Ariella Gilmore
MAT 265 - Winter
Professor Legrady
March 22, 2020

30 Years of Trends Between Ars Electronica's Art Works and ACM's Technical Papers

Previous work

Previously, I focused my priorities on the Ars Electronica data set and creating a visualization that would help enable people to better understand the Ars Electronica archive. This visualization created a space in which people could see projects submitted through the years and see how they corresponded to one another based on word frequency. A regular archive only allows for a static view of one project at a time, but this visualization enables a user to see how a project is important in its dataset.

Exploration

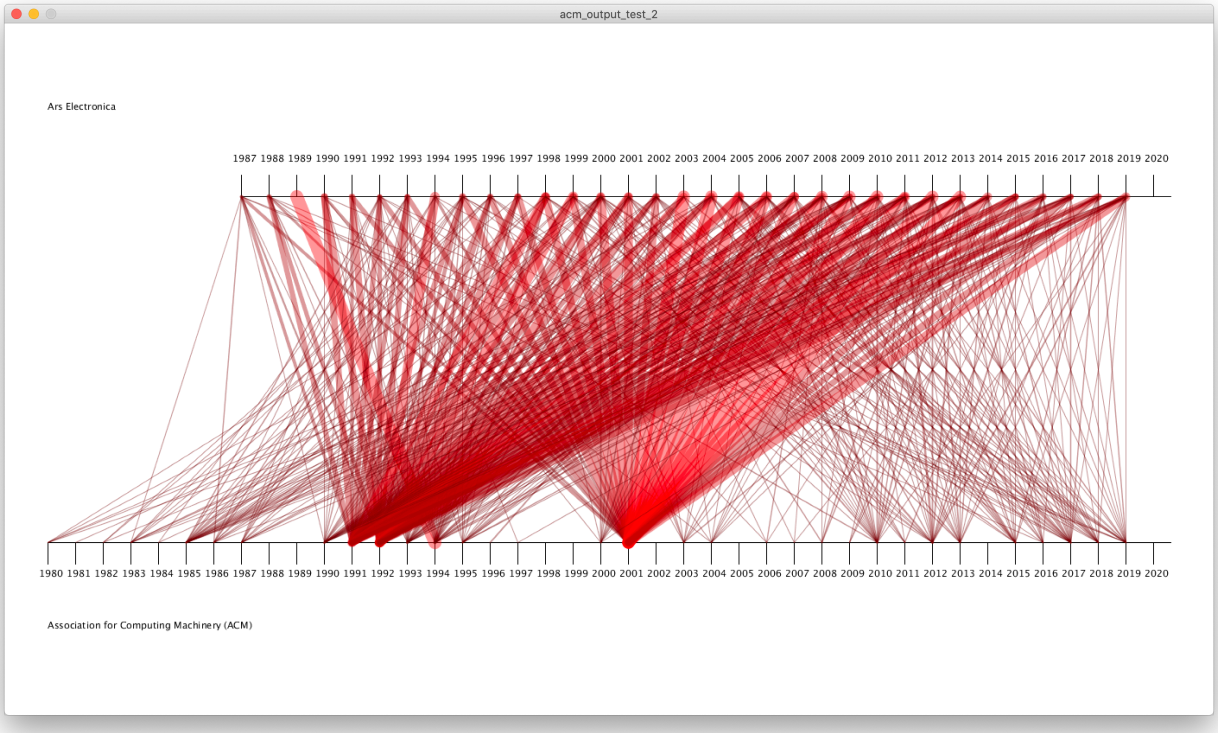
After feeling confident in my results, I wanted to see how works within Ars Electronica compare to the technologies being used in more technical environments/conferences. I explored different types of datasets including resources such as articles from Wired and papers from Google Scholar. I decided to start with using the Association of Computing Machinery (ACM) because of the vast and diverse variety of papers that are submitted through ACM every year and also because it has a reputable digital library that contain all of this information.

Collecting Dataset

Using Beautiful Soup, which is a Python library I used previously, I was able to scrape and download PDF's from each year in ACM's digital library from 1980-2020. The digital library interface allows users to search documents by year and organize by the most downloaded. As an preliminary dataset, I scraped the top 100 most downloaded works from each year.

Analysis

After collecting my basic dataset for both Ars Electronica and ACM, I wanted to develop a new heuristic for comparing these resources. My previous project used a simple word count, but I felt that a word count in this situation would not be accurate enough to convey a relationship between these two mediums. The picture below, shows a word count mapping of Ars Electronica projects to ACM papers. It is noticable that there are many similar words in the years 1991-1994 and 2001, but again this is not able to tell us enough information.



Eventually, the digital humanities department discussed some of their resources to me. The one that stood out the most was a program called the Wikifier.

The Wikifier is a tool that takes in text and, once it finds the keywords from the given text, outputs each of the keywords' best matched wikipedia title. One can try it out here: http://wikifier.org/. The keywords are also ranked depending on both how "important" it is in the document as well as how well it feels the wikipedia title is correctly matched.

This program allows me to take two separate datasets and create a connection between them across one set of possibly similar keywords. The Wikifier allows for different types of language to be used, but still fall under one ultimate category created by the wikipedia titles.

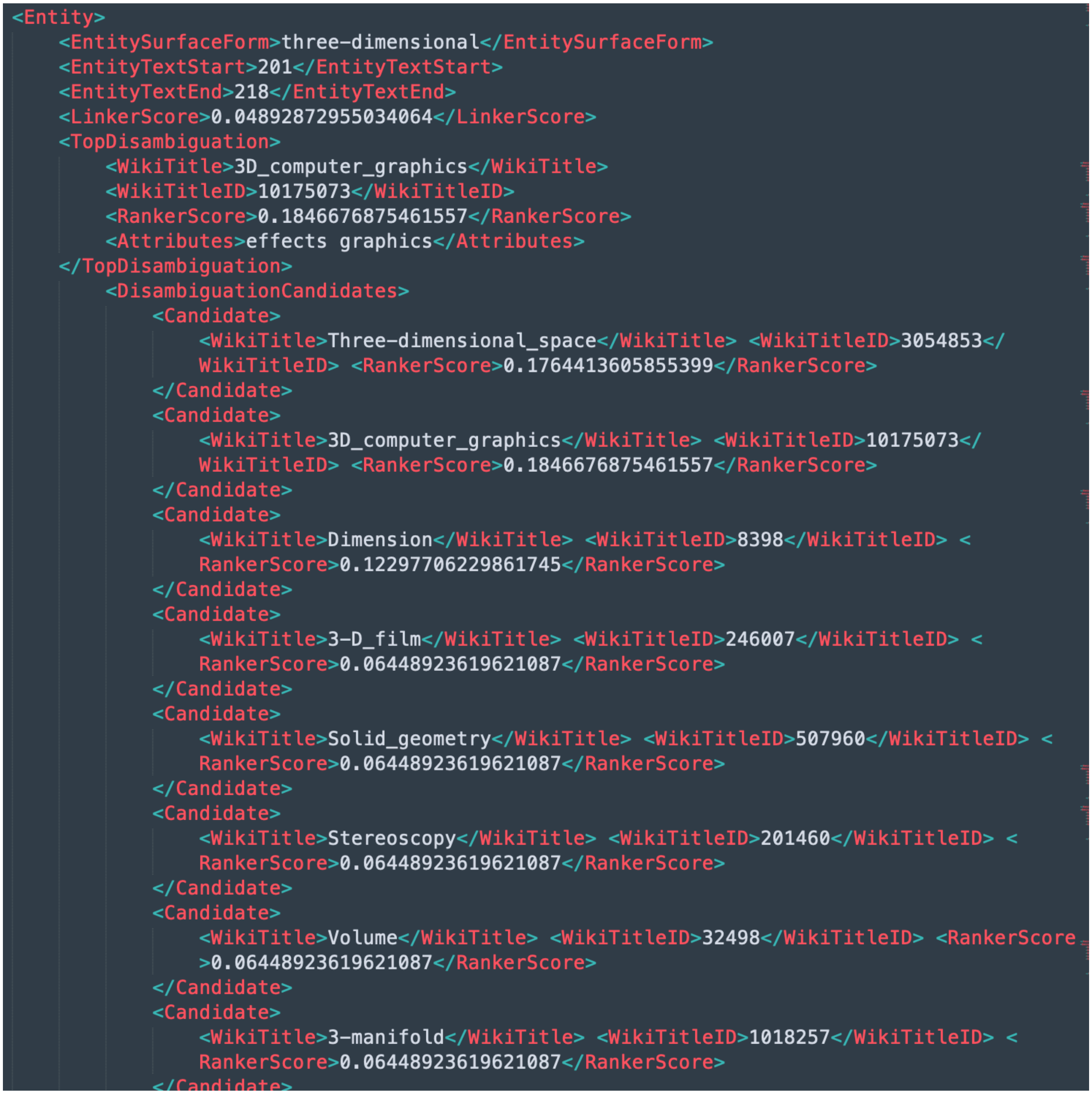
The digital humanities team has a virtual machine, where they gave me an account, so that I can upload all the files I scraped and run the wikifier through this java command line:

```
java -jar dist/wikifier-3.0-jar-with-dependencies.jar -annotateData ACMData
ACMOutput false configs/STAND_ALONE_NO_INFERENCE.xml
```

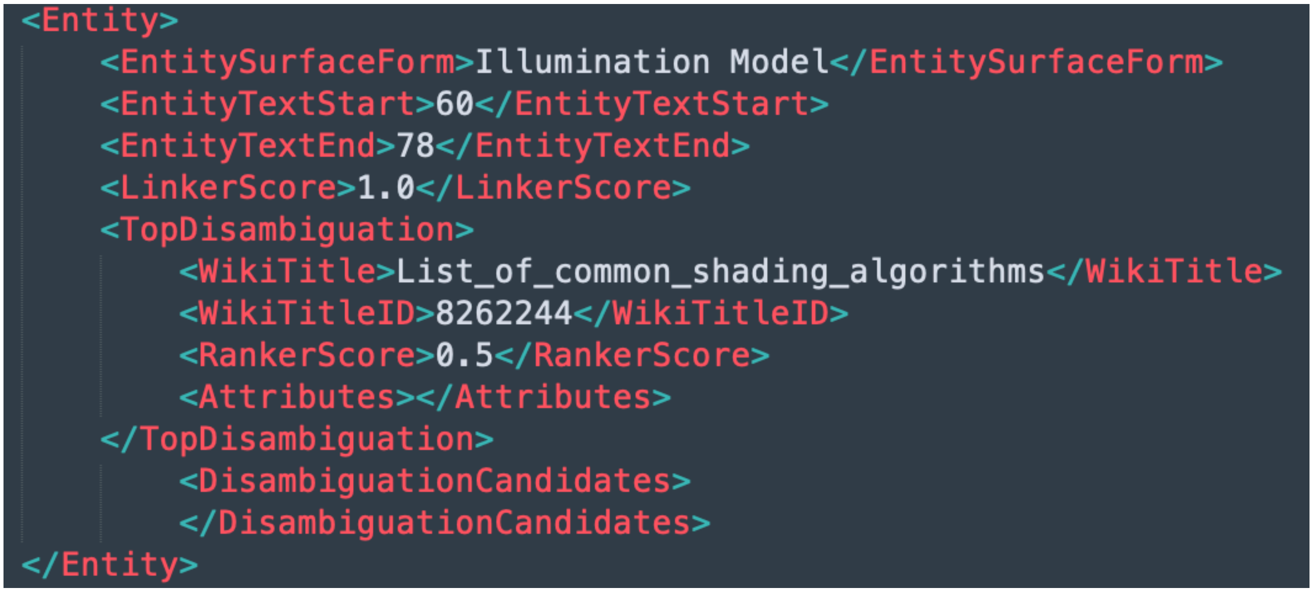
This command line runs the wikifier java file and specify the input folder("ACMDATA") of my data and which folder("ACMOutput") I want my output results to appear.

The wikifer takes in txt files that then get outputed to an xml file with the wikifier data. Because all of my files were originally PDF formatted., I used two different Python libraries called PyPDF2 and pdfminer, which both convert the PDF's into text files. Also, on macs there is an automator application, which is kind of like a visual coding language. One of the features is being able to convert PDF's to text. With the combination of these three tools mostly all of the PDF's will be converted into a normal text output.

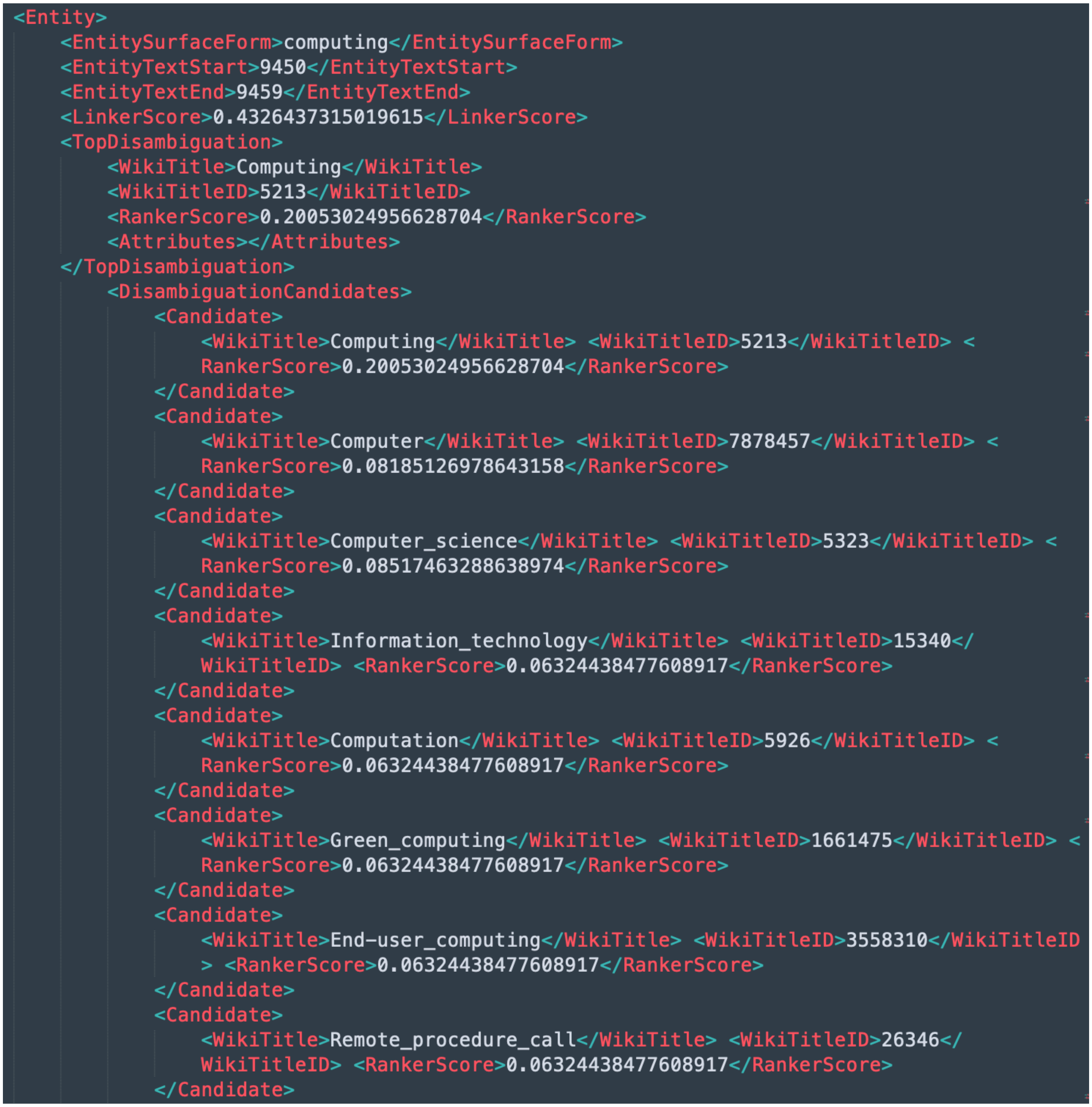
After running the wikifier command line, here are some examples of output I recieve in XML format. The "EntitySurfaceForm" is one of the keywords from the input text. The "LinkerScore" is how strong of a link there is to the Wikipedia article. The "WikiTitle" references the Wikipedia article that the keyword is matched to and the "RankerScore" is how well it feels each Wikipedia title correctly matches the keyword being referenced relative to each other.



```
<Entity>
    <EntitySurfaceForm>three-dimensional</EntitySurfaceForm>
    <EntityTextStart>201</EntityTextStart>
    <EntityTextEnd>218</EntityTextEnd>
    <LinkerScore>0.04892872955034064</LinkerScore>
    <TopDisambiguation>
        <WikiTitle>3D_computer_graphics</WikiTitle>
        <WikiTitleID>10175073</WikiTitleID>
        <RankerScore>0.1846676875461557</RankerScore>
        <Attributes>effects graphics</Attributes>
    </TopDisambiguation>
        <DisambiguationCandidates>
            <Candidate>
                <WikiTitle>Three-dimensional_space</WikiTitle> <WikiTitleID>3054853</
                WikiTitleID> <RankerScore>0.1764413605855399</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>3D_computer_graphics</WikiTitle> <WikiTitleID>10175073</
                WikiTitleID> <RankerScore>0.1846676875461557</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Dimension</WikiTitle> <WikiTitleID>8398</WikiTitleID> <
                RankerScore>0.12297706229861745</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>3-D_film</WikiTitle> <WikiTitleID>246007</WikiTitleID> <
                RankerScore>0.06448923619621087</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Solid_geometry</WikiTitle> <WikiTitleID>507960</WikiTitleID> <
                RankerScore>0.06448923619621087</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Stereoscopy</WikiTitle> <WikiTitleID>201460</WikiTitleID> <
                RankerScore>0.06448923619621087</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Volume</WikiTitle> <WikiTitleID>32498</WikiTitleID> <RankerScore
                >0.06448923619621087</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>3-manifold</WikiTitle> <WikiTitleID>1018257</WikiTitleID> <
                RankerScore>0.06448923619621087</RankerScore>
            </Candidate>
```



```
<Entity>
    <EntitySurfaceForm>Illumination Model</EntitySurfaceForm>
    <EntityTextStart>60</EntityTextStart>
    <EntityTextEnd>78</EntityTextEnd>
    <LinkerScore>1.0</LinkerScore>
    <TopDisambiguation>
        <WikiTitle>List_of_common_shading_algorithms</WikiTitle>
        <WikiTitleID>8262244</WikiTitleID>
        <RankerScore>0.5</RankerScore>
        <Attributes></Attributes>
    </TopDisambiguation>
        <DisambiguationCandidates>
        </DisambiguationCandidates>
</Entity>
```



```
<Entity>
    <EntitySurfaceForm>computing</EntitySurfaceForm>
    <EntityTextStart>9450</EntityTextStart>
    <EntityTextEnd>9459</EntityTextEnd>
    <LinkerScore>0.4326437315019615</LinkerScore>
    <TopDisambiguation>
        <WikiTitle>Computing</WikiTitle>
        <WikiTitleID>5213</WikiTitleID>
        <RankerScore>0.20053024956628704</RankerScore>
        <Attributes></Attributes>
    </TopDisambiguation>
        <DisambiguationCandidates>
            <Candidate>
                <WikiTitle>Computing</WikiTitle> <WikiTitleID>5213</WikiTitleID> <
                RankerScore>0.20053024956628704</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Computer</WikiTitle> <WikiTitleID>7878457</WikiTitleID> <
                RankerScore>0.08185126978643158</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Computer_science</WikiTitle> <WikiTitleID>5323</WikiTitleID> <
                RankerScore>0.08517463288638974</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Information_technology</WikiTitle> <WikiTitleID>15340</
                WikiTitleID> <RankerScore>0.06324438477608917</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Computation</WikiTitle> <WikiTitleID>5926</WikiTitleID> <
                RankerScore>0.06324438477608917</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Green_computing</WikiTitle> <WikiTitleID>1661475</WikiTitleID> <
                RankerScore>0.06324438477608917</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>End-user_computing</WikiTitle> <WikiTitleID>3558310</WikiTitleID
                > <RankerScore>0.06324438477608917</RankerScore>
            </Candidate>
            <Candidate>
                <WikiTitle>Remote_procedure_call</WikiTitle> <WikiTitleID>26346</
                WikiTitleID> <RankerScore>0.06324438477608917</RankerScore>
            </Candidate>
```

Future work

After collecting my results from the wikifier for both Ars Electronica and ACM, I plan on creating a data visualization to represent the connections between these two mediums. One form of representation could be a Sankey diagram. There are multiple ways to create this, but I believe the best way could be using javascript's D3 library. An example of a Sankey diagram created by D3 can be seen here: https://observablehq.com/@d3/sankey-diagram#sankey. Each example provides code explaining how to implement this diagram into your own system. It also allows for uploading your own files and testing that way first. The file should consist of three columns, the first being the source, the second is the target, and the third is the value. This gives the code the information of where the source should connect to the target and how thick of a line should be drawn between them. The D3 library also allows for a lot of parameter adjustment in the stying as well as easy integration for a webpage.