Building Semantic Networks

This section will guide you through the steps to make a network that can be visualized using Gephi and analyzed using MATLAB. If you haven't already, please download and install Gephi. If you'd like to click the hyperlinks, you'll need to download this .pdf from GitHub by clicking the 'Download' button on the top right of the window.

Constructing a network from scientific abstracts

1. **Download word embedding model from the link below (file size: 109.3 MB)**

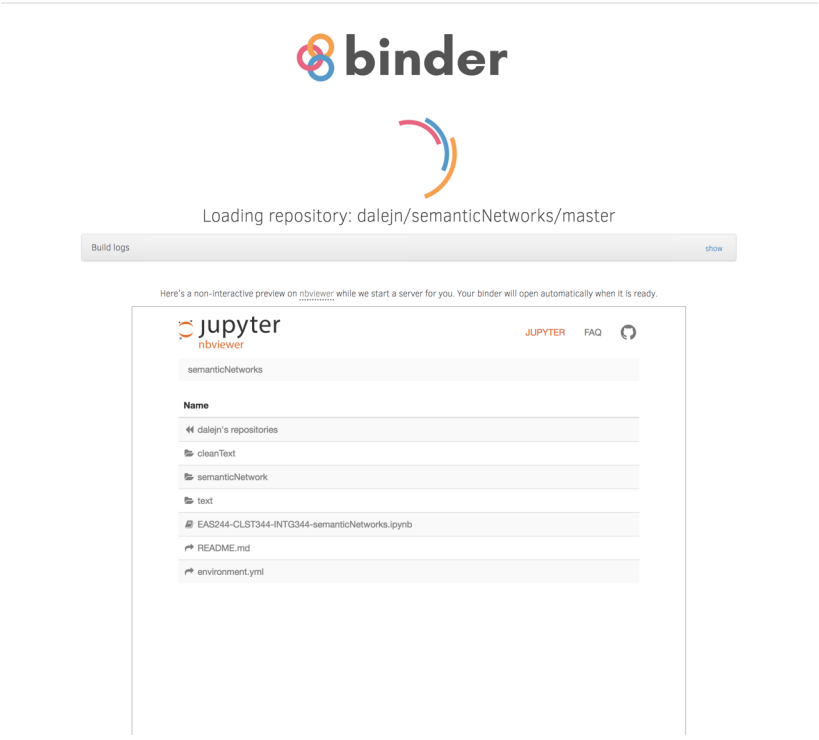
   https://www.dropbox.com/s/j48n6fsavvux5ha/w2v.model?dl=1

   This is a neural network model of word vector embeddings with 52,210 word vectors representing words' semantic meaning trained on 538,238 cleaned articles from PubMed. For more details about the natural language processing an

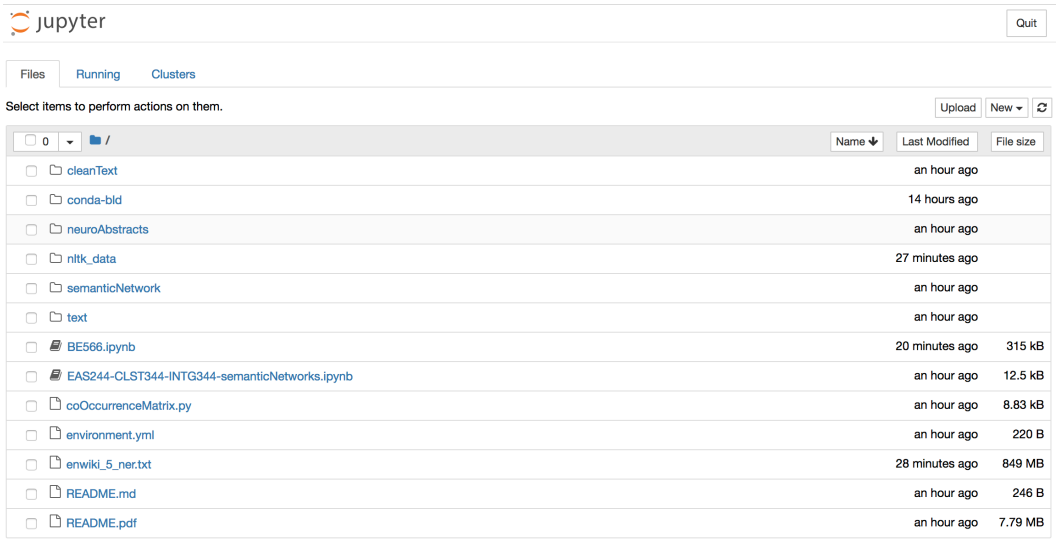
2. **Start programming environment**

   https://mybinder.org/v2/gh/dalejn/semanticNetworks/master

   A. You should see the loading page below. Depending on how many students are trying to access the server, it may take a few minutes or longer to start the programming environment. If it is still loading after 10 minutes, please refresh the page. If you receive an error along the lines of "Event stream failed to load," reload the page. This should be a temporary server error.

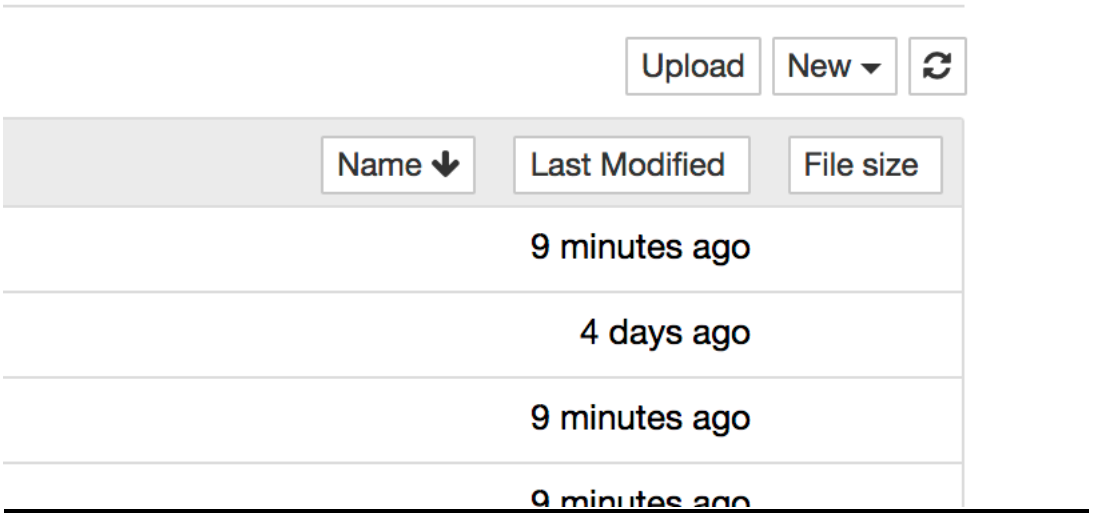


B. When the page is loaded, you will see a screen like the one below. Note, that the environment will automatically shut down after 10 minutes of inactivity (if you leave your window open, this will be counted as “activity”).

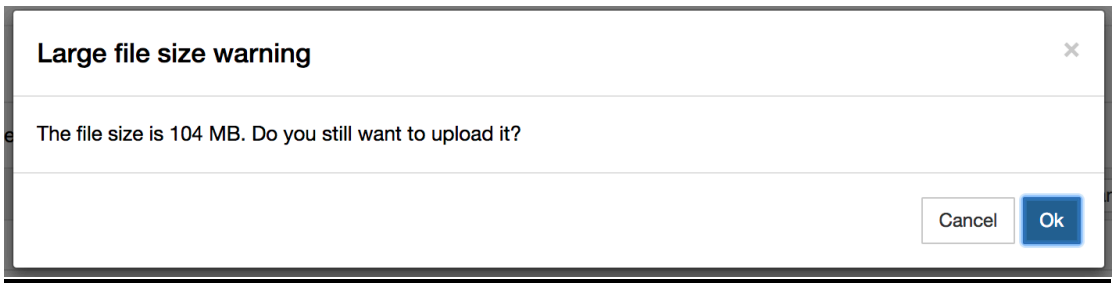


3. <u>Upload the model downloaded from step 1 to the environment</u>

A. Click on “Upload,” go to where you downloaded the file from Step 1 and select it.

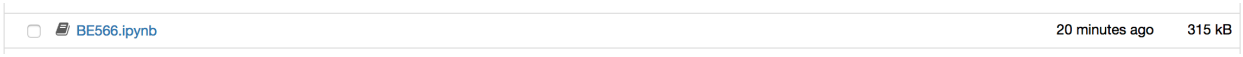


B. Press Ok after selecting the model to upload.

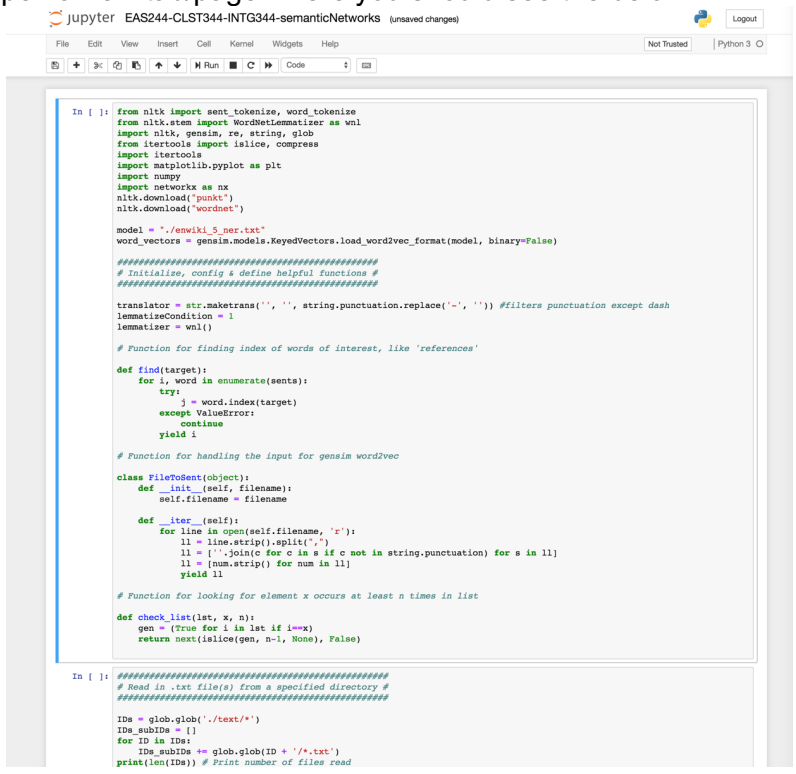


C. Click on the blue Upload button and you should see the percent progress of the upload. Depending on how many students are accessing this environment, this upload can take a few minutes or longer. If the upload progress bar gets stuck at a certain percentage for over a few minutes, cancel and re-upload (overwriting the partially uploaded file).

4. **<u>Once the upload has completed, click on the text link "BE566.ipynb" as seen below to open a programming notebook.</u>**

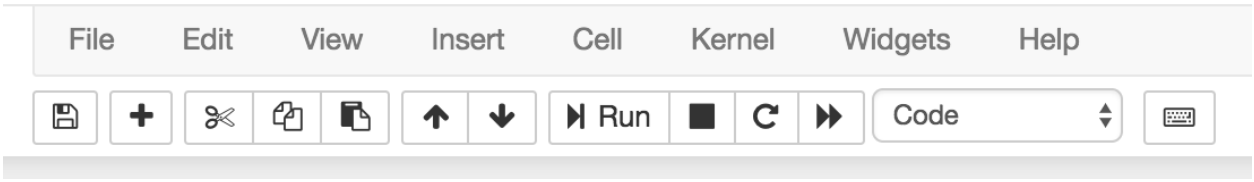


This will open a new tab/page where you should see the below:

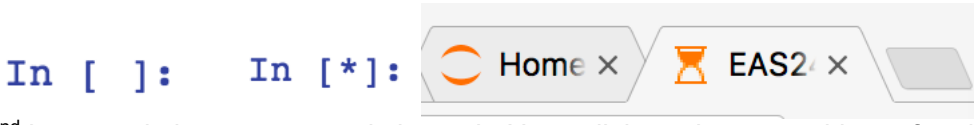


5. **<u>Run the code</u>**

A. 1st box: loads the relevant packages, modules, and functions. Click on the first box, which should then become highlighted along the borders. At the top of the screen in the toolbar, click the Run button.



The text label will change from an empty bracket '[ ]' to one with an asterisk '[*]', which means the code is running. When the code is done running, the asterisk will be replaced by a number. Your tab in the web browser may also show an hourglass to indicate that the code is still running. When it disappears, the code in this box is running. This can take a few minutes or more to finish.



B. 2nd box: reads in your text and cleans it. Now, click on the second box of code, which should become highlighted along the borders.

Change the red text within the single quotes to the author for which you wish to construct a semantic network. The format must be '{Last name}, {First initial}.' If you have the middle initial, then you may use it for more specificity, e.g. 'Satterthwaite, TD'

```
####################
# Clean, lemmatize #
####################

authorName = 'Satterthwaite, T' # 'LastName, FirstInitial'
```

Run this code by clicking the Run button. This will take a few seconds at most.

C. 3rd box: constructs the semantic network matrix from the shortest N cosine-distances (the strongest N associations). Select the third box. Depending on the number of nodes in the graph (i.e. number of unique words in the text), you may want to change the total number of edges (i.e. connections between the words) in the constructed semantic network. I've set this number to be 10 times the number of nodes, but this can be changed depending on how dense you want the resulting semantic network to be.

```
# The number of connections we want: either as a factor of the number of words or a set number
num_top_conns = len(my_words) * 10
```

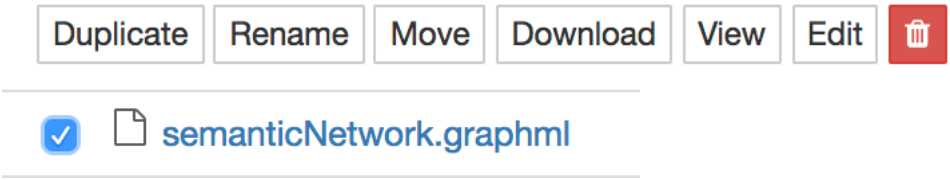
You will see a list of words outputted.

D. 4th box: constructs a co-occurrence network based on 5-gram sliding window (represents a count of words pairs that co-occur with each other in 5-word chunks across the entire text)

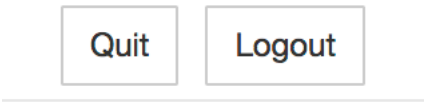
6. Download the relevant outputted files

A. You can close the notebook now. Return to the directory screen and click the "semanticNetwork" link.

B. Check a box (**one at a time**) to download **all** the files in this folder. You must select the files one at a time, or else the Download button doesn't appear.



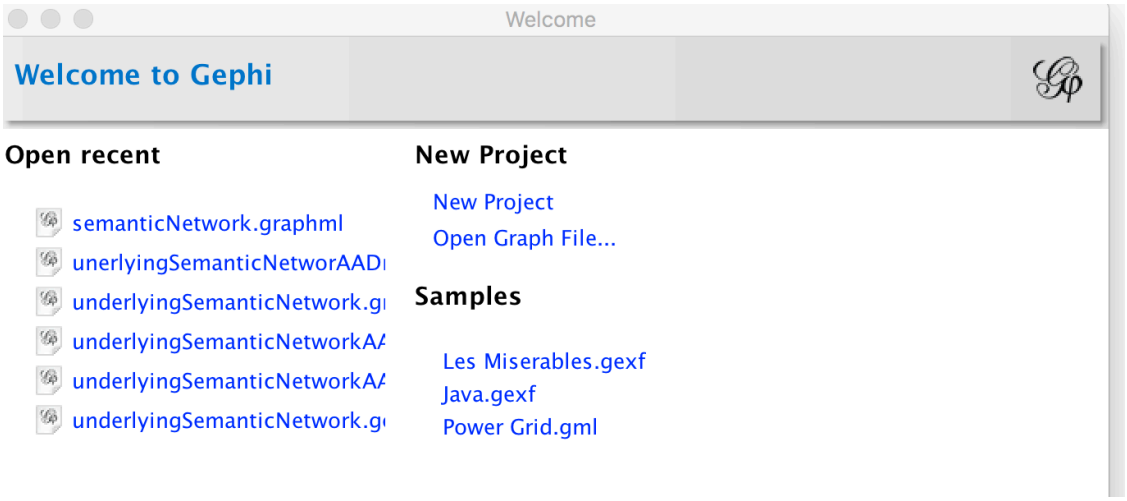
C. Once you've downloaded **all files in this folder**, press Quit at the top right.



7. Visualize the network in Gephi

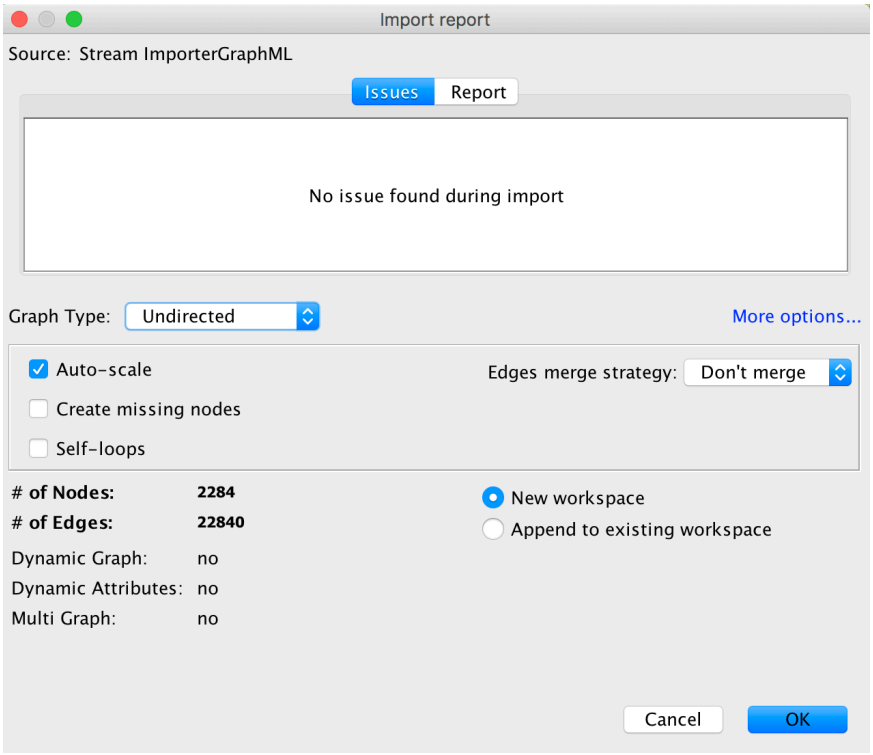
A. Open Gephi. If you haven't already downloaded and installed it, please do so here.

You will see a Welcome screen like below. Click on "Open Graph File…" If you don't see the welcome screen, go to File and Open on your computer's toolbar.

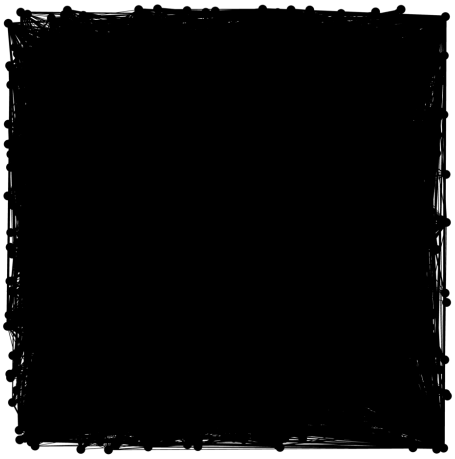


B. Navigate to where you downloaded semanticNetwork.graphml and open this file.

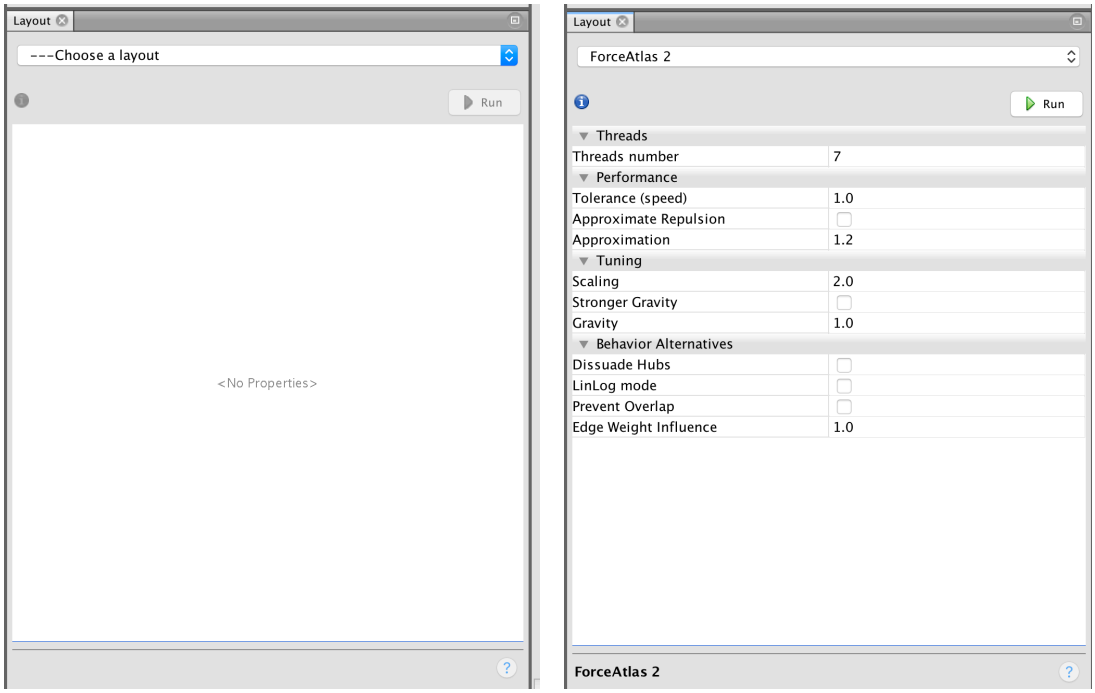
C. A new screen should pop up. Default settings should be fine but please check that they match the ones below and then press OK. Note: If you are loading coOccurrenceNetwork.gexf, there will be many "Issues" listed—you may ignore them and press OK.



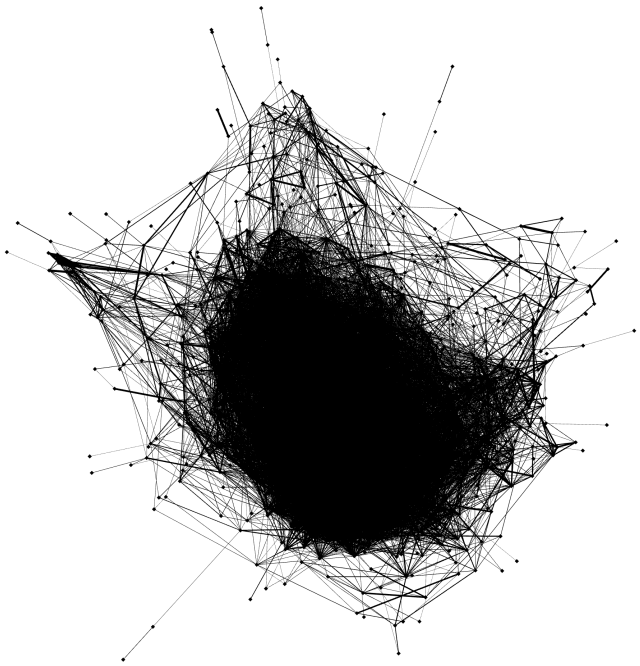
D. The network should be visible. If not visible, you may need to go to Gephi's menu at the top and select Window > Graph. You can scroll your cursor over nodes to see connections. We'll now clean this up to make it more interpretable.



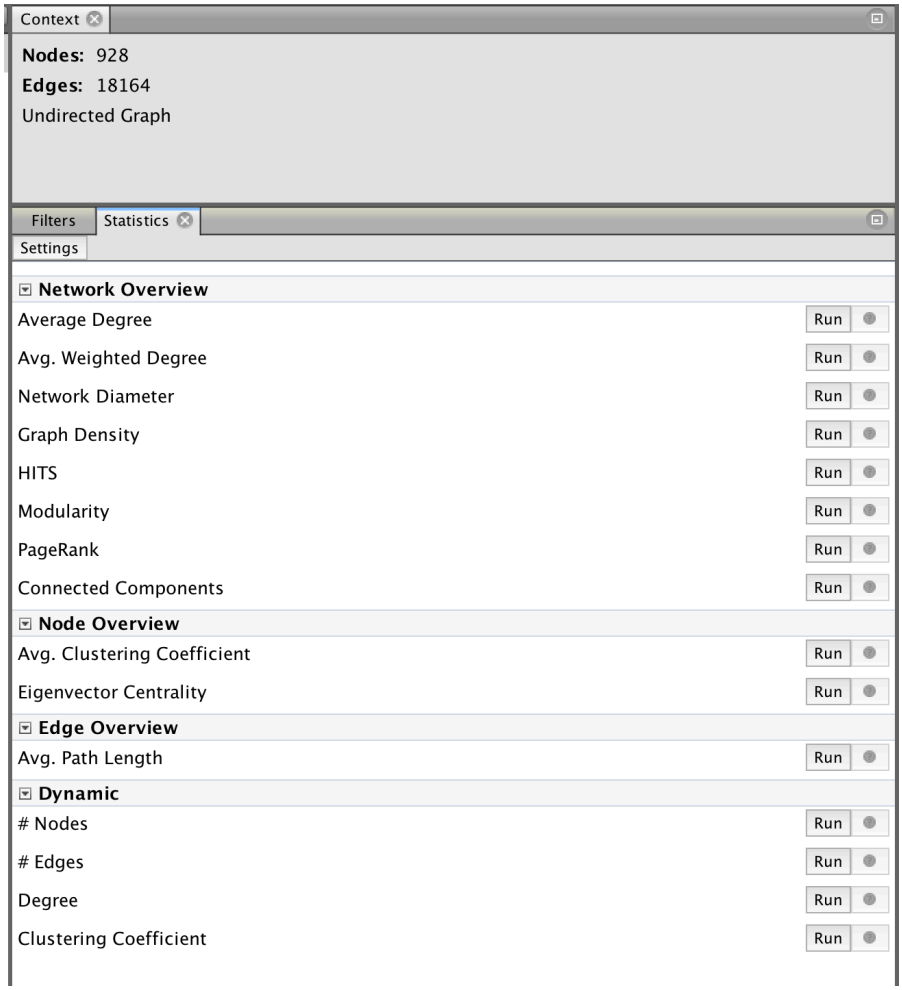
E. On the bottom left, click on "---Choose a layout" and select "ForceAtlas 2." Then, press the Run button, which will then be replaced by a Stop button.

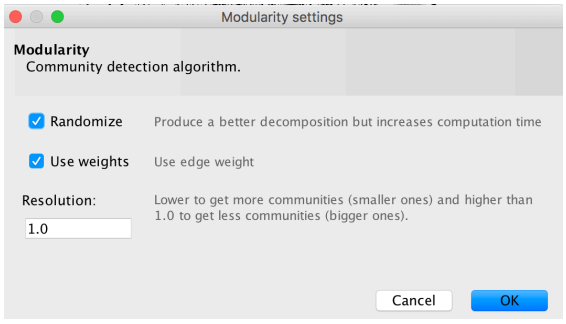


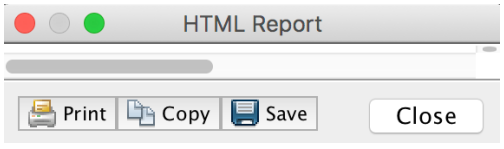
F. Use your mouse wheel to zoom in and out. Hold right click and drag to move the network around. When the network looks like it has stabilized (not moving much), press the Stop button.



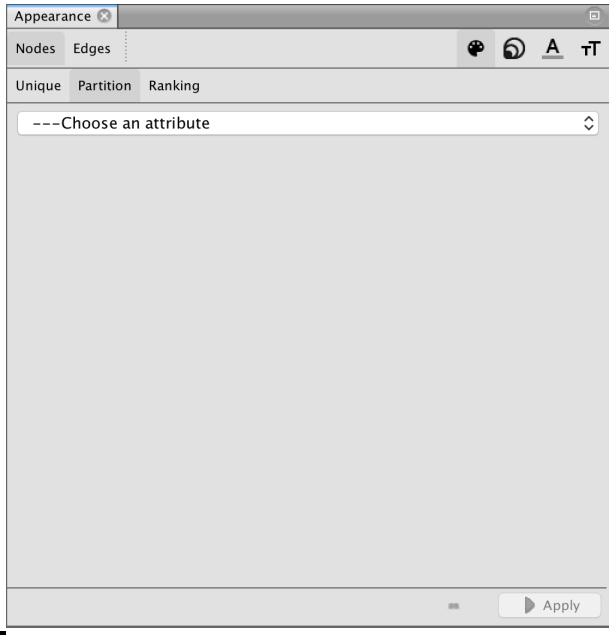
G. On the right-side toolboxes, click on the “Statistics” tab and then click “Run” for the Modularity measure. A box will pop up; click OK. Another pop-up labeled HTML Report will appear; click Close.



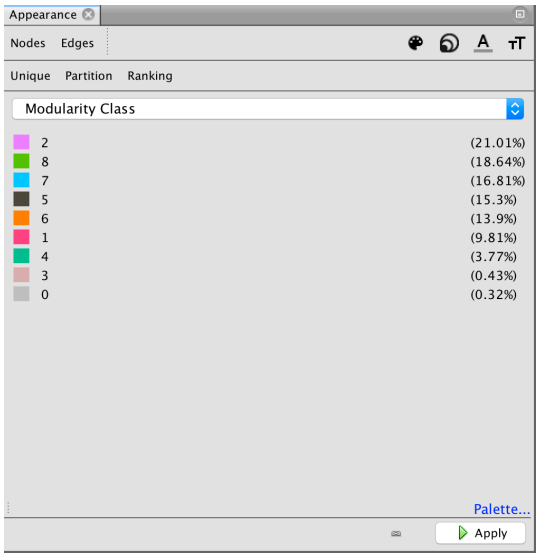




H. At the top-left toolbox labeled Appearance, click on the “Partition” tab.

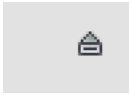


I. Click on "—Chose an attribute" and select "Modularity Class." Click "Apply."



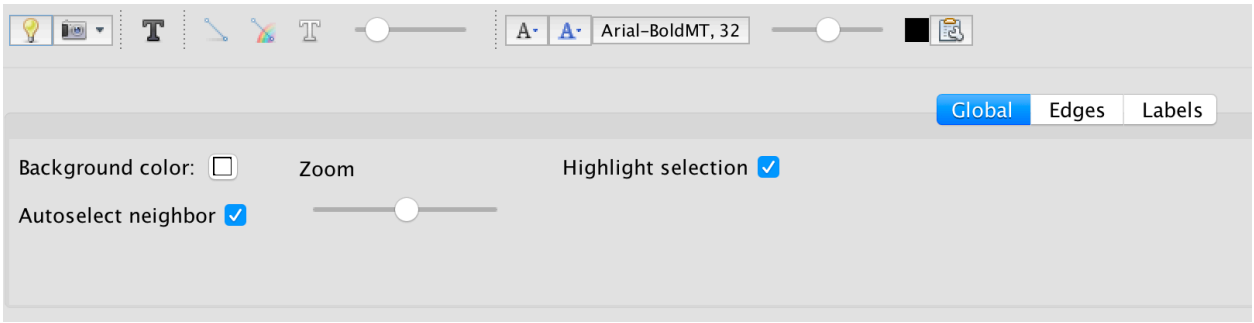
Your graph should now be colored by module.

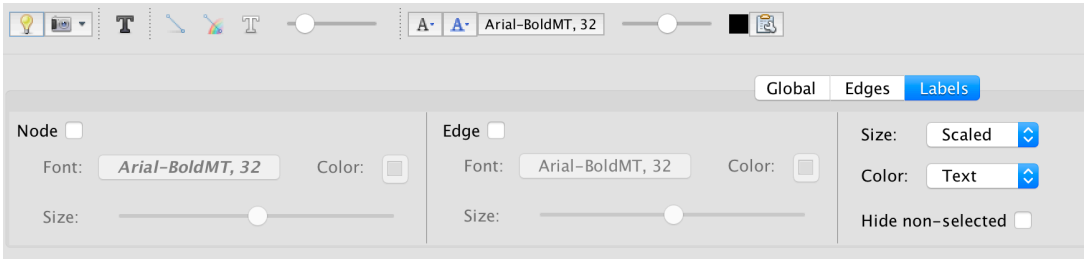
J. Label each node with the unique word it represents. At the bottom, there is a toolbar; click the small button all the way to the right.

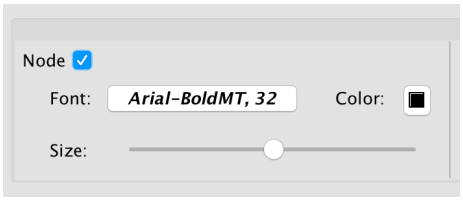


← Small button

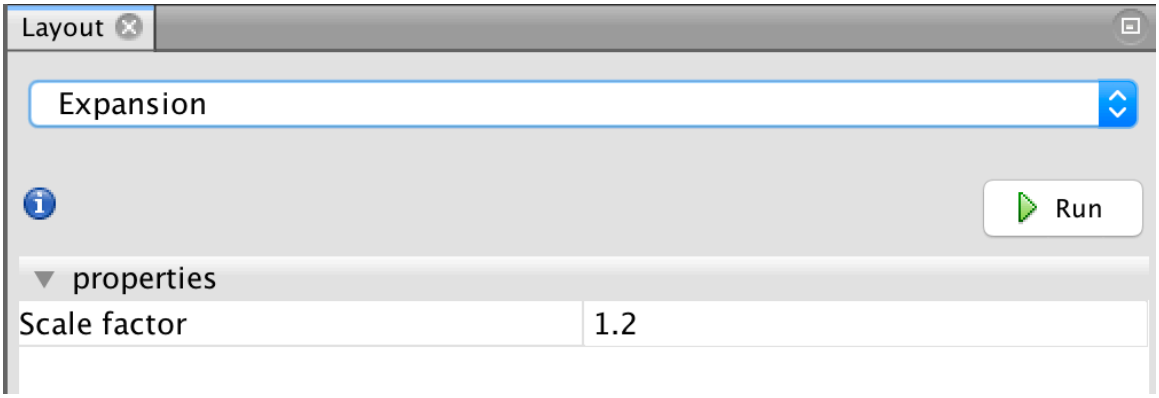
K. Click on the “Labels” tab and click the checkbox for Node.



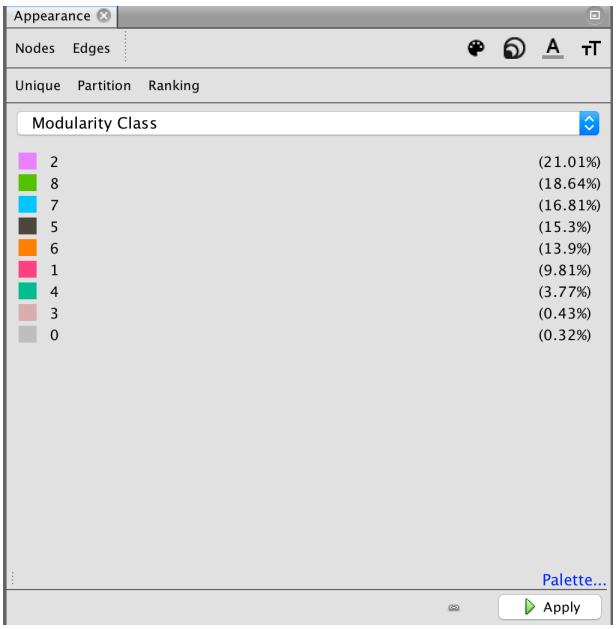


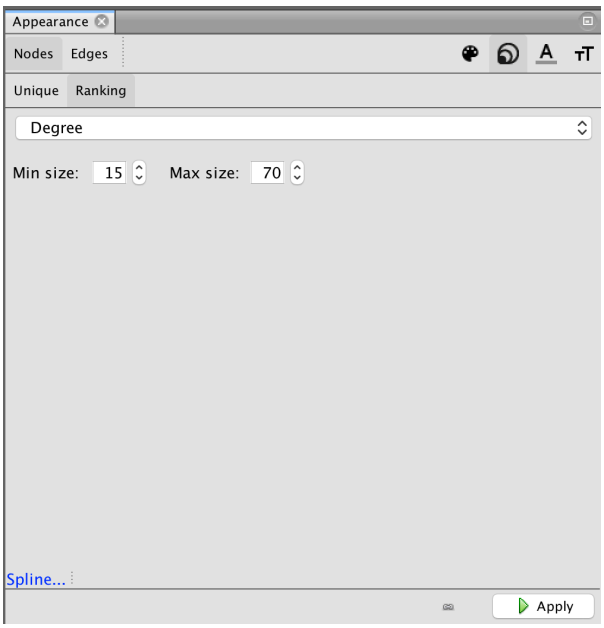


L. Spread out the graph to read the text more easily by going to the "Layout" pane again (the same place where you applied "ForceAtlas 2") and select "Expansion." Click on Run a few times until you're satisfied with the spread of the text labels. Then, zoom out on the graph using your mouse wheel.

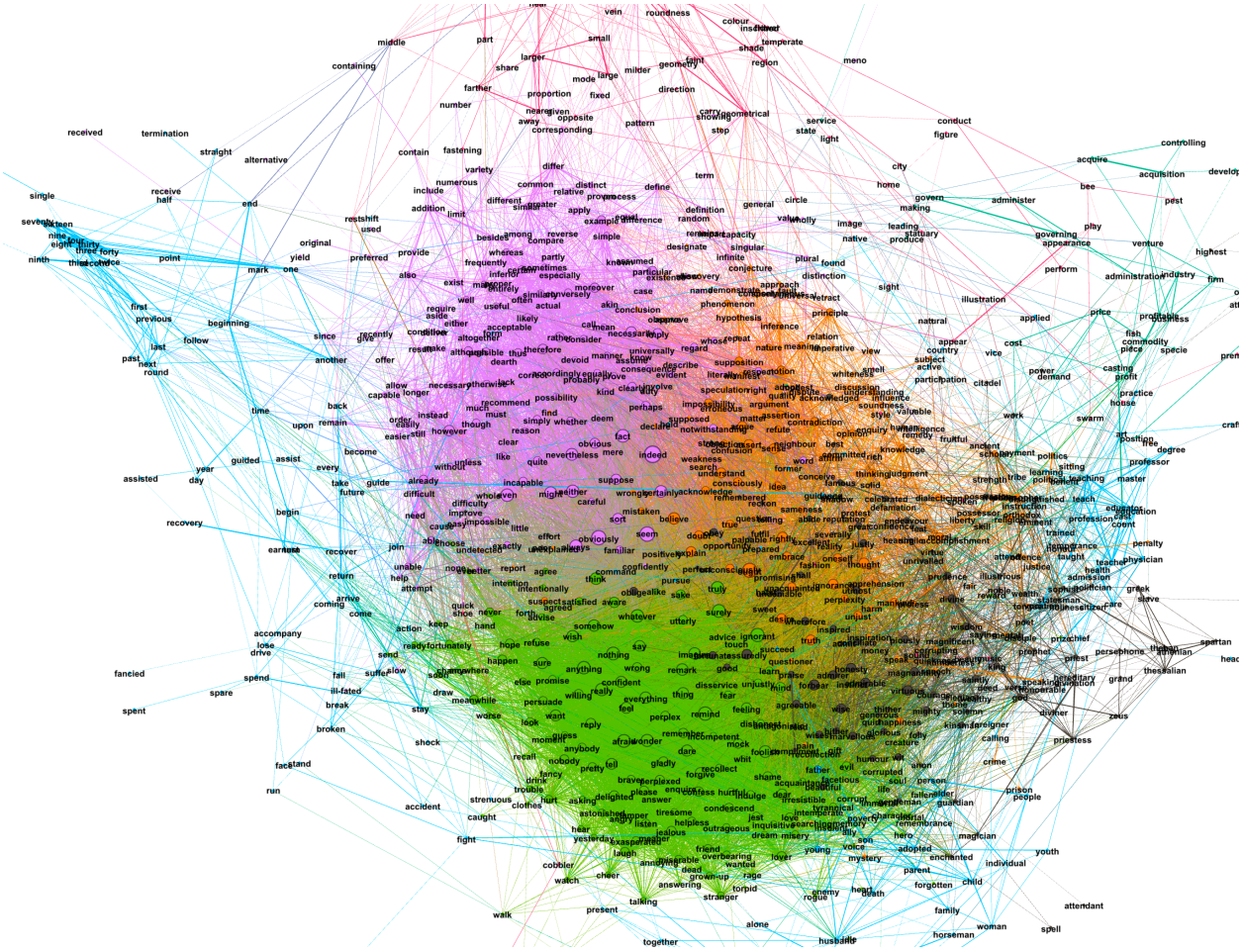


M. Finally, navigate back to the "Appearance" pane and click the  button. Click on the "Ranking" button. Then, select "Degree" from the dropdown menu and change the "Min size" and "Max size" to the ones below. Click "Apply".





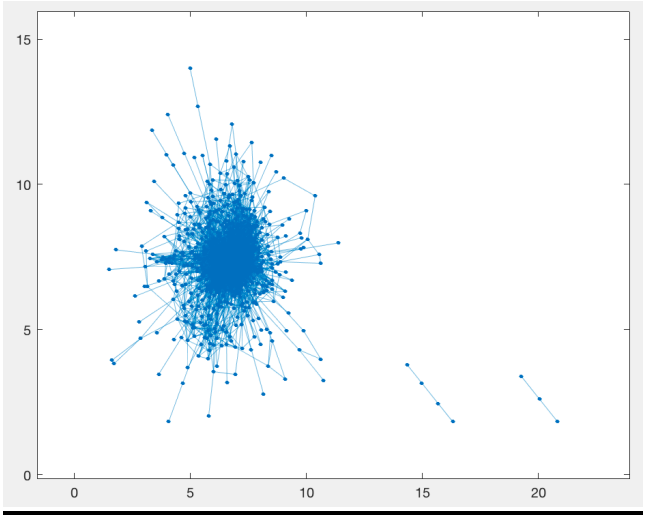
N. Explore your graph! You can scroll over individual nodes to see what words are connected.



O. **Repeat step 7 with the coOccurrenceNetwork.gexf.** How do the networks differ?

8. **Visualize/analyze the network in MATLAB.**

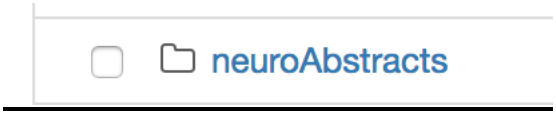
A. Please see semanticNetworkAnalysisReadme.pdf for further instructions.



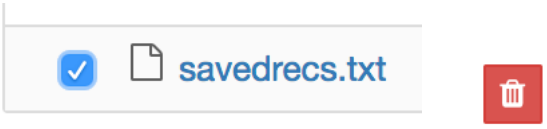
Constructing a network matrix from your own text

1. **Follow the same steps as above, but on Step 3 also do the following:**

A. Click on the "neuroAbstracts" link to navigate to that directory.



B. Check the box for "savedrecs.txt" and then delete it by clicking the red trashcan button.



C. Click on the "Upload" button and upload your text file(s) (must be in .txt format) of the BibTeX-tagged referencss which you previously compiled from Web of Science. Please ensure that the filename ends with .txt

D. Continue Steps 4 and on as described in the above steps.

More details

Math underlying word2vec: https://arxiv.org/pdf/1411.2738.pdf

Nice visualization/demonstration of some toy training data: https://ronxin.github.io/wevi/

Google's original word2vec paper: https://arxiv.org/pdf/1301.3781.pdf

Examples of other implementations of word2vec on biomedical texts (with documentation):

http://bio.nlplab.org/

http://bioasq.org/news/bioasq-releases-continuous-space-word-vectors-obtained-applying-word2vec-pubmed-abstracts