

ANALYZING AND FILTERING FOOD ITEMS IN RESTAURANT REVIEWS: SENTIMENT ANALYSIS AND WEB SCRAPING

Nina Luo¹, Caroline Kwan², Yu Sun³, Fangyan Zhang⁴

¹Webb School of California, Claremont, CA 91711, USA

²Westridge School, Pasadena, CA, 91105, USA

³California State Polytechnic University, Pomona, CA, 91768

⁴ASML, San Jose, CA, 95131, USA

ABSTRACT

Online reviews now influence many purchasing decisions. However, the length and significance of these reviews vary, especially when reviewers have different criteria for making their assessments. In this paper, we present an efficient method for analyzing restaurant reviews on the popular review site known as Yelp. We have created an application that uses web scraping, natural language processing, and a blacklist to recommend customer favorite dishes from restaurants. To test the app, we have conducted a qualitative evaluation of the approach. Through analyzing two different ways to obtain Yelp reviews and evaluating our word filtering process, we have concluded that an average of 51% of nonfood words are filtered out by the blacklist we made. We provide further details of its deployment, user interface design, and comparison to the opinion mining field, which utilizes similar tools to make financial market predictions based on the perceived public opinion on social media.

KEYWORDS

Web scraping, natural language processing, flutter, iOS, android

1. INTRODUCTION

Online user reviews provide feedback about a person's experience with a business [1]. These reviews are often written by customers and generally include an overall score based on a customer's experience and expectations with a service or product. As individual user reviews often provide assessments based on different criteria, it is necessary to piece together a wide range of customer preferences and impressions on different aspects of a service or product to get a complete and accurate picture. This way, we are more well informed and aware of the contrasting opinions regarding the business.

The internet not only provides easy access to large amounts of information but also serves as a global forum and publishing platform [2]. With the rise of fast-paced modern technologies and social networks, immediate satisfaction is granted to users because information can quickly be found and read. As a result, more customers are using the internet to evaluate and comment on services and products [3]. Online website reviews give customers the opportunity to share their experiences and express their observations. Through the first-hand perspective of a customer, user reviews detail the positives and negatives of an experience. Positive reviews are shown to increase revenue, improve a company's reputation, and boost customer loyalty. While positive reviews strengthen the reliability and image of a service or product, negative reviews allow for

improvement and strengthen credibility. When a business replies to a negative review, the business acknowledges the complaint and shows that it values customer feedback. This interaction provides an opportunity for a company to demonstrate its principles and reduces the impact of negative reviews.

Online user reviews are important as they can influence purchasing decisions and determine the reputation of a business [15]. Furthermore, these reviews encourage business and customer relationships by allowing businesses to respond to customers and establish trust through customer engagement. While reviews provide insight about a service or product, some reviews can be extensive or resemble a rant. Although websites have implemented systems to filter out unhelpful reviews, not many reviews are removed and most reviews are published onto the internet for users to read. However, small business websites tend to lack users in the first place and therefore have a low amount of reviews. A limited amount of reviews can mislead users as the few reviews dominate the narrative of a service or product.

Luckily, there are websites dedicated to reviewing restaurants and businesses [14]. A popular review site known as Yelp consolidates business information and allows customers to easily write and read reviews. Large and small businesses alike are given a platform to gain recognition and listen to customer feedback. An automated software finds effective reviews and recommends highly rated restaurants and businesses. As Yelp's automated software regularly tweaks itself and gathers more reviews, the recommended reviews change over time. In order to simplify and ease the task of reading tons of reviews, Yelp implements features known as "Review Highlights" and "Popular Dishes". The "Review Highlights" section points out popular words written in reviews and is a quick way for users to discover the main components of a restaurant or business. While the highlighted words usually refer to a location, item, or name, some words can be irrelevant or unhelpful to users. Another feature is the "Popular Dishes" section. As the name suggests, this section is a gathering of reviews and photos of popular foods. However, this feature does not show up on all restaurant pages, and the dishes are automatically selected by Yelp's machine learning algorithm. Although Yelp uses algorithms to find popular words and dishes in reviews, its sorting process is not always relevant, and the features do not apply to all businesses. To improve their algorithms, Yelp has recently partnered with GrubHub to provide full restaurant menus.

Since a majority of users seek restaurant information from Yelp, this project focuses on facilitating the process of sorting information in restaurant reviews. The approach presented in this paper focuses on recommending highly rated food dishes based on Yelp user reviews aided by web scraping and sentiment analysis. We have developed Dishcovery, a mobile application that allows users to discover the best tasting, most popular foods in nearby restaurants.

Dishcovery uses machine learning to compile Yelp reviews and construct a sentiment analysis rating of food items in restaurants. Unlike Yelp which only provides an overall 5-star rating system for reviews, Dishcovery ranks individual food dishes in restaurants on a precise decimal scale from negative one to one. Using Yelp as a resource to provide restaurant reviews, Dishcovery searches for food items and ranks the dishes based on the positive and negative comments made in reviews. The mobile application is supported by a back end that collects data from Yelp, sorts out food dishes, and creates a numerical rating for each item. The strength of Dishcovery is that through the method of web scraping, we are able to apply our application to find the recommended items of any customer review-based website. This includes already popular websites such as Angie's List, Trip Advisor, Google My Business and much more. Our program can easily expand beyond recommending restaurants on Yelp. Without drastically changing the code, our application can compile a list of favorite aspects from a service, trip, or company.

We have experimented with two different methods to extract user reviews from Yelp. First, we tested the Yelp Fusion API that provides free access to over 50 million businesses [11]. We initially thought this would be a good way to request the Yelp data. However, the API is restrictive and only provides a limited amount of information. For example, API requests are capped at 5000 calls per day and only three partial reviews of 160 characters total are provided for each call. We contacted Yelp to increase our daily calls and get access to more Yelp reviews, but they rejected our request for academic use. These restrictions prevented us from getting enough information to run the machine learning algorithms necessary for finding food items. We decided to try another method to get more data. Second, we used the Python requests library combined with BeautifulSoup to extract all the reviews. Although the Python web scraping gave us a plentiful amount of information from the Yelp URL pages, it was difficult to locate all the reviews because they were in separate places in the HTML code. By using the Python library BeautifulSoup to parse the HTML code, it simplified this process and grouped all the reviews together in a list. The list created for each business has more information than the Yelp Fusion API and allows for better filtering of the food items. BeautifulSoup provides the best results because the Yelp reviews can be quickly found by the HTML tag names and organized together. These experiments have helped us find the best way to obtain the Yelp reviews.

The rest of the paper is organized as follows: Section 2 details the four main challenges and limitations we encountered during the making of our program. Section 3 identifies the solutions and steps we took to combat the challenges mentioned in Section 2. Section 4 showcases the two evaluations we conducted and an analysis of the results; Section 5 cites the relevant related works from other authors for further reading as well as comparisons to our own work. At the end in Section 6, we conclude the paper, restate the main ideas, and explain the future work of our project.

2. CHALLENGES

There are a few of challenges existing in the project. They will be discussed one by one in this section.

2.1. Challenge 1: Limited information from Yelp Fusion API

A big part of Yelp's success stems from obtaining a large amount of user data and business information. Since Dishcovery recommends food dishes from restaurant reviews, it also depends on abundant data sets. We initially used the restaurant information and user reviews provided by Yelp Fusion API through Postman. Although easy to access, the Yelp API limits the amount of data developers can view and only provides developers three incomplete reviews for each business. The limited reviews do not present a complete picture of a business and provide inadequate information necessary for running the natural language processing and sentiment analysis machine learning algorithms. As a result, few food items were identified and the algorithm inaccurately analyzed the overall sentiment of the reviews.

For example, the restaurant review from Yelp Fusion API provides the text "This is one of my favorite restaurants. The food here is so different in a good way. They serve good quality food. My favorite things to get from here are..." The review is cut off and the food recommendations are not shown. This creates a problem for the machine learning algorithms because there are not enough words to analyze and no food items to extract. We addressed this by using the method of web scraping to get the full user reviews from Yelp.

```

{id": "hPuMj09TzX5_Amidkk6e1A",
"url": "https://www.yelp.com/biz/din-tai-fung-arcadia-3?adjust_creative=J4UsXVUs5Myu_cxdioCuq&
hrid=hPuMj09TzX5_Amidkk6e1A&utm_campaign=yelp_api_v3&utm_medium=api_v3_business_reviews&
utm_source=J4UsXVUs5Myu_cxdioCuq",
"text": "This is one of my favorite restaurants. The food here is so different in a good way. They serve good quality
food. My favorite things to get from here are...",
"rating": 5,
"time_created": "2020-05-15 00:01:40",
"user": {
  "id": "Q6s1EKKM1lrE9wStPD7z8w",
  "profile_url": "https://www.yelp.com/user_details?userid=Q6s1EKKM1lrE9wStPD7z8w",
  "image_url": "https://s3-media4.fl.yelpcdn.com/photo/csJ8IfT1Z4LCXWUAE_4_Fg/o.jpg",
  "name": "Sarina T."
}

```

Figure 1: A limited review from Yelp Fusion API that has its food recommendations cut off.

2.2. Challenge 2: Difficulty identifying food items

Reviews address many aspects of a business or restaurant and contain a lot of information. Therefore, it can be difficult to filter out the keywords, like food items and beverages. Additionally, slang and typos often confuse the sorting algorithm. For example, words such as ‘table’ and ‘smile’ are sometimes classified as potential dishes. Although Google Natural Language API has a powerful model that identifies parts of speech, it has trouble with sorting out nouns and often groups them into the other category. The other category requires more implementation of filtering algorithms as it stores the words the natural language processing could not identify. For our purposes, the list needs to be shortened and only contain words relevant to items on a restaurant menu. We fixed this problem by creating a blacklist of common words mistaken by the API to be a dish. The unrelated words in this list will be filtered out, leaving the food dishes.

2.3. Challenge 3: Analyzing positive and negative comments

Most restaurant and business reviews have an overall positive or negative tone. Although Yelp’s 5-star rating system does give us an idea of the reviewer’s opinion, it can occasionally be an unreliable indicator of the feelings expressed in a review. For example, every reviewer has different criteria for rating a restaurant. One customer may express that the food was great but give a three-star review while another customer may express that the food was okay and give the restaurant a 5-star review. Looking at potentially inaccurate star ratings could influence the way users initially view the restaurant. Thus, we decided to focus on the general sentiment of the review instead of using Yelp’s rating system to give us a better understanding of the reviewer’s intended tone. Using Google Natural Language API, we evaluated the sentiment analysis of each review and ranked each dish on their positive or negative attribution from a precise scale of negative one to one.

2.4. Challenge 4: Promotion of app

As a food recommendation program, we wanted Dishcovery to be accessible to the most people and decided to make it an app. We started with building the application in Flutter to make it available on both the Google Play Store and App Store [12]. Then, we improved the back-end data collection to get more information and user reviews. The app has three pages: the main page, the search page, and the about page. The main page and search page suggest nearby restaurants and show pictures of food items. The about page gives a description of the app itself and information about the creators of the app. To promote our app, we decided to create a website. However, we were unsure of how to make it available online and how to build a presentable website design. We solved this problem by finding a premade theme on Envato Market and modifying its HTML code using Sublime Text to fit the Dishcovery app theme [13]. This allowed us to share Dishcovery with more online users.

3. SOLUTION

We have created Dishcovery, an app that recommends popular restaurant dishes from Yelp reviews. When a user selects their desired restaurant or business, information such as pictures and user reviews are collected from the Yelp website. The reviews are then filtered by machine learning algorithms and a blacklist to find the popular dishes from positive Yelp reviews [9]. A JSON file returns the highly recommended food items back to the app for the user to explore.

Main Page: The main page of Dishcovery uses location to recommend the top five most nearby restaurants. Each restaurant includes the following details: restaurant name, distance to restaurant from current location, average Yelp star rating, average cost of the food items on the menu, and a picture of one of the dishes. The information inside each restaurant page includes the top five most popular and highly recommended food dishes that are filtered by machine learning algorithms addressed later in the paper.

The restaurant and business data come from the Yelp website. Yelp reviews are extracted from the actual restaurant Yelp page using the method of web scraping aided by the Python package Beautiful Soup [10]. Google Natural Language API and a blacklist sort out the food and drink items and rank the sentiment of the items on a scale from negative one to one. A higher score means that the item comes from a more positive review.

Search Page: The search page suggests hot searches that are frequently searched by Dishcovery users. A small picture and the price of the suggested food items are shown. At the top of the page, the search bar can be used to find restaurants.

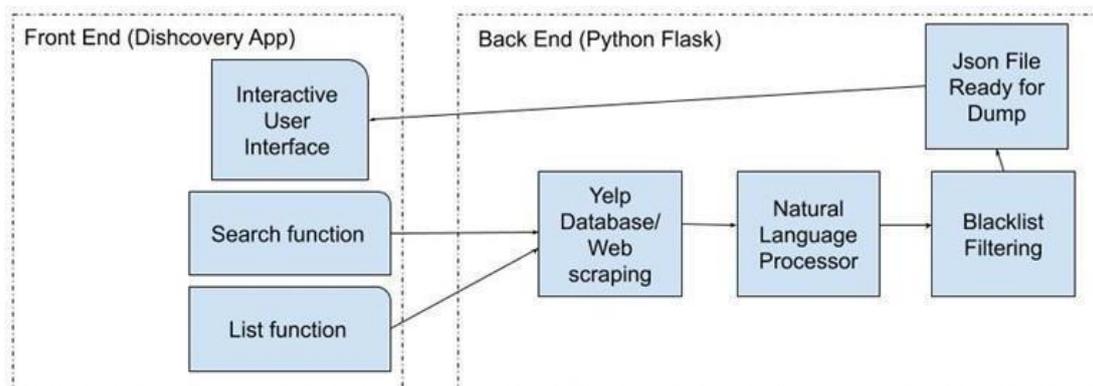


Figure 2: Overview of Dishcovery

Mobile Application: The front end of the mobile application is developed in Dart, a fast and object-oriented language that is used in Flutter apps. Created by Google, Flutter is a UI software development package that can be used to make iOS and Android apps. The interface and design of the three pages on our app are built in Flutter. The back end consists of using Beautiful Soup, Google Natural Language API, and the blacklist we made.

In Beautiful Soup, the program parses through HTML code to find information through tags and HTTP headers. As shown later in experiment 1, this web scraping process provides more user reviews than using Yelp Fusion API. After the reviews are gathered, we use natural language processing to classify food items from the Yelp reviews. We chose the syntax analysis, entity analysis, and sentiment analysis from Google Natural Language API because of its powerful deep learning models and variety of features. Its ease of access and inexpensive price are also added bonuses. The syntax analysis identifies parts of speech and sorts them into different categories.

The entity analysis labels words by type, such as person or location. This is helpful for finding the nouns and food items in the other category. The sentiment analysis creates a score of the feeling expressed in the reviews and determines whether a review has a positive or negative tone. Since the purpose of our app is to recommend popular dishes, we programmed the natural language processing to only include positive reviews that have a positive sentiment score.

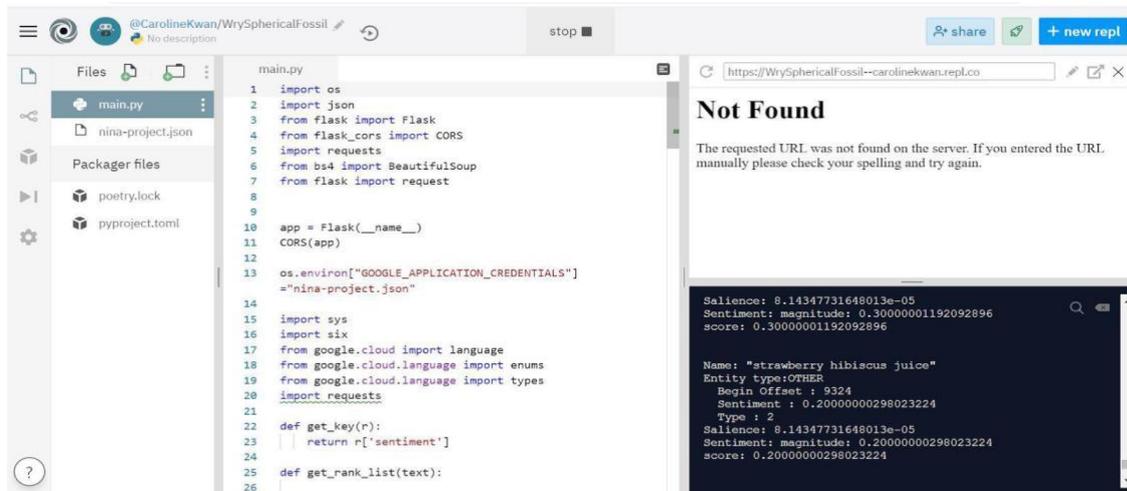


Figure 3: An analyzed food item returned from Google Natural Language API.

Although the entity analysis categorizes words into different types, many words that cannot be identified by the machine learning algorithms are sorted into the other category. Since the food items are in this category, we had to make additional parameters to sort them out. We created a dictionary of words called the blacklist that stores words unrelated to food dishes. This way, the nonfood words are sorted out through the blacklist and remaining words in the other category are limited to the food items. However, this filtering method requires a lot of time as we have to manually write out all the words in the list. After being refined by the blacklist, the condensed list of food words is then returned in a JSON file to the app and shown to the user.

We made a Dishcovery website (dishcovery.app) to promote the services of our app and allow users to easily contact us. The design of the website is created with a premade Envato Market theme. After transferring the files onto GitHub, we hosted our website on GitHub Pages, customized the layout to fit our needs, and bought a domain name. On the main page of the website, we have conveniently included links for people to download the iOS and Android versions of our app.

4. EXPERIMENT

4.1. Experiment 1: testing Data Retrieval Effectiveness

We have analyzed 50 Yelp restaurant reviews from 5 different cities to evaluate the advantages of web scraping with BeautifulSoup over Yelp Fusion API. To count the number of food items, we ran the reviews from BeautifulSoup and Yelp Fusion API through the same machine learning algorithms. As previously mentioned, Yelp Fusion API only provides an average of 160 characters for the reviews in each restaurant. This is not enough for the machine learning algorithms to find food items or accurately analyze the sentiment of the reviews. The solution is web scraping with BeautifulSoup because extracting the full reviews directly from the Yelp restaurant website page provides the most updated and accurate results. Our results show that BeautifulSoup consistently recommends more food items over Yelp Fusion API.

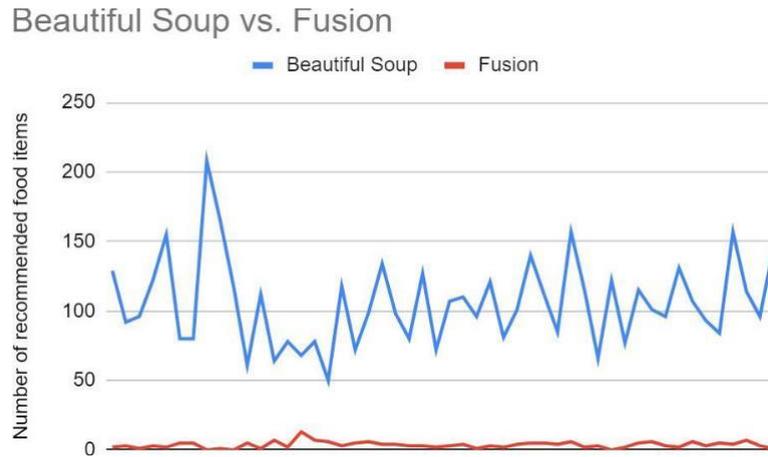


Figure 4: test results of Beautiful soup and Fusion

As shown in Figure 4, there is a significant difference between the number of food items returned by Beautiful Soup and Yelp Fusion API. Since Yelp API provides such limited data, the number of food items in the reviews is consequently very low. This can be seen by the red line (Yelp Fusion API) constantly returning numbers close to 0. The results support our conclusion that another solution is required in which more reviews are provided. We used Beautiful Soup to maximize the amount of reviews we could obtain from Yelp. By using the data shown in a restaurant's HTML page on Yelp, we were allowed to access a lot more reviews. In comparison to Yelp Fusion API, the blue line (Beautiful Soup) constantly outputs an average of 105 food items. This disparity shows that the web scraping method proposed fixes the limited amount of data Yelp Fusion API provides.

Since the main goal of our app is to recommend customer favorite food dishes to users, it is important that the recommendations are accurate. To ensure that unrelated words are not mixed in with the recommendations, we have manually created a blacklist to filter out those unrelated words. In our second experiment, we tested the effectiveness of the blacklist by collecting samples from 50 different restaurants in 5 different locations to ensure the reliability of the final food recommendations. The experiment data can also serve as a future reference for retrieval improvement to the blacklist. The outlier, being the difference of 77.6 in 'City 2', shows that we could filter out more unrelated words.

4.2. Experiment 2: Reliability of Blacklist and Retrieval Improvement

Experiment 2 focuses on measuring the effectiveness and reliability of the added blacklist. Food items in Yelp reviews from 50 restaurants in 5 different cities are collected and averaged. The numbers in the 'Without Blacklist' column signify the mean amount of dish words the algorithm finds from the restaurant reviews in each city, including all unrelated words that the algorithm picked up on as well. The numbers in the 'Blacklist' column include the mean amount of dish words in each city NOT including the unrelated words that have been filtered by the blacklist. Through the results of experiment 2, we have found that the blacklist is very effective in filtering out words that are not food dishes. In some cases, the blacklist even blocks out more than half of the original words in the retrieval sample. This can be seen in the 'Difference' column, which shows how many unrelated words the blacklist has filtered out.

	Without Blacklist	Blacklist	Difference
City 1	236.5	115.4	121.1
City 2	151.4	73.8	77.6
City 3	199.8	97.8	102
City 4	219	103.7	115.3
City 5	219.1	105.8	113.3

The results of both experiments prove that our proposed solutions are valid and beneficial to the application. In experiment 1, we tested the effectiveness of using Beautiful Soup and Yelp Fusion API to output the food words found in the Yelp reviews. We concluded that the web scraping with Beautiful Soup always produced more food words than Yelp Fusion API. The low word count from Yelp Fusion API is attributed to the restrictions from Yelp of only providing three review excerpts for each restaurant. By using Beautiful Soup instead, we are able to gather more reviews and return additional food items. Another experiment analyzed the reliability of the blacklist we created to filter out non-food words. In the results from experiment 2, the large difference between using the blacklist and not using the blacklist was expected because the blacklist is a very long list that encompasses most of the common unrelated food words. The data from this experiment confirmed that the blacklist did improve the relevancy of the food items by a large margin and the reliability of the app overall.

5. RELATED WORK

There are many methods of applying sentiment analysis in online social platforms to observe public opinion and predict potential sways in financial markets [4] [5] [6]. Although both the opinion mining project and Dishcovery use sentiment analysis for prediction, Dishcovery web scrapes from Yelp while the opinion mining project scrapes from Twitter. These two platforms, though formatted similarly in short paragraphs of user input, are functionally distinct and have different algorithms of correlation and accuracy contributing to the final prediction. Both programs are bound to have inaccuracies because they scrape solely from one platform. Public sentiment on Twitter is likely to be different from public sentiment on other social media platforms, so stock market changes cannot be determined from Twitter alone. Similarly, there are other restaurant review websites aside from Yelp that may have different review sentiment.

This paper highlights the use of web-scraping software in searching for grey literature, referring to any research that is either unpublished or non-commercially published [7]. Since searching for grey literature online is often “time-consuming” and “challenging”, web scraping can be very useful in “extracting data from multiple pages of search results.” Unlike Dishcovery, the grey literature search is not limited to any particular website and does not use sentiment analysis to recommend users particular literature. Instead, it finds patterns in the documents to extract. Dishcovery is confined to scraping on Yelp and analyzes the sentiment of reviews. Although the main purpose of these projects is dissimilar, they both use web scraping to gather the bulk of their data.

Although similar to the first related work, this paper discusses the importance and potential for polarity detection and gives more of a general overview of opinion mining. Like Dishcovery, opinion mining would likely utilize both web scraping and sentiment analysis to make predictions [8]. Though opinion mining can be extremely helpful in the financial sector, its predictions need to be very accurate and political and social preferences cannot be determined from the internet alone. The same hurdle applies to Dishcovery: although Yelp plays a big role

in public sentiment for the restaurant experience, good dishes cannot be predicted from Yelp alone. Word of mouth is still very much prevalent, which simply cannot be accounted for in opinion mining or Yelp recommendations.

6. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a simple and efficient way for people to discover food dishes from different restaurants. Our mobile app encourages users to visit local restaurants and go out of their comfort zones to try food items from different cuisines. Aided by data from Yelp, this application gathers reviews, analyzes the overall tone of the reviews, and filters out the popular dishes.

Some of the challenges we faced when making our app include having access to too few Yelp reviews and having difficulty identifying food items. To address these problems and further elaborate on the methods we used to solve them, we performed two evaluations. Our first evaluation of the different approaches to obtain and analyze user reviews shows that Beautiful Soup consistently returns more food items than Yelp Fusion API. To illustrate, the average amount of food words Beautiful Soup analyzes is 105.7 while the average amount Yelp Fusion API analyzes is 3.6. The increased amount of data provided by Beautiful Soup allows for more accurate sentiment analysis of user reviews. In our other evaluation, we found that the blacklist does a good job of removing nonfood words and improves the accuracy of the output. Our results show that the filtering process filters out an average of 51% of the unrelated words from Google Natural Language API. Both of the results in our evaluations, as provided in Figure 5 and Table 1, prove that using web crawling with Beautiful Soup and the blacklist we made are useful solutions for the challenges we initially encountered.

For future work, we plan to improve our food sorting method and blacklist. We noticed that the word sorting groups in Google Natural Language API are too broad for the purposes of our app, so other natural language processing services can potentially be used to improve the efficiency of sorting. Although the current approach is applying a blacklist for narrowing down the food words from Google Natural Language API, this method is inefficient and requires us to look through the result output. Furthermore, it is impossible for the blacklist to filter out all the nonfood items from the Yelp reviews.

In the future, Dishcovery could be capable of identifying bots and fake negative reviews by competing restaurants. Also, we are considering creating a machine learning algorithm that learns to identify food items and works alongside the natural language processing service. This will improve the accuracy of the list of food words for each restaurant and possibly eliminate the blacklist. Thus, a more effective and adaptable solution could be implemented.

REFERENCES

- [1] Duan, Wenjing, Bin Gu, and Andrew B. Whinston. "Do online reviews matter?—An empirical investigation of panel data." *Decision support systems* 45.4 (2008): 1007-1016.
- [2] Chaves, Elisabeth. "The Internet as global platform? Grounding the magically levitating public sphere." *New political science* 32.1 (2010): 23-41.
- [3] Rose, Susan, et al. "Online customer experience in e-retailing: an empirical model of antecedents and outcomes." *Journal of retailing* 88.2 (2012): 308-322.
- [4] Feldman, Ronen. "Techniques and applications for sentiment analysis." *Communications of the ACM* 56.4 (2013): 82-89.
- [5] Pak, Alexander, and Patrick Paroubek. "Twitter as a corpus for sentiment analysis and opinion mining." *LREc*. Vol. 10. No. 2010. 2010.

- [6] Liu, Bing. "Sentiment analysis and opinion mining." *Synthesis lectures on human language technologies* 5.1 (2012): 1-167.
- [7] Haddaway, Neal R. "The use of web-scraping software in searching for grey literature." *Grey J* 11.3 (2015): 186-90.
- [8] Suganya, E., and S. Vijayarani. "Sentiment Analysis for Scraping of Product Reviews from Multiple Web Pages Using Machine Learning Algorithms." *International Conference on Intelligent Systems Design and Applications*. Springer, Cham, 2018.
- [9] Kumar, Naveen, et al. "Detecting review manipulation on online platforms with hierarchical supervised learning." *Journal of Management Information Systems* 35.1 (2018): 350-380.
- [10] Nair, Vineeth G. *Getting Started with Beautiful Soup*. Packt Publishing Ltd, 2014.
- [11] Zunnurhain, Kazi, and Arian J. Gonzalez. "Enhancement of User Experience with Mobile App." *Proceedings of the 47th International Conference on Parallel Processing Companion*. 2018.
- [12] Kuzmin, Nikita, Konstantin Ignatiev, and Denis Grafov. "Experience of Developing a Mobile Application Using Flutter." *Information Science and Applications*. Springer, Singapore, 2020. 571-575.
- [13] Habib, Md. "Web and Android Application Developer" in *Popular IT Limited*. (2018).
- [14] Mellet, Kevin, et al. "A "democratization" of markets? Online consumer reviews in the restaurant industry." *Valuation Studies* 2.1 (2014): 5-41.
- [15] Hlee, Sunyoung, et al. "An empirical examination of online restaurant reviews (Yelp. com): Moderating roles of restaurant type and self-image disclosure." *Information and communication technologies in tourism 2016*. Springer, Cham, 2016. 339-353.