Restaurants in the Madison Area

Lokananda Dhage, Mary Feng, Varun Naik

CS 838 Project Stage 1

1. Questions to Answer

Below are some potential questions that we may answer through our project.

1. Sentiment analysis of the reviews
    - Whether a particular review is positive or negative?
2. Keyword extraction and analysis
    - What are some of the top significant keywords across all the reviews? Does high frequency of a particular keyword give any insights about the kind of restaurants or users? How do these keywords in reviews help us in analysing, understanding the sentiments, favourite restaurants and answering questions such as *which restaurants serve healthy food?*
3. Related restaurants
    - Can two or more restaurants be grouped based on factors such as similarity in menu, quality of service, positive or negative reviews, ratings, and/or other features?
4. Users
    - Do users with more friends write more reviews or rate businesses more favorably on average? More generally, how does amount of interaction (useful/funny/cool votes, number of fans, elite status, number of compliments) affect reviews or tips the user writes?
5. Check-ins
    - What restaurants are more popular for breakfast, lunch, dinner, or late night food?
6. Comparing Yelp & Zomato users/reviews
    - Do Yelp users typically produce higher-quality reviews or vice versa? What factors impact Yelp reviews more than Zomato reviews?
7. Popularity
    - What factors are the strongest predictors for restaurants to be highly rated or receive many reviews?
8. Recommending restaurants
    - Given a time of day and approximate location, what popular nearby restaurants could be recommended?

2. Data Sources

Source 1: Yelp

Yelp was an obvious choice for obtaining information and reviews about restaurants in Madison. While we considered scraping Yelp for reviews at first, we discovered that scraping the site is against Yelp's terms of service (https://www.yelp.com/static?p=tos). Although scraping may still be viable if done carefully, we did not want to be blocked or face legal repercussions. Using the Yelp API was another option that was considered. However, there were various limitations—for instance, only part of one review (around 150 characters) could be retrieved for each business. Given these restrictions, we decided to use the data from the Yelp dataset challenge instead. As part of a challenge, Yelp provides a large dataset for students to use in conducting research. The current round, round 9, of the Yelp dataset challenge contains data about various businesses and their corresponding reviews, users, checkins, tips, and photos (https://www.yelp.com/dataset_challenge). While the whole dataset spans several cities in a few different countries, we wanted to focus specifically on restaurants in the Madison area. Thus, from the original dataset, we first extracted businesses in the Madison area with 'restaurant' as one of the categories. From the resulting businesses, we pulled the corresponding reviews, check-ins, and tips. From the reviews corresponding to businesses in the Madison area, we also retrieved the associated user data. While photos are available, we do not plan to utilize them in our project, so we did not incorporate them in this stage of the project.

Source 2: Zomato

https://www.zomato.com/ is a website with information and user-submitted reviews about restaurants. A quick search on the website revealed that there are 1,759 restaurants in Madison. As with Yelp, we noticed that the use of automated crawlers to extract data from the site is a violation of Zomato's terms of service (https://www.zomato.com/conditions). So, we extracted data from the Zomato API instead. For each restaurant, we have a name, location data, approximate prices, rating data, cuisines, reviews, and other information. In general, restaurants on Zomato have fewer reviews than restaurants on Yelp. Each review is limited to 500 characters, and we have up to 5 reviews for each restaurant.

Rejected Source: Google Places

Google Places API was another choice that was considered for obtaining information and reviews about restaurants in Madison.
However, it was observed that the resulting dataset contained only about 500 restaurants. Due to the limited size of the resulting data we decided not to use restaurant reviews data from the Google Places API even though it offered a structured data with rich schema and many reviews. Instead we decided to use Zomato as the second data source.

3. Extraction of Structured Data

Source 1: Yelp

The whole dataset is fairly large, containing roughly 4.1 million reviews and 947k tips by over 1 million users for 144k businesses, along with check-ins for over 125k businesses and 200k photos. The full dataset from the Yelp dataset challenge contains the following files. Each file contains one json object per line.

- yelp_academic_dataset_business.json: 144072 results
- yelp_academic_dataset_checkin.json: 125532 results
- yelp_academic_dataset_review.json: 4153150 results
- yelp_academic_dataset_tip.json: 946600 results
- yelp_academic_dataset_user.json: 1029432 results

Since our focus is on restaurants in the Madison area, we wanted to extract the relevant data. This was done by first going through the business file, yelp_academic_dataset_business.json. Businesses with a state of "WI" and "Restaurant" as one of its categories were written to yelp_restaurants.json, and a list stored these business_id values. (A state of "WI" was used instead of a city of "Madison", since there are many restaurants near Madison that are in nearby areas such as Fitchburg and Verona. We wanted to include these restaurants too, since the Zomato results include restaurants in Madison and areas nearby Madison like Fitchburg and Verona.) Check-ins (yelp_academic_dataset_checkin.json) and tips (yelp_academic_dataset_tip.json) associated with the aforementioned business_id values were extracted and written to yelp_checkins.json and yelp_tips.json, respectively. This same process was repeated with the review file, yelp_academic_dataset_review.json, to produce yelp_reviews.json and users who wrote reviews of restaurants in the Madison area were kept track of by storing user_id values in a list. Finally, the list containing user_id values from the review file was used to go through the user file, yelp_academic_dataset_user.json, to extract the users who wrote these reviews, resulting in yelp_users.json.

The results of pulling out restaurants relevant to the Madison area are listed below. While each of these resulting files are only about 1% of each original file, we feel the size of the resulting dataset is sufficiently large to work with for this project.

- yelp_restaurants.json: 1403 results
- yelp_checkins.json: 1337 results
- yelp_reviews.json: 61169 results
- yelp_tips.json: 9662 results
- yelp_users.json: 19554 results

Source 2: Zomato

The Zomato API allowed 1000 requests for every 24-hour period since the API key was issued. We gathered data from the Zomato API between 2:00 PM on Saturday, 2/4 and 2:00 PM on Tuesday, 2/7. One API endpoint returned the number of restaurants that matched the HTTP GET parameters, as well as data (not including reviews) for up to

100 of those restaurants. A separate API endpoint returned reviews, given a restaurant id. We chose to receive all data in JSON format.

Since we wanted data for 1,759 restaurants, we needed to limit the number of results for each request, and then perform multiple requests. Initially, we passed an additional parameter to limit search results by cuisine, but several cuisines had more than 100 restaurants. To fetch all data, we limited the size of search results by specifying latitude, longitude, and search radius. We extracted the latitude and longitude coordinates for our results so far and plotted them to determine the approximate spread of results. Using this information, we created a list of (latitude, longitude) pairs such that searches for each pair, with a radius of 2,400 meters, would cover the area where restaurants could be. Iterating through the pairs, we saw that our search areas covered all restaurants, but some search areas had more than 100 restaurants. For these areas, we recursively performed multiple searches with smaller radii. In parts of downtown Madison, where restaurants were very close to each other, the smallest radius we used was 90 meters.

Next, we extracted the restaurant id's from our results. For each id, we sent a request to the second API endpoint to get up to 5 reviews. The request for one restaurant failed to return an HTML response at all, so we omitted this restaurant from our reviews. We summarize information about our results below.

- zomato_restaurants.json: 1759 results
- zomato_reviews.json: 1758 results

4. Extracting from Text Documents (Reviews)

The reviews from Yelp and Zomato are intended to be the text documents for stage 2 of the project. The features/keywords/information we want to extract from the reviews are outlined below.

1. Dining with others
    a. Was the reviewer dining with others? How did the opinions or experiences of others influence the reviewer's perception (rating) of the restaurant?
    b. We could extract words which indicate social occasions and identify who the reviewer was accompanied by extracting words such as "friends", "date", "husband", "kids", "children" from the reviews.
2. Extracting the time of the day the restaurant was visited
    a. Since the time a review is posted need not be the same as when a restaurant was visited, we could extract the time a restaurant was visited by extracting keywords such as "brunch", "morning", "noon", "lunch", "evening", "4pm" and other such temporal keywords which give us the information about the time of visit to a restaurant.
3. Sentiment analysis
    a. Is the text mostly positive or mostly negative?
4. Extracting emotions: words of various categories could be extracted, such as the examples below.

a. Hygiene related words could be adjective words such as “gross”, “mess”, “sticky”, or “dirty”.
b. Service and atmosphere related words such as “dark”, “bright”, “ambient”, “pleasant”, “courteous”, or “romantic”.

5. Open Source Tools Used

1. Yelp

Python 2.7 and the json library were used to extract data relevant to restaurants in the Madison area. The json library, specifically json.loads(), was used to work with json objects in the original files to filter through which results were relevant to keep.

2. Zomato

We wrote Bash scripts that perform HTTP GET requests using curl and stored the JSON output in files. We used Python 2.7 and the json library to combine these separate files and strip away unnecessary information.

3. Google

A simple Node.js client module was written by using node-rest-client library module to make HTTP GET calls to the Google places API by passing in the API key with every request.