Mapstaurant- A Restaurant Recommendation System

1 Introduction

Searching for desired restaurant plays an important role in our social life. Currently, over 178 million unique visitors are using similar applications such as Yelp. Therefore, designing a user-friendly interface to help people find the best restaurants near them is important. Moreover, during the current COVID-19 situation, incorporating related features to provide additional information to users has never been so crucial.

2 Problem Definition

Our project is to establish a new restaurant recommendation website which filters the fake or spam reviews and gives our new ratings for each restaurant. Besides, we will classify restaurants according to the reviews and add new auxiliary information related to COVID-19.

2.1 Innovation

(1) In order to get rid of the different standards and fake review detection, we develop a sentiment analysis and improve the grading system according to the review content.
(2) Incorporate COVID-19 related data, such as hours and delivery options, to give the most up-to-date information about each restaurant.
(3) Take keywords from reviews of restaurants to give individualized features for every restaurant. Thus, save our users from reading through long reviews and help them choose wisely.

3 Academic Survey

3.1 Data Visualization

Since we will process a large amount of Yelp data and show these data visually, the techniques described in [16] will be very useful in visualizing large dataset.

Paper[14] proposed a geographical visualization for fast food restaurants. We will generalize this approach to fit all kinds of restaurants filtered by categories. Moreover, paper[14] highlighted the importance of visualizing proper scope of geographic map.

Similarly, Ji Wang et al.[13] describes a new approach to visualizing word cloud which place semantically related words together. We will mimic this technique to show the review keywords in a word cloud. The downside of this method is that users typically spent more time deciding between two good restaurants because more context information is needed to support the decision.

3.2 Review Interpretation

Paper[11] used Yelp reviews and NLP to extract different taste characteristics and scoring criteria of users in different cities and linked it with geography. Its algorithms and feature extraction methods can be used, and we will improve in how to visualize these features to users.

The average price of a restaurant also matters. We can use the Random Forest Algorithm of [3] to deal with Yelp reviews with filter rankings. But paper[3] is for economic research, we pay more attention to user experience.

3.3 Recommendation Models

Content-based methods such as clustering[1] and classic methods of classification such as SVM and Logistic Regression[9] will be considered based on the availability of training labels in our data set. Also, paper[8] provides a comprehensive introduction and comparison of different recommendation methods.

A recommendation system[12] for event invitations on social networks should consider the distance between the actual event and the person. Similarly, our design can allow users to filter based on distance to the restaurant or incorporate the distance into the total rating formula. Another recommendation system[15] takes into account implicit (visits via check-ins) and explicit (reviews) user feedback. We can mimic these methodologies to form our unique one.

Our basic idea comes from [4] that if labeled restaurants with keywords of reviews, customers will make more efficient decisions and the result will be more suitable for the customers. Luca et al.[6] also analyzed how Yelp reviews affect restaurant revenues. The shortcoming is that the paper did not mention how different presentations of the data affect the businesses. Our project will condense these reviews into a few keywords and display them. Besides, we will use techniques in Text processing such as Term Frequency[10].

Our COVID-19 model can use the timing and equilibrium parameters in model[2] which used restaurant activity data to imply the influence of re-opening. But we will create a more objective platform than the original one.

3.4 Fake Review Detection

In this paper[5], the author provides positive and unlabeled learning based on the user's attributions such as their IP address and features to detect the potentially fake reviews in the unlabeled set. This paper will help us in doing the fake reviews filtration in our project. However, this is mainly based on Chinese language reviews, we have to find a way to apply similar idea to our own project. Besides, this paper[7] also introduces some techniques to do that.

4 Proposed Method

Our methods includes back-end data processing and front-end data visualization. The back-end will produce a csv file which includes extracted keywords, categories, updated ratings, and various attributes for each restaurant. The front-end will take the result csv and directly display the data so there will be no client-server communication since all of our data are ready. The detailed implementations and methods will be described in the below sections.

4.1 Yelp Data Collection

We choose to use Yelp Open Data set from its official website - https://www.yelp.com/dataset as origin data Yelp_academic_dataset_business.json and yelp_academic_dataset_review.

4.2 Data Processing

We need to clean the original data from the Yelp dataset before being processed. Firstly, we clean the data of yelp_academic_dataset_business.json by filter out all businesses with incomplete information (e.g. missing address, business hours). Then we filter out all restaurants by different cities with at least contain the "restaurant" category and sort them to get the cities with most restaurants. We choose Las Vegas for it has the most restaurant contained in the dataset.

Then, we merge the filtered Las Vegas restaurants data with reviews data together on business id for easier reference across each other.

We applied the Pandas module in Python to implement the above part.

Finally, we have collected 5520 restaurants' data and 1458256 reviews from the Yelp dataset.

4.3 Extract Review Keywords

Since we want to provide users with a more intuitive restaurant choice, we decided to extract the keywords for the food, environment, and service comments in each review, and manually select the top five keywords for each restaurant as this restaurant's new features. Because the content of features is extracted from users' real reviews, we believe that giving each restaurant new features can provide users more personalized restaurant choices in addition to restaurant ratings and categories. For example, if users like eating Japanese food, they can choose the "fresh" feature to get a fresh Japanese restaurant or the "authentic" feature to get a more authentic Japanese restaurant.

4.4 Get top reviews

To let our users learn about a restaurant in a short time, we also extracted keywords about restaurant reviews based on the reviews that deleted fake reviews and hoped to show some real reviews after filtration.

We decided to use the random forest classifier provided from the python sklearn library to determine the different feature importance and arrange them in descending order according to the most important features. For each restaurant, we only take the top five reviews corresponding to the order of the most important features.

We used the sum of useful, sum of funny, and the sum of cool as three features to determine their importance for average stars for each restaurant. From the results of the random forest classifier, the sum of useful is the feature that has the greatest impact on the score.

So, for every restaurant, we extract top 5 reviews according to the descending of useful count for each review. In that way, we believe we can give our users a quick yet comprehensive way for understanding a restaurant and help them make their own choice wisely.

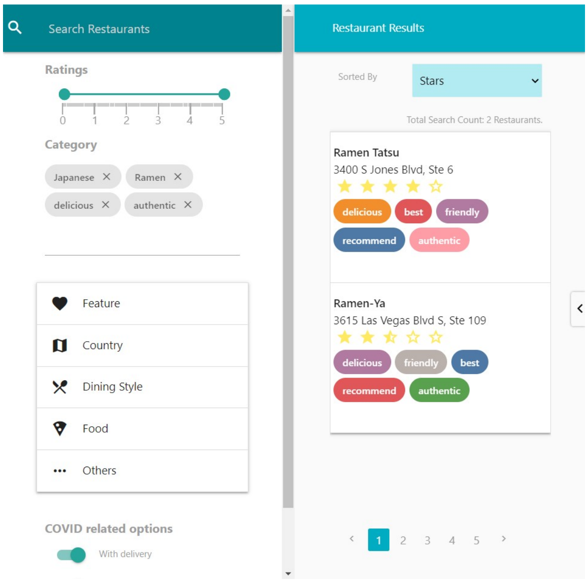

Figure 1: Filter Menu

4.5 Front-end Filter

We need a clean filtering menu which allows the user to find their favorite restaurant. This is best accomplished using Materialize framework since it provides built-in responsive UI.

In the filtering menu, as shown in figure 1, the user can search a specific restaurant based on the name of the restaurant. The search box will then marks restaurants whose name contains the input keywords on the map.

The rest of the menu focuses on filtering restaurants with different filtering criteria. The user can filter by ratings and category tags. Part of these categories comes from the original yelp data set and the rest of the categories are extracted from the user reviews. To construct a list of interested categories from the yelp dataset, we collected all unique categories among the 5520 restaurants which we included in our processed dataset. In total, we collected 459 unique categories and sorted them in the order of frequency of being listed by a restaurant. However, 459 categories are too much for front-end user interface to display in a neat manner. Therefore, we made a choice to filter out those categories with frequency less than 0.1% of 5520 restaurants, i.e. frequency less than 6, and we have 138 categories in our current pool for frond-end filter to select from. We will prioritize the restaurants which satisfies all the tags over those which only contains parts of the input tags.

We have further grouped these categories into five fine-grained sub-categories to improve readability and compactness of the UI. The five sub-categories are "Features", "Country", "Dining Style", "Food", and "Others" as shown in figure 1. The "Features" sub-categories consists of features extracted from each restaurant reviews as mentioned in section 4.3. Which include categories like "authentic" and "delicious". The other four sub-categories are obtained from the yelp dataset as mentioned in the previously. "Country" includes categories like: "Mexican" and "Chinese". "Dining Style" includes: "Bars" and "Buffets". "Food" includes: "Pizza" and "Seafood". And "Others" includes some special needs such as: "Food Delivery Services" and "Gluten-Free". These sub-categories are implemented as collapsible blocks in our filtering UI to make our webpage even more succinct and also allow users to choose their filtering keywords more systematically.

The filtered restaurants will be listed under a scroll-able area as shown in the right-hand side of figure 1. Each restaurants will be shown with their name, address, star ratings, and top-5 keywords extracted from the reviews. Furthermore, as shown in figure 4, each restaurant result can be expanded when clicked by the user (green arrow), displaying information such as restaurant hours, original yelp ratings, number of reviews, and some meaningful reviews determined by methods in section 4.4 (red rectangle). Results are collapsible and scroll-able, and top-5 reviews are displaying with useful rating as a reference provided for users. Open hours are obtained by the current day of user time.

Lastly, these results will be shown in the order of most matched categories first, and then by star ratings. Users could choose to sort by other methods too. (orange rectangle). Also, the orders are shown separately by pagination (blue rectangle).

4.6 Front-end Map

There are four functions for our map: (1) restaurant map visualization, (2) map zooming,(3) interactive marker,(4) markers clustering.

(1) We use Leaflet and jQuery as our map visualization tool. Since for Leaflet, a json object is needed, but it's easier to get a csv file from our back end. Thus, we use Papaparse to transform and read csv file into json object. By using leaflet,

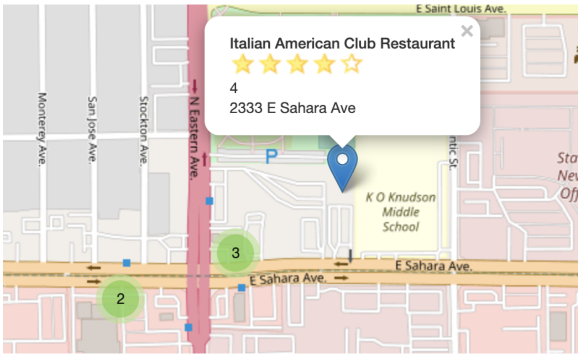

Figure 2: Marker and Clustering

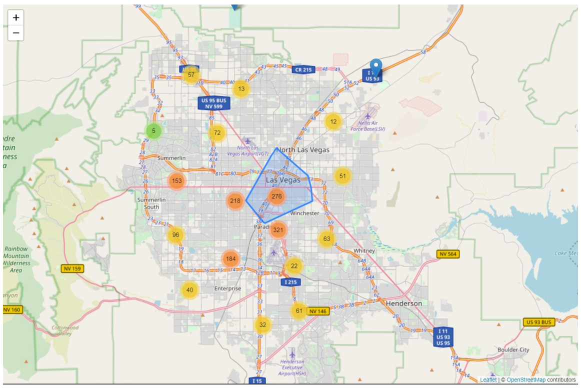

Figure 3: Map displaying all restaurants in Las Vegas

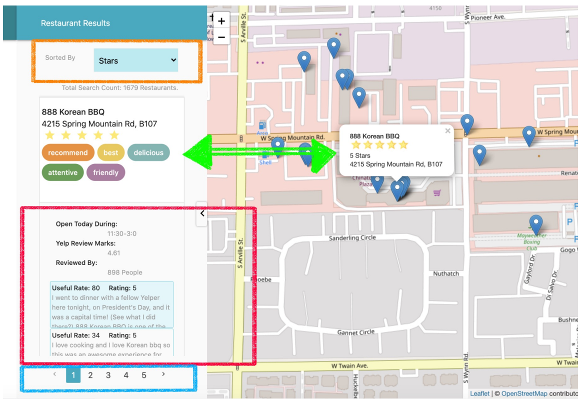

Figure 4: Interactive Search Result

we extract longitude and latitude and all restaurant information for markers. We also use image sprites to display stars in visualized form.

(2) Our map provide zoom in and out interaction for overall and detailed scan. We also provide interaction on each marker when hovering to see details of our restaurant.

(3) Not only could users choose by search result to focus on one restaurant, they can also choose markers to see detailed results. Functions are made when hovering to see brief information and clicking to locate to the right page and see detailed definition.

(4) Since our platform provides a large number of restaurants. For many areas such as shopping malls, markers would occur and overlap with each other. Therefore, we choose to use clustering to represent restaurant number for an area.

The method we use is MarkerClustering.js. It calculate and cluster nearby markers, bound them as a group, and shows the region border when hovering. Also, it build a tree with parent cluster node and cluster children. As what we can see in Figure 3 & 2, a circle represents the count number, range shape and range color of clustering. And it is also interactive when you click on one cluster, it would zoom in and display children cluster. In this way, the interaction could be more user friendly.

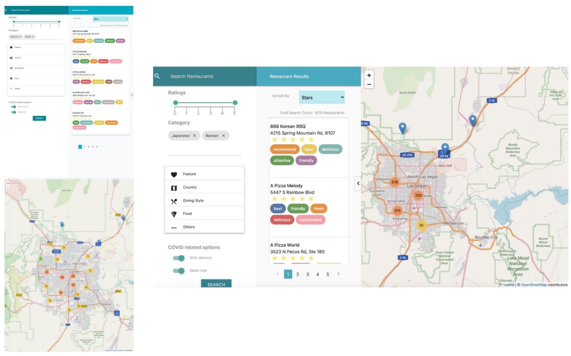

Figure 5: Mobile Layout(Left) and Laptop Layout(Right)

4.7 Mobile Responsive Platform

We also use Materialize.js to make our platform responsive for mobile devices. Figure 5 shows the different layout proportions displayed under different screen size conditions.According to different screen characteristics, we made the mobile one into the upper and lower structure, and the laptop into the left and right structure.

4.8 Filter fake review analysis

The main drawback of Yelp grading system is it just takes the average of all the users' marks and round up to a number. The potential problem is it doesn't consider the effect of peer ratings, such as useful marks, which will be largely affected by the fake reviews or fake marks. Besides, another existing defect is that everyone's grading standard varies a lot, which makes the final grade no sense. The analysis results of original grading system are shown as Fig. 6.

As what we can see, the distribution is not even and we can't see any useful information simply from the final grades. Alternatively, we can use sentiment analysis to overcome such disadvantages, combining with the usage of useful marks.We can directly give each restaurant a new mark based on the sentiment analysis of reviews. For the sentiment analysis, the model is based on Naive Bayes, which use a trained model with a built-in dictionary of positive and negative words. The likelihood function is shown as below:

$$l(\theta_{c,k}) = \sum_{i=1}^{m}\sum_{k=1}^{d} x_k^{(i)} log\theta_{y^{(i)},k} \qquad c = -1, 1$$

$\theta_{y^{(i)},k}$ is the parameter gained from the trained data, $x_k^{(i)}$ is the review's feature vector, c is the cluster to define whether it is a positive or negative review. The $log\theta_{y^{(i)},k}$ is each word's contribution. The simplest sentiment analysis is to sum all words, the formula is like:

$$Sentimentscore = \sum_{i=1}^{m} c \cdot log\theta_{y^{(i)},k} \qquad c = -1, 1$$

The final Sentiment score is to choose the larger absolute value and normalize it to between -1 and 1. By considering the useful score as a weighted parameter, our modified score is shown like this:

$$Modi\_score = \sum_{i=1}^{m} \frac{(useful[i] + 1) \cdot Sentimentscore}{numberofreview + useful[i]}$$

To make modified score more sensible, we use the quantile grading system. To be specific, if the modified score is larger or equal to 95 percent quantile, then we give it 5 stars.

5 Experiments and Observation

5.1 Sentiment Usefulness Grading System

In order to implement the Sentiment Analysis and the proposed Sentiment-Usefulness grading system, we used the Python package TextBlob for the sentiment analysis purposes. Then we used the proposed function in Chap 4.8 to get the new scores. When getting the new scores, we will compare new scores with the Yelp's original scores to see the difference. The comparison is made by making the original data into quantile and compare it with new score. The result is shown as below:

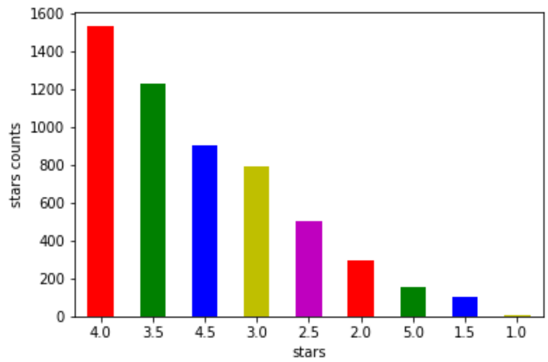

Figure 6: Yelp Restaurants Star Distribution

Firstly, the original scores in figure 6 show a large group of restaurants get 4.0 points. Since the Yelp just takes the average of all the comments, So it hard to see whether this average grading is useful or not. Besides, the original scores might be biased due to the existence of many fake reviews which tends to give high scores to a restaurant.

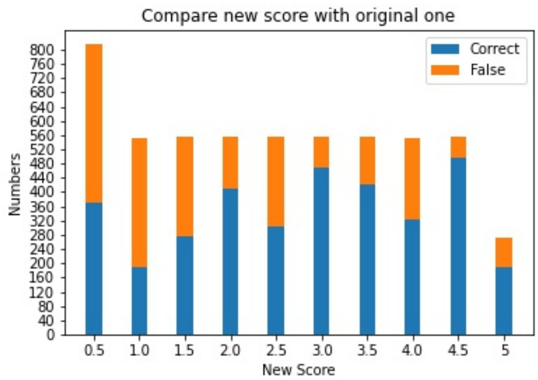

Figure 7: Sentiment-Usefulness and Yelp Comparison

The figure 7 shows the result for comparing proposed Sentiment-Usefulness Grading and original Yelp Grading. The results shows when the new score is larger than or equal to 3.0, the Yelp grading shows a high consistency with Sentiment-Usefulness Grading. But when the new score is low than 3.0, the Yelp grading doesn't match

Sentiment-Usefulness Grading. By combining figure 6 and 7, we could conclude that many high score restaurants in original Yelp grading are now at the low score region in proposed Sentiment-Usefulness Grading. We believe the difference is resulted from the existence of fake reviews which give a high score to the poor restaurant, making these restaurants have a biased higher score.

6 Evaluation and Survey

6.1 Survey Design

In order to evaluate the user experience for our website, we designed a Google questionnaire: https://forms.gle/Pi1beLz3tdi3YDNd6 and distributed this survey to students who have the access to github.gatech.edu to collect feedback. During November 19-21, we collected 44 responses for this survey and the majority of the participants are male graduate student from Georgia Tech. In this survey, we designed 11 questions for users to evaluate our features, and 2 questions for users to compare our website with their mostly used restaurant finding application. Lastly, 3 questions are designed to acquire the basic information such as gender and degree level to know the characteristics of our participants.

6.2 Feature Evaluation Results

From question 1 to 11 (Q1-Q11), we specify each feature in our user interface with brief tutorial to help user experience the feature. Then, we ask users to evaluate how helpful they think about the feature from very helpful (score=3.0), moderately helpful (score=2.0), slightly helpful (score=1.0), to not helpful (score=0.0). And figure 8 shows users are impressed the most by our rating filter, and clustered markers zooming. Among the 4 fine-grained categories for filtering, "Dining Style" and "Food" are the more popular ones. Moreover, we examined the 95% confidence interval for these average score and found each of the lower bound is greater than 1.5 which means all of our designed features earned positive feedback above the average score that we can get from the scoring range 0.0-3.0.

6.3 Application Comparison

From question 12 to 13 (Q12-Q13), we asked users' mostly used application for finding restaurants and asked users to choose any (multiple) feature that our website is superior by comparison. We

Feature	Average Score
Rating Filter (Q2)	2.82
Map Zooming (Q10)	2.73
Restaurant Information on Map (Q11)	2.64
Category Filter: "Dining style" (Q5)	2.55
Category Filter: "Food" (Q6)	2.45
Result List Display (Q9)	2.41
COVID19 Related Filter (Q8)	2.36
Category Filter: "Country" (Q4)	2.27
Search Bar (Q1)	2.18
Category Filter: "Feature" (Q3)	2.09
Category Filter "Others" (Q7)	1.95

Figure 8: Sorted Average Score for Website Features (score ranging from 0.0 to 3.0)

found 77% of the participants use Google Map the most and 23% for Yelp when trying to find restaurants. And figure 9 shows Google Map users admire our rating filter the most, and not surprisingly the map browsing feature the least. The results also shows our website out-performed Yelp in filtering and map features. These results convince us that our website is competitive to the existing popular applications for finding restaurants.

Our Feature	% of participants thinks better than : Google Map	Yelp
Search Bar (Q1)	41.18	60
Rating Filter (Q2)	47.06	80
Category Filter (Q3~7)	29.41	80
COVID19 Related Filter (Q8)	35.29	80
Result List Display (Q9)	29.41	20
Map Browsing (Q10~11)	23.53	80

Figure 9: Comparison with Similar Applications

7 Distribution of Effort

All team members have contributed similar amount of effort.

8 Conclusions & Discussion

We proposed an interactive restaurant recommendation map with multiple filtering methods and our survey results showed our application is competitive to existing popular restaurant finding applications. In the future, our application shows greater potential if we can implement more features such as incorporating the user's distance to the restaurants as filtering criterion, or improving the category filtering options instead of displaying at most 56 keywords at a time for our user to choose from.

References

[1] Gediminas Adomavicius and Alexander Tuzhilin. 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*, 17(6):734–749, 2005.

[2] Edward L Glaeser, Ginger Zhe Jin, Benjamin T Leyden, and Michael Luca. Learning from deregulation: The asymmetric impact of lockdown and reopening on risky behavior during covid-19. Technical report, National Bureau of Economic Research, 2020.

[3] Edward L Glaeser, Hyunjin Kim, and Michael Luca. Nowcasting the local economy: Using yelp data to measure economic activity. Technical report, National Bureau of Economic Research, 2017.

[4] Wei Jin, Hung Hay Ho, and Rohini K Srihari. Opinionminer: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1195–1204, 2009.

[5] Huayi Li, Bing Liu, Arjun Mukherjee, and Jidong Shao. Spotting fake reviews using positive-unlabeled learning. *Computación y Sistemas*, 18(3):467–475, 2014.

[6] Michael Luca. Reviews, reputation, and revenue: The case of yelp. com. *Com (March 15, 2016). Harvard Business School NOM Unit Working Paper*, (12-016), 2016.

[7] Michael Luca and Georgios Zervas. Fake it till you make it: Reputation, competition, and yelp review fraud. *Management Science*, 62(12):3412–3427, 2016.

[8] Masoud Mansoury, Bamshad Mobasher, Robin Burke, and Mykola Pechenizkiy. Bias disparity in collaborative recommendation: Algorithmic evaluation and comparison. *arXiv preprint arXiv:1908.00831*, 2019.

[9] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[10] Elizabeth Poché, Nishant Jha, Grant Williams, Jazmine Staten, Miles Vesper, and Anas Mahmoud. Analyzing user comments on youtube coding tutorial videos. In *2017 IEEE/ACM 25th International Conference on Program Comprehension (ICPC)*, pages 196–206. IEEE, 2017.

[11] Sohrab Rahimi, Sam Mottahedi, and Xi Liu. The geography of taste: using yelp to study urban culture. *ISPRS International Journal of Geo-Information*, 7(9):376, 2018.

[12] Diego Saez-Trumper, Daniele Quercia, and Jon Crowcroft. Ads and the city: considering geographic distance goes a long way. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 187–194, 2012.

[13] Ji Wang, Jian Zhao, Sheng Guo, Chris North, and Naren Ramakrishnan. Recloud: semantics-based word cloud visualization of user reviews. In *Proceedings of Graphics Interface 2014*, pages 151–158. 2014.

[14] DL Widaningrum, I Surjandari, and AM Arymurthy. Visualization of fast food restaurant location using geographical information system. In *IOP Conf Ser Earth Environ Sci*, 2018.

[15] Fuzheng Zhang, Nicholas Jing Yuan, Kai Zheng, Defu Lian, Xing Xie, and Yong Rui. Exploiting dining preference for restaurant recommendation. In *Proceedings of the 25th International Conference on World Wide Web*, pages 725–735, 2016.

[16] Nick Qi Zhu. *Data visualization with D3. js cookbook*. Packt Publishing Ltd, 2013.