CS573 Data Visualization Final Project Process Book

Project title: Visualizing Yelp Dataset
Project repo: https://github.com/stels07/DataVisFinal
Team:
Yihao Zhou, yzhou2@wpi.edu, iihaw
Hongzhang Cheng hcheng3@wpi.edu, hcheng3
Shi Wang, swang11@wpi.edu, stels07

Background and motivation
Our team members have been yelp users for a long time and found the platform very useful. The app offers list view and map view of the businesses around, and users can apply different filters according to their preference. The mobile app seems to be more useful, but there is limited screen for displaying data. We thought it'd be interesting to explore how data could be better visualized on a bigger screen. We'd like to help user find what they are looking for more quickly, and at the same time give more information about rating and an overall picture of the businesses around. Based on our own experiences, we concluded that the key factors for decision making are: average rating of the business, category, location, price, number of reviews. Number of reviews is a strong indicator to how reliable the average rating is. We designed to encode all key factors into our visualization and it provides a better user experience for exploring and selecting restaurants.

Project objectives
We have a few objectives for our project. First our visualization needs to provide a general picture of the businesses in the area. Second our visualization needs to include average rating, category, location, price, busy hour for a business. Third we need to show details about customers rating and reviews. Finally users who using our design are able to filter easily.

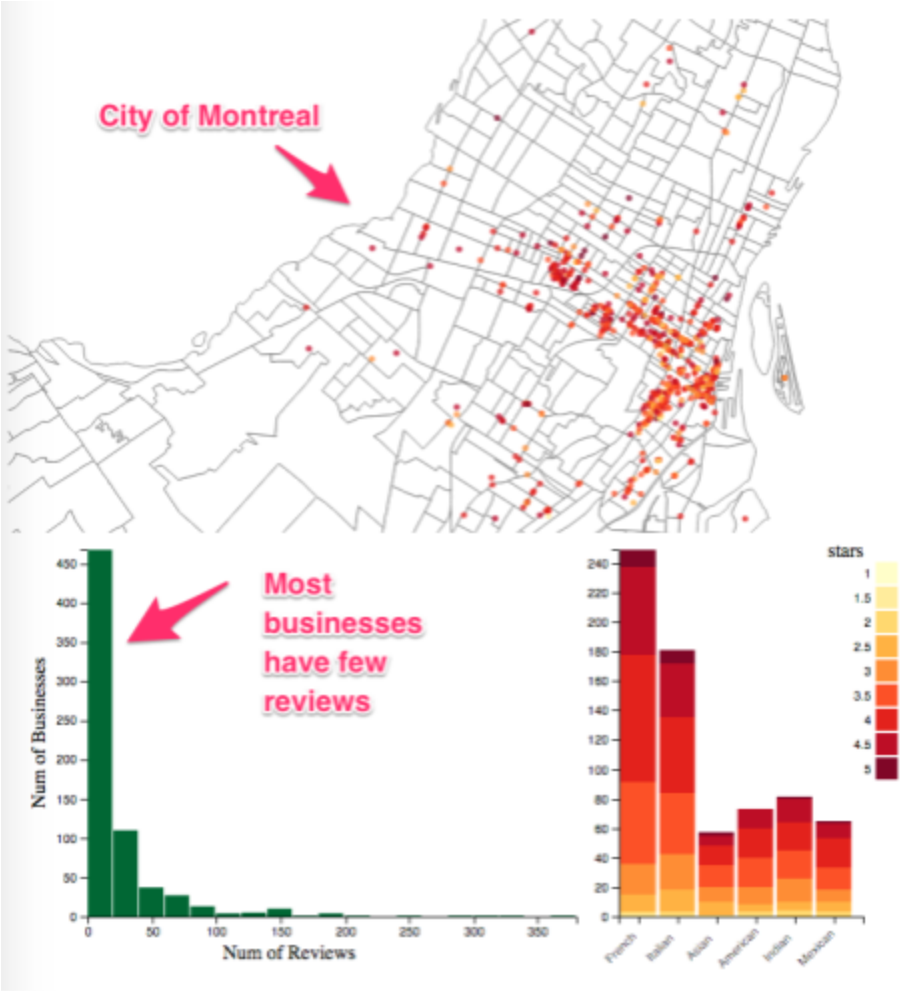
Data
We use the data from yelp's data challenge: https://www.yelp.com/dataset_challenge, which including businesses, reviews, and user data from ten large cities worldwide formed into json files.

Data processing
The data we get from Yelp includes multiple large json files. It contains data for selected cities. The dataset is pretty clean. We just need to filter out data for a certain city (Montreal for instance). After using R for some preliminary assessment, we found out there were 4371 businesses in Montreal. Because it will be too dense to show all the businesses on the map and because of our objective is to help users find the restaurants, we decided to only use restaurant data, which reduce the number of businesses to 2146.

In the initial dataset we downloaded business, review and user data are in different json files. So we joined the data by business_id, which is an unique id across all the file. The joining process is written using Python.

Initial Data Exploratory

After completing cleaning and joining our data, we did the initial exploratory of our dataset by using the the dataset on assignment 5.

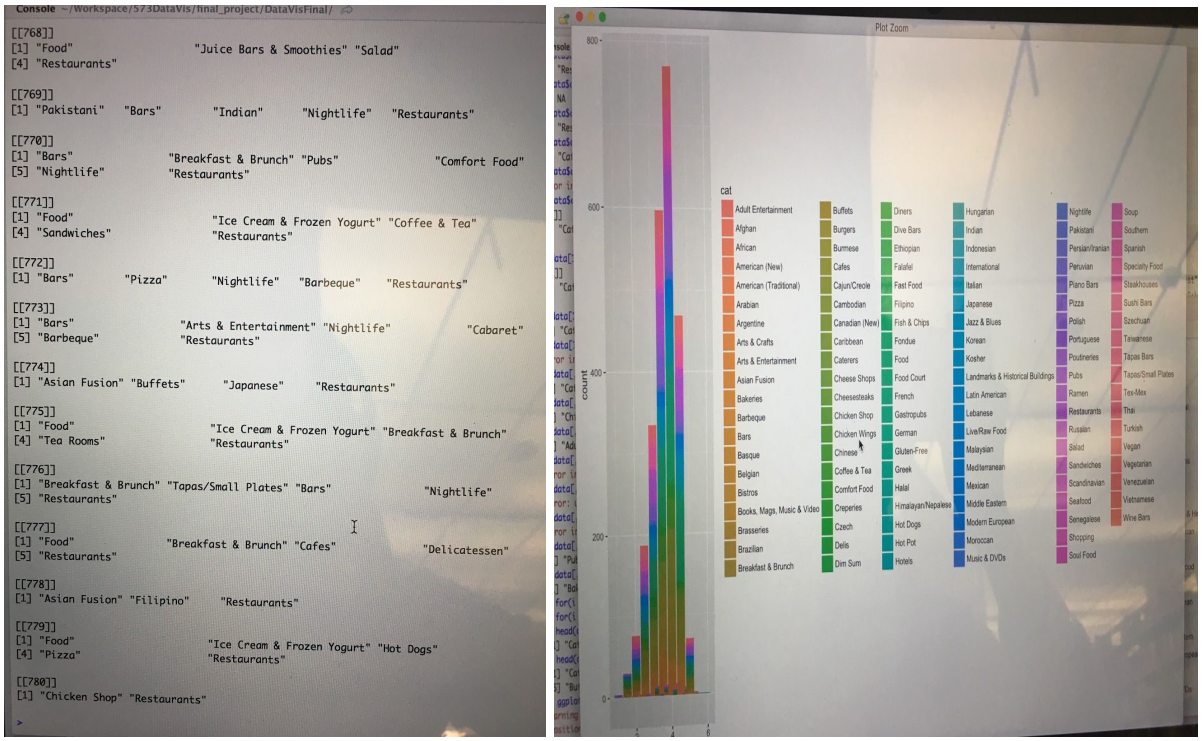


#1.1 Graph for assignment 5

As shown in the graph #1.1, we created three interactive graphs using d3.js. The map shows the city of Montreal and dots represent all the restaurants in Montreal. The color of dots maps to the star rating of restaurants from 1 to 5 stars. The bar chart on the lower left side shows the number of restaurants with different number of reviews. The idea behind this chart is to show how the overall popularity of all the restaurants in this area. As we can see most of the restaurants in Montreal have very few reviews, which could be because most restaurants are

not popular or Yelp is not popular in Montreal, therefore not many people leave reviews on Yelp. The stacked bar chart on the lower right shows number of restaurants in different categories and with different rating stars. As we can see in Montreal French category has the most number of restaurants and most French restaurants are within rating between 3.5 to 4.5. With initial data exploration, we had a brief idea of how the data looks like in a large scale. We found out that restaurants are too dense in some area of the city and many dots are overlaid with each other. Therefore it is difficult to select one particular restaurant in high density area. We then decided that we are going to make a zoomable map so user can zoom in to select the restaurant easily.

One thing we realized during data exploration is that one restaurant usually have more than one category labels. Putting all categories in a stacked bar chart is not applicable.



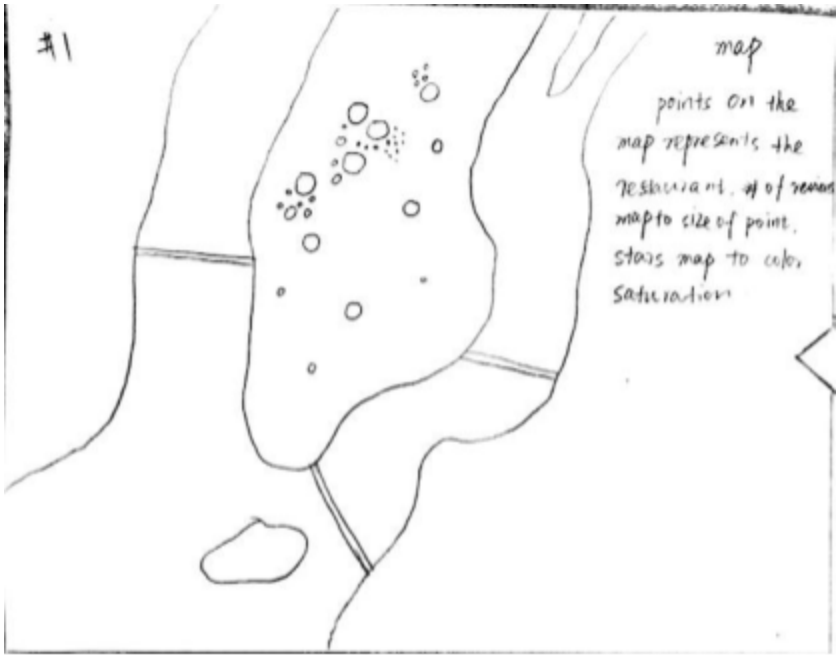
Later in our design, we use treemap to filter category. A restaurant with multiple category labels will be counted multiple times in the treemap. It makes sense because when user filter by category, we want to present all the businesses in that category.

Visualization Design Evolution

1. **Map**

At first we represent all the restaurants on a static map (see Map Version 1), mapping the number of the reviews for each restaurant to the size of the point and map the average star of the restaurant to the saturation of the color. From this encoding readers could rapidly perceive the location information about different restaurants, and restaurants with more reliable rating in the map,(number of reviews is a strong indicator to how reliable the average rating is)which is

the point with the bigger size and high saturation. This design gives readers a general information about the topic of our data visualization project.



#2.1 Map Version 1

Then we explored our data by drawing all the restaurants on a simple map (see Map Version 2). We then quickly realize that in some area restaurants are too dense that the dots are overlaid with each other. We immediately decided not to map the number of reviews to the size of dots anymore because it will make map too messy.

#2.2 Map Version 2

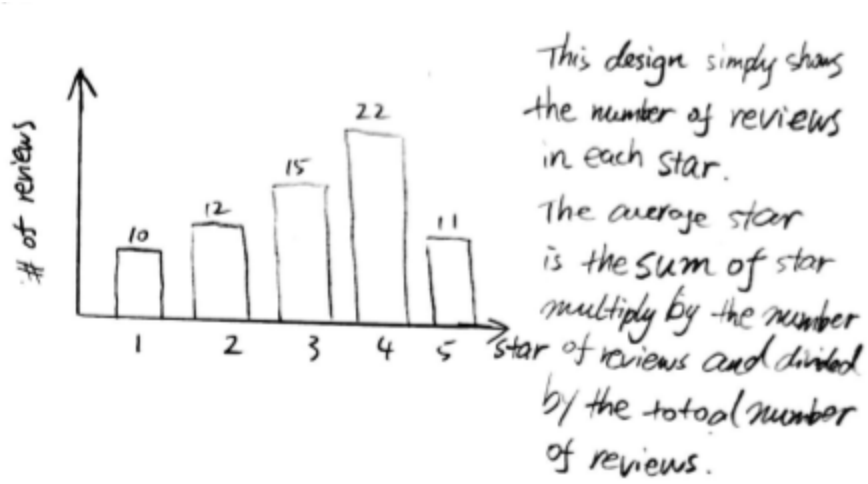
However, in some area dots are still heavily overlaid even with the same size, making it difficult to select the restaurant. Therefore to solve this issue, we decide to use scalable map so we can zoom in the map in some high density area. After research online, we found an open source javascript library called leaflet.js, which provide functionality to create an interactive map. We then plug this map into our design and draw dots on this map (see Map Version 3). As you can see we can zoom in to the street level and see every restaurant clearly. We made the dots to increase the size slightly when zoom in on the map so we can select the dot easily. We also made the dot to be transparent so when two dots overlaid with each other, you can still see them.

In our final design, when mouse over a restaurant, detailed information of the restaurant will show up on the right panel, as well as updates on treemap and histogram. The updates will disappear when mouse out. When click on a restaurant, the right panel will stay. It makes it easier for comparing restaurants.

#2.3 Map Version 3

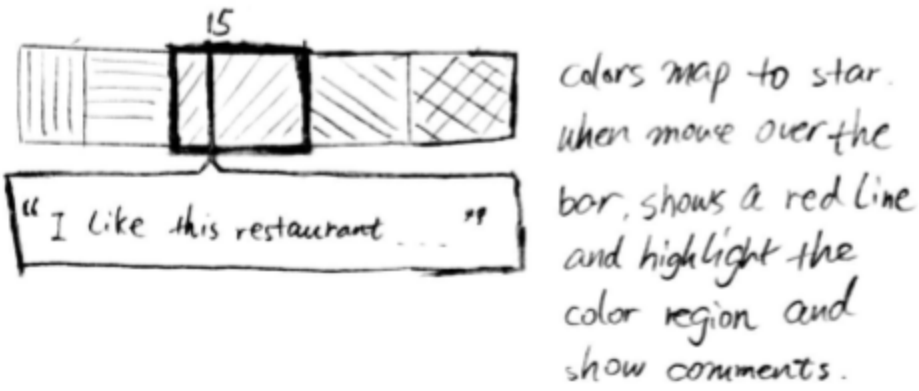
2. Number of Reviews Barchart

In Yelp, when users leave comments, they also need to give a rating from star 1 to 5. Based on this information, we designed a bar chart that shows a distribution of reviews over different ratings. When the user mouse over the restaurant on the map, the charts about this restaurant will show up on the right side of the map. One chart is showing the number of reviews that made for each star. The first version of design simply shows a bar chart as seen in graph #3.1. The x axis is the star and the y axis is the number of reviews.



#3.1 Number of reviews version 1

The second version of the design to show a single horizontal bar representing the total number of reviews for the restaurant. We then map the star to five colors on the bar. When mouse over the bar, the area where mouse is located will be highlighted and a number of reviews for this star will display on the top of the bar. When mouse over the bar, a vertical line will display on the bar and follow the mouse movement. A text box will show up below the vertical line with the actual review sentences.



#3.2 Number of reviews barchart Version 2

In the third version of the design, we borrowed the review designed by Amazon which shows the percentage of review distribution on each star. We think this is the best design because it puts all the ratings on the same level to compare and shows the percentage instead of number because users care more about which rating has the most reviews. We also show the total number of reviews on the top of the panel.

#3.3 Number of reviews Version 3

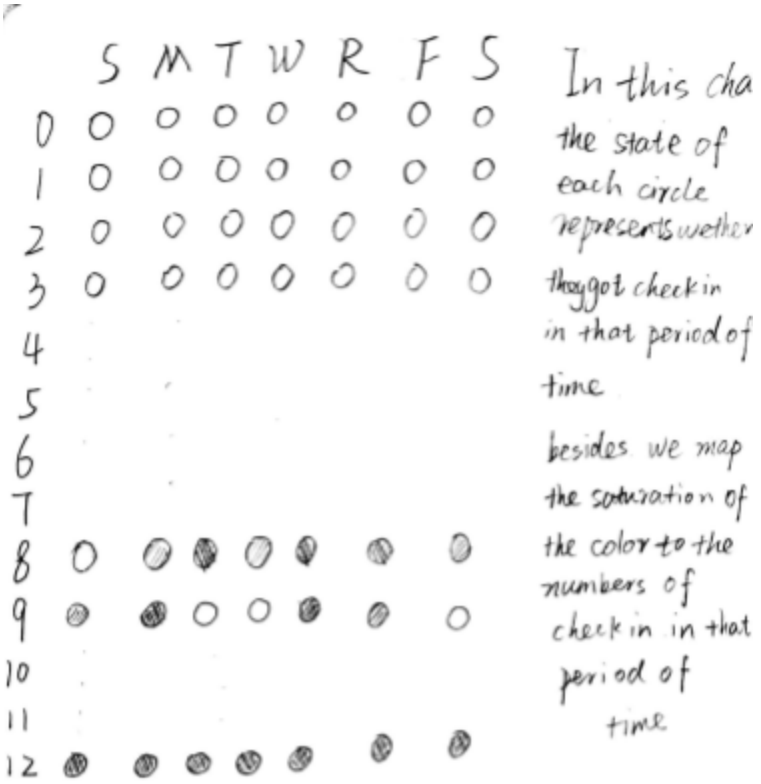
When user click on the blue bar of each rating, a second panel shows up with all the reviews in this rating level (see graph #3.4). When finish reading reviews, user can click on the review panel to hide the review.

#3.4 Review Panel

Check-in Heatmap

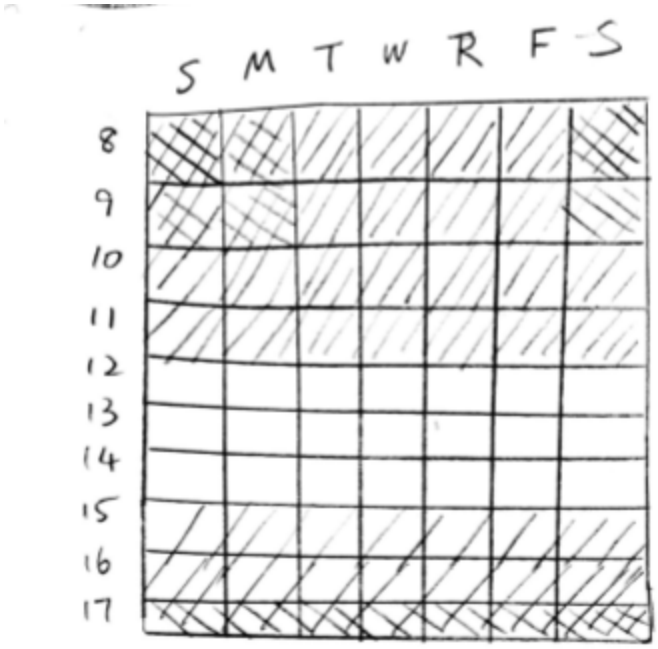
We also designed the check-in heatmap that shows the number of check-ins for each hour in a day and seven days a week. The check-in heatmap allows user to visually find out what day and time a restaurant is busy so they can maybe avoid that period of time. The first version of design, we use circles to represent each day and hour. The saturation of the color in each circle

represents how many check-ins in that period of time. The darker the color of the circle, the more check-ins in that hour.



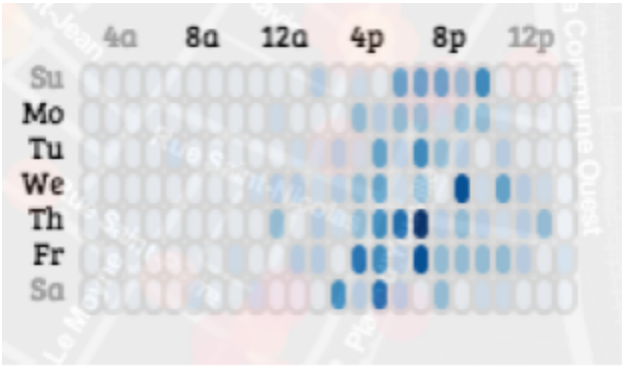
#4.1 Check-in Heatmap Version 1

In version 2 we use rectangle to represent the number of check-in in one hour period of time and the overall graph is a square with blocks of different color saturation. Using heatmap allows to visually show at what time in a day and in what day in a week this restaurant is busy. Use rectangle instead of circle can save us many space because rectangles can line up perfectly one by one.



#4.2 Check-in Heatmap Version 2

In our final design version, we rotate the table so the horizontal axis shows the time and vertical axis shows the days in a week. The saturation of blue color shows the number of check-ins in that period of time.

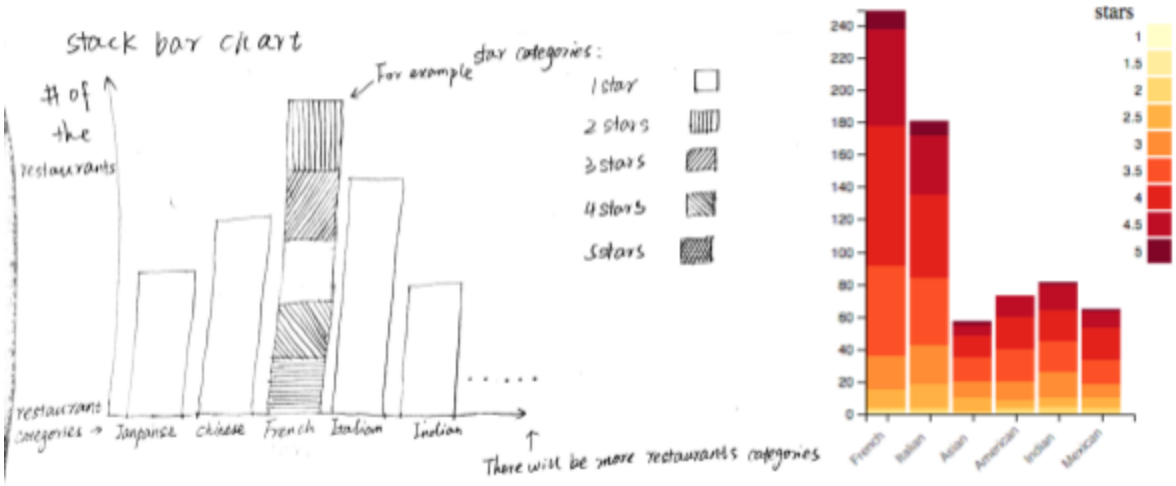


#4.3 Check-in Heatmap Version 3

Stacked Bar Chart (Not Used)

This is a stacked barchart (see graph #5.1) to show information about in different restaurant categories the distribution of the average stars. In this chart, we grouped the restaurants in

several categories,such as japanese, chinese, italian, french, indians and so on, then we map the average star to the color to show the star distribution of different restaurant categories.This chart is also an interactive chart with the map, when we click somewhere inside a bar,the related restaurants in that categories and with that average star will highlight in the map. Besides, it also interactive with the chart # 5.1, This design helps reader more easily find their desired restaurant. But eventually, we give up this design because we want to integrate other information in our design: the suggest price of the restaurant, so we chose the tree map (see graph #6.2 or #6.3) instead.



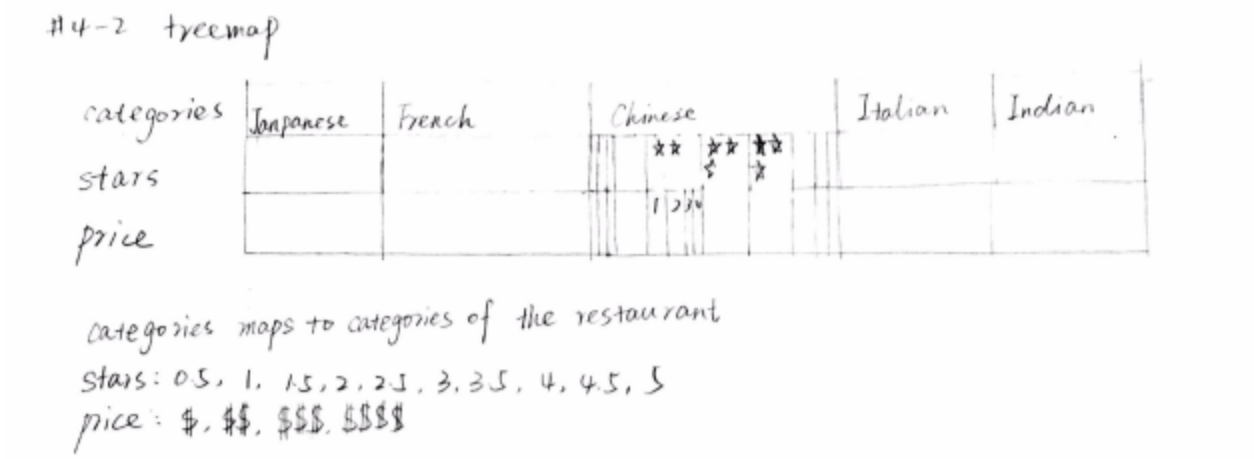
#5 Stack bar chart to show number of restaurants in different categories

Treemap

Below (Graph #6.1) is our initial treemap design. In this design, we encoded the categories of the restaurants in the first line, average stars in the second line and its suggested price in the third line.This design helps the reader get the information about the distribution of the price in a certain star category which in a certain restaurant category.This tree map also interactive with the map (graph#1), when we mouse over one point in the map, simultaneously the related category, star, price will highlight in the this treemap, and when we click somewhere in the treemap, the related restaurants will highlight in the map.

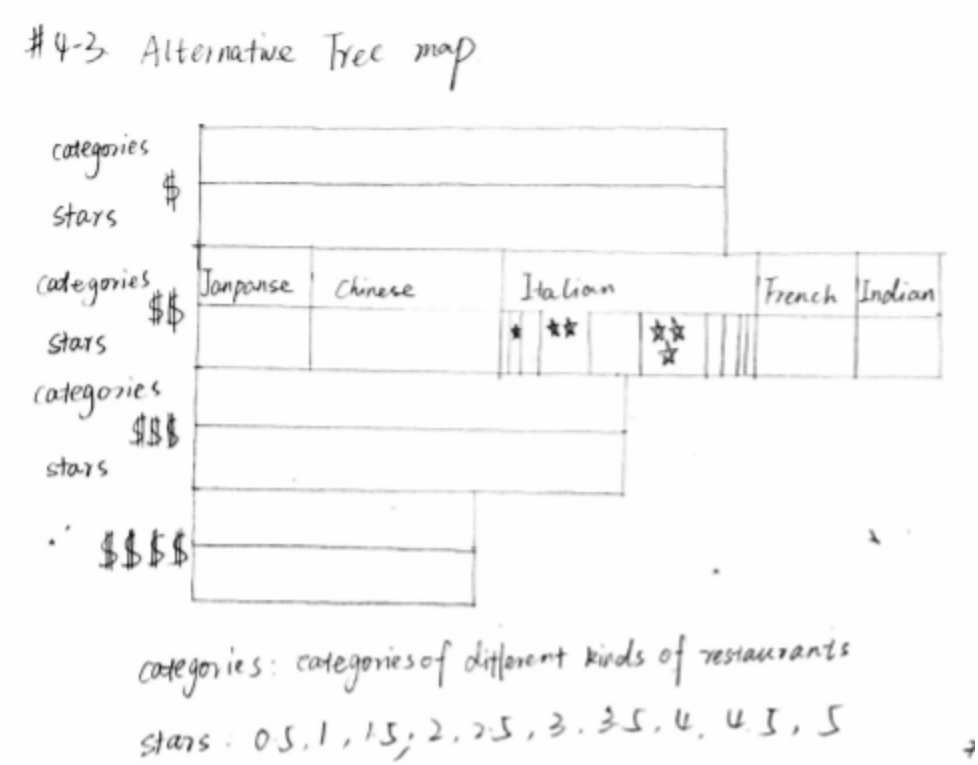
This map also interactive with the graph #7.1 and graph #7.2, when we click somewhere in the treemap, graph #7.2 will show us 10 restaurants with largest numbers of reviews in that category, and the way it interactive with graph #7.1 is when we click one point in the map,not only the treemap will show its related category information, graph #7.1 will also show its number of reviews distribution. This design makes the way reader choose restaurant easier, all kind of information is categorized, and from the distribution, reader could know more accurate information about the quality of this restaurant. For example, when user filter the chinese food, 4 star, two price sign, all the restaurants qualified for those filter will appear in the map, then user could click the one with largest number of reviews in graph #7.2, that restaurant will highlight in

the map. When the reader mouse over one point in the map, the related information for that restaurant will highlight in the treemap and also its distribution in the graph #7.1 will appear.



#6.1 Treemap Version 1

This (graph #6.2) is the 2 version of treemap we designed, the only difference between this one and last one is we take the suggested price out, so the there will only be 2 layer in the treemap,but there will be four tree maps. the reason we did this is because in before one （graph#6.1）, there are three layers, and in total, the bottom layer will be the numbers of the restaurant categories times categories of stars times price categories, that will be a huge number and we worried about the resolution will be not enough and the each single element will be too tiny in the last layer for reader to choose. So in this case, the number of the bottom layer will significantly decrease, but this graph will become bigger and hard to integrated in final design, now we are still thinking about which one to use.



#6.2 Alternative Treemap

Before we implement the our new treemap, we needs to build the json data with the right structure of treemap layout from our yelp dataset. We use the category hierarchy information on the yelp data api website to process our data and build the structure.
Here is the information of the category:
https://www.yelp.com/developers/documentation/v2/all_category_list
By using the category information from the yelp dataset, we could directly pass the data we got from the yelp api to our program and build the tree. So it will be suitable as long as yelp give us access to their data.

When we actually begin to implement the treemap, we felt the shape of all our treemap designs are not fit for our whole page design. For the first one #6.1, it needs a decent space in width due to its width is mapping to the numbers of restaurants fits the filters applied and it's very hard for us to compress it(some part in the treemap will be too small and can not be clicked). Another fact is the order of the filter applied is fixed and not flexible, so we discard it.
For the second one #6.2, its space wise, but the order of the filter is fixed too, we could do multiple choices in this treemap, but we then just feels it will be a lot of work to choose, since the category is too many.

Finally, we combined the treemap with rating filter, and come up with a new design as graph

#6.3 Implemented Treemap

In this design, the treemap is more space wise, and the slider bar will apply the rate filter on the treemap and the multiple choices dollar buttons will apply the price filter on the treemap. Those two filters also coordinate with restaurants showed in our map and the review histogram.
When applied the filters, the structure of the tree map will change into the graph shown below:

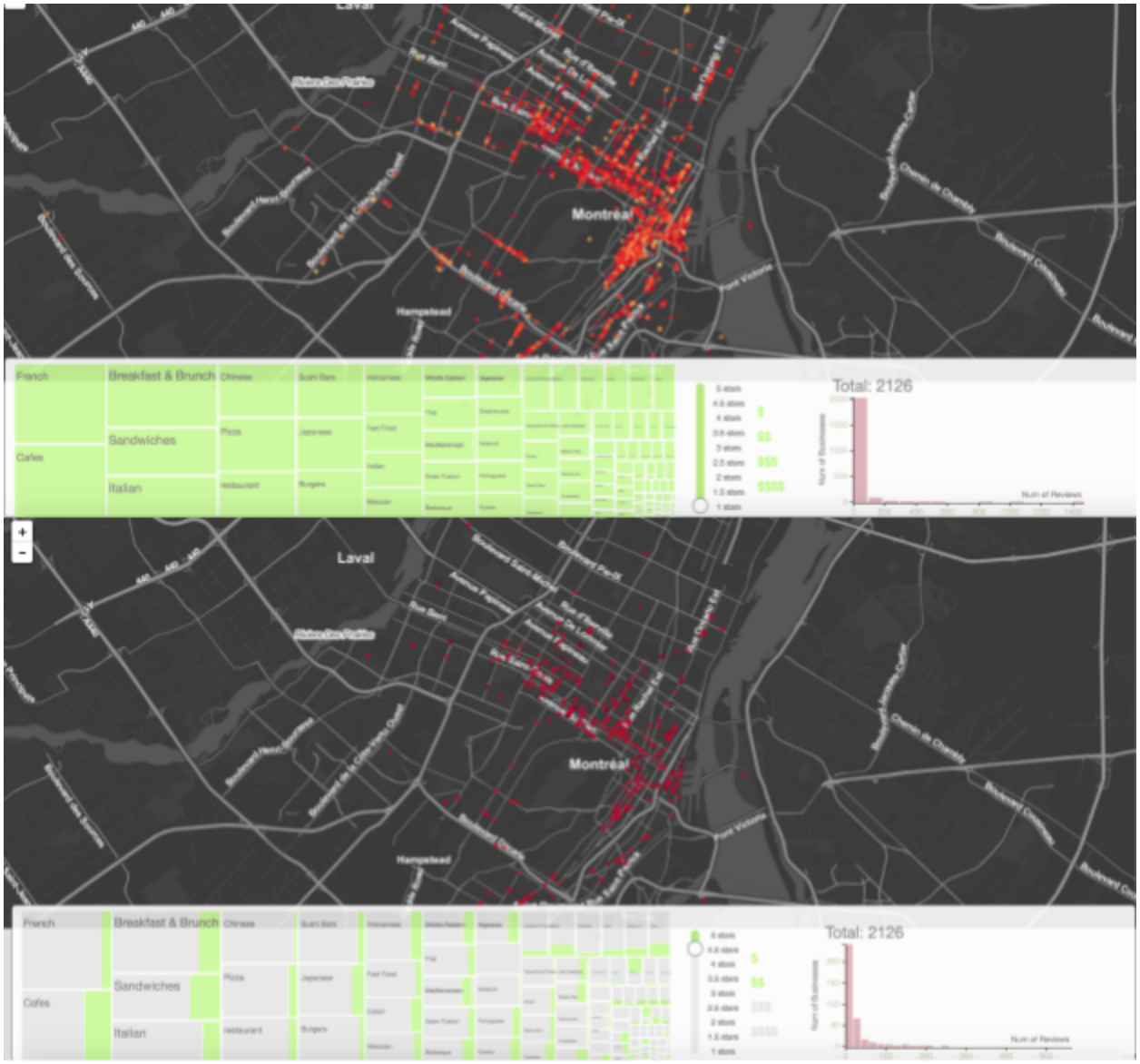
6.4 Implemented Treemap 2

The treemap above is applied the star filter, the one below is applied the price filter after apply the star filter.

#6.5 Implemented Treemap 5

In the graph above, you could see the green and grey rectangles in one category, the green rectangles represent the numbers of restaurants in that category which satisfy the filters, and the grey rectangles represent the numbers of restaurants doesn't satisfy the filters, this distribution information in one category could help user adjust their expectations, such as when they found out there are only very few or none restaurants when applied the five star filter and one dollar filter,they will begin to find something more realistic.

Like mentioned before, the treemap also coordinates with the map and the review histogram which means after applied the filters, the map and the histogram will also change accordingly.



Graph above is the before we applied the filter, and the graph below is when we applied the filter, as you can see, there is less points in the map and the histogram changes accordingly.

The element in the treemap also provide a click function for the user, when clicked the rectangle in the treemap, it will apply a category filter to the data, the restaurants on the map and the histogram will change accordingly. when you clicked the same rectangle again, the filter will be canceled.

As you can see in the graph above, when you clicked one category, the corresponding rect will be highlighted and a category filter will be applied, the points in the map and the histogram are only shown those filtered data. After this click event, user could either cancel this filter by click in the same category again or click other category, user could also change the rating slider bar and or multiple price buttons to filter the data more, the clicked effect will still be there to let user know the category filter is still there as you can see below.

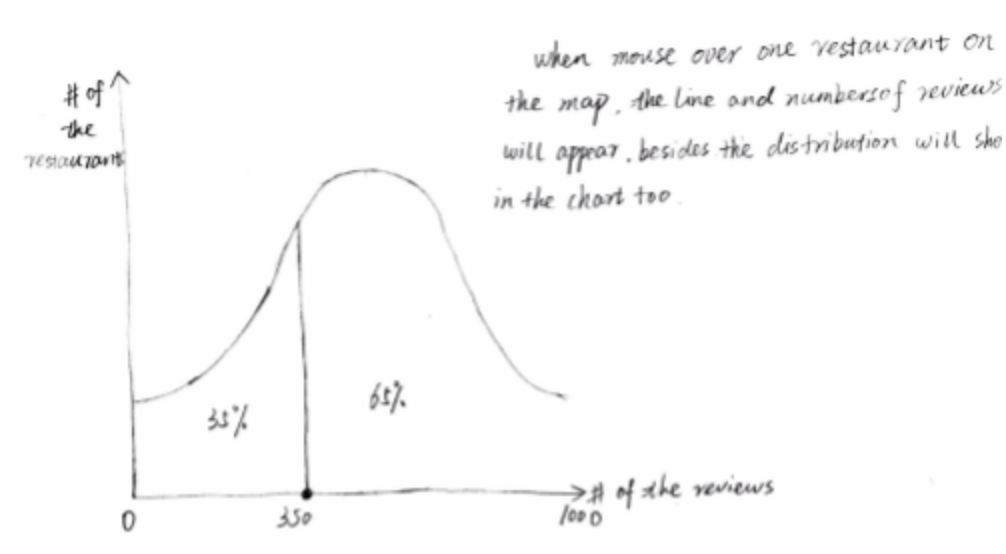
As we could see above the clicked function in the treemap also shows user a suggestion panel above the treemap, which lists the most popular restaurants within the filtered data. The suggestion board gives developer an opportunity to apply recommendation algorithms. For

example, we can suggest newly opened restaurants with high ratings but fewer reviews. Although user could customize their own filter using different filters we provided. However, sometimes users didn't want to customize their own, so we provide them a suggestion functionality which provide users the top 5 restaurants with most reviews, and the restaurants with high rate but low reviews.

Another thing about the treemap is the slider filter we used, at first it's all transparent, then we feel it's confusing for the user to know the different color in the treemap, so we add color to it to make the user intuitively understand the green means it's fits the filter. As you can see the comparison as graph below

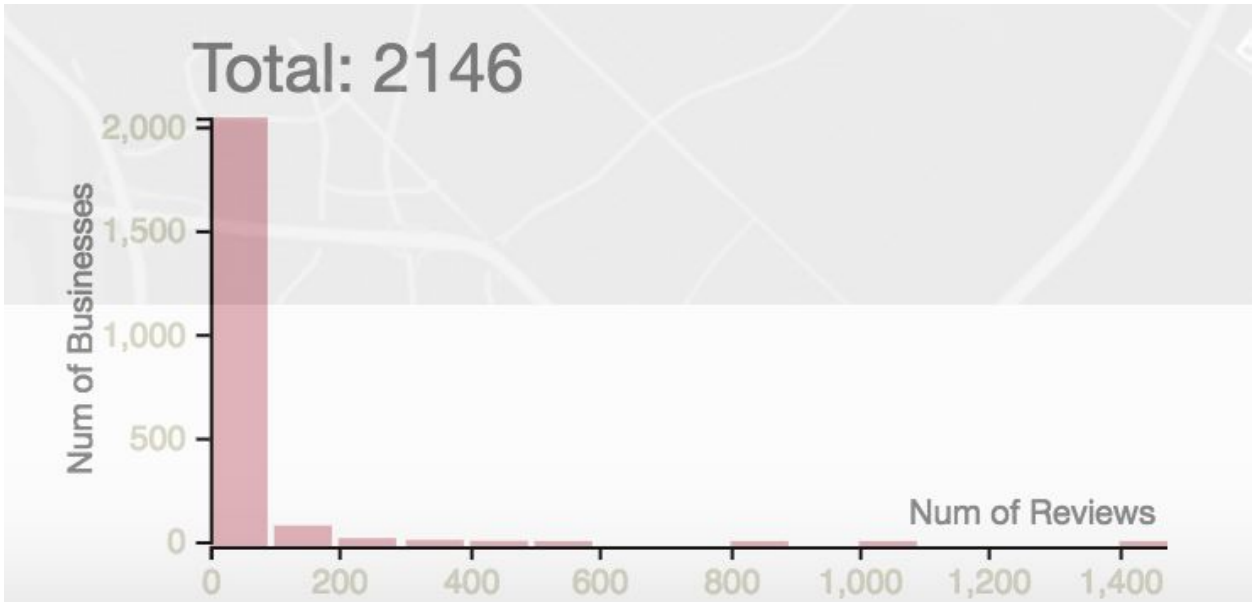
Histogram

This graph shows the distribution of the number of restaurants over the number of reviews. We believe the graph will show as a normal distribution. Therefore we will curve-fitting.

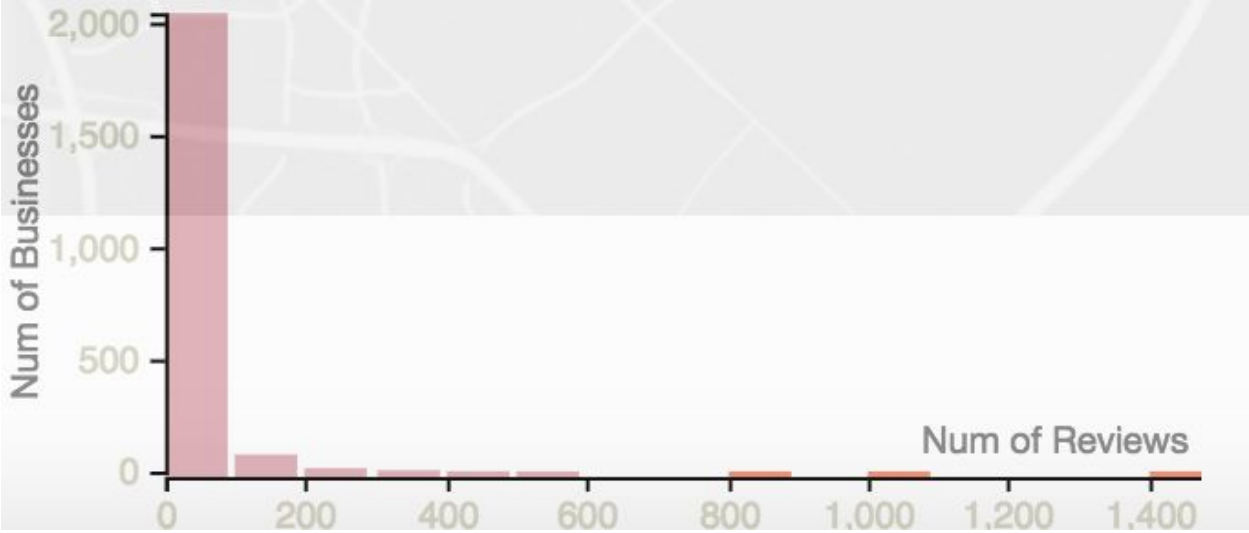


#7.1 Number of restaurants distribution over number of reviews

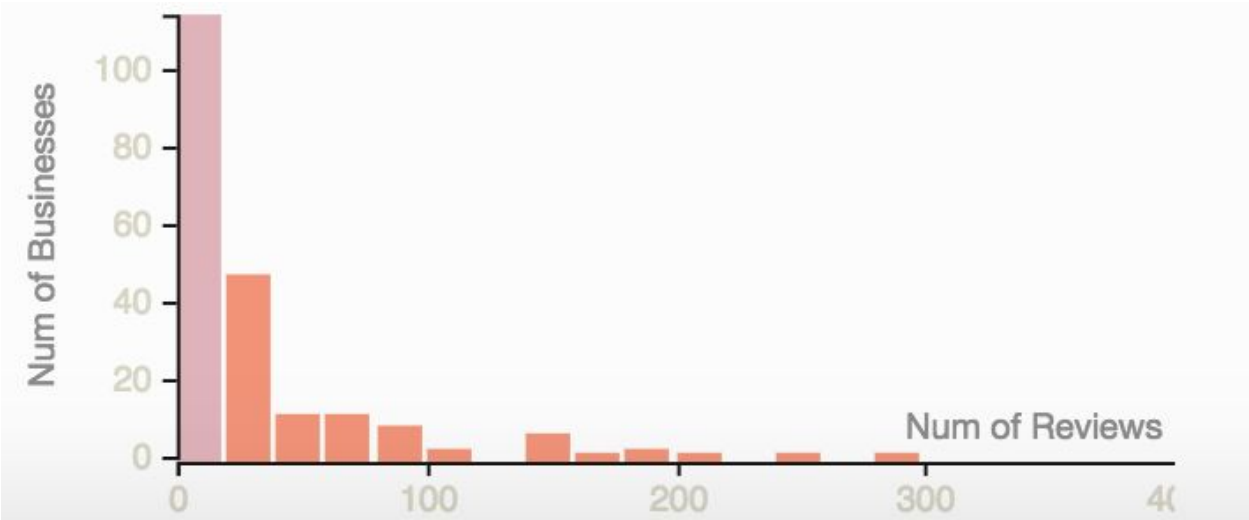
When we actually implement this functionality, we replaced it with a histogram, since the data is not normal distributed as we thought, most of the restaurants have very few reviews, so the curve-fitting will make most of the information in the left of the chart which will be hard to read and choose. This is the histogram we made instead as shown in the below graph.



As you can see, this is the total restaurants of the Montreal, and most of them has fewer than 100 reviews. It will be very hard for users to choose the bars with very few restaurants, so we add a minimal height to our bar which means even it's only one restaurant there, user could still select that bar, as you can see below the 3 small bars in the right are red.



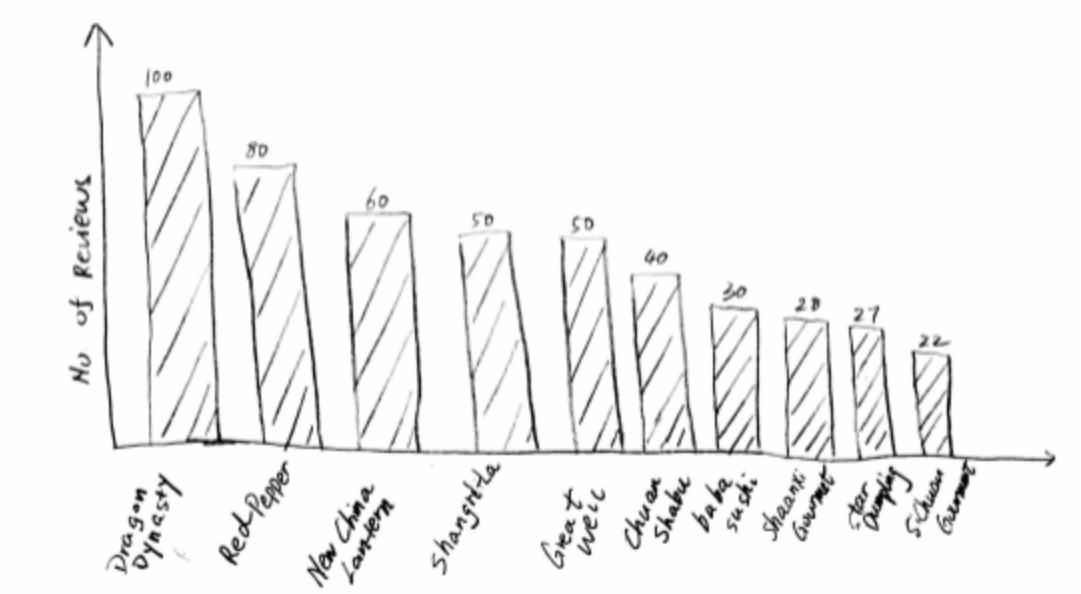
The histogram is also coordinated with the filters in the treemap and the map,the data shown in the histogram is filtered after the reset of the map and all other filters related with the treemap. The histogram itself also works as a filter, when user click one bar inside the review histogram, all the bars in the right side of the clicked bar will be highlighted and selected,which means all the restaurants (satisfy the other filters before) with higher review count than the selected review count bar were filtered out,the reason we did this is we think for review information, the more the better, nobody will specifically focus on restaurants with 500 review if there are restaurants satisfy the other filters but with 1000 reviews are there, as you could see the graph below. After applied the review count filter, the points in the map will updated too.



At first we plan to update the treemap when we apply the review filter, but then we realize that will create a circle event. The review bar chart will update the treemap, the treemap will update the review, then it will be too confusing, so we discard this design.

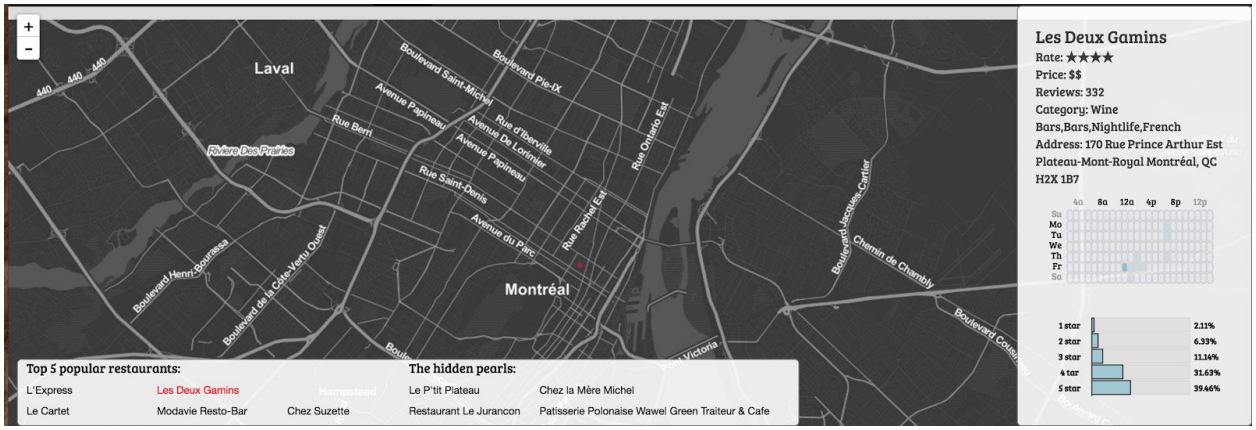
Suggestion panel and chart

This graph shows the list of 10 restaurant with the largest number of reviews after filtering in tree map. It interacts with the treemap and map as described in the treemap chart.



#7.2 List of first 10 business after filtering in Treemap

When we actually try to implemented this chart, we felt the review information is shown in the detail panel, so it's not necessary to draw a chart for it, besides the layout of our final design has no space for another bar chart. Then we discard this design and adopted the suggestion panel as I described before, as you can see in the below graph.

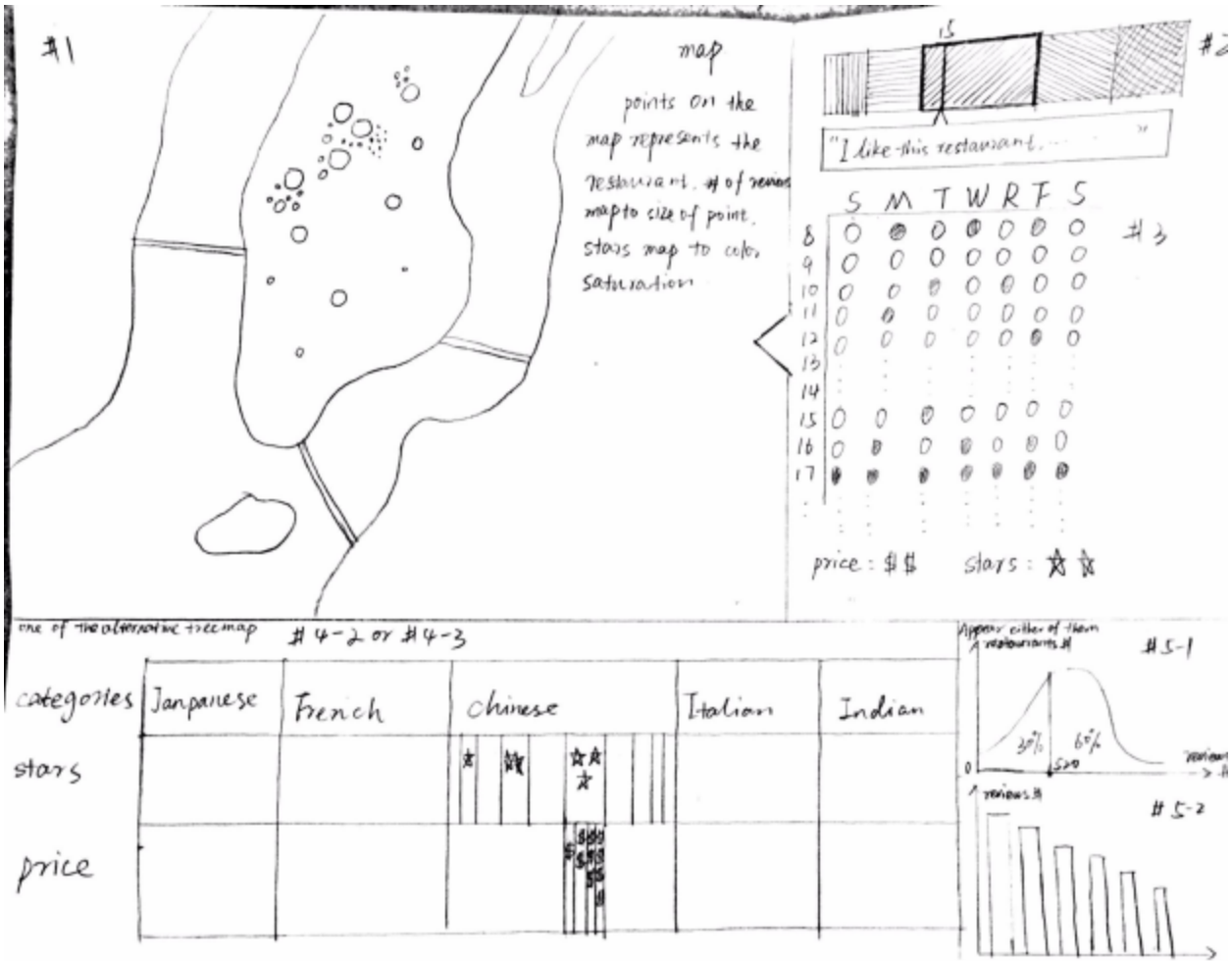


When clicked the element inside the treemap, a suggestion panel will be shown above the treemap. Inside the suggestion panel, there are at most 10 restaurants, five are the restaurants

with the most reviews after applied the customized filters, other 5 are the restaurants with low review but high rating which normally happens for those newly opened nice restaurants. Users could also click the restaurant in the suggestion panel, then the detailed information will be showed in the right panel. When user click the map or the same category rect again, the right panel will be hide and the suggestion panel will be hide too.

Final design

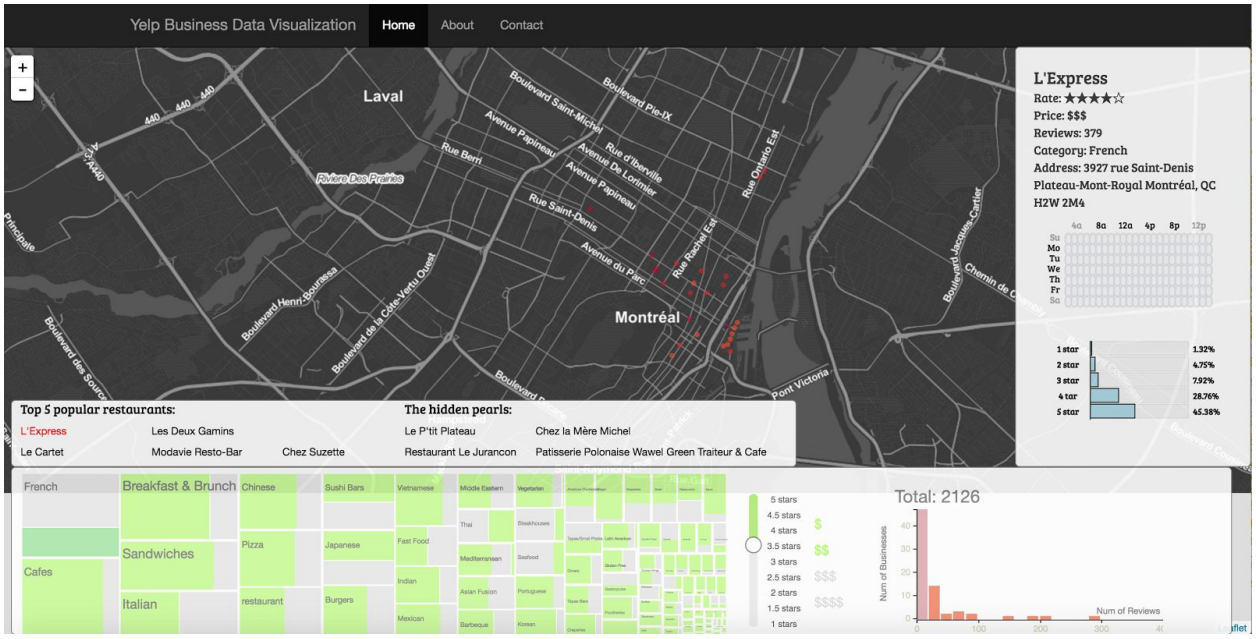
This is the comparison of our initial design and what we implemented, we implemented all the functionality as we designed, maybe the form was different,but we think its better.

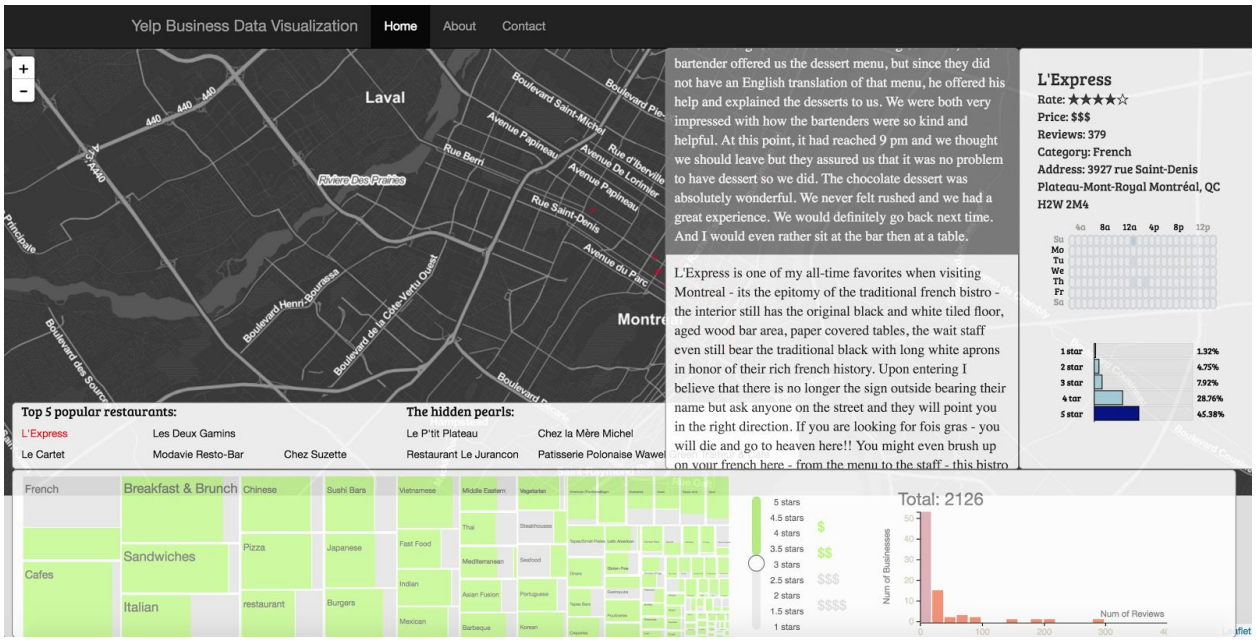


#6.1 Final Design Graph

The final design put together #1-7, and adds interactions among the graphs. Detailed interactions are listed in the must-have and optional features section. Above sketch shows the layout of the whole visualization. #1 is the main graph that have business location, average rating and popularity (reflected by number of reviews it get) encoded. So it occupies most area. #2 and #3 shows more specific information of one business if the user is interested. #4 and #5

provide more information of the whole picture. #4 shows the category and price information of that chosen business. It can also function as a filter to #1. #5-1 and #5-2 occupy the same space. #5-2 will show when #4 is used as a filter.





Graph above is our final result, all the interaction between different part of the graph was described in the previous section. And all those things combined leads to our final design.

Thank you for reading such a long process book!