Project title: NYU Professors Citation Network and Personal Profiles

Team: Cinny Lin (ycl461), Jerry Ding (yd1158), Huanci Wang (hw1685)

Overview

Finding an advisor is a crucial decision for students who plan to pursue research careers. However, many students struggle to find the relevant information that helps them understand a university faculty well. To address these students' needs, we visualized NYU New York, Shanghai and Abu Dhabi computer science professors' publications and citations over the years, as well as the diversity of their coauthors and research labs (based on gender and ethnicity).

Data

We created our own dataset by scraping and querying from Google Scholar, Google map, Ethnicolr, Namesor. It includes 42 professors/coauthors, and 2750 publications details (name, year, citation, number of co-authors) by those professors. For each co-author, the dataset includes their gender, ethnicity, and location, inferred from their names and affiliation.

Specifically, we have those separate data files. “cites_per_year.csv” records each professor’s citation per year (quantitative). “gender_dict.pkl” and “race_dict.pkl” records all professors’ and co-authors' predicted gender and ethnicity (categorical). “location.pkl” records all affiliations and their locations by Google Map API (longitude and latitude, quantitative). “publication_all.csv” records every publication of the professors’ and their co-author amounts (quantitative), year (ordinal) and citation counts (quantitative). “connected_names.csv” records professors who have co-authorship before. The other data files seen in the project are also inferred from these dataset and are used for graphs.

We preprocess the data for each graph. For different graphs, we might use csv files, json files, or directly pass data in flask. We make sure the data is properly fed to each graph, as shown in the demo.

Goals and Tasks

The goal of this project is to reduce the time students spend on looking into different professors by developing a series of visualizations that could facilitate students’ search on professors’ research information. Specifically, the user could search (locate) for a specific professor, and information about a professor’s research works would be presented, and additional information is derived from original data (eg. ethnicity, gender, location). Our race chart also serves the purpose for users to compare and enjoy.

Visualization

Our visualization is composed of a main view (network graph) and two sub views (race chart and individual profiles).

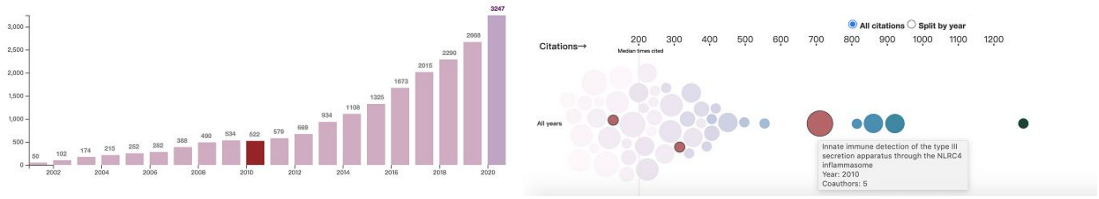
1. **Network graph:** On opening our visualization, the users are presented with a network graph as below which shows CS department professors of NYU New York, Shanghai and Abu Dhabi.

- Mark: Node
- Channel: Link, size, color
- Rationale: We choose a network graph to represent the connections

between professors. Two professors are connected if they have co-wrote papers together. We think network graphs can best represent relationships, and in this case co-authorship. Additionally, the size of the node is a good visual channel for the number of publications, since bigger nodes can easily catch users' attention. Moreover, different campus locations of professors are distinguished by different color scales so that users can see the interaction among the three NYU sites.

- Interaction:Users can click on a node, which opens up to a new page of the professor's publications and coauthors. Users can use the dropdown list to filter the network graph to only show results from either school branches. They can also drag each node to better see its connection with other nodes.

2. **Individual profile:** On clicking each node, the users will be guided to the professor's individual profile, which is the second view. There are 2 bar charts, a pie chart, a bubble chart and world map on this page.



2.1 Publications / Citations

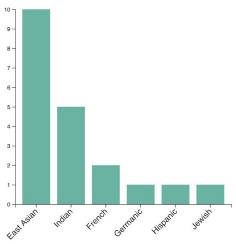
1) **Bar chart -- Citations**

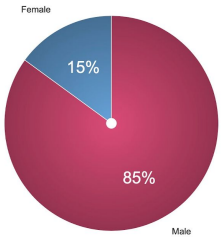
- Mark: bar
- Channel: color,vertical and horizontal position
- Rationale: this graph aims to show the number citations that a professor receives over the years. The year with the highest number is highlighted in a different color. A bar chart like this clearly shows the trend and differences over the years thus we chose this view.
- Interaction: They will be an animation at first showing each bar one by one. On hovering, the bar will be highlighted and the other bars will have a higher opacity. Also, the linked view between this bar chart and the bubble graph below will show and the citations of the same year will be highlighted.

2) **Bubble Chart -- Publications (linked with bar chart 1)**

- Mark: Circle
- Channel: Color, size, position
- Rationale: The x axis shows the number of times each paper was cited, and the y axis can be expanded and represents the year the paper was written. The size of the circle represents the number of co-authors in that paper. Bubble chart is a good choice because the size and color channel are very informative and it allows this graph to show 3 pieces of information for each publication.
- Interaction: Users can click on "Split by year" or "All publications" to choose to see publication by year or all publications of this author. On hovering, users can see a tooltip showing the title, year of publication and number of times being cited. Also, the linked view of the citations in the same year will appear in the bar chart above. On clicking, it opens up a new window looking up the paper on Google scholar.

2.2 Coauthors



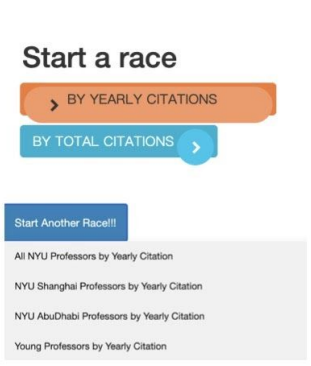


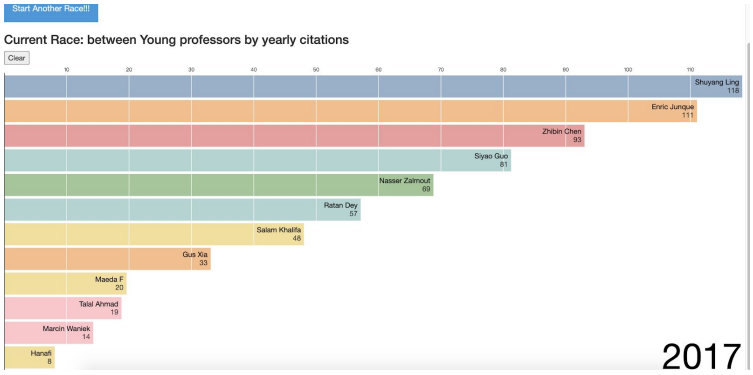
3) **Bar chart -- Ethnicity**
   - Mark: bar
   - Channel: length, horizontal position
   - Rationale: this graph aims to show the number of different ethnicities of a professor's coauthors. A bar graph is the best way to demonstrate distribution among a few categories. Moreover, the graph is ranked in descending order, which makes it easier to compare and see the top-ranked ethnicity.
   - Interaction:
     When hovering over a bar, the bar will be highlighted. Additionally, if you click on the bar, it will open up a new window which searches up that ethnicity on Google.

4) **Pie chart -- Gender**
   - Mark: area
   - Channel: angle and color
   - Rationale: this graph aims to show the gender distribution of a professor's coauthors. A pie graph is best to contrast the proportion of gender and the two colors can help clearly distinguish between the two genders.
   - Interaction: On hovering, a tooltip with the detailed number of people of this gender will appear. If clicked, the corresponding area will pop out.

5) **World map -- Location / Affiliation**
   - Mark: Circle
   - Channel: Spatial region, size, vertical and horizontal positions
   - Rationale: We choose a world map to represent professors and where they are based, because a map is best at representing geo-location and clustering of locations. The size of the circles represent the number of professors from that location. The location is inferred based on their affiliation.
   - Interaction: On hovering, the circle will be highlighted and a tooltip with the names of the professors from this location will appear.

3. **Bar chart race -- Citations:** If users click on "start a race by yearly citations/total citations" on the main view, they will enter the third view, the race bar chart.





- Mark: bar

- Channel: vertical and horizontal positions, color does not encode any features
- Rationale: This is an animation showing the number of citations professors received each year (accumulated if choose total citation, separated if select yearly citation). It presents an interesting animation which shows how the number of citations each professor received has changed over the years, and how the ranking relative to other professors have changed.
- Interaction: In this part, there is a drop down list that allows users to filter the data. One can choose to see the race of a single campus site, all professors or young professors who only have publications in the recent 10 years only (because the ranking of young professors are often hindered by professors with more citations)

Reflection

We had the idea of what we wanted to do very early on, and realized that there were no publicly available datasets for what we wanted. We used “scholarly Python library” to support our web scraping process, and several other API for data augmentation, and eventually made our graphs accordingly. We first created all the graphs separately and then combined them together into multiple connected pages. For better front-end communication, we decided to use Flask and d3 together. The process of putting data and graphs together is a challenging process, we encountered many problems such as caches, IP proxies, using different D3 versions, data transmission in flask, and graph modification. Fortunately, we managed to solve them eventually.

How have your visualization goals changed?

We initially intended to clearly demonstrate information for users mainly statically, and we ended up with adding more buttons, animations and user interface to increase interactivity and user engagement. We added:

1. Head bar in the main view: we initially designed that the main view should include a network graph, and we ended up adding a head bar and start-a-race buttons. Users can return to the main view, choose between sites using the head bar so that it looks more like a real website and is much more user friendly.
2. We managed to create an interesting racing bar chart which allows users to play with the data in a variety of ways and see an interesting animation.
3. We added animations to the many charts, and a hovering effect for almost all charts.

How have your technical goals changed?

During our data generation process, we dealt with challenges of IP being blocked, and solved it by using Luminati to change IP addresses every 10 queries. We also had to do data cleaning for the scraped data.

We initially created several graphs to put together, and later on we added more dropdowns and buttons for interaction.

1. In our bubble chart, we managed to build a “split by year” function which allows users to see each year’s publication. This improves the interaction between users and our graph.
2. We added a lot of choices for the race chart in a dropdown list
3. We added a drop down list to choose between sites
4. We had more animations and hovering effect

Also, we changed the layout of our view. Initially, we thought of putting everything in one page, and now we have a main view and other two views will appear on click. This is technically more complicated but it is more user friendly because now the layout is cleaner.

In the process, we had to fix many small but important issues. For example, we disabled any cache for real-time update of data; we enabled multiple versions of D3 to work together; we managed to send different types of data successfully through Flask; disable “click” when you do “drag”; we fixed a lot of small bugs due to many graphs presenting on one page. These are all unanticipated problems. We learned a lot about D3 by fixing them.