**Overview**

This assignment is meant to give you the opportunity to do non-coordinate data visualization, specifically using a network of your choosing.

**Goals**

Network data lends itself easily to graph visualization. This assignment will allow you to explore the means of network visualization that you have encountered in the course, to decide how best to visualize data, and to analyze other network visualizations.

**Time Estimation**

The time this assignment takes will vary greatly and depends a lot on your choice of data set, on the tools you use, and on the customization of your data. Try to leave yourself at least three hours to devote to this assignment.

**Instructions**

* Find some network data that you think is suitable and that you would like to visualize. Here are some sites that provide links to a wide variety of different graph/network datasets:

1. [Stanford Large Network Dataset Collection](http://snap.stanford.edu/data/index.html" \o "Link: http://snap.stanford.edu/data/index.html" \t "_blank)
2. [UCI Network Data Repository](https://networkdata.ics.uci.edu/" \t "_blank)

* Choose a visualization platform and parse the data into a format suitable for the tools you will use.

1. For non-programmers, there are downloadable programs for creating graph visualizations at Graphviz. The program "neato," which creates a layout for an undirected graph based on multidimensional scaling, is a good place to start. The main challenge with using these tools is converting the graph data into the input text file format used by the tool, and understanding (and experimenting with) the various tool settings.
2. For programmers, there are graph visualization tools available in D3 for JavaScript, such as force-directed graphs, treemaps, collision detection, and a nice graph drawing tutorial. Feel free to use any other libraries or languages as well.
3. Visualize the data in a meaningful way, keeping in mind the requirements of the rubric.

**Submission**

* You must upload an image of your visualization for peer evaluation.
* In addition to your visualization, please include a paragraph that helps explain your submission. A few questions that your paragraph could answer include:

1. What is the data set that you chose? Why?

2. Did you use a subset of the data? If so, what was it?

3. Are there any particular aspects of your visualization to which you would like to bring attention?

4. What do you think the data and your visualization show?

**[Submit Programming Assignment 2](https://www.coursera.org/learn/datavisualization/peer/zUX82/programming-assignment-2-submission" \t "_blank)**

**Evaluation**

Your peers will use the [Programming Assignment 2 Rubric](https://www.coursera.org/learn/datavisualization/supplement/F1vg5/programming-assignment-2-rubric" \t "_blank) to evaluate your submissions. You will also evaluate four of your peers' assignments after you have submitted your assignment. This assignment is worth 15 points.

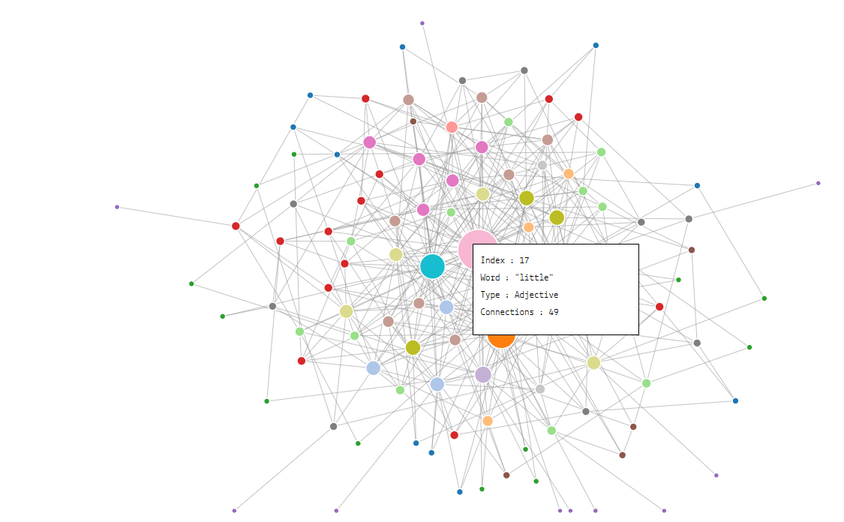
[Evaluate Programming Assignment 2](https://www.coursera.org/learn/datavisualization/peer/zUX82/programming-assignment-2-submission/give-feedback" \t "_blank)

**Q & A**

Please post any issues or questions about this assignment to the [Programming Assignment 2 Help Forum](https://www.coursera.org/learn/datavisualization/discussionPrompt/7YoNA/programming-assignment-2-help-forum" \t "_blank).

**Sample Submissions**

**Sample Submission 1:**



**Explanation:**

This is a graph visualization representing the most commonly occurring adjectives and nouns in the novel *David Copperfield* by Charles Dickens. The nodes are both adjectives and nouns, and the edges connect words that appear in an adjacent position in a book. Hovering over any node brings up the data associated with that node.

The size of the node is determined by the number of edges it has, and the positions are determined by a force-directed layout algorithm.

**Grading Rubric**

|  |  |
| --- | --- |
| **Proximate Layout** | 5 points – Related data is connected properly, and the varying sizes helps to properly separate data. |
| **Design of the Visualization** | 4 points – Only one problem with the use of color. |

*Note: The Contest part of the rubric is up to you.*

**Sample Comments from Instructor:**

Even though there are many edge overlaps, this is a large non-planar graph. The items are well distributed and the edges are easy to follow visually through the overlaps.

The frequency of words is nicely mapped to the area. The colors are assigned randomly. While the random colors help differentiate the nodes, there is no differentiation between adjectives and nouns, and so the visualization misses the opportunity to easily incorporate this dimension in its use of color at the nodes.

**Sample Submission 2:**

One way to create a graph from ordinary data is to create an edge between items and the categories they belong to. For example, this is a map of the CS faculty at UIUC. We had a list of faculty and the areas of CS they worked in. For example:

...

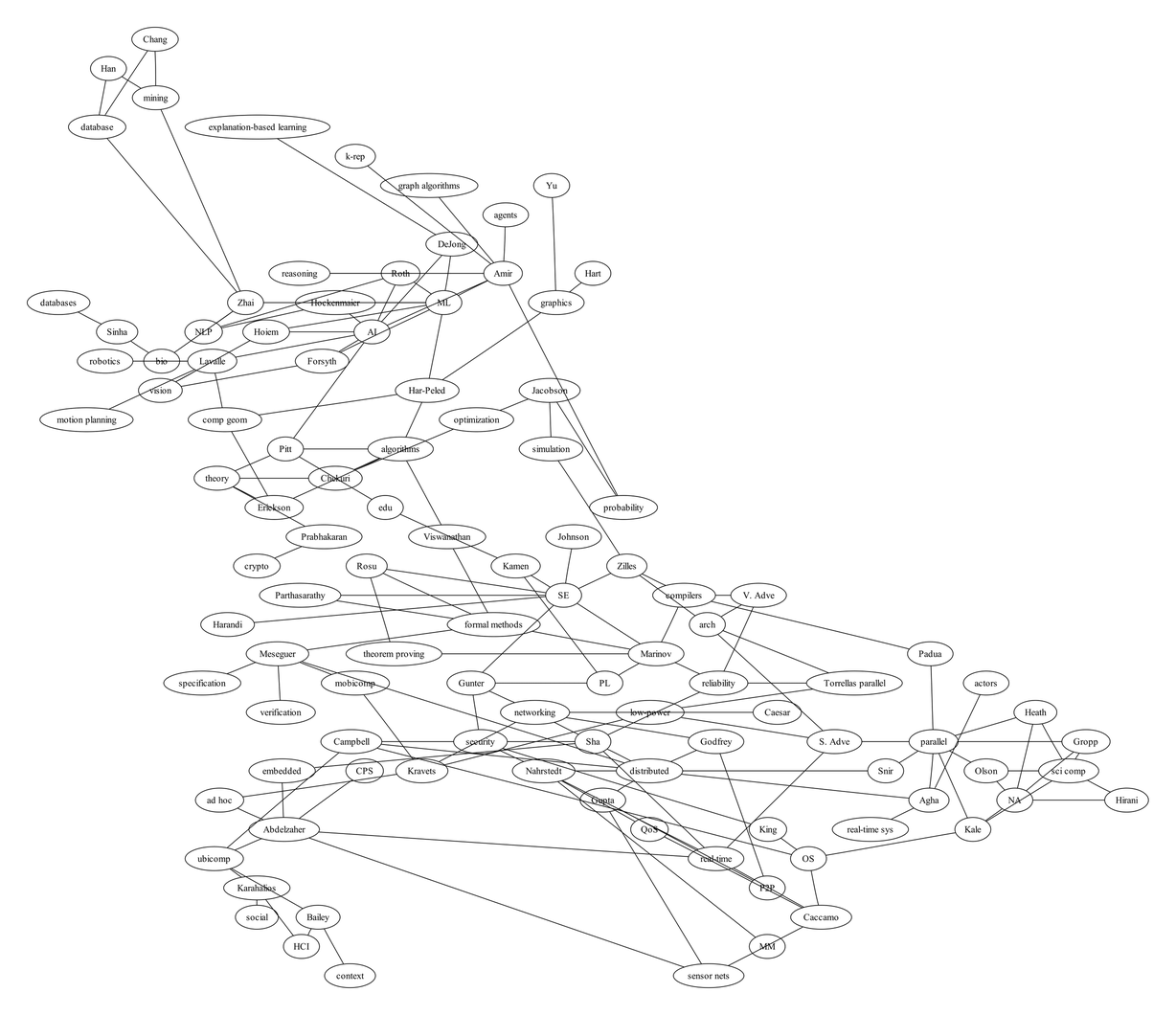
Han: database, mining

Hart: graphics

...

Zhai: database, mining, bio, ML

Using a dot file (for the “dot” layout program available from Graphviz) that assigned each faculty member and each research area to a node, we then made edges from each faculty member to the research areas they had listed. The result was the attached map of the CS department that clustered faculty by the areas they worked in.



**Grading Rubric**

|  |  |
| --- | --- |
| **Proximate Layout** | 4 points – Unnecessary and distracting overlaps of the edges and nodes |
| **Design of the Visualization** | 4 points – No differentiation between faculty and areas. |

*Note: The Contest part of the rubric is up to you.*

**Sample Comments from Instructor:**

Even though the instructor made this graph, it is far from perfect. While the edges do not overlap each other, the edges do overlap the nodes, which makes it difficult to see. For a graph with this few number of elements, one could adjust the placement by hand, perhaps using the free vector graphics editing utility "inkscape." Also, the graph does not differentiate between faculty and areas, but could with node shapes and/or colors.