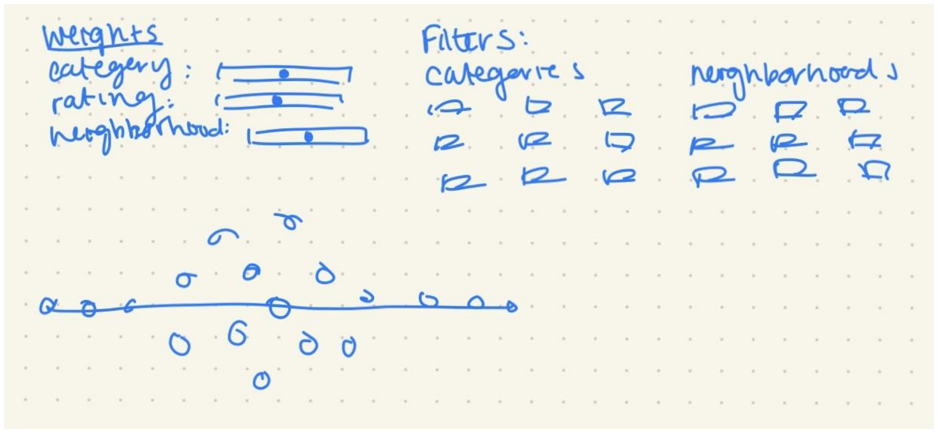
Olivia Zhu oz28, Connie Liu, cl2264
INFO 4310
Homework 3 Design Document

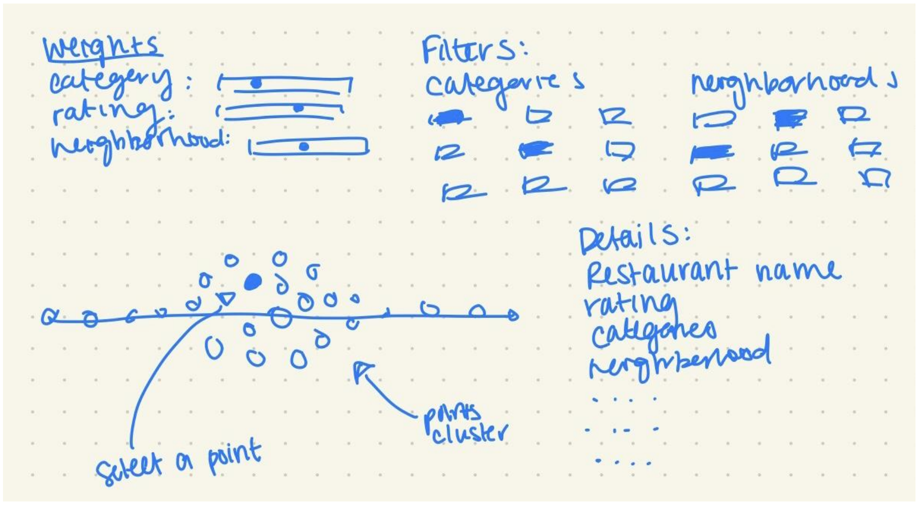
Data Description

The data we chose was the Yelp dataset of 343 restaurants in the Boston area. The user is seeking to find a restaurant to eat at. They are most likely new to Boston and don't have hard set preferences, or are a local who is seeking to try something new. As a result, the user would like to explore the different food possibilities aligned by category of food, optimal ratings, and neighborhood since Boston is a large area. These criteria for picking a restaurant are rather broad, giving the user the ability to browse multiple options and then further look into them by viewing their details or viewing them on Yelp. Also, users want to explore similar options and browse. With these needs, we were able to begin storyboarding.

Storyboards

For our initial storyboard, we chose to use a bubble chart on a single axis. We planned to implement a scoring metric to evaluate how close the restaurant is to the user's selected criteria. The reasoning for this is because each restaurant is multifaceted and it is difficult to show all of their attributes, therefore we decided to quantify them into a single measure that represents matchiness. The scoring metric is used to determine each point's position in the cluster. We decided on the bubble chart because it easily conveys which restaurants have the best scores, as well as restaurants with similar scores. It's likely there would only be a few restaurants truly matching the user's criteria, however the user might be interested in adjacent restaurants similar to their interests. The bubble chart also follows the visual metaphor that more important results would be in the center and less important results are in the periphery, which is intuitive to the user.





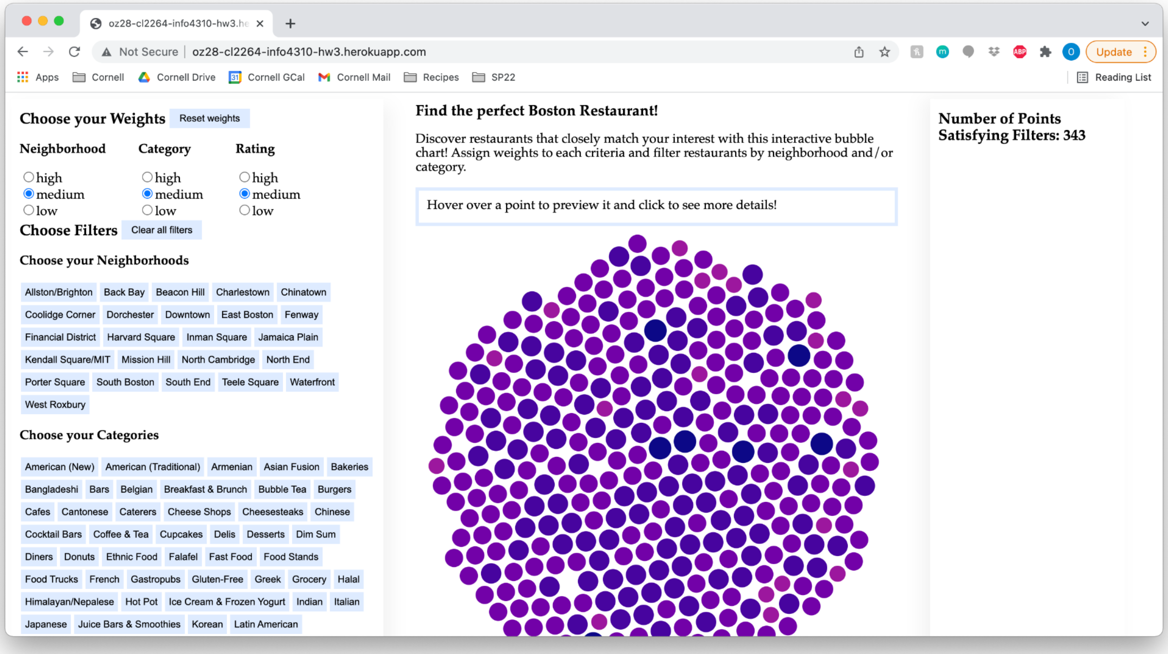
We then chose to have weights for the scoring metric so that users can show how strict certain preferences they have are. For example, users may weight rating more heavily as opposed to neighborhood or category to indicate higher importance. We used sliders to determine weight, as they provide an intuitive visual way to adjust weights, as a more full slider would indicate larger weight, while a less full slider indicates smaller weight.

Users can input their preferences using the filter options we chose: neighborhood and ranking. We chose these two properties as we felt that these were most relevant to choosing a restaurant among the attributes given in the dataset. We felt that the clicking of buttons to indicate preference is obvious and standard, therefore we used this to follow the user's intuition. These preferences will adjust the scoring metric we use to position points, therefore it will bring points that satisfy preferences closer to the middle and points that do not further to the periphery.

We chose to have a details panel as well because just having points would not provide the user with enough useful information on how to make a choice. Therefore we designed an interaction where users can hover over a point to show more details about it. By hovering over a point, users indicate their interest in it, therefore it makes sense to show them more details about it.

Overall, we decided to follow the overview, filter, details on demand framework, because we thought this was the best way to allow users to explore the large amount of data.

Final Interactive Visualization Application



Choose your Weights

Neighborhood	Category	Rating
high	high	high
[✓ medium]	[✓ medium]	[✓ medium]
low	low	low

Choose Filters

Choose your Neighborhoods

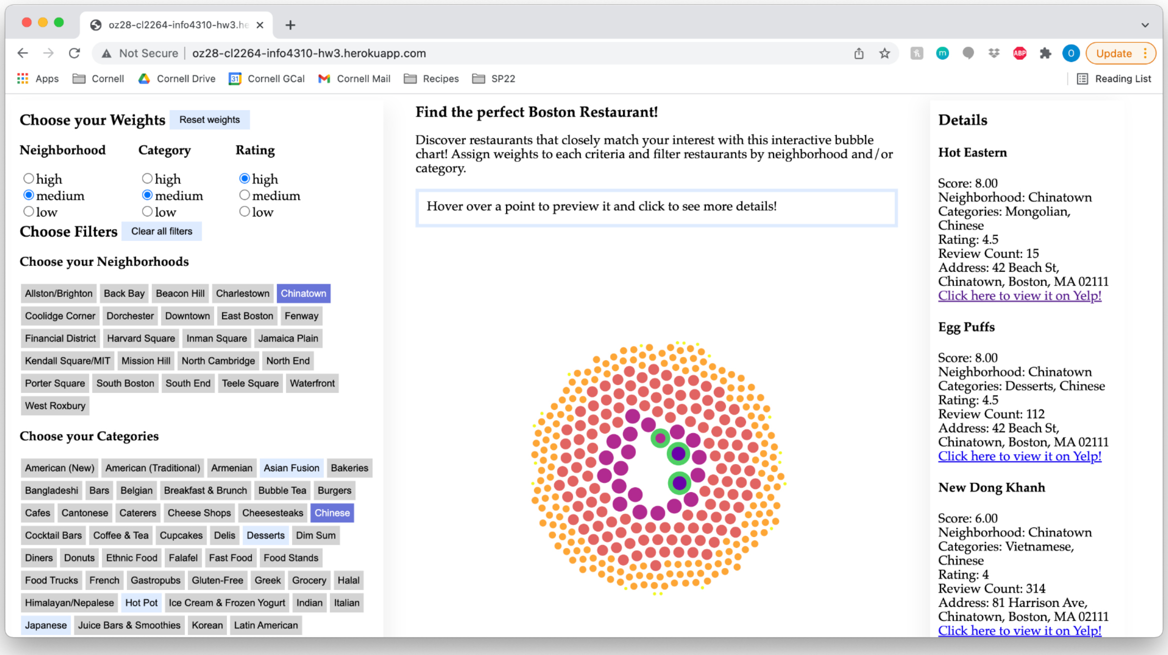
Choose your Categories

Find the perfect Boston Restaurant!

Discover restaurants that closely match your interest with this interactive bubble chart! Assign weights to each criteria and filter restaurants by neighborhood and/or category.

Hover over a point to preview it and click to see more details!

Number of Points Satisfying Filters: 343



Choose your Weights

Neighborhood	Category	Rating
high	high	[✓ high]
[✓ medium]	[✓ medium]	medium
low	low	low

Choose Filters

Choose your Neighborhoods: [✓ Chinatown]

Choose your Categories: [✓ Chinese]

Find the perfect Boston Restaurant!

Discover restaurants that closely match your interest with this interactive bubble chart! Assign weights to each criteria and filter restaurants by neighborhood and/or category.

Hover over a point to preview it and click to see more details!

Details

Hot Eastern

Score: 8.00
Neighborhood: Chinatown
Categories: Mongolian, Chinese
Rating: 4.5
Review Count: 15
Address: 42 Beach St, Chinatown, Boston, MA 02111
Click here to view it on Yelp!

Egg Puffs

Score: 8.00
Neighborhood: Chinatown
Categories: Desserts, Chinese
Rating: 4.5
Review Count: 112
Address: 42 Beach St, Chinatown, Boston, MA 02111
Click here to view it on Yelp!

New Dong Khanh

Score: 6.00
Neighborhood: Chinatown
Categories: Vietnamese, Chinese
Rating: 4
Review Count: 314
Address: 81 Harrison Ave, Chinatown, Boston, MA 02111
Click here to view it on Yelp!

The final interactive visualization builds off of the storyboards. It consists of a bubble chart, filters and weights for user preferences, and details for selected points. Initially, users are able to see an overview of all of the restaurants, although there is not much insight as they all have similar scores since there is no user input. Upon inputting weights and preferences, the user can see which restaurants best satisfy their criteria. They can then hover and preview points, and click to select and add to the details panel to better inform their decision.

The story we seek to tell is one of restaurant variety in Boston. As seen in the visualization, selecting several criteria rapidly limits the number of relevant points in the visualization. Also, when clicking on one criteria, more than several other possible buttons are grayed out, no longer being editable. The main goal of the visualization is to provide users with multiple restaurant options depending on their specific preferences with different degrees of relevance. As a result, users will be able to pinpoint and weigh different varieties and categories of restaurants according to their taste. Our visualization effectively helps these users explore Boston's rich restaurant scene, and decide and gain insight on a restaurant.

Issues & Tradeoffs

One of the tradeoffs we made was with the weighting system, as we allow users to choose any of the three options thus leading to the possibility of redundant choices, such as having all three be high at the same time which would result in no change. However, we chose to keep that there as it would be further confusing to the user if they were limited in their decisions and also it was hard to engineer.

Another tradeoff was the user interface of the filters, as we were not able to fit all the filters on one screen. Although this meant users had to scroll down to see all the categories, we felt that this was necessary as it would allow more space for the visualization itself and the details panel, two views that the user will find themselves interacting with more than the filter panel. Additionally, once the user chooses the weights, they can scroll down to see all the filters since they won't need to edit the weights.

Upon final examination and exploring our visualization we found that there are some duplicates in the data. This is because if a restaurant showed up in multiple search categories, it showed up multiple times in the raw data but under different search categories. As this was something we discovered late in our development process, we were not able to address it, but this is an area we would fix by filtering out the duplicate data.

We only chose to filter location by neighborhood instead of using a map because we felt that neighborhood category was general enough to allow users to browse within their range of interest. Because of the focus on the bubble chart, we didn't want to add a map view which would clutter up the visualization and make it more difficult to compare multiple points at once.

We chose not to allow users to select multiple neighborhoods because we saw single neighborhood exploration to be a more common use case than multi-neighborhood selection. If a user selects too many neighborhoods and categories, they then begin to include many more possibilities, making a lot of points score relatively high. This is not helpful to the user, because by including so many possibilities, it becomes more difficult to see which are the best in the visualization. Therefore, we confine users to selecting one neighborhood first, then selecting categories present in that neighborhood. This is a trade-off as some users may want to explore multiple neighborhoods at once, but as an alternative they could also restart their search with a new neighborhood and keep the previously selected points to compare. Also, allowing for multiple selection of neighborhoods was difficult to develop on our timeline, so this is an area we would improve in further.

The final tradeoff we'd like to address is that of using the bubble chart as our main visualization. Using the bubble chart is unconventional for this kind of visualization that helps consumers make decisions, but we feel that it is effective. Since it is not commonly used, the user requires more practice or accommodation, but we provided additional text and effective visual metaphors to help guide the user.

Development Process

Bubble Chart

We began with implementing our bubble chart. We tested out a number of forces from d3.force to figure out which one was best for our use case. We began with using x and y forces to create bubbles based on score, this ended up creating multiple bubbles along the score axis, but this was not the visual metaphor we wanted. Instead we wanted our bubbles to cluster towards a single center. Therefore we tested out the center and radial forces. We found that using the radial force worked best because we could better control how far a point should be from the center based on its score. We also included a collision force to prevent the points from overlapping. This collision force, of course, adds some variability to the points proper distance from the center, but this is an acceptable trade-off because the general position of the points is still distinguishable and useful, and it is more important that the user can distinguish relative positions of the points (i.e. which points are closer to the center than others) rather than actual distance. We initially were only going to use distance from the center as a channel for the score but also decided to use size and color hue to emphasize this factor more.

Relevance Score

We also began with implementing the relevance score as a weighted sum between rating, category matching, and neighborhood matching. We initially scored out of 5 points since the ratings are out of 5. For category matching, we calculated it as the proportion of selected categories a restaurant includes out of the total number of selected categories. For neighborhood matching, we did a binary metric of 1 if the restaurant was in the selected neighborhood and zero if not. After feedback from Professor Rzeszotarski, we used a normalized rating instead. We also made the score out of 10 points so that there is a larger distribution and the users can better understand differences between scores.

Filters

Next, we developed the filter buttons so that when users select a category or neighborhood, the scores for each point will adjust according to that preference. We do not actually filter the dataset, but rather we use the clustering to visually “filter” out the points with low scores by making them smaller, lightening in hue and saturation and pushing them towards the edge. This is a tradeoff because the points are not physically removed, but by drawing less attention towards them, we are effectively "removing" them from the user's gaze. We decided to color the buttons if they were being hovered over as well as if they were clicked. This helps the user identify where their mouse is as well as which filters they have already selected. We also decided to disable and grey out filters that when clicked on would result in no points matching all of the already selected filters. This helps us make a tightly coupled system that is more usable because the user will know which combinations of filters will produce an empty set.

Details Panel + Point Selection

We originally implemented the details panel and point selection with the intention that the user would select one point at a time. After implementing this, we thought that it would be useful for the user to be able to select and compare multiple restaurants, so we tripled the details panel such that the user can select three points at a time. Given this, we thought it would be nice for the user to preview the point before selecting it to compare to others. Initially, when the user hovered over a point it would show it's details. In our multi-select version, we implemented a mouseover such that when a user hovers, it'll show the name and score of the restaurant as a preview. If they clicked on it, it would add that restaurant's details to the details panel.

Weight Inputs

We initially implemented the weight inputs with sliders as discussed in our storyboarding. We were attempting to create linked percentage sliders as these would make sense to the user because the other sliders would adjust as they move one slider. These had implementation issues where the sliders were not adjusting properly and they caused negative percentages. Based on feedback from Professor Rzeszotarski, we decided to do high, medium, and low user inputs instead, therefore there is more abstraction to the user and we can calculate the weights on the backend. In our final weight inputs, we used radio buttons as input for each weight. Based on the relative inputs the user gives, we can then calculate weights that follow these expectations.

Styling

We put the filters on the left side panel as it would be the first area users would be interacting with. Then we had the visualization be the largest chunk and in the middle, in order to place larger emphasis on it and its changes. Then, we had details on the right side as that was the last step and what users would be examining least frequently. We chose green to be the point selection color as it was a color that did not appear anywhere else on the visualization and caused the highest contrast. We chose to have all the filters available as buttons instead of a drop-down so users would know the full variety of their options.

Work Breakdown

Connie

- Ideation (1hr)

- Weighting mathematics and filter buttons (3 hour)
- Calculating the common categories (1 hour)
- Sidebar details (2 hour)
- Office hour feedback (0.5 hr)
- Styling (2 hr)
- Writeup (2.5 hr)
- Total: 12hrs
- The part that took the longest time was figuring out the mathematics of how the button selected and weights interplayed with each other as there were many different cases to consider.

Olivia

- Ideation (1hr)
- Data-processing (0.5hr)
- Bubble chart (3hrs)
- Multi-select details + hover (2hrs)
- Weight inputs + configuring score (3hrs)
- Disabling filters (1.5hrs)
- Office hour feedback (0.5hrs)
- Setup heroku (0.5hrs)
- Writeup (2hrs)
- Total: 13hrs
- The weight inputs and configuring the score and creating the bubble chart took the longest because it took a lot of experimentation to get each of these parts perfect.