Rajiv Mahajan, Idriss Dem, Lennard Evertz

**Change in Sentiment of Customer Reviews Pre-Post COVID-19**

**“Change in Sentiment of Customer Reviews Pre-Post COVID-19”**

1. **ABSTRACT**

Customer satisfaction has always been at the core of a service business. Restaurants belong to this class of businesses. In the modern days of the internet, customers do not rely on mouth-to-mouth recommendations anymore as online reviews are easier to create and access. Websites for online reviews are used to get an estimate of the quality of a product or restaurant. Popular examples are Yelp, TripAdvisor, and Google Reviews. Generally, there are two options to be a customer at a restaurant. One can either visit the restaurant, experience their customer service, and enjoy the food in their vicinity, or order food to one’s location of choice. In both instances, the quality of food will likely be part of a review’s content. Besides, the first option can include comments on the service of a restaurant, while the second option can incorporate information about delivery pace. In April 2020, the nature of how we can consume food from restaurants has drastically changed. Because of the Covid19 virus, restaurants had to close completely or offer delivery options, if they wanted to serve their customers. With our research, *we want to determine the change in sentiment of customer reviews in the transition of two separate periods*. We define the pre-covid timeframe as the time before April 2020. Our first analysis will focus on the change in sentiment from pre-covid times to covid-times, and from covid-times to post-covid times. The second boundary would be to comprehend the usefulness of reviews. What is the general sentiment for each of the restaurants? How does the sentiment of a review play a role in the usefulness of reviews? Is there generally a significant change in customer sentiment in between these periods?

1. **INTRODUCTION**

Yelp users give evaluations and compose reviews about organizations and administrations. These reviews and ratings help other yelp users to assess a business or assistance and settle on a decision. The issue most users face these days is the absence of time; the vast majority can't peruse the reviews and simply depend on the business' ratings. This can be misdirecting. While evaluations are helpful to pass on the general insight, they don't pass on the setting that drove users to that experience. For instance, if there is an occurrence of an eatery, the food, the vibe, the service, or even limited offers can regularly impact the user’s evaluations. The type of service has especially been impacted in the past year, due to the closure of indoor dining for many restaurants. This data isn't visible from rating alone, nonetheless, it is available in the reviews that users compose. The arrangement of Yelp restaurant reviews into one or more "Food", "Service", "Delivery", and other "Value" classes are the issues in thought. Sources of info are the Yelp restaurant reviews and accompanying data on the reviewer. Consider Yelp reviews: "They have not the best happy hours, but rather the food is acceptable, and administration is far better. At the point when it is winter, we become regulars". It is effectively deduced that this survey discusses "food" and "administration" in a positive opinion, and "arrangements/limits" (cheerful hours) in a negative feeling. Separating arrangement data from the reviews and introducing it to the users, will assist the users with understanding why a commentator appraised the eatery "high" or "low" and settle on a more educated choice, keeping away from the tedious course of perusing the whole rundown of restaurant reviews.

Sites, such as Yelp, have a star rating framework that allows clients to perceive what the overall assessment of a specific business is without pursuing every one of the reviews for that specific business. Be that as it may, there is a wealth of client-created data online from web-based media stages like Twitter, Facebook, or blog entries where customers express their opinion about a specific item or business. Since the reviews on Yelp gives both the star rating and the text for each review next to additional information like how useful people think a review is or how many friends a reviewer has, we will determine how the overall content of reviews as well as quality of reviews has changed during the transition of pre covid to covid times, followed by the transition “back to normal”.

1. **LITERATURE REVIEW/RELATED WORK**

There is an enormous number of papers on related subjects, for instance, proposal frameworks (Adomavicius, G., Tuzhilin, A., 2005), useful companion expectation strategy (Nolan Miller et al., 2005), and rating forecast.

Adomavicius, G. and Tuzhilin, A. (2005) presents us with an outline of suggested frameworks. Furthermore, it portrays the current version of suggestion strategies that are fundamentally partitioned into three classes, content-based, collaborative, and hybrid recommendation approaches. In any case, there are impediments to these methodologies. This paper talks about a few potential augmentations that can further develop suggestion abilities, just as make proposal frameworks material to a more extensive scope of use (Nolan Miller et al., 2005).

Michael J. et al., (2007) presents us with an essential content-based suggestion framework; it suggests a food item dependent on the description of the food item, just as the profile of the client's advantage. These two factors together decide the last proposal. Although the subtleties of the food item might contrast in various suggestion frameworks, certain features are remaining in a similar manner. For instance, the necessary resources to analyze food items are highlighted (Michael J., Pazzani and Daniel Billsus, 2007).

Gayatree Ganu et al., (2009) gave us a more comparable model. A free-text design survey is hard for PCs to break down, comprehend, and analyze. To recognize the data in the message reviews, this paper presents new specially appointed and relapse-based proposal strategies that into the text-based part of client reviews (Gayatree Ganu et al., 2009).

Previously utilized methods for feeling grouping can be characterized into three classes. These incorporate AI calculations, connect investigation techniques, and score-based methodologies. The adequacy of AI procedures, when applied to opinion arrangement errands, is assessed in the spearheading research by Pang et al, 2002.

Ziqiong Zhang et al., (2011) utilized standard AI strategies gullible Bayes and SVM are joined into the space of online Cantonese-composed eatery reviews to naturally arrange client reviews as positive or negative. The impacts of component introductions and element sizes on characterization execution are talked about.

In a recent study, ReviewTrackers (2021) analyzed gathered online reviews statistics and compared trends with previously dominant behavior. Gathering reviews from a broad range of industry and from different review websites, they determined that review interaction is up by 50% from pre-pandemic levels. Additionally, the study highlights those reviews are significantly shorter now than ever.

1. **RESEARCH QUESTION**

This research work looks at the claims of ReviewTrackers (2021) in more detail by utilizing a scraped dataset from Yelp, containing Italian restaurant reviews in the northern New York City area. More precisely, we will analyze how reviewing behavior changed from pre-pandemic levels, and how it is developing now as restaurants opened for indoor dining once again. Our claim is supportive of the findings of ReviewTrackers (2021). However, we will analyze the reasons behind this more thoroughly. For instance, we are analyzing the sentiment of reviews in these different time periods. Which reviews were deemed useful by other site visitors? What other things have changed?

In the heart of it, we will determine the change in sentiment of customer reviews in the transition of two separate periods. We define the pre-covid timeframe as the time before April 2020.

Our first analysis will focus on the change from pre-covid times to covid-times. Also, the change from during covid to post covid will be studied. The second boundary of interest starts with this year, on January 01, 2021. Precisely,

* + What elements of a review play a role in the usefulness of it?
  + Is there generally a significant change in customer sentiment in between these periods? What other things have changed?

1. **Data Analysis and Results**
   1. **DATA CRAWLING/COLLECTION**

We will scrape openly available reviews from Yelp.com. Doing this, we utilize Selenium to scroll and click through the pages to scrape all available reviews from a search URL on Yelp. This resulted in scaping reviews of 20 restaurants that appear when one searches for “Italian Food” near New York City, New York. Our scraper will open restaurant pages, scroll down, and click on “next page” button until all the review pages are scraped. Our python function returns a data frame, which will be saved as a .csv file for further analysis. The script used can be examined in appendix [2].

* 1. **DATA PRE-PROCESSING**

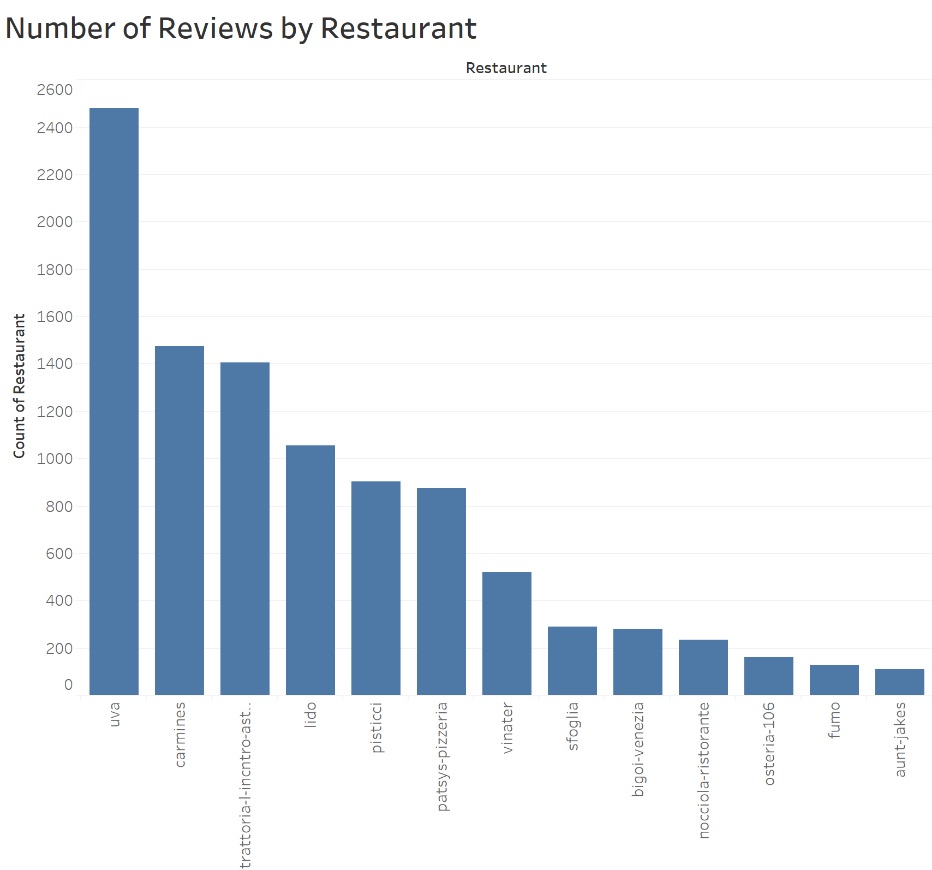
Diverse pre-handling procedures were applied to eliminate the noise from our datasets. It assisted with lessening the component of our datasets, and henceforth assembling more accurate data significantly quicker. The goal is to improve the overall data quality for enhanced results when applying algorithms. In our case, preprocessing falls into 2 parts: data cleaning and data transformation.

Data cleaning is the most common way of fixing or eliminating mistaken, ruined, inaccurately designed, copy, or inadequate information inside a dataset. When consolidating different information sources, there are numerous chances for information to be copied or mislabeled. In case information is erroneous, results and calculations are questionable, even though they might look right. There is no one specific way of endorsing the specific strides in the information cleaning process because the cycles will shift from dataset to dataset. Yet, it is urgent to set up a format for your information cleaning process, so you realize you are doing it the correct way without fail. The main steps involved in data cleaning are as follows: replace null value in columns useful, funny and cool with 0 (meaning nobody clicked on either of these buttons, which will result in not existing html tags); convert date column into pandas datetime format in order to apply time flags. Our review aims to compare different time periods; remove punctuation as a step to allow for text mining; convert reviews to lower case to avoid misleading results in text mining; and finally remove stop words. Indeed, stop words are commonly used words that do not add meaning to sentences and thus can be ignored without affecting the meaning of the sentence, i.e., ‘a’, ‘the’ and so forth (Singhal, G., 2020).

Data transformation is the process of changing the format, structure, or values of data. For data analytics projects, data may be transformed at two stages of the data pipeline. Transforming data yields several benefits: time flags, tokenization, lemmatization and anonymizing reviewers. As far as time flags are concerned, because it is important to compare time periods, it was decided to apply 3 time flags characterized by the letters A, B and C and corresponding to 3 periods: before covid, during covid and after covid (Singhal, G., 2020). To carry on, tokenization is a needed process allowing to break reviews into words, contents or meaningful elements called token. Lemmatization is applied to normalize the data for a more uniform form of every word and for improved text matching (Singhal, G., 2020). Lastly, reviewers were defined as numbers instead of their names because the study should be anonymous.

Further, following steps were taken in python to complete pre-processing phase. Refer to APPENDIX 1 for the python code:

* The NLTK library was used to download all stop words in the English language and then filtered out from every review.
* The NLTK library was utilized to tokenize every element from every review. The NLTK’s word tokenizer is very simple to use (as it comes as a package to be applied to concerned sentences) and can split for standard contractions, separate appearing at the end of lines, and treat most punctuation characters as separate tokens. The “word\_tokenize” built-in function is used on single sentences or on columns of sentences through an “apply” function along the desired axis or column.
* The Spacy package was used for lemmatization. It is a popular natural language processing framework that provides easy application of natural language processing tasks such as lemmatization. After downloading the Spacy modules, the built in “.lemma\_” can be applied on a sentence or included within a defined function to lemmatize a column of different sentences or reviews (via an “apply” function on the selected column).
  1. **EXPLORATORY DATA ANALYSIS**



*Figure 1: Count of restaurant reviews scraped from Yelp.com*

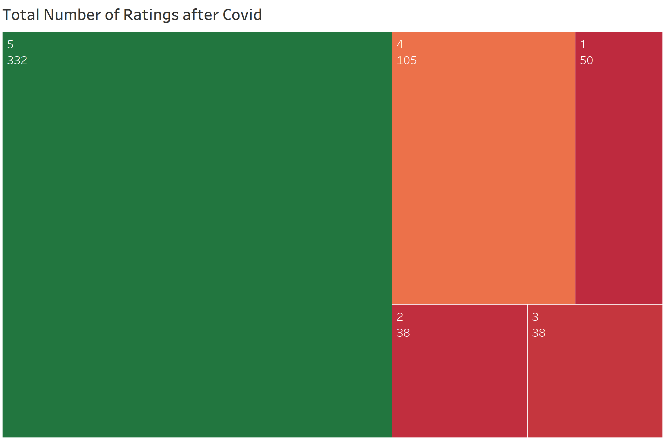
For analysis purposes, we are only looking at restaurants with more than 100 reviews, cutting the dataset down to 13 restaurants with a total of 9924 reviews. The reasoning behind this is that most restaurants with fewer reviews do not have any reviews from time period A, making a comparison infeasible.

We define a review as an ensemble of elements. Each review contains the elements listed in table 1. The variable that can be most intuitively understood is arguably rating. The rating system is a five-level value system taking integer values for each review, where 1 denotes the worst and 5 represents the best possible grade. The average of all reviews is the most dominant indicator for people to choose from the vast variety of available restaurants. ReviewTrackers (2021) show that the most often applied filter when looking for restaurants is “4 stars and above”.

*Table 1: Scraped data columns in Yelp data set. Code retrieving this information can be found in Appendix B.*

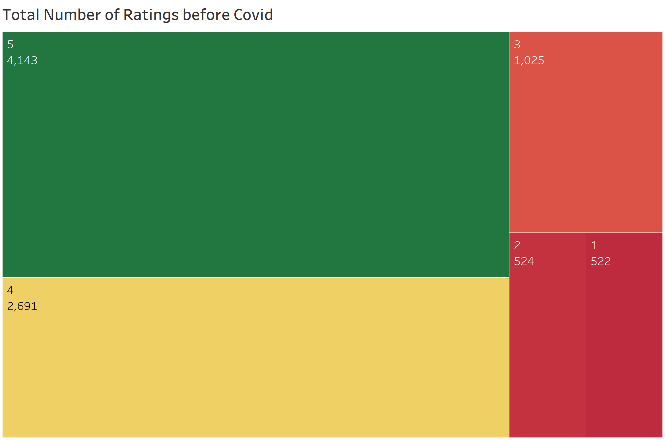
|  |  |
| --- | --- |
| **Element** | **Meaning** |
| Star rating | Rating from 1-5, integer |
| Review text | Review text content from user, string |
| Date | Ranging from 08/09/2014 – 10/11/2021 |
| Useful | Number of people that clicked on “useful”. None if 0. Ranging from 0-91 |
| Funny | Number of people that clicked on “funny”. None if 0. Ranging from 0-93 |
| Cool | Number of people that clicked on “cool”. None if 0. Ranging from 0-52 |
| Friends | Number of friends a reviewer has. Ranging from 0-4999 |
| ReviewCounts | Number of total reviews written by this reviewer. Ranging from 1-16352 (!) |

Hence, it comes of no surprise that the average rating in our dataset, which consists of the top 20 search results, is above 4.0. Figures 2-4 show the count of star ratings for each of the described time periods. As seen in figure 2, the number of reviews pre-pandemic unsurprisingly greatly outnumber the number of reviews in the other two timeframes.

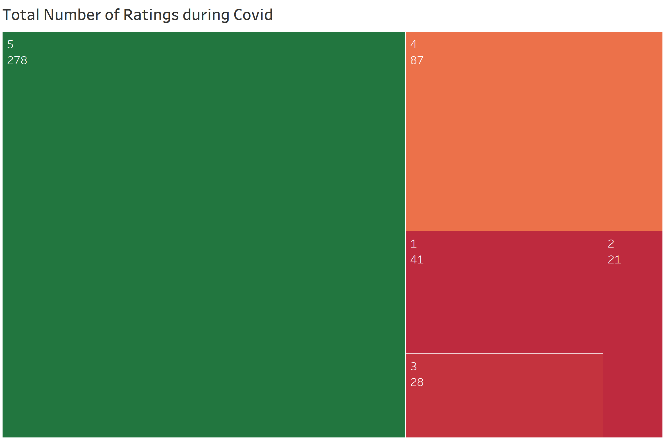


*Figure 2: Number of ratings pre-pandemic with regards to number of stars.*

Looking at the average ratings by quarter from the beginning of 2019, we see that the reviews written during the pandemic show a slightly better conception of restaurants. Figure 5 also shows how reviews in the post-pandemic timeframes tend to fall back to pre-pandemic levels.



*Figure 4: Number of ratings pre-pandemic with regards to number of stars.*



*Figure 3: Number of ratings during pandemic with regards to number of stars.*

Chart, background pattern

Description automatically generated

*Figure 5: Average rating by year and quarter, cut at 2019. We see those ratings during covid tend to be better than both pre-*

*and post-pandemic.*

Supporting the claim that pre-pandemic reviews were longer on average (ReviewsTrackers, 2021), figure 6 shows how the length of reviews for different star levels have changed in the three periods of interest. However, we see reviews from timeframe C (post-pandemic) experience a slight increase of length compared to reviews from timeframe B. This change seems not as significant when looking at the aggregation of all ratings of those periods, disregarding the level of stars (figure 7). This might be because the number of reviews in different star categories are different in different timeframes and require further investigation.

Chart, bar chart

Description automatically generated

*Figure 6: Average review length by star rating and timeframe. Orange represent the fraction of reviews deemed useful,*

*while blue reviews did not get any indication of usefulness.*

*Figure 7: Average review length by timeframe.*

*Figure 8: In this basic regression, it looks like the length of a review has the most influence on usefulness in 4-star ratings.*

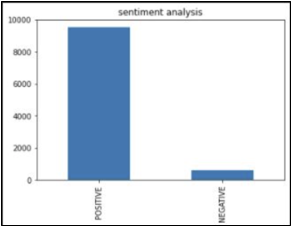
* 1. **Methodology**

Chart, bar chart

Description automatically generatedIn order to analyze the change in sentiment between the time periods, it was necessary to first perform sentiment analysis on all scraped reviews. For this, we compared two separate approaches. First, we looked at a dictionary based counting approach, containing three lists of words. The first list would include positive words such as “delicious”, “great” and “fantastic”, while the second list would deal with negative words such as “terrible”, “bad” and “worst”. Finally, the third list included negations. For each review we would then count the number of positive and negative phrases (considering negations) and compare the two resulting counts. A review was classified as positive whenever the count of positives was greater than the count of negatives.

Figure 7: Average review length over time periods A, B and C.

Second, we utilized the VADAR package. VADAR is a trusted tool for sentiment analysis as it is built on well-established sentiment word-banks, incorporates lots of lexical features for comparison, and so forth. Two main metrics were extracted from this package: sentiment (whether a review is positive or negative) and intensity (strength and compound) of emotion. To get the intensity of a review, the absolute value of the compound is computed. This results in a value between 0 and 1, where a value close to 1 means that a review is either very positive, or very negative. Values closer to 0 indicate a neutral review.

The sentiment analysis identified positive reviews with a very small portion of negative sentiments (figure 8) in the overall dataset.

This sentiment distribution totally makes sense as most restaurants selected for this review analysis have very good overall grades on Yelp. This can be compared in figures 2-4.

Figure 9 shows the intensity value distribution in our dataset.

Figure 8: Most reviews are classified as positive.

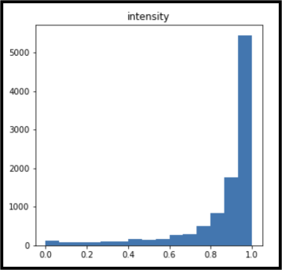


Figure 9: Intensity of reviews.

Intensity should be differentiated from sentiment. When looking at a review intensity, it does not show any signs of how a review rates a restaurant. Only with the additional analysis of our sentiment distribution, we know that most of the intensity scores closer to 1 originate from very positive reviews.

Comparing these two approaches of sentiment analysis, we calculated Cohen’s Kappa to see if they are comparable. With a value of 0.28 the two methods are not strongly agreeing on the classified class (Landis, JR., Koch, GG., 1977). We therefore printed all reviews that were classified as negative for both approaches to compare them through human coding. Ultimately, we decided to go forward with the VADAR based sentiment analysis approach.

Afterwards, the reviews could be investigated to collect and grasp some valuable information about the restaurants. The sentiment alone would only tell if a review is positive or negative.

* 1. **Results**

This subsection of the report will present concrete modelling approaches and the most relevant findings of our analysis. The first part aims at answering the question of whether there is a significant difference in sentiment in the transition of different time frames. More precisely, we defined the pre-covid time frame as the time up to March 14, 2020 (A), the time frame of covid from March15, 2020 to December 31st, 2020 (B) and the post-covid time frame as everything after January 1st, 2021. Obviously, Covid was not over with the beginning of 2021. However, we saw more and more restaurants opening for indoor dining from right around that time, which is why we expect a shift in restaurant reviews from then on. The second part of the analysis focuses on the question of significance of review intensity to predict usefulness of a review.

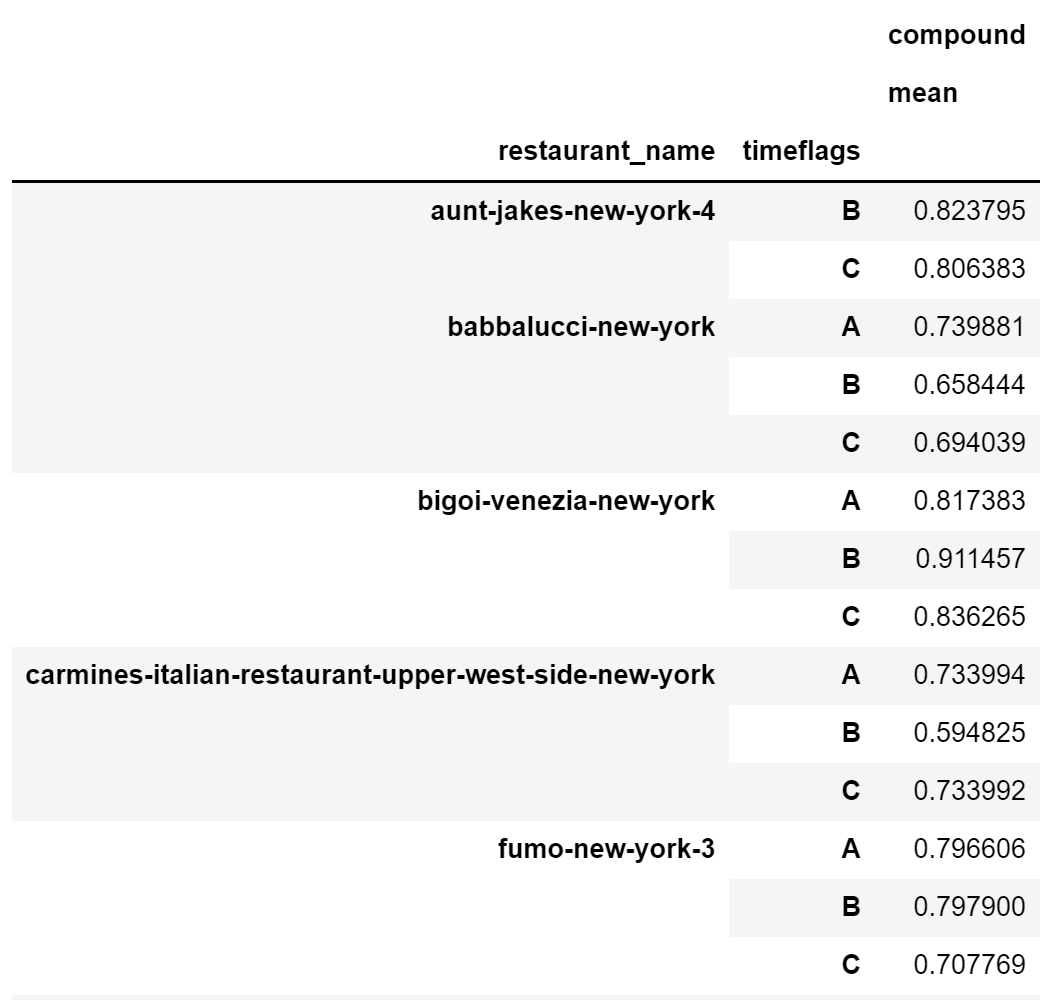
To compare the different time frames and their respective sentiment score, we looked at the compound value provided by the VADAR library. According to their documentation, this measure is the “most useful metric if you want a single unidimensional measure of sentiment for a given sentence” (cjhutto, 2014). Aggregating the mean compound score for restaurants in our data set yields a result that can be seen in figure 10. As can be observed, the mean compound scores do not seem to differ significantly from time frame to time frame. To analyze this behavior, a paired t-test was conducted and confirmed the suspicion: with p-values of 0.25 and 0.20 for tests of the transition from A to B and B to C respectively, we cannot conclude that there is enough evidence that the difference in sentiment in different time frames was significant. We additionally tested for a shorter time frame for period A, as the original time frame A has a different length for different restaurants and goes back to 2014 for some restaurants. As a comparison, we created a lower boundary of June 15, 2019 for time frame A. This date was chosen to even out the length of different time frames in the data set. Again, we cannot find evidence in support of our original claim that the sentiment of different time frames varies. The p-values here are 0.15 for the transition from A to B and 0.20 for the transition from B to C, respectively. Obviously, changing the length of time frame A only influences the p-value of the first t-test.

Figure 10: A selection of mean compound values per restaurant and time frames.

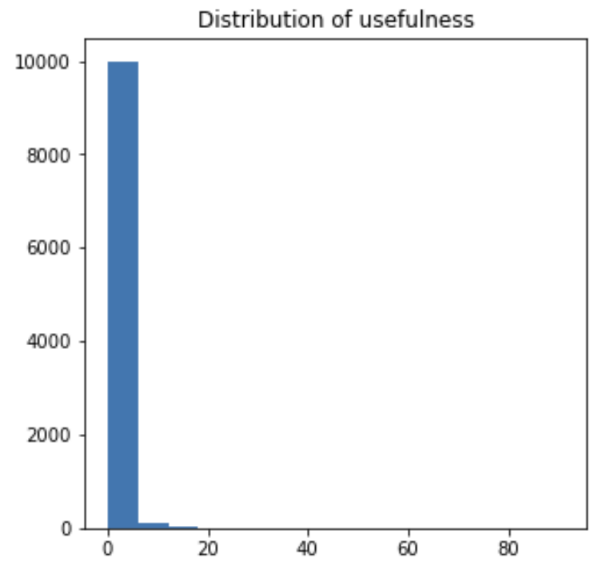
To check for the significance of review intensity when predicting usefulness of a review, we first identify what it means that a review is “useful”. Platform users of any platform can share content, comment on posts and also express their (positive) gratitude towards the content creator. On platforms like Facebook and Instagram, the latter is expressed through a click on the button “like” underneath a post. On Yelp, people have the choice of three buttons for each review: one for “cool”, one for “funny” and one for “useful”. This analysis aims at assessing the significance of intensity when predicting the number of times people clicked on “useful” for any given review. The intensity of a review is assessed by the absolute value of “compound” and taken as a new input variable in several predictive models. As seen in figure 11, the usefulness of a review in our data set is highly focused on the lower end of the number scale. To slightly spread out the target variable, we transformed usefulness into log-usefulness (Belinda, Z., Roseanne, F., Zahra, M., 2015) (figure 12).

Figure 11: Skewness of usefulness of reviews.

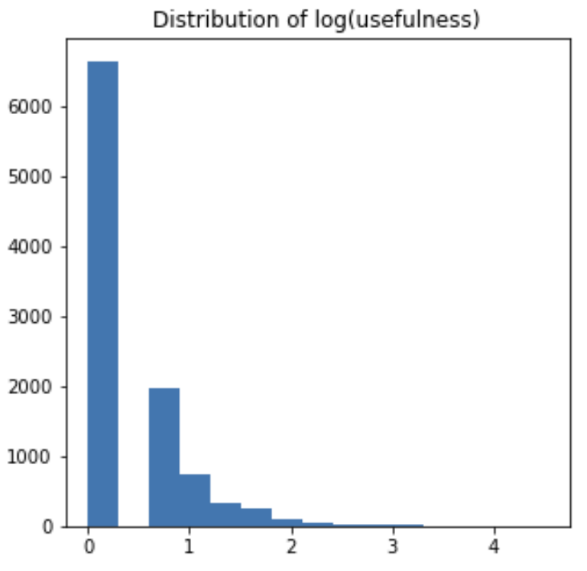
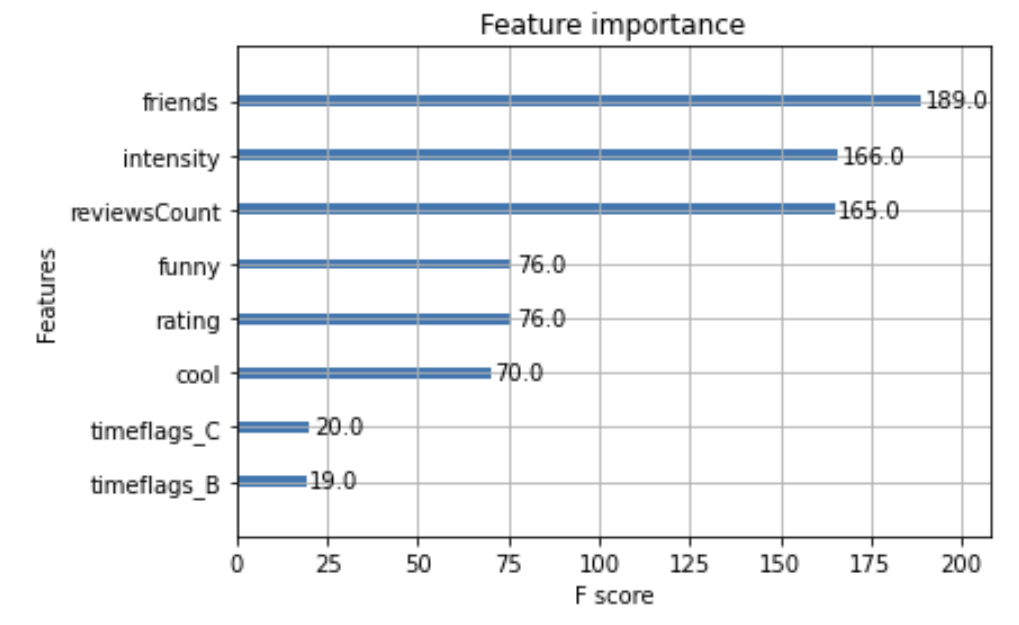
Using the variables in our dataset, we conducted three different analyses to assess significance and importance of intensity. First, using a mixed linear model regression analysis, we found that intensity is indeed a significant factor when predicting log-usefulness of reviews. The full results can be seen in figure 14. Intensity turns out to have an estimate of 0.275 and is significant at the 1% level. Second, using 10-fold CV LassoCV regularization, we found that intensity again is an important estimator, with a coefficient of 0.22. Finally, using XGBoost, we found that intensity belongs to the top 3 most important variables in our dataset. The ranking in figure 13 shows feature importance for the respective variables in our dataset. The displayed F-score is a count of the number of times a feature was split on across all trees in the boosting process (xgboost developers, 2021). Next to intensity, the number of friends on the platform and the number of reviews written by that reviewer are the other two variables in the top 3. This makes sense as friends get notified when someone is posting a new review and more active people are usually associated with more friends.

Figure 13: Variable importance according to the XGBoost algorithm. Intensity is in the top 3.

Figure 12: Taking the log of the target variable spreads out the variable space.

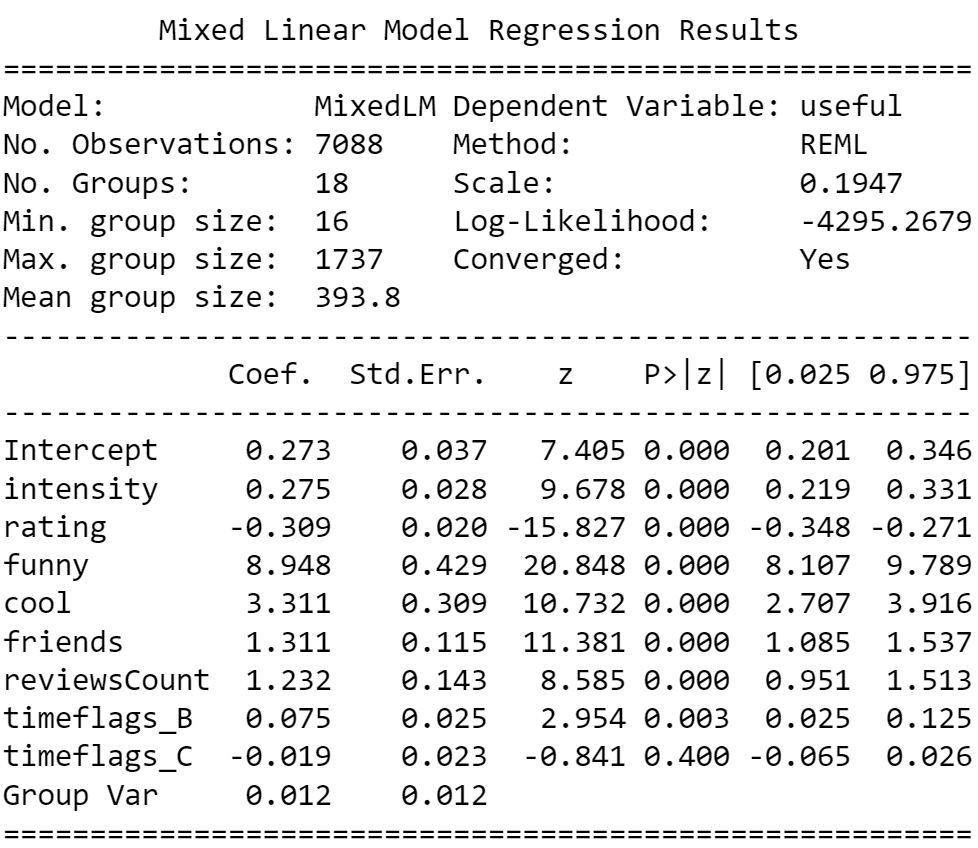


Figure 14: Results of mixedlm algorithm. Intensity is significant along with rating, funny, cool, friends, reviewsCount and timeflags\_B.

1. **DISCUSSION**
   1. **Why does our methodology work or not work?**

While our analysis of significance of intensity when predicting usefulness of reviews showed the expected behavior, the hypothesized results of different sentiment values through the covid-pandemic were not holding true. There can be several reasons for that. First, our initial hypothesis could be wrong and there was indeed no difference in sentiment over the different periods. In this case the average customer felt approximately the same about the service or quality of a restaurant over each period. Second, one can argue that the selection of our restaurants is a very homogeneous group of restaurants. This is resulting due to the sorting algorithm Yelp is using. When searching for a type of restaurant, Yelp outputs a list of restaurants you might be interested in. Naturally, they tend to have better reviews. As seen in subsection 5c, the average review score of all of our restaurants is above 4 stars. In future analysis, it would be interesting to incorporate restaurants from the full spectrum of review ratings into the model. Additionally, it would be interesting to see how the sentiment regarding different aspects of a restaurant has changed. This way, one could analyze how the sentiment regarding the food quality has changed in different time periods. This type of aspect-based sentiment analysis has shown to provide useful insight about several aspects of review sentiment analysis (Kamonkorn BuangSoong, 2020).

While our regression analysis of predicting usefulness of reviews tends to underestimate the actual number, we saw that intensity of reviews is in fact a significant predictor. This result is consistent with other research (Belinda, Z., Roseanne, F., Zahra, M., 2015) and across multiple approaches we have analyzed.

* 1. **Why are the findings meaningful?**

Customer satisfaction is one of the most important features in the service industry. Therefore, it is very important for restaurants to keep the sentiment of their customers high, independent of the situation we are facing. Our research shows that restaurants did not let their customers down in the pandemic and through the time when only delivery options were available. It also shows that customers had no problem transitioning from dine-in to delivery and then most importantly, back to dine-in options at their favorite restaurants.

In terms of review intensity, restaurants should value enthusiastic positive reviews more than just objective 5-star reviews. As seen in our analysis, a higher intensity score is associated with a higher (log-) usefulness of a review. The more useful a review is, the higher up it appears in the restaurant's review listings on Yelp. For restaurant owners, we recommend filtering out high intensity, negative reviews and respond to them publicly. As these reviews will be seen more prominently, the restaurant’s answer and potential justification for what went wrong is also more visible to potential customers.

* 1. **What are limitations and how to improve?**

For future research we suggest looking at a greater data set of restaurants with a higher variance in review sentiment and average rating. This could eliminate some of the limitations outlined in the previous subsection 6a regarding the analysis of difference in sentiment over several time periods. More reviews can be collected from different websites such as Google, Facebook and TripAdvisor. Additionally, we suggest investigating aspect-based sentiment analysis specifically for the aspects of food quality and delivery.

As for the intensity significance analysis, we would further look into time-dependent importance of this feature. Is intensity of a review something that is more important in certain time periods, for instance in a remote, dine-out period (B)?

1. **REFRRENCES**

[1] Adomavicius, G. ,Tuzhilin, A. (2005), "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions", IEEE Transactions on Knowledge and Data Engineering, Volume 17, Issue 6.

[2] Nolan Miller, Paul Resnick, Richard Zeckhauser, (2005), "Eliciting Informative Feedback: The Peer-Prediction Method" management science Vol. 51, No. 9, pp. 1359–1373

[3] Michael J. Pazzani and Daniel Billsus, (2007), "Content-Based Recommendation Systems", Lecture Notes in Computer Science Volume 4321, pp 325-341

[4] Gayatree Ganu, No´emie Elhadad, Am´elie Marian, (2009), "Beyond the Stars: Improving Rating Predictions using Review Text Content", Twelfth International Workshop on the Web and Databases, Providence, Rhode Island, USA

[5] B. Pang, L. Lee, S. Vaithyanathan, (2002), “Thumbs up? Sentiment classification using machine learning techniques”, Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 79–86.

[6] Ziqiong Zhang, Qiang Ye, Zili Zhang, Yijun Li (2011), “Sentiment classification of Internet restaurant reviews written in Cantonese”, Expert Systems with Applications: An International Journal, Volume 38 Issue 6, Pages 7674-7682.

[7] 2021 online reviews statistics and trends: A report by Reviewtrackers. ReviewTrackers. (2021, October 8). Retrieved October 29, 2021, from https://www.reviewtrackers.com/reports/online-reviews-survey/.

[8] Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. biometrics, 159-174.

[9] cjhutto (2014), retrieved 12/09/2021 from <https://github.com/cjhutto/vaderSentiment>

[10] xgboost developers (2021), retrieved 12/09/2021 from <https://xgboost.readthedocs.io/en/latest/python/python_api.html>

[11] Kamonkorn BuangSoong, 2020, retrieved 12/10/2021 from <https://www.kaggle.com/kamonkornbuangsoong/restaurant-reviews-aspect-based-sentiment-analysis>

[12] Belinda, Z., Roseanne, F., Zahra, M. (2015), Predicting Popularity, retrieved 12/10/2021 from <https://belindazeng.github.io/goingviral/>

[13] Gaurav Singhal, (2020), Importance of Text Pre-processing. Retrieved 12/10/2021 from https://www.pluralsight.com/guides/importance-of-text-pre-processing

1. **Appendix**

**Appendix [A]** **Pre-processing code (for one restaurant, redo for every restaurant)**

import pandas as pd

import numpy as np

import spacy  
nlp = spacy.load("en\_core\_web\_sm")

import nltk  
nltk.download('stopwords')  
from nltk.corpus import stopwords

[nltk\_data] Downloading package stopwords to  
[nltk\_data] C:\Users\didil\AppData\Roaming\nltk\_data...  
[nltk\_data] Package stopwords is already up-to-date!

from nltk.tokenize import word\_tokenize  
nltk.download('punkt')

[nltk\_data] Downloading package punkt to  
[nltk\_data] C:\Users\didil\AppData\Roaming\nltk\_data...  
[nltk\_data] Package punkt is already up-to-date!

from datetime import date  
d1 = date(2020, 3, 15)  
d2 = date(2021, 1, 31)

*#function for time flags*  
**def** label\_race (row):  
 **if** row['date'] < d1:  
 **return** 'A'  
 **if** d1<=row['date']<=d2:  
   
 **return** 'B'  
   
 **if** row['date'] > d2:  
 **return** 'C'

*#function for lemmatization*  
**def** lemmatize(text):  
 *"""Perform lemmatization and stopword removal in the clean text*  
 *Returns a list of lemmas*  
 *"""*  
 doc = nlp(text)  
 lemma\_list = [str(tok.lemma\_) **for** tok in doc]  
 **return** lemma\_list

df1= pd.read\_csv(r'C:\Users\didil\OneDrive\Bureau\BIA660\final project\done\aunt-jakes-new-york-4.csv', header =0)

df1['date'] = pd.to\_datetime(df1['date'])

df1['timeflags'] = df1.apply (**lambda** row: label\_race(row), axis=1)

*# replace text with only numbers for ratings*  
df1.replace('5 star rating', '5', inplace=True)  
df1.replace('4 star rating', '4', inplace = True)  
df1.replace('3 star rating', '3', inplace = True)  
df1.replace('2 star rating', '2', inplace=True)  
df1.replace('1 star rating', '1', inplace =True)

*# replace null value in columns useful, funny and cool with 0 (meaning nobody clicked on either of these buttons)*  
df1['useful'].fillna(0, inplace = True)  
df1['funny'].fillna(0, inplace = True)  
df1['cool'].fillna(0, inplace = True)  
df1['friends'].fillna(0, inplace=True)

df1.isnull().values.any()

False

*# defining reviewers as number instead of their names*  
df1.drop(columns = ['Unnamed: 0'] , inplace=True)  
df1['reviewer']=np.arange(df1.shape[0])  
df1['reviewer']= df1.index+1

*# get reviews in lower case*  
df1['review']=df1['review'].apply(**lambda** x: " ".join(x.lower() **for** x in x.split()))

*# remove punctuation*  
df1['review']= df1['review'].str.replace('[^\w\s]','')

*# remove stop words*  
  
stopwords = stopwords.words('english')  
df1['review'] = df1['review'].apply(**lambda** x: " ".join(x **for** x in x.split() **if** x not in stopwords))

*#tokenization*  
df1['token review'] = df1['review'].apply(**lambda** row: word\_tokenize(row))

*#lemmatization*  
df1['review\_lemmatized'] = df1['review'].apply(lemmatize)

**Appendix [B] Scraping Code**

import requests  
from bs4 import BeautifulSoup   
import pandas as pd  
import time  
from selenium import webdriver  
from selenium.webdriver.common.by import By  
from selenium.webdriver.support.ui import WebDriverWait  
from selenium.webdriver.support import expected\_conditions

**def** getReviews(page\_url):  
  
 reviews=[]  
 end = False  
   
 names=[]  
 ratings=[]  
 dates=[]  
 review\_contents=[]  
 useful=[]  
 funny=[]  
 cool=[]  
 friends=[]  
 reviewsCount=[]  
 driver = webdriver.Chrome()  
 driver.get(page\_url)   
 time.sleep(5)  
 driver.execute\_script("window.scrollTo(0, document.body.scrollHeight);")   
 time.sleep(2)  
   
 **while** not end:  
 html = driver.page\_source  
 soup = BeautifulSoup(html.encode('utf-8'),"html.parser")  
 reviews = soup.find\_all("div", class\_="review\_\_373c0\_\_3MsBX border-color--default\_\_373c0\_\_1WKlL")  
 **for** review in reviews:  
 **try**:  
 *## names*  
 name=review.find("a", class\_='css-166la90').text  
 names.append(name)  
 **except**:  
 names.append(None)  
  
 **try**:  
 *## ratings*  
 rating=review.find("div", class\_='arrange\_\_373c0\_\_2S3dc gutter-1\_\_373c0\_\_3zF2l vertical-align-middle\_\_373c0\_\_7yhdw border-color--default\_\_373c0\_\_1WKlL').find("span", class\_="display--inline\_\_373c0\_\_1gaV4 border-color--default\_\_373c0\_\_1yxBb").find("div")["aria-label"]  
 ratings.append(rating)  
 **except**:  
 ratings.append(None)  
  
 **try**:  
 *## dates*  
 date=review.find("div", class\_='arrange\_\_373c0\_\_2S3dc gutter-1\_\_373c0\_\_3zF2l vertical-align-middle\_\_373c0\_\_7yhdw border-color--default\_\_373c0\_\_1WKlL').find("span", class\_="css-e81eai").text  
 dates.append(date)  
 **except**:  
 dates.append(None)  
  
 **try**:  
 *## friends and number of reviews of reviewer*  
 divs = review.find("div", class\_='user-passport-stats\_\_373c0\_\_2nBtY border-color--default\_\_373c0\_\_r305k').find\_all("div")  
 **for** div in divs:  
 **if** div["aria-label"] == "Friends":  
 **try**:  
 friends.append(div.find("span", class\_="css-1dgkz3l").text)  
 **except**:  
 friends.append(None)  
 **elif** div["aria-label"] == "Reviews":  
 **try**:  
 reviewsCount.append(div.find("span", class\_="css-1dgkz3l").text)  
 **except**:  
 reviewsCount.append(None)  
 **except**:  
 friends.append(None)  
 reviewCount.append(None)  
 **try**:  
 *##review contents*  
 review\_content=review.find("span", class\_='raw\_\_373c0\_\_tQAx6').text  
 review\_contents.append(review\_content)  
 **except**:  
 review\_contents.append(None)  
  
 **try**:  
 *## useful, funny, cool*  
 mixed=review.find\_all("div", class\_='arrange\_\_373c0\_\_2S3dc vertical-align-middle\_\_373c0\_\_7yhdw border-color--default\_\_373c0\_\_1WKlL')[-1].find\_all("span", class\_="css-1ha1j8d")  
 **try**:  
 usef = mixed[0].find("span", class\_="css-3fag8g").text  
 useful.append(usef)  
 **except**:  
 useful.append(None)  
 **try**:  
 fun = mixed[2].find("span", class\_="css-3fag8g").text  
 funny.append(fun)  
 **except**:  
 funny.append(None)  
 **try**:  
 coo = mixed[1].find("span", class\_="css-3fag8g").text  
 cool.append(coo)  
 **except**:  
 cool.append(None)   
 **except**:  
 useful.append(None)  
 funny.append(None)  
 cool.append(None)  
   
 **try**:  
 driver.find\_element\_by\_css\_selector("a.next-link").click()  
   
 time.sleep(5)  
 **except**:  
 end = True  
   
  
 reviews\_df = pd.DataFrame()  
 reviews\_df["reviewer"]=names  
 reviews\_df["rating"]=ratings  
 reviews\_df["date"]=dates  
 reviews\_df["review"]=review\_contents  
 reviews\_df["useful"]=useful  
 reviews\_df["funny"]=funny  
 reviews\_df["cool"]=cool  
 reviews\_df["friends"]=friends  
 reviews\_df["reviewsCount"]=reviewsCount  
 driver.quit()  
   
 **return** reviews\_df

driver = webdriver.Chrome()  
search\_url = "https://www.yelp.com/search?find\_desc=Italian%20Food&find\_near=new-york-city-new-york-14&ns=1&start=10"  
driver.get(search\_url)   
time.sleep(2)  
html = driver.page\_source  
soup = BeautifulSoup(html.encode('utf-8'),"html.parser")  
  
links = soup.find\_all("a", class\_="css-1f2a2s6")  
restaurant\_links = []  
**for** link in links:  
 **if** link["href"].startswith("/biz"):  
 restaurant\_links.append(link["href"])  
   
driver.close()  
   
init\_url = "https://www.yelp.com"  
appendToURL = "?sort\_by=date\_desc"  
  
reviews=[]  
**for** restaurant in restaurant\_links:  
 reviews.append(getReviews(init\_url+restaurant+appendToURL))

**for** i in range(len(restaurant\_links)):  
 midPart = restaurant\_links[i].replace("/biz/", "").split("?")[0]  
 reviews[i].to\_excel('Midterm\\'+ midPart +'.xlsx')