Samantha Lee

Professor Aanjaneya

Computer Graphics (16:198:523)

8 April 2022

<center>Project Milestone</center>

Current Status

As per the project proposal, I am on track with my intended goals. As of now, I am able to generate a basic version of my 3D force-directed graph with group-based colors and labels on hover, and I can continue to use April to implement various additional features.
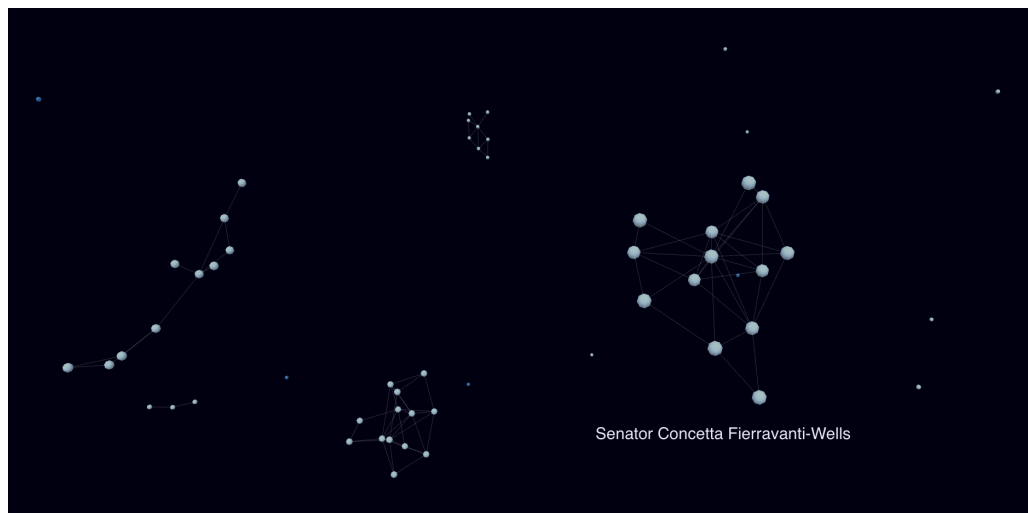
Note: all data manipulation and logic can be found in *fdg/src/index.js*.

Problems Encountered

There were some minor setbacks I had to work around.

1. Rendering the visualization
    a. I was hoping to render my visualization in pure HTML and JavaScript, but the package is only supported if I run using a local host. Thus, I had to create a React application and learn how to set up a server and run my graph through the frontend framework.
2. Data formatting
    a. Documentation of the 3D FDG package outlined that my data must be formatted like so:

```
{
    "nodes": [
        {
            "id": "id1",
            "name": "name1",
            "val": 1
        },
        {
            "id": "id2",
            "name": "name2",
            "val": 10
        },
        ...
    ],
```
```
    "links": [
        {
            "source": "id1",
            "target": "id2"
        },
        ...
    ]
}
```

However, the datasets I had in mind merely listed the nodes and edges in CSV files. The datasets I could find that were already in this dictionary format were a part of existing FDG projects or had arbitrary topics, but I wanted to create a unique FDG that concerned my own interests. Thus, I ended up choosing a network dataset on politician Facebook pages and reformatting the data in a way the package could interpret.

3. Package Bug

   a. To reformat the data, I looped through each line of the nodes CSV and pushed a corresponding dictionary to an array containing all nodes. Then, I did the same for links. For some reason, this method of generating the link data caused an error. After some investigation, I found that while the package transformed each node in the nodes array into an object supported by Three.js, it did not do the same for the links. I hope to resolve this issue, but in the meantime, I have hardcoded the link data. If I'm unable to find a fix, this method will suffice.

4. Data volume

   a. My dataset has 5.9K nodes and 41.7K edges. I was successful in rendering all the nodes (albeit with a bit of lag), but it took several minutes to render the edges appropriately. Thus, I chose to filter the dataset to include only American politicians (using "senator" and "congress" as keywords to indicate nationality in FB page titles).

5. Limitations in data

   a. The dataset itself only includes names of politicians and numeric IDs. In my final FDG, I'd like to color the nodes based on partisanship and include more information on the politicians. To do this, I will have to find

an API that provides this data or build a web scraper to collect it. Alternatively, I can add it manually.

Objectives

Within the next two weeks, I hope to

1. Double check that the filter-via-keyword approach accurately includes all American politicians in the datasets and excludes non-American ones.

2. Collect additional information about the politicians in the dataset

   a. In particular, I would like to find party affiliations. Currently, I group the nodes into two categories based on the keywords "senator" and "congress." This isn't the most constructive grouping (the former is a subset of the latter), but it demonstrates the use of color. By including affiliations, I can group by party and customize colors to better represent the distribution of nodes and links.

3. Play around with additional FDG features

   a. Highlighting nodes on hover

   b. Click to focus

   c. Fix nodes after dragging