Estelle Hooper, Eva Ruse, Gabriella Chu
INFO 4310
HW3 Final Report

“Boston Bites” Consumer Visualization

Data Description & Exploration Needs

final csv: new_boston.csv
pre-processing code: boston_preproc.ipynb
topojson of Boston: output-6.json

We chose to focus on the “yelp_boston.csv” that has information from Yelp about restaurants in the Boston area. This interactive visualization is meant for users that want to find a place to eat in the Boston metropolitan area by providing a way to explore establishments spatially on a map as well as providing information like rating and address for specific places.

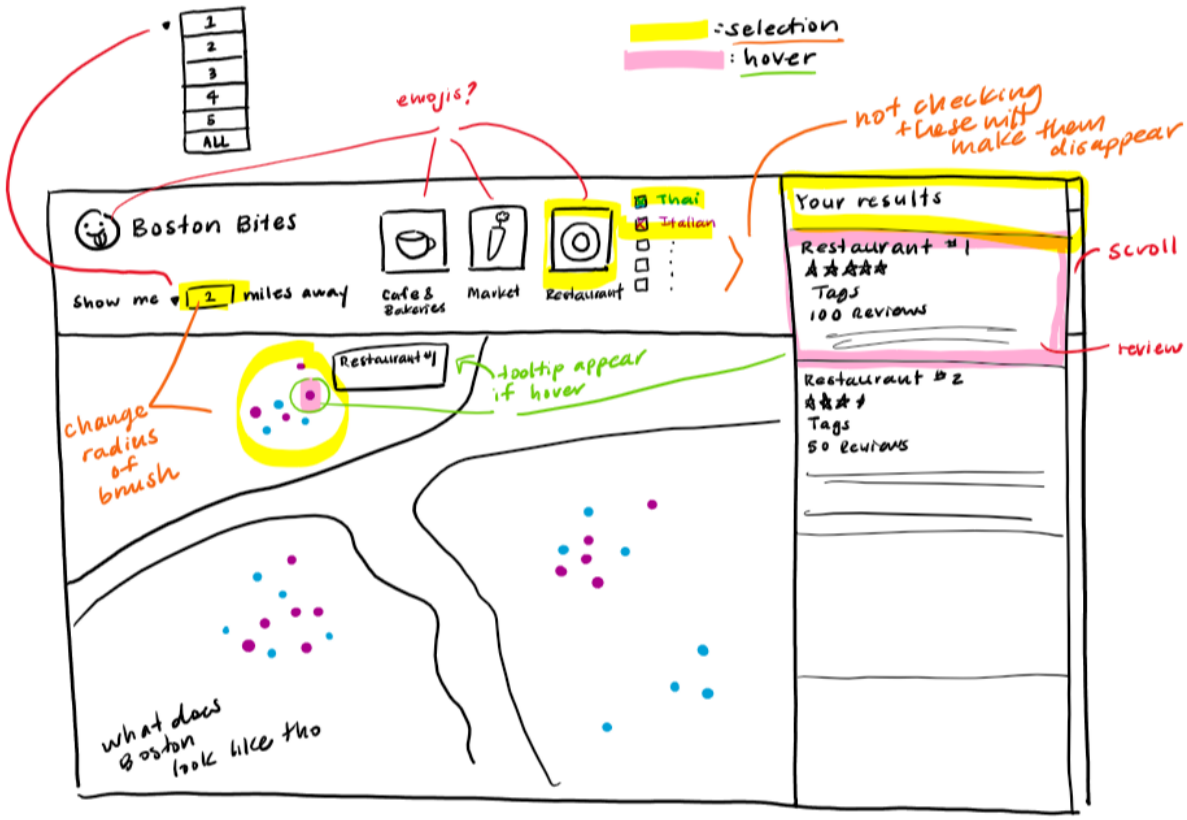

Figure 1. Initial Sketch

Our initial sketch (fig. 1) provides several interactions given the dataset, including a zoomable and pannable map to locate restaurants spatially, buttons to filter for different kinds of

establishments and different kinds of food, and a scrollable list of places to eat based on the filtered results. There is also a draggable brush available on the map for users to filter restaurants based on distance. The user would be able to click the circular brush in order to place it somewhere on the map and drag it around. The establishments that are contained within the radius of the brush would appear on the side in the form of a list. There is a button that the user can use in order to change the size of the radius of the brush, mapping to different distances across the map. We also designed an on-hover element for the list of places, where a tooltip would appear on the map next to the circle corresponding to the restaurant on the list containing the name of the establishment. The circle that represents the place would also change in size and opacity.

We decided to use a **map** interaction because we thought this would be the most intuitive way to display the restaurants to the user. We thought about what the overview should be for the users of the interaction and decided that the target audience should be presented with all of their options at first glance, and the user may be interested in knowing where most types of food establishments and cuisines are concentrated. As a result, we decided to plot all of the points in the Boston metro area upon loading. The map is **zoomable and pannable** to allow users to inspect a certain area further.

To ***initially filter*** the data, users have the option to use several **buttons**. We split the data points into four main categories—markets, cafes and bakeries, desserts and drinks (this category is not in the sketch but in our final vis), and restaurants. Since these are the main categories, we also wanted to use buttons (rather than a dropdown or radio buttons) so we could include icons/emojis that make sense with the label but also draw attention to these initial filters.

For these buttons, because there are only four categories, we decided to only allow the user to select one at a time; if the user selects restaurants, they can not also select cafes and bakeries. However, for the categories that had a substantial amount of points, like desserts & drinks and restaurants, we included a ***secondary filter*** in the form of **checkboxes**. Here, the user is able to select more than one option. This is because we thought that if a user wanted to find a restaurant, they would most likely want the option to browse through multiple types of restaurants, yet have the option to limit the ones they wanted in particular or did not want in particular. However, we thought that if a user wanted to find a restaurant, they would most likely not also be looking for a market, and thus did not give the option to choose both of those.

There is also a **dropdown** to ***change the radius*** of the brush on the map, which correspond to walkable distances. We used a dropdown because we wanted the selection to look different from the existing buttons and checkmark selections. Additionally, from the top to bottom, the options for the distance increase, which seemed intuitive. The dropdown also only changes one element on a vis, the brush, so presenting all distance options at the same time seemed unnecessary compared to a task like filtering al the dots on a map.

We chose to use a **circle brush** on the map because we wanted something to represent radial distance. Since circle brushes are not common, placing it can feel less accurate, so we also included drag. Points in the brush dynamically update a results list. Furthermore, the results have an on-hover interaction where the user is able to see the ***corresponding place*** on the map and its name with a **tooltip**.

These interactions follow Ben Shneiderman's general VIS mantra. Overview first with the map, zoom and filter with the buttons and checkboxes, and the details-on-demand with the circle brush, tooltip, and dynamically populating list.

Final Visualization Application

Our final visualization is centered around the map component. The map focuses on Boston metropolitan area. Users can filter the data on display in the map by whether they are looking for a market, cafe & bakeries, desserts & drinks, or restaurant. Of these filters, desserts & drinks and restaurants can be filtered further. Users also select a distance they are willing to travel (within walking distance), creating a circle brush they can move around once they click on the map. Locations within the range they are willing to travel populates into a table on the right under "Results".

Marks: dots representing food locations in Boston, paths representing streets in Boston, polygons representing land area and shape of Boston.

Channels: hue corresponding to button colors, horizontal aligned position, vertical aligned position

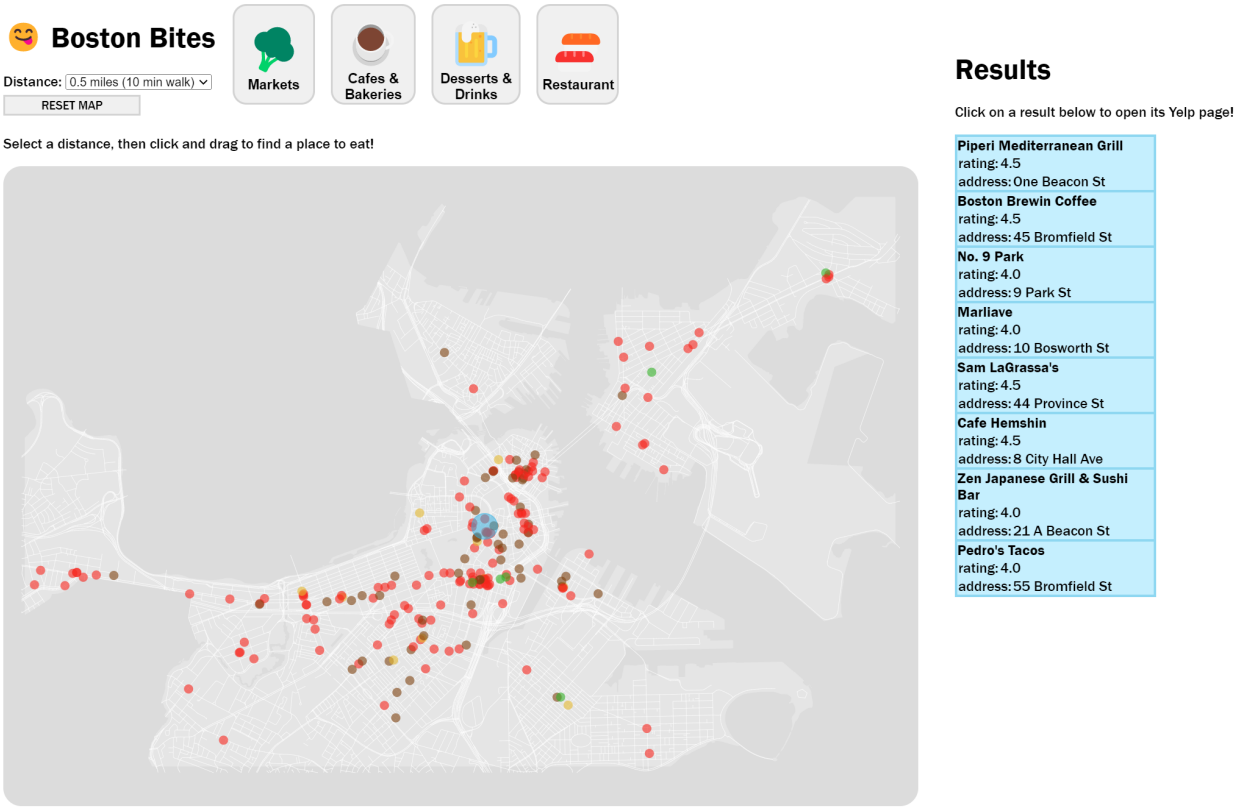

Figure 2. Final Visualization. All data points with brush.

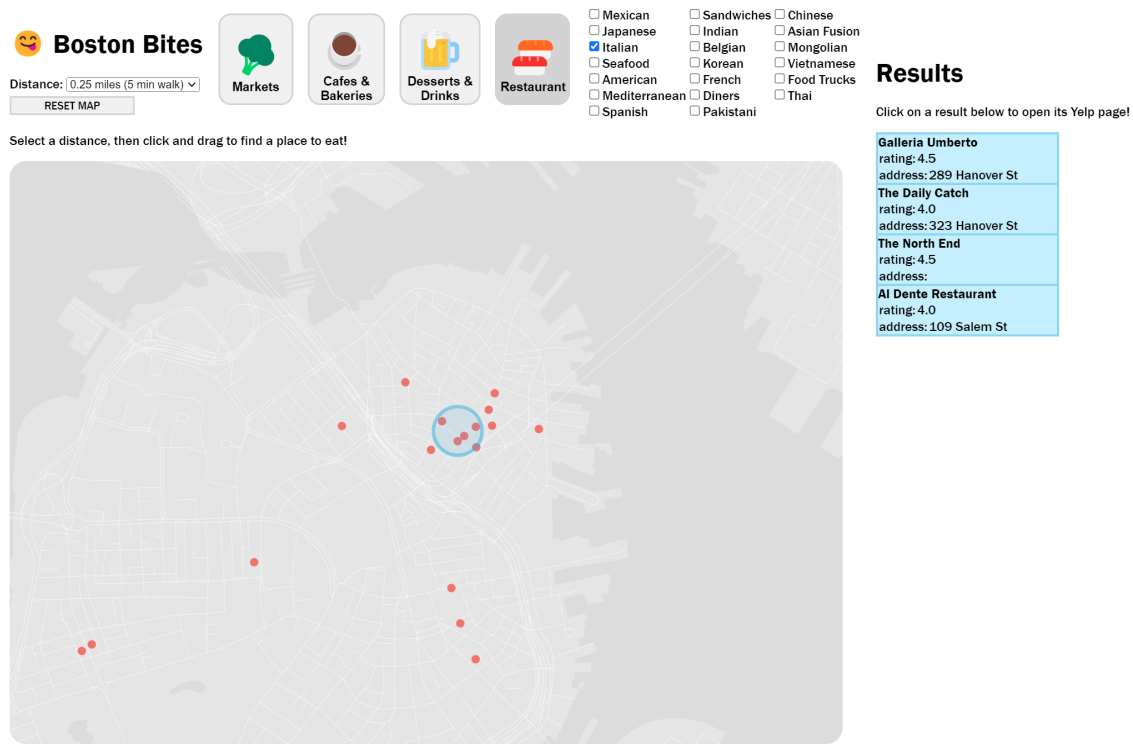

Figure 3. Restaurant filter and Italian subfilter selected. Dots scale with zoom and Brush gets more transparent with zoom as well.

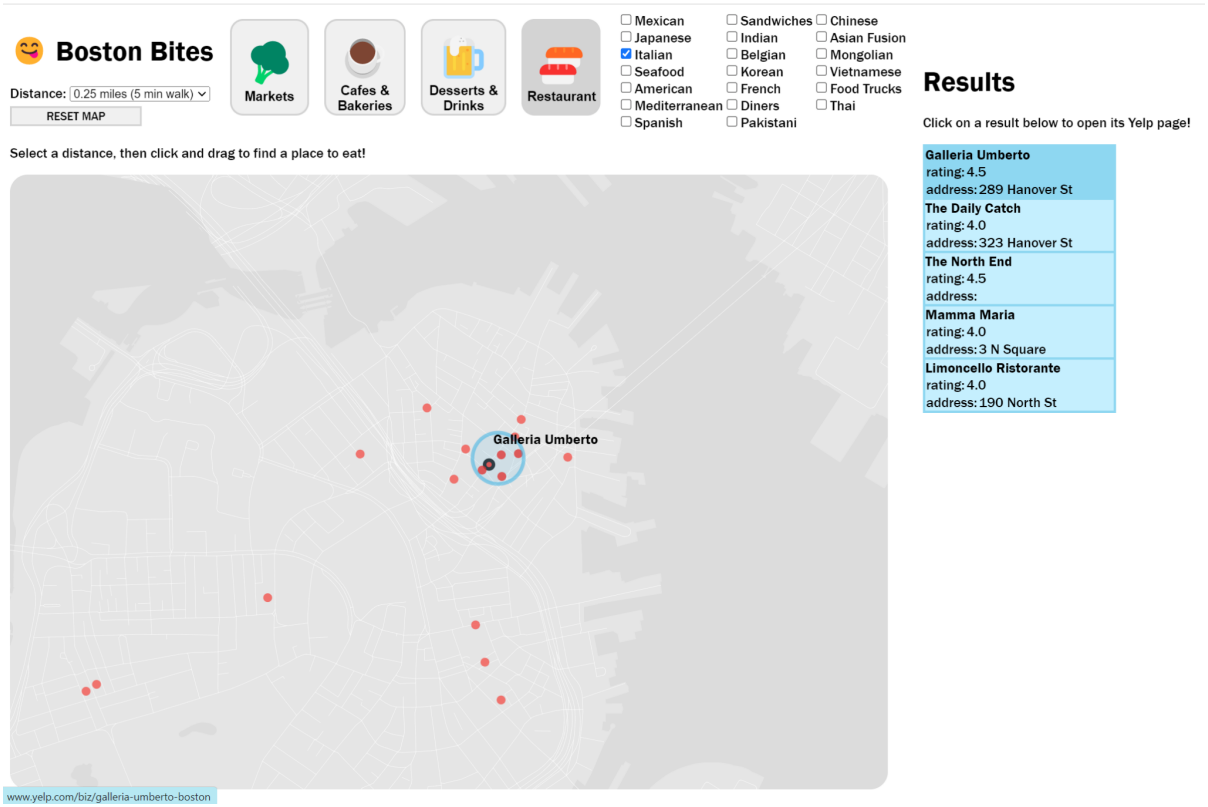

Figure 4. Hovering on a result provides a tooltip to the corresponding point.

Tradeoffs:

Yelp data preprocessing/Filter categories

We decided on the filter categories based on the “categories” and “search categories” column of the original dataset. The 4 main categories we chose would have been 6 by separating desserts and drinks or cafes and bakeries. However, we thought it was fine to be concise with these 4 categories because we noticed that all the categories that were not “restaurant” had few data points anyway, and it would be easy for the user to explore with zooming and brushing. For the checkboxes to cover more specific categories, we only used the first item in the “categories” column that contained a list of many categories. For restaurants, there were 26 unique category values. To avoid making that many checkboxes, we combined some of the cuisines, like Japanese = [Japanese, sushi, ramen] and Italian = [Italian, pizza]. Even though we know this generalizing the categories, we still finished with 20 checkboxes. Combining the filters for some of the cuisines seemed fine because many of these categories have few restaurants.

Topojson preprocessing/Removing Boston’s surrounding area

One tradeoff is that despite having a dataset that included points in Boston and the surrounding area, we chose to drop any points that were outside of the Boston metropolitan area in order to improve how the map looked for users. Before dropping these points, our map was a very zoomed out version of Boston and the surrounding area. Since we wanted users to be able to

look at locations within walking distances, our circle brush was really small on the map. We dropped points, thus losing data, in order to improve user experience and scope.

User knowledge of Boston geography
Another tradeoff is that we rely on the users to pick their current locations or areas of interest. This might not be the best design choice for users who haven't been to Boston before and aren't familiar with the geography and where they will be staying. However, after picking a location of interest, we provide relative distances with the brush to eventually provide feedback on the restaurant's proximity to them.

Clicking on specific dots
An issue with the interactions we chose might be that a user may want to click on a specific point on the map and see details that way. Unfortunately, this conflicted with how our brush works. We chose not to allow this and instead to have the corresponding circle radius increase when a user moused over the location in the results list. We thought this tradeoff was fine because our visualization emphasizes exploration, and the tooltip from the results list supplements finding specific points.

Development Process

We started by implementing the map because it is the focal point of our visualization. It provides an overview of the data as well as several interactive elements. Our initial design included the review snippets in the table under results, however, we ended up only including the location name, the rating, and the address. We made this decision because including the review snippet made the rows of the table too cluttered. After examining the data, we found that the review snippets were usually positive even if a rating was low; we speculated this is because people tend to leave really positive reviews with a positive rating, but no reviews with a negative rating. Instead of including the review directly on the table, for the final implementation, we decided to instead link the yelp page for the restaurant to our table to give users the option of finding out more information if they wanted to. This gives the user the ability to seek out additional information without overcrowding the existing visualization.

Design Choices

- **Brush and drag:**
    - We were concerned about if the user knew that they could drag the brush. Upon hover, the cursor changes to the **"move" cursor**. Though we know many people don't read instructions on visualizations, we also include text at the top to indicate that it can be dragged.
- **Brush and zoom:**
    - We keep the brush at high opacity when the map is small so that the user can identify it easily through **preattentive processing.** Once the user zooms in on a brushed area, the brush gradually becomes more transparent. This is because zooming is an action of **smooth pursuit,** in which the user is focused on

following the brushed area. We make the brush more transparent so that the user can easily see the dots in it, and has passed the phase of preattentive processing. Because the visual system is attuned to fine discrimination and detecting edges, we find making the brush more transparent necessary once the eye switches from its pursuit of the brush to the points in it while zooming in. We leave the border of the brush so that the user will still easily know the bounds of their original selection if the brush becomes too transparent.

- **Color:**
    - corresponding data:
        - We use the colors from the buttons (emojis) for the color scale for the dots on the map to build semantic meaning.
        - We make the brush and the results list blue to indicate the relationship between the two because all results displayed were generated because they are the points in the brush.
    - red-green colorblindness **tradeoff**:
        - We realize that the red and green used for the dots on the map are similar shades and that they are not differentiable for people with that type of colorblindness. We decided to keep these colors because of their semantic meaning, and believe that this is fine because the user can filter on those categories using the buttons. However, it would be difficult for someone to understand the distribution of the data when looking at all the categories at once.

Work Breakdown

Everyone
- Brainstorming/planning: 2 hr

Estelle
- filters buttons: 3 hr
- data cleaning: 1 hr
- filtering data: 2 hr
- styling: 3 hr
- write-up:1 hr

Eva
- map: 2 hrs
- Circle brush: 4 hrs
- Linking lists and points: 2 hrs
- Write-up: 1 hr

Gaby
- map: 2 hr
- Data cleaning: 2 hr

- Table/tooltip: 1.5 hr
- Write-up: 1 hr

Work Citations

Prof. Rzeszotarski Lectures

- INFO 3300
    - 10-5 Choropleth maps, for plotting a map
- INFO 4310
    - 2-13 SPLOM + Responsiveness, for brushing

Boston Street Segments geojson: https://data.boston.gov/dataset/boston-street-segments

City of Boston Boundary geojson: https://data.boston.gov/dataset/city-of-boston-boundary

Boston neighborhoods geojson: https://bostonopendata-boston.opendata.arcgis.com/datasets/3525b0ee6e6b427f9aab5d0a1d0a1a28/explore?location=42.312358%2C-71.056800%2C12.77

Mapshaper: https://mapshaper.org/

Brush: https://d3-graph-gallery.com/graph/interactivity_brush.html

Checkboxes: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/checkbox

- with a data join: https://stackoverflow.com/questions/52598074/d3-checkbox-label-input-order