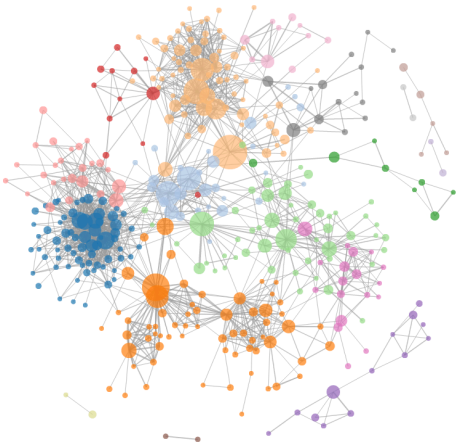
Coding Test

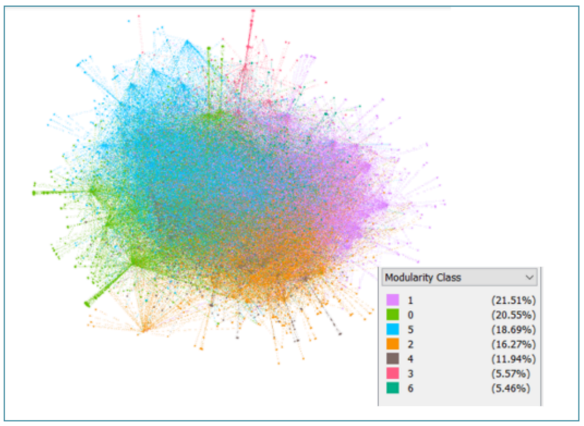
- **Submit**
  - Please create two GitHub repositories for each project. For each project:
    - a `frontend` repo
    - a `backend` repo (including notebooks or other code for data preprocessing, and add explanations about how you process the data)
  - Email Yifang:
    - `links` to your four Github repos (two repos per project)
    - Two `demo videos` showing the final results of both projects (preferably with your audio explanation)
  - **DDL: Jan. 2nd, AOE.**
  - You are encouraged to complete as many tasks as possible, but feel free to skip any you cannot finish.
- **Dataset**
  - Use the open dataset SciSciNet to analyze and visualize important information.
- **Project 1: Full-Stack Web Development**
  - Tools: Frontend (React.js, Vue.js, TypeScript, or other frameworks), Backend (Flask and Python, or other frameworks), Visualization (D3.js)
  - T1: Create two interactive node-link graphs showing (1) a paper citation network and (2) an author collaboration network for `computer science` at `your current university` over the `past five years`.
    - Output:
      - Interactive networks with **draggable nodes** using traditional graph layout methods such as force-directed layout
      - Hoverable nodes and edges that display additional information via tooltips

- Clear titles and legends to improve readability

- Note: You may encounter scalability issues due to the large number of data points. Apply your own solutions to address this problem, and embed a paragraph of your solution into the website
- Example visualizations (for illustration)



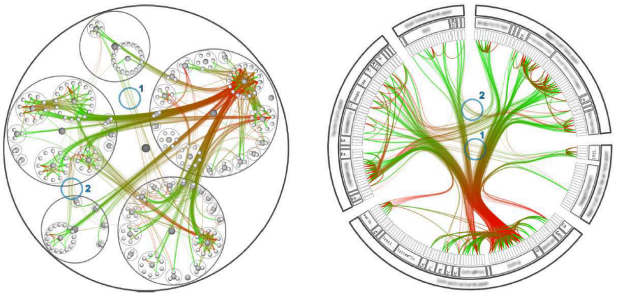
small network



large network

- T2: Create two coordinated dashboards using D3.js: (1) a timeline (line or bar chart) showing the number of CS-related papers at your university over the past 10 years; (2) a histogram showing the patent citation (`Patent_Count`) distribution of a collection of papers.
  - Interaction: When a user clicks a data point (a dot or a bar) in the timeline, the histogram updates to show the patent citation distribution of all papers from that selected year.
- T3: Select one of the networks you created above and refine it:
  - Output

- Apply (hierarchical) edge bundling technique to make the links easier to read



(a) (b)

Fig. 7 Hierarchical edge bundling[15] applied on: (a) a balloon layout; (b) a radial layout.

- Improve graph layout approach (either change the layout algorithm or adjust the parameters in the algorithm) to make the node clusters more distinguishable

- **Project 2: LLM and AI Agent**
  - Tools:
    - LangChain and LangGraph, or other LLM frameworks such as AutoGen, Crew;
    - The same full-stack development tools mentioned in Project 1;
    - Vega-Lite: a grammar of interactive graphics (DO NOT use D3.js in this task)
      - Vega-Lite online editor: https://vega.github.io/editor/#/examples/vega-lite/bar
  - T1: Create a small sample of data from SciSciNet, for example, all papers from your current university (and filter other tables accordingly).
  - T2: Build a multi-agent LLM framework to automate the data analysis process: data filtering, data analysis, and data visualization.
    - Output:
      - Perfect: A web-based chatbot with a chat interface and one or more dashboards
      - Or you can also provide a Jupyter notebook to show the workflow.

- Interaction:
  - User asks an example question such as: `show me the number of papers by field` or `show me the number of papers by year`. The system returns an interactive dashboard using Vega-Lite (e.g., a bar chart with filtering and hovering interactions)