Final report

Project title: Pittsburgh Restaurant Rating System
Team members: Cecilia Cai (zc1151), Yichen Wei (yw3092), Yuhan Wang (yw3091)

1. Overview
    a. Screenshot of your visualization

    b. Summary of your project briefly

    There are a lot of restaurants in Pittsburgh but it is hard to find the “best” places to eat when searching online. To better understand the food scene in Pittsburgh, we are interested to show the different types of restaurants on map and sort the restaurant in different rating criteria to help people find the top ranking restaurants to eat in Pittsburgh.

2. Data
    a. description of your data (e.g., dataset type, scale/cardinality)
        i. Business: flat table with 14 attributes
        "Business_id":category, 2149 levels
        "Name":category, 1784 levels
        "Address":category, 2149 levels
        "City":category, 1 level
        "State":category, 1 level
        "Latitude":quantitative, ranging from 40.3148776 to 40.591735
        "Longitude":quantitative, ranging from -80.259831 to -79.705
        "Stars":ordinal, 9 levels
        "Review_count": quantitative, ranging from 3 to 2001
        "Is_open":category, 2 levels
        "BusinessAcceptsCreditCards":category, 2 levels

"BikeParking":category, 2 levels
"GoodForKids":category, 2 levels
"ByAppointmentOnly":category, 2 levels
"RestaurantsPriceRange2":category, 3 levels
"Monday":quantitative
"Tuesday":quantitative
"Wednesday":quantitative
"Thursday":quantitative
"Friday":quantitative
"Saturday":quantitative
"Sunday":quantitative

ii. Review_total: flat table with 9 attributes:
"Business_id":category, 2149 levels
"Stars":ordinal, 9 levels
"Useful":quantitative, ranging from 0 to 92
"Funny":quantitative, ranging from 0 to 57
"Cool":quantitative, ranging from 0 to 86
"Text":category, 53494 levels

iii. Pittsburgh_2014_CDBG_Census_Tracts: geometry dataset with geometric longitude and latitude of each neighborhood in Pittsburgh

iv. Pittsburgh_park: geometry dataset with geometric longitude and latitude of each park in Pittsburgh

v. Derived datasets:

1. freq_type.json: flat table derived from "Review" table's text attribute
"Text": category, 288 levels
"Size": quantitative, ranging from 1 to 2149

b. Include a URL linking to the source of your data

i. Business, review_total:
https://www.kaggle.com/yelp-dataset/yelp-dataset?select=yelp_academic_datas

ii. Pittsburgh_2014_CDBG_Census_Tracts
https://catalog.data.gov/dataset/pittsburgh-neighborhoods-map

iii. Pittsburgh_park
https://data.wprdc.org/dataset/pittsburgh-parks

c. Briefly describe your current data preprocessing pipeline, if there is one.

i. In the "Business" database, there are 290000+ business in total but since we only care about restaurants in Pittsburgh, we first filter out the city "Pittsburgh", category "restaurant". We can get 8000+ items.

ii. Then, we screen the raw data in a small scale by narrowing with the opening date. In another word, if a restaurant lacks the information of open hour or close for the whole week, the restaurants will be screening out.

iii. After clearing the basic restaurant data, we merge the other datasets, such as user information, reviewer and check in information. The joint key for merging is business_id which is unique for each restaurant.
iv. Finally, the original files are Json files and we have transferred it into csv files.
v. For the derived dataset freq_type, we focus on the "category" attribute in "business" database. We select all the words in the field, separate them into individual phrases, and then count the frequency for each word. The derived table is sorted based on the frequency of each category label.

3. Goals and tasks
    a. Description of your intended task(s) in both domain-specific language (based on the usage scenarios) and abstract language (visualization language).
        i. Domain language: Our target users are the general public who are interested in the restaurants at Pittsburgh. The users can explore the local restaurant on the geometric map based on their preference. More specifically, they can make their dining choice based on their personal weight of price, review counts, and other factors influencing the restaurants' rating tailored for this user. They can also see the relative percentile of the different dimensions used to rate this restaurant.
        ii. Task abstraction: Users can explore restaurants in Pittsburgh, browse restaurants in each neighborhood, identify restaurants for each category, compare restaurant characteristics with others', locate the top 10 restaurants for the dimension he cares about.
4. Visualization
    a. Describe the visualization interface that you have built. What views are there and what do they allow users to do? For each view, describe your visual encoding choices and include the rationale for your design choices. How can users interact with your project within each view, and how are views linked?

        VIEWS:

        There are three major views: geometric map view, line chart glyph view, and stacked bar chart view

        - Geometric map allows users to have an overall view of the popular restaurants across Pittsburgh; users can zoom in and out to see more clearly the distribution and select particular restaurants. We use the point mark with color and size channels to encode restaurant category and review counts. We adopt this encoding choice because restaurants are categorical, so we want to separate them according to different types, so the color channel is efficient to convey category information. We use the size channel to encode review count because it gives an intuitive scale of quantitative attributes. Since the interaction of hue and size can be distracting, we separate the two channels by allowing users to filter out one category, so he can focus on popularity. We use the area mark with the color channel to encode neighborhoods. Since we don't have much information to

show neighborhoods itself, it just helps separate the restaurants from others based on the geometric location.

- Line chart glyph allow users to see five main characteristics that determine a restaurant's rating. We use link and point marks with position channels to encode the numerical attributes. We use this design because position is the most effective to represent numerical values. We design the axis to be cyclic so that the line can form an area for easy comparison between restaurants. It is linked to the geometric map. If a user hover on the point on map, the glyph will show the information of this restaurant. If the user hover on each axis, the node will show the percentile of this dimension as long as its original value. For example, if a restaurant is rated 4 stars, and 4 stars is 80% percentile among all Pittsburgh restaurants, then in the glyph when users hover on the star axis's node, it will show 80%, and also 4.
- Stacked bar charts allow users to rank the restaurants according to its preference on review counts, opening days, prices and other factors. We use link marks with position channels to encode the factors' quantity. We use the color channel to encode the type of ranking because it is easier to contrast with color. We use stacked bar charts because we think users are willing to see each factor on bars separately, and see the rankings directly from the total length of each bar. The sliders will be used to control the weight on each dimension.

INTERACTIONS BETWEEN THE VIEWS:

The interactions among three views are linked. When the user hovers over a point on the map, a glyph showing the information for the corresponding restaurant will appear on the left bottom corner of the window, along with the detailed information about the selected restaurant in text, and the stacked bar chart will show information for top 10 restaurants of the same category to the selected restaurant.

When the user clicks on a category of the restaurant on the legend, the points of the same category on the map will be filtered out and pop up, while others will be hidden behind. Users can then hover over the popped up points to check for information of specific restaurants of his/her desired category (the one selected on the legend). Meanwhile, the bar chart will also show information of the top 10 restaurants of the selected category.

On the other hand, when the user hover over the bar chart, the corresponding restaurant will pop up on the map, and its information will show on the left hand side.

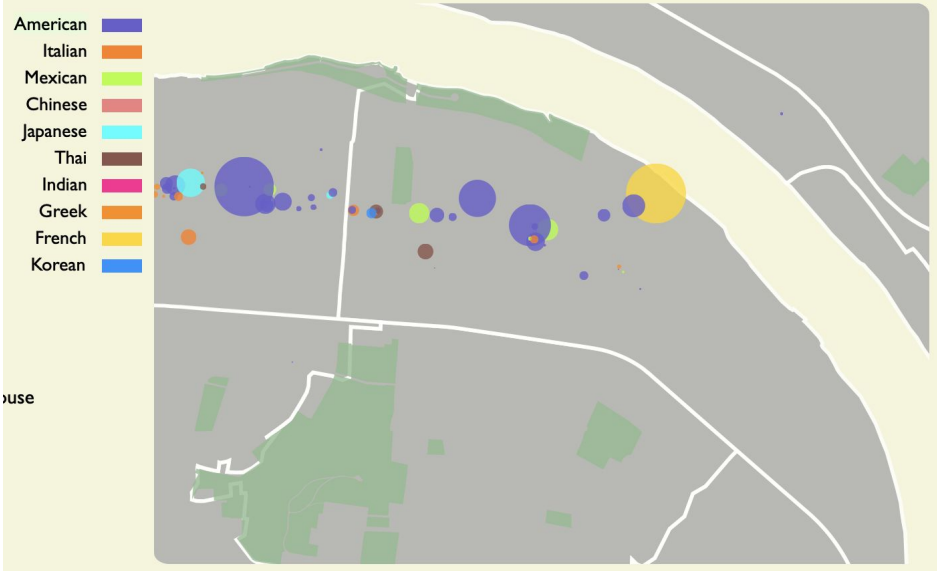
ADDITIONAL FEATURE:

In addition, we generated a Word cloud, which is a view for fun. We use size channels to encode the frequency of each category. When the user clicks on one word, for example, bars, they will be redirected to Google search for "Pittsburgh bars".

b. Include screenshots to support your words

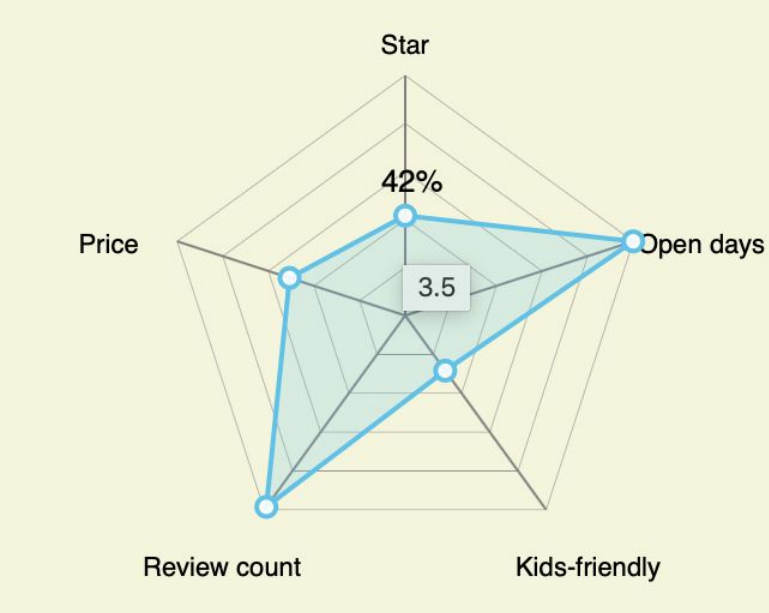
1. Select certain category (for example, select Thai)

2. Zoom in



3. When hovering over a point on the map, the corresponding category will pop out on the legend, and its glyph view will show up as well.

4. Hover on the glyph's node, both the percentile information and the absolute value will show up



5. The bar chart will change according to the choice of different rating criteria. If hover on the bar, the corresponding restaurant will turn red on the map, and the glyph will show the detailed information.

5. Reflection
    a. Describe how your project has developed from your initial proposal to your final product

Our initial proposal is very similar with our final product. With our initial proposal, we started to implement each function, but we also changed our interface so it is more attractive. For example, we initially want to have an “other” category so that users can type in some keywords to search for this type of restaurant. However, after our implementation we think the function is not effective because it can be very distracting to see different colors and sizes popping out all together. For example, if the user type in Asian, both Thai and Chinese restaurants will show up, but this can also be done easily by selecting both Chinese and Thai checkboxes. Therefore, we think it’s a redundant function and delete it. Another function we abandoned is letting the user select the specific time he will come to the restaurant. It is because the opening honor in our dataset is not consistent. Also, the time data is like “11:0-1:0”, and cleaning time data will be hard. Thus, we allow users to select the specific day on which he would like to visit the restaurant, and we provide the opening hour information for users’ reference.

b. How have your visualization goals changed?

   We did not change our goal. We stick to our initial goal and implement them step by step. However, we do have little adjustments. For the glyph view, we initially want it to cover the text information about the restaurant because glyph contains the numerical information. However, we decided to keep the words because users are interested in information such as address and related categories.

c. How have your technical goals changed?

   Because our dataset is from different sources, they are not consistent with each other. For example, the geometric map is not the same scale with the restaurant's location scale, so some restaurants show outside the map. Therefore, instead of showing the whole map of Pittsburgh, we have to strictly exclude its suburb and let the users explore the region. Additionally, we initially want to draw the word cloud by ourselves, but we find the three major views are already complicated, so we decided to adopt open resources to draw the word cloud and do not incorporate important information in this view.