SMU Data Science Review

Volume 2 | Number 1 Article 13

2019

Leveraging Reviews to Improve User Experience

Anthony Schams
Southern Methodist University, aschams@smu.edu

Iram Bakhtiar
Southern Methodist University, ibakhtiar@smu.edu

Cristina Stanley
Yelp, cristina.stanley@gmail.com

Leveraging Reviews to Improve User Experience

Anthony Schams[1], Iram Bakhtiar[1], Cristina Stanley[2]

[1]Master of Science in Data Science,
Southern Methodist University,
Dallas, TX 75275 USA
[2]Yelp, 140 Montgomery Street,
San Francisco, CA 94105 USA
{aschams, ibakhtiar}@smu.edu
cristina.stanley@gmail.com

Abstract. In this paper, we will explore and present a method of finding characteristics of a restaurant using its reviews through machine learning algorithms. We begin by building models to predict the ratings of individual reviews using text and categorical features. This is to examine the efficacy of the algorithms to the task. Both XGBoost and logistic regression will be examined. With these models, our goal is then to identify key phrases in reviews that are correlated with positive and negative experience. Our analysis makes use of review data publicly made available by Yelp. Key bigrams extracted were non-specific to the restaurants examined, but key trigrams were specific to restaurants, including menu items. While the models were successful in predicting high-rated reviews, they struggled to identify negative reviews with acceptable accuracy. The method outlined in this paper proved successful in extracting positive trigrams that are highly specific to the restaurants examined, and we propose these phrases be emphasized on Yelp pages to allow users to quickly learn the items of highest quality at a restaurant.

1 Introduction

The rise of technology and social media has led to an increase in consumer review sites such as Yelp.com, which allow users to share their personal experience at a restaurant or business. Yelp is a leading online platform for consumers to find and submit reviews on businesses and services. It covers numerous cities across the United States as well as 31 other countries and categories ranging from health care providers, dry cleaning services, restaurants to pet sitting and tutoring. Yelp provides information about a business, as well as allows users to rate their experiences (by assigning number of stars as they deem appropriate for their level of satisfaction/dissatisfaction). Users are also able to upload and share photos from their experiences with other members of the online platform. Importantly, this sharing is done publicly, so all Yelp users have access to these reviews. This paper and its analysis will focus on the restaurant category.

We believe that as the volume of online reviews increase, in today's culture of 'instant knowledge' and 'instant gratification', it will become necessary to apply machine learning algorithms to manage and analyze this abundance of data in a timely manner. This becomes ever more evident when we take into consideration the fact that online reviews not only include assignment of star ratings but also text reviews which depict the user's sentiments of their experience more clearly.

Due to the weaknesses of star ratings, we will largely be focused on the text reviews. Star ratings can be difficult to act upon, because of the level of arbitrariness inherently found in them. A 3-star experience for one user may be 4 stars for another user. They can also be a challenge to analyze because a rating represents the overall experience, which can be composed of both positive and negative individual features [1]. Examining text data however allows us to evaluate individual facets of the user experience and hopefully see how those fit into the overall experience.

We will be performing sentiment analysis on Yelp user reviews, specifically focusing on recommended reviews. A variety of machine learning algorithms will be used for classifying the user reviews. We will also limit the analysis to a robust metro area, Las Vegas. This was chosen because it is the most represented metropolitan area in the publicly available dataset being used. Our goal is to first use the features of the restaurant and the review to try and predict the star rating of the review. This is to test the efficacy of our model. The feature importance will then be examined. We believe this feature importance will allow us to help determine what is important in a user's experience at a restaurant. This can then be leveraged to improve both user experience on Yelp and can be provided to businesses to help improve themselves.

The paper is organized as such: Section 2 contains background information on sentiment analysis and the machine learning techniques employed in our analysis. Section 3 contains an overview of the data used. Section 4 provides a brief overview of the methods we use to complete our analysis. Section 5 goes over the results of our analysis, while section 6 discusses some of the ethical concerns with the analysis we performed. Section 7 is the conclusion, while section 8 is saved for a discussion of future work to extend upon the analysis completed here.

2 Background

2.1 Online Reviews and Yelp

Yelp was founded in 2004 by Jeremy Stoppelman and Russel Simmons and is headquartered in San Francisco. It was founded with the idea of helping communities find local businesses that were highly or poorly rated by others. Yelp has branched into other business verticals as well. Besides reviews on local businesses, users can now find events, make reservations, order food, have food delivered using Yelp. Yelp earns revenue via advertising. This makes understanding the qualities of a restaurant important as it is in Yelp's best

interest. Understanding these qualities can improve the efficiency of the advertisements delivered to specific users, as better targeted advertisements can be placed.

Yelp's data encompasses millions of reviews from millions of users. Per Yelp as of Q2 2018, it had an average of 32 million unique mobile app users a month, an average of 72 million unique mobile visitors a month, and an average of 74 million unique desktop visitors a month [2]. It is however important to note that not all reviews are treated equally. Yelp makes use of a proprietary software that labels reviews as 'Recommended' or 'Not recommended'. While specific details are not made public, Yelp uses the quality of the review and the reliability and activity of the user in order to determine if the review is trustworthy. The goal of this is to ensure the reviews published are in fact representative of a business. This practice specifically eliminates voting manipulation from users and bots trying to game the system. Only recommended reviews are factored into the overall Yelp rating of a business. While they do not affect the overall Yelp rating, non-recommended reviews are available to view.

Users have the option of providing ratings, which includes both a star rating and a text review. Text reviews are required as part of this rating. Text ratings provide a clearer representation of the service and food the reviewer received. This ultimately provides more information to other users to make informed decisions on restaurants. As such, a major focus of our analysis will be on the text reviews.

2.2 Opinion Mining/Sentiment Analysis

Semantic orientation (SO) is a term denoting the subjectivity and opinion expressed in a text. Sentiment analysis is the process of extracting the semantic orientation of a text. This analysis typically involves identifying an opinion, often positive or negative, and an intensity, the degree to which the words express that opinion. Sentiment analysis can therefore be applied to online reviews, which can allow us to measure the popularity and overall sentiment towards a product. [3]

There are two main approaches to automatic sentiment analysis. The first is a lexicon-based strategy, in which the semantic orientation of the words are used to find the orientation of the document as a whole. This involves building a lexicon of words with their individual semantic orientations. The second strategy is a more machine learning approach, in which a classifier is built using labelled instances of text. Both of these strategies make use of words as features, but the method of assigning semantic orientation to these features is different.

The lexicons mentioned above can be made manually or using seed words to grow the list of words. Lexicon-based approaches often focus on adjectives, where a dictionary of adjectives is first assembled with semantic orientations. A text to be analyzed then has its adjectives extracted and annotated with the corresponding SO values from the dictionary. These SO scores are then aggregated to score the text.

The majority of machine-learning approaches involve training Naïve Bayes or Support Vector Machine classifiers. These strategies will be outlined later in the paper. The features

they work on are often either unigrams or bigrams. While these techniques have been shown to have high accuracy classifying the domain they were trained on, their performance drops when used on text outside of this domain [3].

2.3 Classification Techniques

XGBoost is a gradient tree boosting algorithm that is widely used. It is highly scalable and is built for handling sparse data, such as the feature vectors of online reviews. In this case, the features can be attributes of the restaurant being reviewed or the words, both single words (unigrams) and pairs of words that appear together (bigrams), that are used in the review. A tree-based method such as this one functions by splitting variables into bins such that the purity within each node is high. As more and more splitting occurs, a tree-shaped structure appears. This process produces a decision tree that minimizes the error (by some chosen metric such as RMSE or MAE, discussed shortly) between predicted and actual values. XGBoost produces multiple trees via this method, which are then used to 'vote' on a final prediction. Because it is a tree ensemble model, prediction is a fast operation [4].

For example, if we wanted to predict the star rating of a Yelp review using the neighborhood the restaurant was in and the type of food they served -a first tree might first split on whether the restaurant is in Queens, NY and then split on whether Ethiopian food was served. (The trees have binary splits). A second tree might split first on Italian food being served, and then on being in Saint Laurent, QC. In both trees, these splits are chosen such that they maximize the purity of the resulting nodes. In the case of numerical values, splits occur based on thresholds determined by the algorithm such that purity is highest. In the case of actual Yelp reviews, there are many more features. Because of the abundance of features, different trees can split on different explanatory variables. In addition to this, switching the order splits occur can also yield slightly different trees.

XGBoost was used as it has additional features that allow it to perform better than traditional tree ensemble methods. It uses a randomization parameter to reduce the correlation among trees, making their splits more orthogonal which will ultimately increase accuracy. Highly complex trees are also penalized, as more complex trees struggle to accurately predict unseen data. This reduces overfitting.

One weakness of XGBoost is that the complexity of the model generated reduces interpretability. While regression coefficients have clear interpretation with respect to their effect (in both magnitude and direction), the combination of many variables in a decision tree make interpretation less intelligible at first glance. One strategy to interpret a tree-based method is its feature-importance metric. When a node in a decision tree splits, we can take the decrease in impurity as a measure of that feature's significance. And because the features and splits are selected based on the purity that results, we can use the prominence of a feature being used as a splitting factor as a proxy for its importance.

Predicting numerical ratings comes with an interesting decision that must be made. Because rating is numerical and ordinal, it is possible to treat it as a continuous variable and

perform regression. However, because Yelp ratings are limited to integers between 1 and 5 inclusive, this task is more of a supervised classification problem. Luckily a linear regression objective function in XGBoost will yield integer predictions because all of the training data have integer targets. We would therefore like to compare the two objective functions, linear regression and multiclass classification.

Logistic regression is another machine learning algorithm that can be used in supervised classification problems. It can be used because it provides coefficients for all features trained that can be easily interpreted. In multiclass (more than 2) classification, each possible classification level has its own set of coefficients corresponding to the features in the dataset. When a data point is to be classified, a logistic function is evaluated for each class using its learned weights and the data point's feature vector. The probability of the data point x being in class c is:

$$P(\mathrm{x} \in \mathrm{c}) = \frac{1}{1 + \exp(-\beta_{0c} - \sum \beta_{\mathrm{ic}} x_i)} \quad (1)$$

Where β_{0c} is the bias term of class c, β_{ic} is the coefficient of the i^{th} feature for class c, and x_i is the value of the i^{th} feature of the data point [5]. Class prediction is assigned as the class whose calculated probability is greatest [6]. With regards to coefficients, a positive coefficient is associated with a feature being more common with members of that class, while a negative coefficient is associated with that feature being uncommon with members of a given class.

2.4 VADER

This project also made use of VADER, or Valence Aware Dictionary for sEntiment Reasoning, a standard lexicon built specifically for microblog-type content. It combines lexical features with five generalizable rules relating grammatical and syntactic tendencies used by humans to express sentiment intensity [7]. This includes text-based tendencies such as multiple exclamation points and heavy use of capital letters. It was developed with the goal of being a gold-standard for sentiment analysis. It can even outperform humans on certain tasks and outperforms other lexicons and machine learning techniques on a variety of tasks. VADER's strength comes from its valence awareness and deep lexicon, which makes use of more subtle indicators of sentiment such as punctuation (such as a number of exclamation points) and even emoticons. It will be used here to evaluate the sentiment of reviews. VADER produces four numbers to measure sentiment: positive, neutral, negative and compound. Positive neutral, and negative measure the levels of positive, neutral, and negative sentiment of the text respectively, while compound is a combined score that estimates the sentiment of the text as a whole. The higher the compound score, the more positive overall the text analyzed is. These scores will be used as features in our models.

2.5 Representation of Text

Our text data will be represented using term frequency-inverse document frequency (tf-idf.) Tf-idf is a statistic that represents how important a given word is within a corpus. It increases proportionally with how often it appears in a given document of the corpus (a review) as well as how rare it is in other samples [8]. It is stored as a sparse matrix, in which each row of the matrix corresponds to a document and each column represents a word (or bigram, trigram, etc.) being in the document. Because this becomes relatively storage-intensive, we will restrict our training and test data set to a small subset of total reviews when working with tf-idf.

When performing these algorithms, it is important to reduce the feature space to one that makes computations both effective and efficient. It is therefore common to reduce the vocabulary size by removing common words not indicative of class (called stop words) and by only including words that give the most mutual information on class [9]. The feature space can be further reduced by only including words that have appeared in the training data multiple times. [10]

When building a tf-idf vectorizer, there are a number of parameters that can be customized to improve both performance and computation speed. With such a large corpus of documents, it is likely that the vocabulary is very large. A larger vocabulary becomes harder to store and analyze, so we can restrict our final vector to words that either appear a certain number of times or appear in a certain proportion of documents. For this analysis, we will be restricting our tf-idf vectors to words that appear in at least 0.1% of documents. This will remove very rare words from our vocabulary, which saves both storage space and prevents overfitting. We also remove common stop words, as well as words that are not commonly stop words but appear in over 30% of documents (such as 'food') in our corpus because they are functionally stop words.

An additional parameter for the vectorizer that we will be examining is the number of words being examined as a unit. While analysis might look at only single words, we also have the option of looking at bigrams (pairs of words found together), trigrams (three words found together), or higher n-grams. We will be looking at the effect of looking at bigrams and trigrams and their effects on prediction results.

2.6 Performance Metrics

We will evaluate the quality of the regression using root mean square error (RMSE) and mean average error (MAE). Both of these metrics will increase the less accurate the predictions are. The equations for RMSE and MAE are shown below as (2) and (3) respectively.

$$\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2\right]^{\frac{1}{2}} \tag{2}$$

$$\frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x}_i| \tag{3}$$

As we can see, the RMSE involves the square of the difference of predicted values, which will ultimately mean that large errors will be more pronounced in this statistic [11].

Logistic regression and XGBoost supervised classifiers were evaluated by their accuracy (4).

$$\text{Accuracy} = \frac{\text{True Positive(TP)} + \text{True Negative(TN)}}{\text{Total Number of Observations}} \tag{4}$$

3 Data

The data used in this study are publicly provided by Yelp. The data is divided into seven subsets or tables.

One table includes all metainformation on a business entity, including business ID (designated by Yelp), name, address, review count, and the category or categories it is listed under. Another table houses 82 features including business ID (generated by Yelp), kinds of parking available near the business, if its family-friendly and of the restaurant has easy access for disabled patrons to name a few. A third table contains user reviews on all business categories. There is also data on the business hours of each business, user information, as well as Yelp check-ins and tips.

We will be utilizing these three databases for our analyses. The Yelp data includes reviews from 5,200,000 users on 174,000 businesses across 11 metropolitan areas. As mentioned earlier, we will be limiting the scope to include data for only restaurants in Las Vegas.

The data set was cleaned prior to analysis. XGBoost can only operate on numerical and Boolean data, so a number of features in the original data set had to be one-hot encoded. This involved converting some of the data columns from strings or dictionaries into multiple, one-hot encoded variables. In operating with the text data, we elect to represent each document using tf-idf. This cleaned dataset could also easily be passed into a logistic regression or another machine learning model, and one-hot encoded features are very easy to interpret.

4 Methods

4.1 Predicting Rating using Sentiment

VADER is used to measure the sentiment in reviews. These sentiment features, along with the other features of the restaurant, are then used in predicting rating using the techniques discussed above, XGBoost and Logistic Regression. Luckily, the original training set is large so we will use 60-40 train-test split of these to build and evaluate the tf-idf-using models.

4.2 Identifying Key Words and Phrases in Reviews

XGBoost and Logistic Regression each have their own methods of expressing feature significance in the predictions they make. In XGBoost, feature importance is measured by number of times a feature is split on in the ensemble, and the gain from these splits [12]. Gain is measured as the change in entropy as a result of the split occurring. It is noted that this statistic does poorly when the features can take on a large number of values, but in this case, many are one-hot encoded so that is not a concern [13]. Logistic regression in contrast provides a coefficient for each feature that can be directly interpreted as the change in the natural logarithm of the odds of being in that class, all other features equal [5]. Our strategy is to use tf-idf features in a logistic regression or XGBoost models to identify key phrases in reviews that are indicative of a high-quality or low-quality review. These phrases can then be emphasized on Yelp to allow users to quickly and easily learn about the experience a restaurant would provide. To do this, we now look at each restaurant individually and build models accordingly. This yields significant features that are unique to each restaurant. Both algorithms are used to provide a well-rounded view of each restaurant's review corpus. This will be discussed further in the results section.

5 Results

5.1 Predicting Star Rating

General models to predict star rating were attempted using both XGBoost and logistic regression. Table 1 shows the results of predicting star rating of Yelp reviews using features from the restaurants, the VADER analysis of the review, and the tf-idf vectorization of the review looking at n-grams of certain sizes. (n-grams not listed means no tf-idf data was used for the reviews. With respect to XGBoost, the first thing that we see is that including n-grams, even just unigrams, improves review prediction, with RMSE decreasing from 0.881 to 0.829 (~5.6% decrease.) This is a lower RMSE compared to previous studies [14]. A similar improvement was seen in the multiclass classifier, where error dropped from

47.8% down to 43.8% (an 8.4% decrease). This is strong evidence that inclusion of unigrams in analysis can improve prediction and classification when text data is involved.

It is observed that considering bigrams however did not improve performance significantly. In fact, including bigrams with unigrams increased both the RMSE and MAE of the linear regression models. It also improved performance in the classifier only slightly, increasing accuracy from 56.2% to 56.3%. Considering the vocabulary is 75% larger, this marginal increase in performance and significant increase in model size are not promising. Indeed, there is a great deal of previous research into whether the inclusion of bigrams and higher n-grams in a bag-of-words model improves performance. Initial research found that performance increases, while other studies found the opposite [12]. This work falls into the latter category. There is also previous work that shows a small increase in accuracy when limiting the number of n-grams in the model [13]. XGBoost regression performed worse when limiting the number of n-grams, while its performance increased when attempting classification. Logistic regression performed slightly worse with this limitation, its accuracy dropping from 62.9% to 61.0%.

Table 1. Performance of XGBoost and logistic regression in predicting star rating on 10% of the restaurant reviews in the dataset.

Algorithm	Objective Function	n-grams	Vocabulary Size	RMSE	MAE	Accuracy
XGBoost	Linear	1-2	6943	0.831	0.631	-
XGBoost	Linear	1-2	500	0.844	0.635	-
XGBoost	Linear	1-2	1000	0.837	0.632	-
XGBoost	Linear	1	4098	0.829	0.627	-
XGBoost	Linear	0	-	0.881	0.685	-
XGBoost	Softmax (multiclass)	1-2	6943	-	-	56.3%
XGBoost	Softmax (multiclass)	1-2	500	-	-	55.6%
XGBoost	Softmax (multiclass)	1-2	1000	-	-	56.6%
XGBoost	Softmax (multiclass)	1	4098	-	-	56.2%
XGBoost	Softmax (multiclass)	0	-	-	-	52.2%
Logistic Regression	-	0	-	-	-	51.1%
Logistic Regression	-	1	500	-	-	61.0%

Logistic Regression	-	1	4144	-	-	62.9%
Logistic Regression	-	1-2	6935	-	-	63.5%
Logistic Regression	-	1-3	7032	-	-	63.5%

Interestingly, logistic regression outperforms XGBoost in this task. While logistic regression performs worse when not using any text data, it outperforms XGBoost significantly when n-grams are used, resulting in an accuracy of 62.9% compared to 56.2%. Much like XGBoost, performance is not greatly improved with the inclusion of bigrams. Inclusion of trigrams also offers no improvement in classification accuracy. This is particularly notable because it is a comparable result to Conneau et al. (2016), who used a deep convolutional neural network to perform this task with 64.7% accuracy.

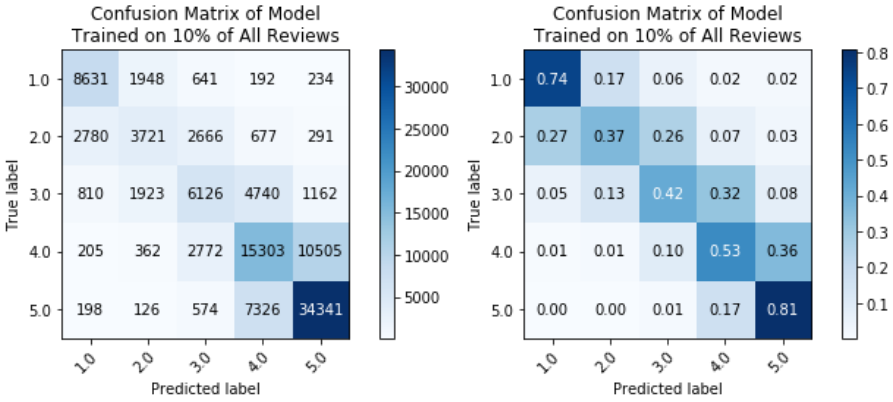

Fig. 1. Confusion matrices of logistic regression models trained on 10% of all reviews.

Looking at the normalized and non-normalized confusion matrices for the model trained on 10% of the data, we see that logistic regression is strongest at predicting 1-star and 5-star reviews but struggles with other ratings. Importantly, because the rating system is ordinal, misclassifications are most commonly within 1 star of the true rating. This reinforces the idea that the star rating is imprecise. There is clearly however some merit to the system and our model, as we can still predict the star rating of a review within 1 with high confidence. This indicates that the noise in ratings does not completely drown out the signal. Our model also tends to overestimate the rating when it does make a mistake. This could be interpreted as people generally more likely to be generous with their star rating given their text review.

5.2 Identifying Key Words and Phrases

Logistic Regression is used to find key phrases for two reasons. The first is the interpretability of its coefficients over XGBoost's interpretability. Its coefficients contain information on both the direction and magnitude that the feature influences classification. This will leave us with easily interpretable results. Logistic Regression also out-performed XGBoost in classifying reviews, so there is inherently more confidence in this interpretation.

It is also important to note that this model uses only bigrams or trigrams and does not include any smaller n-grams as features. When smaller n-grams are included in the feature space, the coefficients for these shorter n-grams dominate over the larger n-grams. We therefore limit our model strictly to either bigrams or trigrams. Looking at the confusion matrices of this model, we see that we are able to predict high-rated reviews (4 and 5 stars) well but struggle with lower-rated reviews. Because our model performs poorly on low-rated reviews, we do not have a high confidence in the coefficients that logistic regression provides for these class levels. We will therefore focus specifically on the model's insights on 5-star reviews.

Table 2 shows the results of our analysis for the most reviewed restaurant in the Yelp dataset, Mon Ami Gabi in Las Vegas. Here we can see that using bigrams instead of trigrams results in a much larger vocabulary size and a small increase in accuracy of the model. Although not the principle concern, it is important to note that the accuracy of the model restricted to this restaurant is significantly less accurate than the model looking at all restaurants. We hypothesize that this is because of the smaller, more specific vocabulary when looking at only a single restaurant. This specific vocabulary such as the targets of this study may not strictly be associated with specific class levels. This weaker association however can still be valuable in improving user experience of Yelp. With the smaller sample size, the risk of overfitting is also present. Each phrase being counted as a feature may only need to appear in a handful of reviews, which could result in an inflated coefficient if all these features happen to be of a certain rating. However, increasing the minimum document frequency for an n-gram to be included in the analysis did not improve accuracy of the model. This is evidence that these less common features are superfluous but not detrimental to the model.

Looking at positive bigrams, we see that there is sadly not very much information. These phrases are mostly vague and non-specific to Mon Ami Gabi, such as "highly recommend" and "favorite place." We are still however able to extract some information, including the fact that Bloody Mary's are associated with positive experiences. We are also able to conclude that the restaurant is in some way associated with the Bellagio Fountains. These could be a marketing opportunity for Mon Ami Gabi, as they could advertise themselves to people that look up Bloody Mary's or the Bellagio Fountains online.

Positive trigrams however are much more informative and specific to Mon Ami Gabi. We learn that mentioning the Bloody Mary bar in reviews is strongly associated with positive reviews. We also find more menu items in the positive trigrams, and positive reviews

of the wait staff. These are both actionable insights for Yelp. The high-quality wait staff and the menu items associated with positive reviews could be emphasized on the restaurant page.

By using these positive n-grams, we could improve the usability of Yelp by providing a list of the most high-rated items on the menu. Currently, Yelp does have a system that they call 'review highlights.' While the mechanisms of this are not made public, it appears to be a more naïve method of selecting key phrases to emphasize about a restaurant. It appears to select phrases that are relatively specific to a given restaurant that are mentioned in a large number of reviews[1]. Yelp also currently has a 'Popular Dishes' system that takes the results from 'review highlights' and uses machine learning to "gather and organize photos and reviews of popular menu items.[2]" While Yelp is confident that the best items on a menu are most likely to appear in these highlights[3], our method offers up some improvement to these two systems. For example, our method provides a quantification of quality in terms of an interpretable regression coefficient. Our method also provides the opportunity to find notable items that are not heavily reviewed, that would likely be missed by Yelp's current 'review highlights' system.

Table 2. Positive bigrams and trigrams of Mon Ami Gabi. Each section includes the 5 phrases with the highest coefficients, as well as additional informative n-grams.

Bigrams		**Trigrams**	
Accuracy	53.5%	Accuracy	50.8%
Positive Bigrams		**Positive Trigrams**	
"highly recommend"		"directly street Bellagio"	
"favorite restaurant"		"great service great"	
"favorite place"		"bloody mary bar"	
"bloody mary"		"trip las vegas"	
"Bellagio fountains"		"restaurant las vegas"	
"superb service"		"flourless chocolate cake"	
		"bananas foster crepes"	
		"ordered prime steak"	
		"say good things"	

1 https://www.yelp-support.com/article/What-are-Review-Highlights?l=en_US
2 https://www.yelp-support.com/article/What-are-Popular-Dishes?l=en_US
3 https://venturebeat.com/2018/06/19/yelps-popular-dishes-ai-highlights-the-food-everyones-talking-about/

5.3 Predicting Features for Cuisines

In predicting star ratings of reviews, logistic regression saw a marked increase in accuracy when looking at all restaurants of a specific cuisine in Las Vegas. This increase in accuracy is about 12% at the cuisine level compared to the restaurant level. (~56% compared to 50%.) This is likely due to the larger sample sizes pulled for the cuisines, as each phrase counted likely has more reviews associated with it. The increase in accuracy comes despite the decrease in vocabulary size compared to individual restaurants. This is evidence that some overfitting occurred at the restaurant level, as we have decreased the number of features and improved accuracy. This implies that those additional features at the restaurant level were likely incorrectly learned due to the small number of reviews in which they appeared. Unfortunately, it also implies that phrases that are unique to certain restaurants or a subset of restaurants will be less likely to appear as features because they don't meet the 0.1% appearance threshold to be counted.

Besides running logistic regression on the top restaurants in Las Vegas, logistic regression was also run on the top with the intent to predict key words and phrases that may associate with positive reviews specifically for those cuisine types. The results of Chinese cuisine are shown in table 3.

Table 3. Results of logistic regression models looking at bigrams and trigrams of Chinese cuisine restaurants in Las Vegas.

Bigrams		**Trigrams**	
Accuracy	56.4%	Accuracy	51.6%
Positive Bigrams		**Positive Trigrams**	
"wait come"		"highly recommend place"	
"highly recommend"		"time las vegas"	
"best ve"		"excellent customer service"	
"like home"		"visit las vegas"	
"restaurant place"		"best dim sum"	

The methodology applied for logistic regression at a cuisine level follows the same premise as that for the individual restaurants. Table 3 shows the top five positive and negative bigrams for Chinese cuisines. It may be noted here that we were expecting the bigrams to be a bit vague since they are cuisine specific, and not restaurant specific.

As expected, positive bigrams like "really good" and "highly recommend" are strongly related to positive experiences. However, like the analysis done at the restaurant level, these bigrams are not very informative. These phrases could all be used to describe restaurants of high quality of any cuisine. They are simply generic praise for restaurants. Unfortunately, unlike at the restaurant level, positive trigrams are not very informative. They do not include menu items. While dim sum is mentioned in one trigram, it simply means that whichever restaurants warrant reviews mentioning 'best dim sum' has very high ratings.

This is in contrast to the phrase "dim sum" being found as positive, which would indicate that dim sum across the many restaurants is good.

6 Ethics

Various ethical issues with regard to how people maintain their privacy and a business functions arise with such a system in place.

These reviews are published online and are effectively available forever with their publication in the Yelp Dataset. This makes the personal information found in the dataset very hard if not impossible to become compliant with GDPR guidelines, as it is difficult to provide the right to be forgotten.[4]

There is also a major concern with the source of this information being crowdsourced that all facts and opinions being shared are true and honest. This analysis is entirely dependent on the users of Yelp to share their honest experiences at these restaurants and businesses. This makes the usefulness of the reviews published as informative or misinformative as the reviewers intend it to be. This is the purpose of filtering reviews as recommended or not recommended, but it would be foolish to assume that the system in place to do this was flawless.

This type of analysis even lends itself to potential abuse. While the data obtained for this study has had identifying information removed, it is possible to query quotes from reviews in order to find original posts on Yelp and the people who posted them. This opens up the possibility of harassment for negative or unwanted posts.

A small or new less popular business can potentially be negatively impacted by a system such as ours in place. A shift of users become expecting of a quick and easy-to-digest phrases that our system provides could occur. Because some businesses may not have a sufficient number of reviews, review highlights may not be trustworthy or usable at all, ultimately impacting business for the less-reviewed.

Another major concern with the system outlined here is the possibility of abuse from restaurants to get specific dishes to be highly recommended, or attacks from competitors to get specific dishes deemphasized. Currently, Yelp allows restaurants to highlight specific menu items that are mentioned often in reviews. This combined with Yelp's own categorization of reviews as "recommended" and "not recommended," makes it difficult for a restaurant to game the system and artificially get an arbitrary dish to be recommended for the sake of profit. Our system as described does not contain additional safeguards against abuse of this. If the informative n-grams are updated frequently however, artificially recommended dishes would ultimately be penalized as future reviews would indicate the true, lower quality of the item.

[4] https://gdpr-info.eu/art-17-gdpr/

This analysis could also inspire others to scrape data from Yelp in order to better understand reviews. This practice is explicitly against Yelp's Terms of Service. Because the data used was obtained from Yelp with the goal of education, there is no concern that we are breaking any Terms of Service.

7 Conclusions

This study finds that logistic regression outperforms XGBoost in classifying restaurant reviews on Yelp. This was done using tf-idf vectors as features for the algorithms. As mentioned in section 5, including text data in the form of unigrams improves this performance with RMSE decreasing ~5.6%. Inclusion of bigrams and trigrams do not improve upon this metric. Logistic regression, which outperformed XGBoost in this task, was then used to find specific phrases to highlight on Yelp pages to improve Yelp's usability. This is because it is easier to interpret coefficients of the features provided, since the logistic regression coefficients provide information on the direction as well as the magnitude that the features influence classification. We believe this is an improvement on Yelp's current systems, as our system can be used to provide quick feedback on a restaurant using the consensus established in its reviews. It also provides an improvement over Yelp's current 'review highlights' system as our system allows items mentioned fewer times to still be highlighted as of high quality.

8 Future Work

Looking towards the future, one of the major focuses of work to be done is looking at how these results can be applied to other types of commerce. The strategy outlined here could be used for other business verticals on Yelp, or on other review sites online. There is also room for improvement to our model. The simplest improvement would come from using the full corpus of Yelp reviews, of which we only had access to a subset of. Also, currently, the phrases found to be significant are treated independently from one another at both the restaurant and cuisine-type level. A more complex system that pools all of these important phrases together could in turn be used to automatically select phrases unique to an individual or small subset restaurant, which could create an identity and help differentiate between similar restaurants. Knowledge of a restaurant's menu could also be incorporated into the model to more confidently select key phrases as menu items. This could be easily incorporated on Yelp's side, as they have knowledge and in fact provide menu information to their users. However, because Yelp does not allow scraping of these pages, we did not incorporate it into our model. An additional topic to examine would be potential clustering of the phrases found describing each restaurant. Restaurants could then be clustered based on their best attributes, allowing users to potentially find new restaurants both locally and while travelling.

References

1. Guzman, Emitza, and Walid Maalej. "How do users like this feature? a fine grained sentiment analysis of app reviews." *2014 IEEE 22nd international requirements engineering conference (RE)*. IEEE, 2014.
2. Yelp, "Q2 2018 Shareholder Letter". *Available at http://www.yelp-ir.com/static-files/73d9c17f-0936-4ab3-ad98-3a0ab329a5b6. Released August 8, 2018*
3. Taboada, Maite, et al. "Lexicon-based methods for sentiment analysis." Computational linguistics 37.2 (2011): 267-307.
4. Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.
5. Peng, Chao-Ying Joanne, Kuk Lida Lee, and Gary M. Ingersoll. "An introduction to logistic regression analysis and reporting." The journal of educational research 96.1 (2002): 3-14
6. Karsmakers, Peter, Kristiaan Pelckmans, and Johan AK Suykens. "Multi-class kernel logistic regression: a fixed-size implementation." 2007 International Joint Conference on Neural Networks. IEEE, 2007.
7. Gilbert, CJ Hutto Eric. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14. vader. hutto. pdf. 2014.
8. Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.
9. McCallum, Andrew, and Kamal Nigam. "A comparison of event models for naive bayes text classification." AAAI-98 workshop on learning for text categorization. Vol. 752. No. 1. 1998.
10. Joachims, Thorsten. "Text categorization with support vector machines: Learning with many relevant features." European conference on machine learning. Springer, Berlin, Heidelberg, 1998.
11. Willmott, Cort J., and Kenji Matsuura. "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance." Climate research 30.1 (2005): 79-82.
12. Zheng, Huiting, Jiabin Yuan, and Long Chen. "Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation." Energies 10.8 (2017): 1168.
13. Quinlan, J.R. Mach Learn (1986) 1: 81. https://doi.org/10.1007/BF00116251
14. Tang, Duyu, et al. "User modeling with neural network for review rating prediction." Twenty-Fourth International Joint Conference on Artificial Intelligence. 2015.
15. Bekkerman, Ron, and James Allan. Using bigrams in text categorization. Technical Report IR-408, Center of Intelligent Information Retrieval, UMass Amherst, 2004.
16. Fan, Mingming, and Maryam Khademi. "Predicting a business star in yelp from its reviews text alone." arXiv preprint arXiv:1401.0864 (2014).
17. Conneau, Alexis, et al. "Very deep convolutional networks for text classification." arXiv preprint arXiv:1606.01781 (2016).