0.015\*"rice" + 0.015\*"roll" + 0.013\*"soup" + 0.013\*"order" + 0.013\*"dish"'),

**Topic 3:**

'0.024\*"pizza" + 0.015\*"chees" + 0.013\*"dessert" + 0.013\*"steak" + 0.012\*"bread" + 0.012\*"salad" + 0.008\*"sauc" + 0.007\*"dish" + 0.007\*"wine" + 0.007\*"potato"')

**Topic 4:**

'0.067\*"buffet" + 0.011\*"pour" + 0.007\*"cest" + 0.006\*"mai" + 0.006\*"nicht" + 0.005\*"station" + 0.005\*"plu" + 0.005\*"trè" + 0.005\*"dan" + 0.005\*"sehr"')

**Topic 5:**

'0.024\*"order" + 0.012\*"time" + 0.012\*"food" + 0.010\*"like" + 0.010\*"wait" + 0.010\*"would" + 0.010\*"tabl" + 0.009\*"even" + 0.008\*"came" + 0.008\*"didnt"'

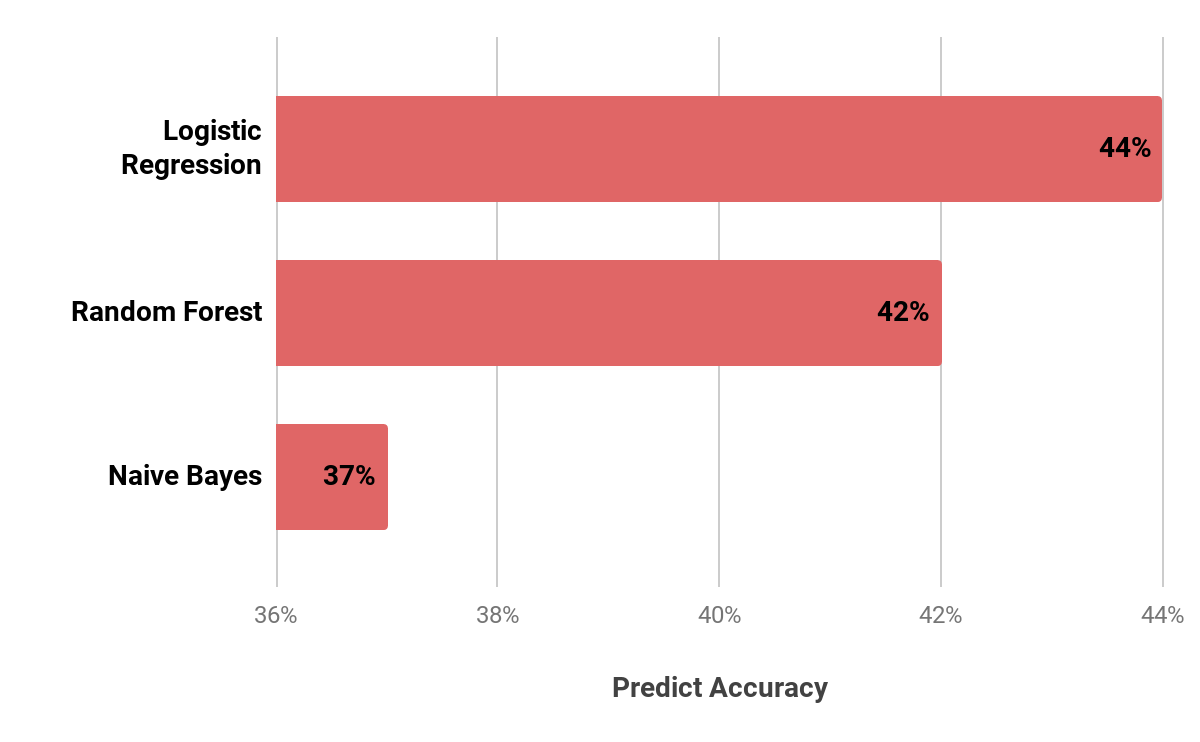
In each review, we were able to get the scores of each topic. For instance, the first review is : “Super simple place but amazing nonetheless. It's been around since the 30's and they still serve the same thing they started with: a bologna and salami sandwich with mustard. Staff was very helpful and friendly. “

And here are the output scores of the 5 topics (0 to 4 stands for topic 1 to topic 5): (0, 0.627663), (4, 0.17196113), (2, 0.14566064), (1, 0.05252065), (3, 0.0021945399)

Since topic 1 has the highest score, around 0.63 among all the scores, which means topic 1 is the most representative for this review and it talks a lot of great things for this restaurant. And these 5 scores of each topic can be used as features to predict the review rating.

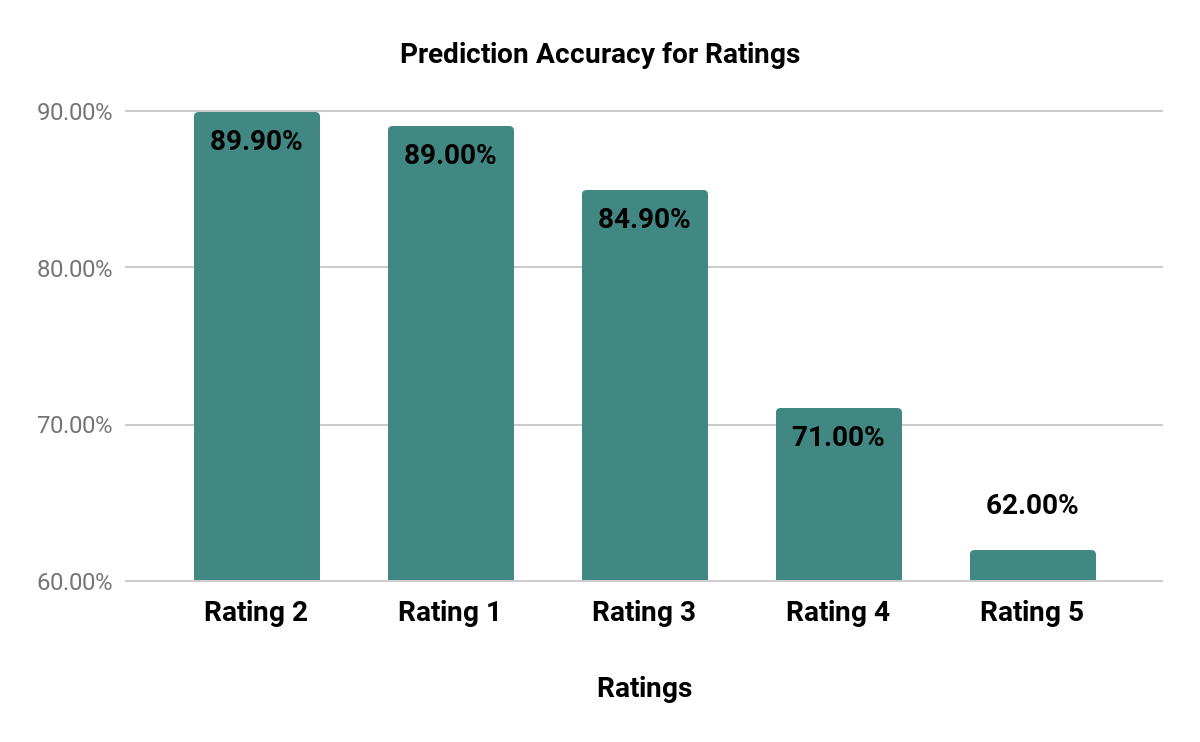
LDA model for different ratings

For 499517 reviews about restaurant, we ran the LDA model and used the output scores as machine learning features. We tried 3 machine learning algorithms: Logistic regression, Naïve Bayes and Random Forest Models, and obtained their accuracy separately, 44%, 37%, and 42% for all the categories rating 1 to rating 5.



**Figure 14**

Here’s prediction result for each rating. From the result, we can see that rating 2 and 1 have the best prediction accuracy over 89% and rating 5 has the lowest accuracy.



**Figure 15**

### Insights of LDA Model

This mythology for LDA model in restraint reviews can be used for other categories, such as food, hotel and travel, shopping, financial services, home services etc. It’s very helpful for us to grasp the topics being talked in the reviews and understand the context of reviews under specific category. We can also run the machine learning models for every category and cross check their prediction accuracy and customize the strategies to increase the prediction accuracy.

Furthermore, the LDA output (topics scores) can be combined with other features generated from topic modeling's and to be applied in the machine learning algorithms development, such as features from sentiment analysis and other topic modeling's can be combined with LDA output to train the machine learning models. It might potentially increase the prediction accuracy.

# Appendix

iPython Notebooks

# Works Cited

[1]. Alexandr, B. (n.d.). Rating prediction with sentiment analysis.

[2]. Asghar, N. (n.d.). Yelp Dataset Challenge: Review Rating Prediction.

[3]. Ganu, G., Elhadad, N., & Marian, A. (n.d.). Beyond the Stars: Improving Rating Predictions using Review Text Content.

[4]. Jong, J. (n.d.). Predicting Rating with Sentiment Analysis.

[5]. Kavousi, M. (n.d.). Estimating the Rating of Reviewers Based on the Text.

1. Figure [↑](#footnote-ref-2)