Process Book
Yelp Data Challenge

Marissa Stanvick and Maxwell Lloyd

CS171 Final Project

Spring 2015

1) Table of Contents

2) Visualization Idea Evolution:

2.1) Visualization Idea 1: Visualization of Ice and Fire

Our initial design proposal and sketches (Appendix 8.1) detailed a “Game of Thrones” based visualization. The Game of Thrones books and television shows are a complex story following dozens of characters in tens of locations. The complexity lends itself to confusion for the readers and viewers. To help clarify where a character spends their time, we aimed to create a map based visualization. One could select characters and move through time in the book or television show to see where they moved and how much of their scenes were spent in a specific location. The user could select more than one character to compare the path each has taken. The initial proposal in Appendix 1 has more detail on the initial visualization.

However, based on the feedback from our TF, we decided to move to a different plan. The major problem for this proposal was that the data would be difficult to acquire. Neither Max nor I have experience with Python or data scraping. Additionally the format of data isn’t straightforward to obtain. For example, just because a location is mentioned in a recap, does not mean any of the characters were actually in the location. We tried to contact the creator of http://quartermaester.info/ to find out more about their data set, but we were not successful. Without data, our project would not have made it far! Using a fictional map would have also added a an extra layer of complexity to the mapping especially because we could not utilize D3 built in mapping functionalities. Because of these difficulties, we generated a new idea.

2.2) Visualization Idea 2: Yelp Data Challenge

For our new project proposal (Appendix 8.2), we focused on an idea with easily accessible data. From Section 7, we learned of the Yelp academic data set. Using this data set we wanted to generate an interactive map that would help users decide where to eat. You could filter the restaurants based on location. Within a specific location, the restaurants could be filtered on restaurant category. Each restaurant would be represented on the map by symbols that could be colored by # of reviews or # of stars. Finally, if the user clicks restaurants, they could compare them based on the aggregated review data in a Word Cloud. This proposal can be found in Appendix 8.2 for more detail. Based on feedback from the TF, we amended our idea to be less application based and more data focused. These ideas will be explored in the rest of the process book.

3) Visualization Overview and Motivation

Yelp is a popular website used to rate and review business of all types. The website stores a very large amount of data ranging from business information to review details to user profiles. All of this data could be explored together to garner insights and trends about businesses.

4) Data & Data Processing

Our data set is from https://www.yelp.com/academic_dataset. Yelp provided data for academic purposes, which is a compilation of business, review, and user data for businesses near 30 different schools. The data set includes businesses near Harvard University and MIT. The data is provided as a tar file which contained all of the business, user, and review objects in a single json file. We then separated the business, user, and review data into three files for easier processing, both by us, and chrome. A significant amount of further data processing was implemented:

4.1) Businesses

The initial businesses file was filtered to include on businesses associated with Harvard and MIT. This data set included approximately 1000 businesses.

4.2) Business Data Categories

Each business initially contained an array of multiple categories. In our visualization plan, we hoped to color code based on the category. This would be problematic because 1) each business included more than on category and 2) there were nearly 400 unique categories in the data set. It would be unreasonable to generate that many different colors. To solve this problem, every category from the entire data set was sent to an array. Each category was then counted for frequency. We then analyzed the top 20 most frequent categories. Overlapping categories such as restaurants and food or shopping and fashion were eliminated. We then updated the data set to only include one category if any of the categories matches one of the top 9 categories. If the business did not include one of the top nine categories, it was given a category of other.

4.3) Review Data Size

The review data was initially 300mb large. It included data from 30 different universities and all of the text of the Yelp reviews. Because of the size of the file, we could not use chrome to load and manipulate the data. To minimize the data size, we used nodejs to run our javascript locally to create a new JSON file which only included reviews of businesses associated with Harvard and MIT. The smaller data file was only 50mb in size. This dataset could be loaded successfully with and Chrome

4.4) Review Data Counts Processing Speed

The smaller review data set (50mb) was used to generate an area graph which plotted the number of reviews over time. Initially, the filter of this code was done in the visualization javascript (areavis.js). However, the load time was very slow. To make this process faster, the data was pre-processed with nodejs. A new JSON file was generated. The JSON file included each business. Within each business, the date and cumulative number of votes on that date were included. This file was passed into the area visualization. The area graph is then generated by summing the number of reviews for the brushed businesses. Because this file was still slow to node, we limited the scope to a two year time frame.

4.5) Review Text for Word Cloud

To generate the Word Cloud, the review data had to be pre-processed significantly. This data was processed using nodejs and written to a json file to increase the visualization speed. First, an empty array was created where for each business id, there was an empty array of text and an empty array of top words. We looped through every review and pushed all of the review text to a single string (for each unique business ID). To process the data, we removed all non-letter characters, and made everything lowercase. We then split the large string of review text into an array of individual words which we sorted alphabetically, this was to make counting unique words easier since they would all be together. We then looped through every word in each business. We counted the number of times each unique word appeared and pushed this to a new array. We sorted this array by count and included only the top 50 words. We then attempted to get an average rating for each of the 50 words for each business. We did this by taking each array of 50 words, then searching through the raw review data for those words, and incrementing the total rating by (occurrences of word * star rating of the review it was found in). Unfortunately this did not bear fruit because the data processing took far too long (for each business, for each word, look through all the reviews and match). Without the rating information, the words in the word cloud were colored arbitrarily, and the size of the text was determined by the number of times each word was mentioned by any reviewer. The word cloud is only populated when a single business is selected.

5) Exploratory Data Analysis

For our initial proposal we did look at some other Game of Thrones data visualization, but since our data set was destined to be very different, we did not glean much information from these.

Due to the time constraints brought on by our change in project proposal. We sadly did not have the luxury of exploring our data through visualization before we began our implementation. To inform our design we used our experience from the homework psets to determine what techniques work well for showing different data types. For instance, force layouts, and layouts with nodes as the objects are very good at displaying nominal data, then sorting can be implemented to gain better resolution of the data at hand. Two dimensional area plots are very good at showing very large amounts of continuous data with limited dimensions. We decided to employ these along with a word cloud in order to fit a very large amount of data onto one screen, and have the user be able to get useful information about a specific business, without actually having to read through all the reviews.

6) Design Evolution

6.1) Map Layout:

6.1.1) Design:

The map layout (Figure 1) is the first step in the Yelp data visualization. On the map of Cambridge, we will plot a heat map of all the businesses in the data set. The user will have the option of choosing whether the heat map is populated by the number of reviews, the cost, or the rating of the business via a dropdown menu. The user can brush a certain area of the map to filter the force layout. Although conveying continuous information via color is not that effective, in this case the range of number of reviews for each business is quite large, and the population is dominated by businesses with only a handful of reviews. This leads me to believe that the Cambridge area is potentially full of hidden gems, so to speak.

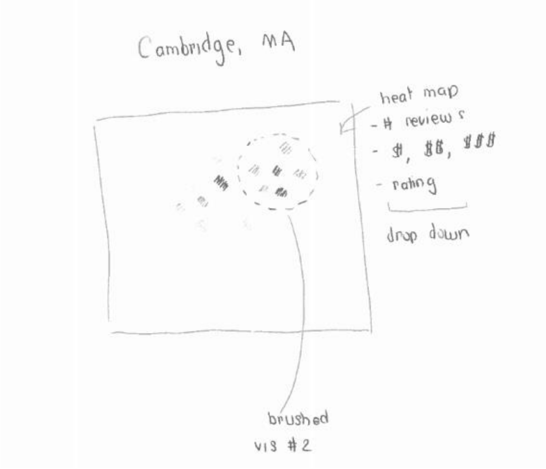


Figure 1: Map Layout

6.1.2) Questions:

6.1.2.1) Where are businesses located near Harvard and MIT?

6.1.2.2) Are there clusters of businesses with a high number of reviews in a certain area?

6.1.2.3) Are similarly priced businesses located in the same area?

6.1.2.4) Are there differences in the rating of business between the Harvard and MIT areas?

6.2) Force Layout

6.2.1) Design: The force layout (Figure 2) will have nodes represented by each business. The user can select whether to group the nodes by category. The user can select whether to color the nodes by # of reviews, # of stars, or category. A user can select a business to filter the area graph.

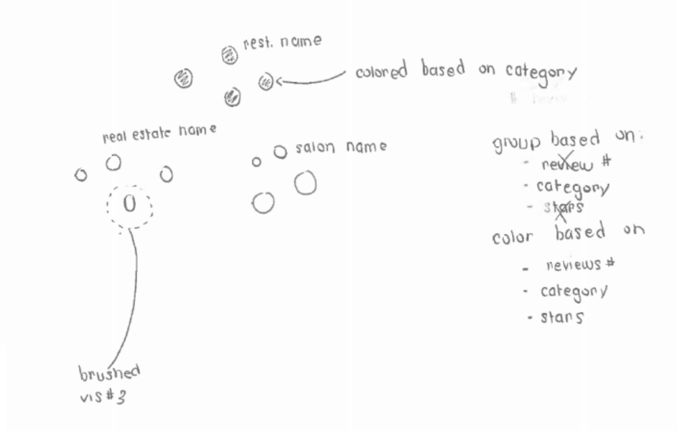


Figure 2: Force Layout

6.2.2) Questions to Answer:

6.2.2.1) Are there a large number of restaurants in a certain brushed area? Are there a large number of bars near restaurants? What other types of businesses are in the brushed area?

6.2.2.2) Are the businesses with a lot of reviews all restaurants?

6.2.2.3) What type of businesses are most common?

6.2.2.4) Are all business with a high # of stars located in the same area?

6.3) Area Graph

6.3.1) Design: The area graph (Figure 3) is specific to the selected business from the force layout. The area graph will plot the # of reviews over time. The user can brush a certain time period and receive the average # of stars and the total # of reviews in that time period.

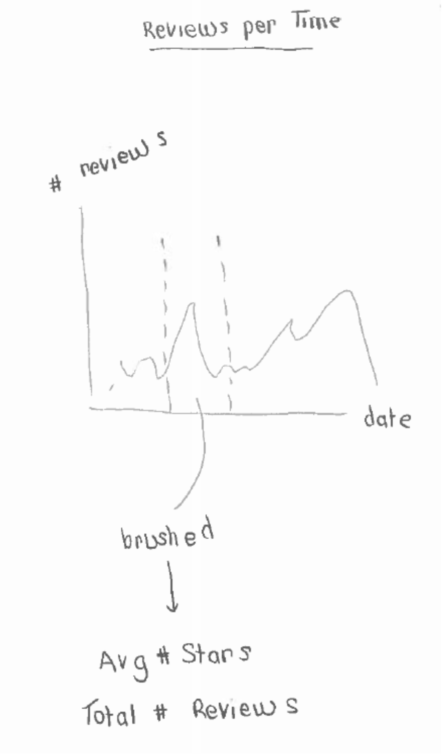


Figure 3: Area Graph

6.3.2) Questions

6.3.2.1) What do the # of votes look like over time?

6.3.2.2) Are the reviews for a specific business clustered around a certain period of time?

6.3.2.3) What is the average # of stars given in the summer?

6.4) Word Cloud

6.4.1) Design: The word cloud will be generated when a business is selected in the force layout. The word cloud will show the most commonly used words in the center, in the largest font size. This will allow the user to quickly know what specific things are being mentioned in the context of all reviews for a given business.

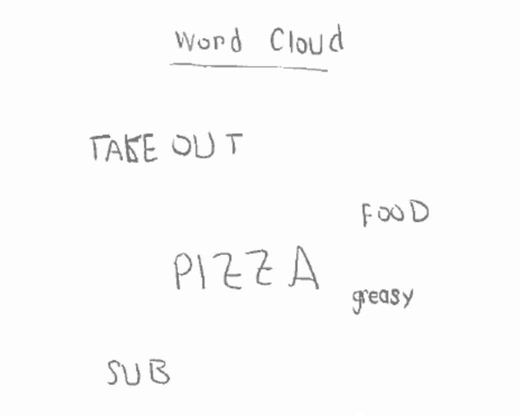


Figure 4: Word Cloud

6.4.2) Questions:

6.4.2.1) What should I order from this restaurant?

6.4.2.2) Who is the best manicurist at this beauty salon?

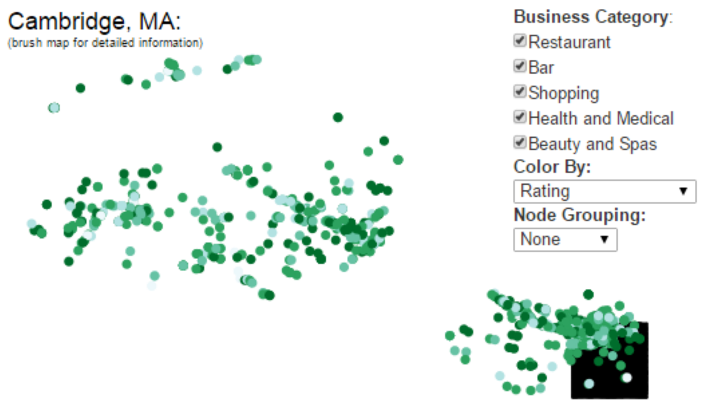
6.4.2.3) What are the common words associated with very poor, or very good reviews?

6.4.2.4) Does this bar show a lot of games for a certain sports team?

7) Implementation & Challenges:

7.1) Mapping

The first map layout will be based on a map of Cambridge. The Cambridge GitHub webpage (https://github.com/cambridgegis/cambridgegis_data) provides topojson files of many different aspects of the city. We would like to use the boundary topojson to generate a map of the neighborhoods in Cambridge. To make a map, we followed Mike Bostock's example (http://bost.ocks.org/mike/map/). We successfully generated a map of the US using a SHP file from the US census (https://www.census.gov/geo/maps-data/data/cbf/cbf_counties.html). Currently, we are in the process of generating a map of Cambridge. It is only generating a square and we are trying to troubleshoot this.



Cambridge, MA:
(brush map for detailed information)

Business Category:
☑Restaurant
☑Bar
☑Shopping
☑Health and Medical
☑Beauty and Spas
Color By:
Rating ▾
Node Grouping:
None ▾

Figure 5: Map Vis Problem

7.2) Milestone 1 Timeline Update

Because we revised our idea twice, we are behind schedule for generating our visualization. Below is an updated schedule:

Week 1-2: April 4-17 – Done

- Acquired data
- Processed and cleaned up data
- Created initial SVG elements and JS files
- Outlined JS functions and general functions to do
- Sketched out visualizations

Week 3: April 18 – 24

- Generate Map Layout
- Generate Force Layout
- Generate Area Graph

- Interactivity:
    - Brush map layout to update force layout
    - Click filter to update map layout and force layout
    - Click node grouping to update force layout
    - Click node colors to update force layout
    - Click node to update area layout and word cloud
    - Brush area layout to update data aggregation outputs

Week 4: April 25 - May 2

- Generate Word Cloud

Final Push: May 3 – May 5

7.3) Area Layout Challenges

When our area graph was first generated, it looked like Figure 6. We determined that the dates were not sorted properly and thus caused the area graph to plot out of order. We fixed this by formatting the data in javscript using "new Date". This ensured we were working with date objects and not strings.

Our review count area graph is re-generated upon brushing the map. However, when we brushed we were getting the graph represented in Figure 7 below. The problem turned out to be from the code in Figure 8. Instead of using reviewsByDate we were using this.reviewsByDate. Therefore, each time a new selection was picked, this.reviewsByDate was getting longer and longer. To fix this we used reviewsByDate and reset the variable each time the function was entered. This fixed our graph and significantly increased the speed of our code.

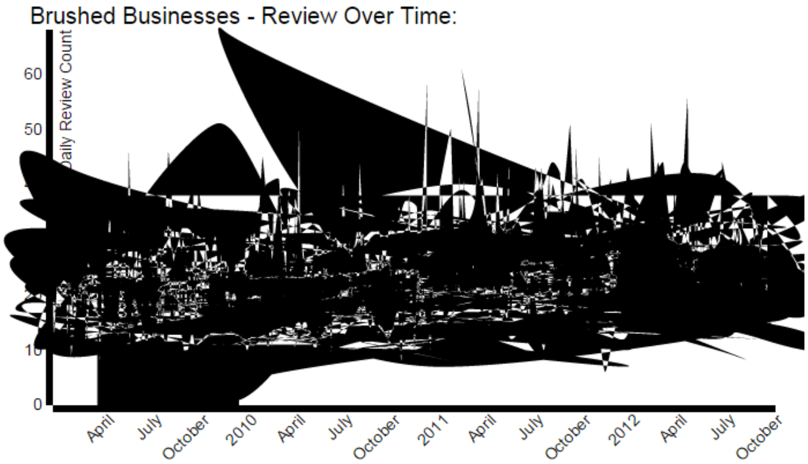


Figure 6: Area Vis Problem 1

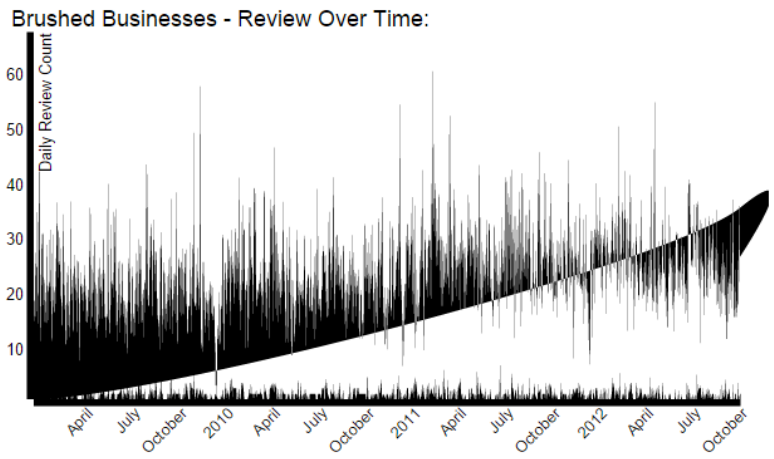
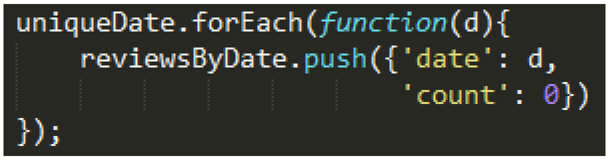


Figure 7: Area Vis Problem 2



```
uniqueDate.forEach(function(d){
    reviewsByDate.push({'date': d,
                        'count': 0})
});
```

Figure 8: Area Vis Problem Code

7.4) Visualization Changes:

Over the course of designing our visualization we changed different aspects of our design.

- Initially (Figure 1) we intended to create a heat map of Cambridge MA. However, to take advantage of the map functionality when plotting latitude and longitude, we altered our visualization slightly. We created circles for each business and colored similar to a heat map. Therefore, the user can see where an individual business is located and how it compared to others around it by the coloring.
- Initially (Figure 2) we intended to let the user select to color the nodes based on number of stars or number of reviews. Because this encoded the same information as the map visualization, we removed this from the options
- Initially (Figure 3) we intended to update the area graph (reviews per time) based on a user selecting a node in the force layout. However, many of the businesses had a small number of reviews which rendered the graph meaningless. Instead, we updated the area graph based on the brushed area so that it would be plotting a larger number of businesses.

8) Evaluation

Through the development and use of our visualization we learned several interesting things about the various businesses in Cambridge. The first being that the type of businesses being reviewed in Yelp are largely restaurants and retail (a cursory look at the ‘other’ category reveals mostly bars and fast food).

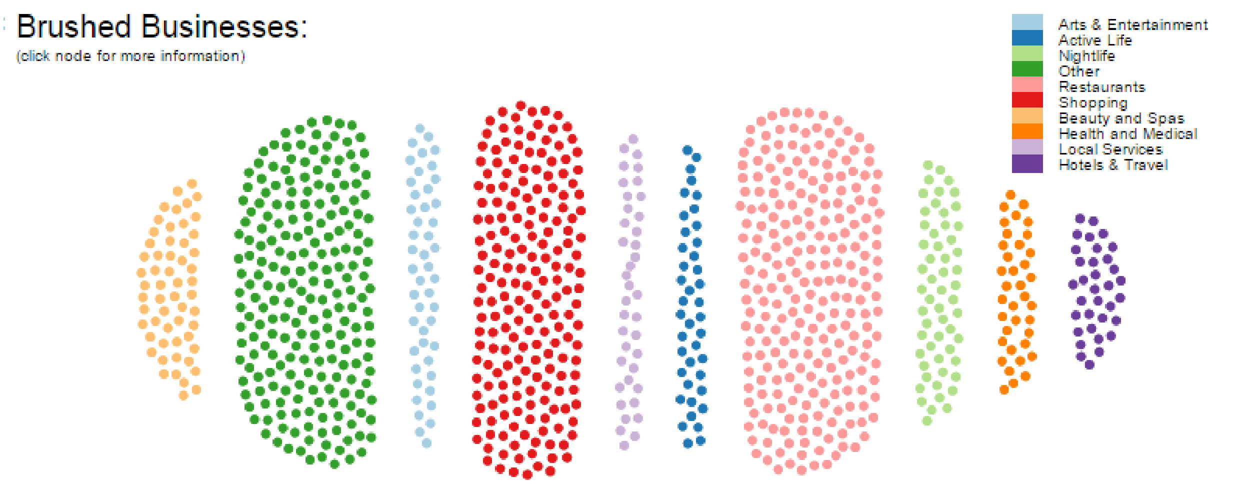
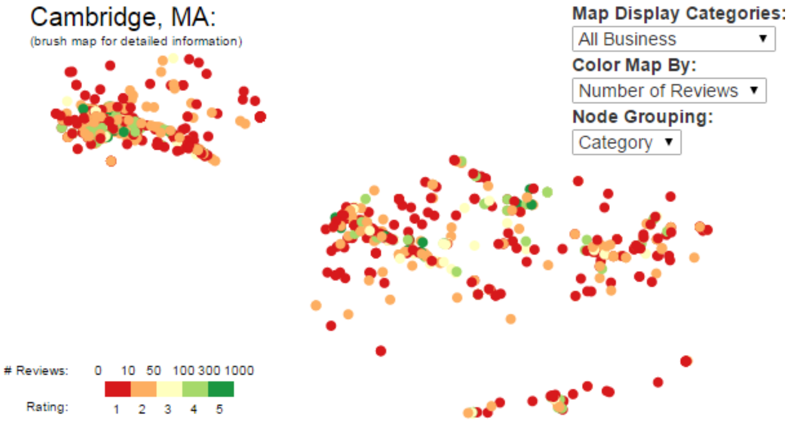


Figure 9: Types of businesses

If you color the nodes by the number of reviews for each business, you’ll notice that the vast majority of the businesses in yelp have very few reviews. My initial thought that businesses with more reviews would be clustered around the red line. But the data shows that Yelp is actually full of entries with very few reviews. The high end of the review color scale is a vast minority in the context of all of the businesses (Figure 10).


Cambridge, MA:
(brush map for detailed information)
# Reviews: 0 10 50 100 300 1000
Rating: 1 2 3 4 5

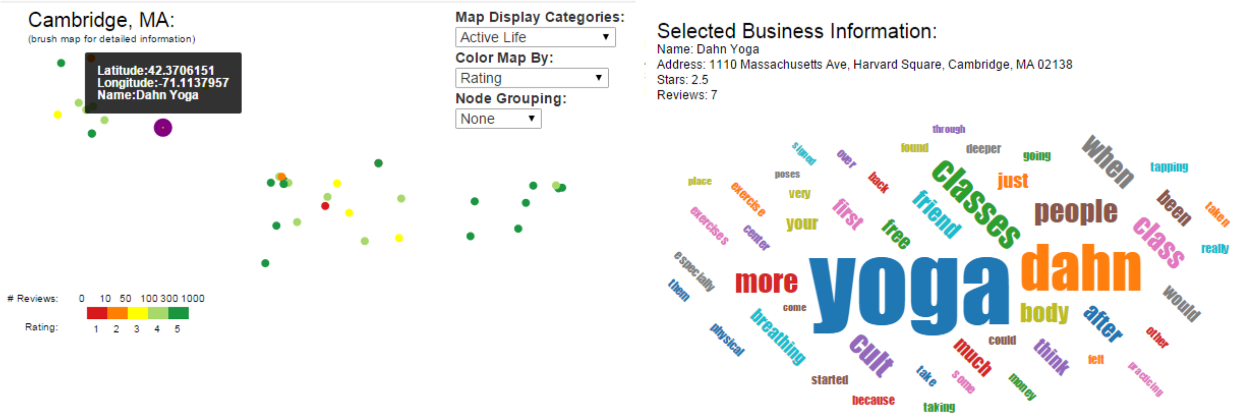
Map Display Categories: All Business ▾

Color Map By: Number of Reviews ▾

Node Grouping: Category ▾

Figure 10: Businesses colored by number of reviews

Our visualization also allows for a condensed review of specific businesses, typically when people use yelp to learn more about a business, they spend time reading through the long winded ramblings of strangers. Our word cloud allows you to condense all of the reviews for a specific business and see the words that are used most commonly across all reviewers. Figure 11 shows the word cloud for a yoga studio, the name of the business are the two most mentioned words, but if you look below the “O” in “yoga”, youll notice that the word ‘cult’ has likely been used in multiple reviews. Coupled with the 2.5 star rating, I would conclude that Dahn Yoga is a place to be avoided. By using the filters on our business map you can now easily review other gyms and yoga studios in your area.



Map Display Categories: Active Life

Color Map By: Rating

Node Grouping: None

Selected Business Information:
Name: Dahn Yoga
Address: 1110 Massachusetts Ave, Harvard Square, Cambridge, MA 02138
Stars: 2.5
Reviews: 7

Figure 11: Dahn Yoga Word Cloud

9) Appendix

9.1) Visualization of Ice and Fire Initial Proposal & Sketches

Group Members: Marissa Stanvick and Maxwell Lloyd
Github Repository: https://github.com/maxwelllloyd/cs171-finalproject.git

1. **Background and Motivation.** Discuss your motivations and reasons for choosing this project, especially any background or research interests that may have influenced your decision.

A Song of Ice and Fire is a series of books written by George R. Martin. The first book, A Game of Thrones, was published in 1996. The most recent book, A Dance with Dragons, was published in 2011. There are two books planned which will complete the 7 book series. The books detail a complex fictional universe. There is a large assortment of characters that are followed throughout a multi-year journey which details a war over the king of Westeros. The books were adapted into an HBO television show, A Game of Thrones, in 2011. Since then the popularity of the books and television show has sky rocketed.

One of reader's and viewer's biggest complaint with the show is the overall complexity of the story. There are dozens of narratives going at any given time spanning across two continents and many kingdoms. Both members of this project team are fans of both the books and television show. The interest in the series coupled with the complex web of characters and locations prompted the idea for this visualization.

2. **Project Objectives.** Provide the primary questions you are trying to answer with your visualization. What would you like to learn and accomplish? List the benefits.

Because Game of Thrones involves a large number of different characters and locales, this visualization will focus on questions pertaining to those areas. This project will specifically try to address the following questions:

1) How does the location of a character change over time? Where do they travel to?
2) What percentage of time do characters spend in a certain location?
3) How do the books and tv show differ in terms of character locations?

3. **Data.** From where and how are you collecting your data? If appropriate, provide a link to your data sources.

There are several other Game of Thrones visualizations available on the internet, so the first place we will look for data is these existing visualizations. We will either ask the creator how they collected data or (if applicable) ask permission to use their raw data (cited in full of course).

Our second option will be to write a python program to scrape web sites such as (http://awoiaf.westeros.org/) and (http://towerofthehand.com/). Since neither of us have any experience with web scraping or python, our expectations may need to be altered to reflect the data we're able to gather.

The third option is the simplest, but also most time consuming, manually culling data from both show and books by rewatching and rereading. We can also manually catalogue data from the above websites in CSV files.

4. **Data Processing.** Do you expect to do substantial data cleanup? What quantities do you plan to derive from your data? How will data processing be implemented?

The answer to this question is highly dependent on the method chosen from question 3. Assuming that we will be scraping our own data, I don't think we'll have to do a large amount of data cleanup, but there will be quite a bit of processing involved. Javascript will be used to implement the data processing we will have to do. From the character location and time data, we will have to calculate the percentage of time spent at each location. The quantities we will need to derive are on a per character basis;

1) Physical location: Calculated from string data
2) Location at a given time: Time scale calculated from chapter/episode index
3) Total amount of time spent at each location: Time and location will be normalized by chapter/episode index

5. **Visualization.** How will you display your data? Provide some general ideas that you have for the visualization design. Include sketches of your design.

The sketches for the design are located in the PDF file "Initial Project Proposal Sketch". Because the visualization is trying to answer location based questions, the basis for the main portion of the visualization will be a map of Westeros and Essos. The map will be divided into the major regions of the book. There will be a menu on the page that allows the user to select characters and timeline. As a user selects a character and time point, the regions will populate with symbols representing the character. As the time slider is progressed, more symbols will populate the map as characters move. The symbols will encode what percentage of time the character spends in a certain location. To provide more detailed information on region specific time percentage, when characters are selected, a bar graph will also appear that encodes the percentage of time that is spent in each regions.

6. **Must-Have Features.** These are features without which you would consider your project to be a failure.

The most critical aspect of our visualization will be the movement of each node to specific locations at specific times across a fictional map. Because it is a fictional location, we cannot use d3 geoscaling. This feature encompasses all of our must have features: we must be able to align the movements of characters to a time scale which is not explicit in either book series or television show. We will also need to assign physical locations to string data gathered from the book/show.

The next important feature of our visualization will be the deep dive into character location. When you click a character, there will be a graph detailing the percentage of time the character has spent in each region.

The third must have feature will be the ability to show multiple characters at once in both the map visualization as well as detailed view. We will select a group of the most important characters to begin our visualization with. If we are able to implement data scraping, we will be able to include a larger number of characters.

7. **Optional Features.** Those features which you consider would be nice to have, but not critical.

One optional feature will be the character path. The path would represent the route the character travels. Because travel is never by air, the route each character takes between locations will not be straight, so we will attempt to move (or create the illusion) characters non-linearly between locations.

A second optional feature would be to compare the book series to the television series. This would require a sync of timelines, since one episode of the show covers a variable number of chapters in the book. It would also require displaying two sets of data on the graph at once.

A third optional feature would be to include additional data about each character. For example, how old the characters are or if the characters are alive.

8. **Project Schedule.** Make sure that you plan your work so that you can avoid a big rush right before the final project deadline, and delegate different modules and responsibilities among your team members. Write this in terms of weekly deadlines.

April 3rd: Project Proposal Due

April 5th – April 11th:

- Data: (Max)
    - Gather data source
    - Format data source
- Map: (Marissa)
    - Acquire map
    - Specify map regions
    - Map regions to specific SVG locations
- Visualization: (Marissa)
    - Set up data visualization elements
        - SVG 1: Map
        - SVG 2: Selections / Options
            - Slider
                - TV slider
                - Book Slider
            - Book and television buttons
            - Groups of character buttons
            - Character expansion on selection
        - SVG 3: Bar Chart

April 12th – April 18th:

- Data Wrangling: (Max and Marissa)
    - Input 1: Specific Character
    - Input 2: Time Location
    - Output: Location & Percentage of Time

April 17th: Milestone 1

April 19th – April 25th:

- Interactivity:
    - Movement of Slider (Marissa)
        - Addition / Subtraction of Map Symbols
        - Update of data corresponding to each map symbol
        - Update of bar chart with data
    - Check/Uncheck Character (Max)
        - Addition / Subtraction of Map Symbols
        - Update of data corresponding to each map symbol
        - Addition / Removal of Bar Chart
    - Meeting with TFs

April 26th – May 5th :

- Interactivity:
    - Check/Uncheck Book / TV Buttons (Marissa and Max)
        - Merging / Separation of Map Symbols
        - Update of data corresponding to each map symbol
        - Addition / Removal of Second Series to Bar Chart

May 5th: Final Project Due

9.2) Yelp Data Challenge Proposal & Sketches

Group Members: Marissa Stanvick and Maxwell Lloyd
Github Repository: https://github.com/maxwelllloyd/cs171-finalproject.git

9. **Background and Motivation.** Discuss your motivations and reasons for choosing this project, especially any background or research interests that may have influenced your decision.

Going out to eat is a common occurrence in many households. When going out to eat, many people base their decision on

1) Proximity of Restaurant
2) Cost of Restaurant
3) Type of Cuisine
4) Restaurant Reputation

Yelp is a website which tracks a lot of the above data. They allow users to rate restaurants and write reviews. This allows others to gain more information about the restaurant before visiting.

Currently, Yelp has a map functionality. However, you can only view 10 restaurants at a time and cannot easily view all of the restaurants in a certain area. Currently, you also can't compare restaurants. The comparison of restaurants is critical in selecting where to dine, and the aggregation of reviews is useful in deciding if the restaurant is right for you. Perhaps you are looking for a pizza joint, but the reviews specify that the restaurant actually specializes in subs. There is also no aggregation of the text in the reviews. Our Yelp visualization intends to build on these areas.

10. **Project Objectives.** Provide the primary questions you are trying to answer with your visualization. What would you like to learn and accomplish? List the benefits.

The primary goal of this visualization is to be able to choose a place to dine based on Yelp reviews. From this visualization, you will learn:

1) What restaurants are in the area
2) What restaurants have the best rating

3) Where specific cuisine restaurants are located
4) Which restaurants have the highest number of reviews
5) What the major topics in the reviews are for each restaurant
6) What the specifics of each restaurant are

11. **Data.** From where and how are you collecting your data? If appropriate, provide a link to your data sources.

The data for this visualization will be from the Yelp data set challenge (https://www.yelp.com/dataset_challenge/dataset). This data is a collection of JSON files that include information about each business, the reviews, and the reviewers.

12. **Data Processing.** Do you expect to do substantial data cleanup? What quantities do you plan to derive from your data? How will data processing be implemented?

Because we are focusing on our design on restaurants, we will have to clean-up the Yelp data to include only business that serve food and beverages. Currently, the data set includes other businesses such as hospitals or nail salons. By cleaning up the data, our data set will be smaller and help with the load time for our website.

From the data set we will be deriving the following quantities:

1) Average # of Stars
2) Total # of Reviews
3) # of Times Specific Words are Mentioned in Reviews for a Restaurant

13. **Visualization.** How will you display your data? Provide some general ideas that you have for the visualization design. Include sketches of your design.

The sketches for the design are located in the PDF file “Yelp Project Proposal Sketch”. Because our design is trying to answer the question of “where should I eat”, it will primarily be map based. The user will have the opportunity to interact with the map to select a certain area to eat. They will be able to filter the restaurants based on type of cuisine or other important features such as delivery. They will then be able to look at a small subset of restaurants to compare the number of reviews and what words are mentioned most often in the reviews.

14. **Must-Have Features.** These are features without which you would consider your project to be a failure.

Our design must be map based. The user needs to be able to first select the general section of the country they are eating. They then must be able to brush a smaller area.

The user must be able to filter based on the type of cuisine. This will be crucial to limiting the amount of restaurants the user sees. They could also filter on delivery or cost.

The nodes on the area map (visualization 2) need to be colored by rating so that the user can have a general idea of how a restaurant is regarded.

The user must be able to compare restaurants based on the number of ratings because the number of ratings is an important marker of how popular a restaurant is.

We must be able to aggregate the top words from each review. For example, if a review mentions pizza a lot but the user wanted a sub, that would be useful in selecting the restaurant they want to eat at. At minimum, we should be able to provide this information in a bar graph or force layout.

15. **Optional Features.** Those features which you consider would be nice to have, but not critical.

One optional feature would be to display the review information in a word cloud. A word cloud is a neat visual way to quickly pick out which words are mentioned most often in the reviews. Ideally, we would color these words based on the average number of stars given in the reviews which include the word. For example, if 100 reviews mentioned pizza but the average rating of these reviews was 1 star, the pizza at the restaurant may not be good.

An additional optional feature would be for the user to be able to zoom in and out of the selected area similar to Google Maps functionality. The required feature is the ability to brush a certain area.

An additional optional feature would be to have a rating and # of reviews filter on the area map visualization. The user can then select if they only want to look at restaurants with a X number of reviews and an average rating of greater than Y.

16. **Project Schedule.** Make sure that you plan your work so that you can avoid a big rush right before the final project deadline, and delegate different modules

and responsibilities among your team members. Write this in terms of weekly deadlines.

April 3rd: Project Proposal Due

April 5th – April 11th:

- Data: (Max)
  - Gather data source
  - Format data source
- Map: (Marissa)
  - Learn D3 map functionality
- Visualization Pieces: (Marissa)
  - SVG 1: Map
  - SVG 2: Selections / Options
    - Cuisine
    - Other Attributes
  - SVG 3: Detailed Visualization
    - Bar Chart for # of Reviews Comparison
    - Word Cloud for Review Text
    - Overview Box with Specific Info

April 12th – April 18th:

- Map
  - Plot restaurants on map
  - Select map area
  - Plot restaurants on specific map area
  - Color points based on review

April 17th: Milestone 1

April 19th – April 25th:

- Map
  - Filter restaurants based on cuisine and attributes
  - Brush map to select a small number of restaurants to compare
- Comparison Visualization
  - Bar chart to compare the number of reviews
- Detailed Info Visualization
  - Word cloud to determine most popular words
  - Text box with detailed restaurant information

April 26th – May 5th :

- Final details!

May 5th: Final Project Due