MIE 451/1513 Decision Support Systems
Lab and Assignment 5:
Social Network Analysis

This assignment involves social network analysis based on twitter data. Through this assignment you will have better understanding of graph analysis methods, as well as different centrality measures in the graph, and edge prediction.

- Programming language: Python (Google Colab Environment)
- Due Date: Posted in Syllabus

Marking scheme and requirements: Full marks will be given for (1) working, readable, reasonably efficient, documented code that achieves the assignment goals, and (2) for providing appropriate answers to the questions in a Jupyter notebook (named `sna-assignment.ipynb`) committed to the student's assignment repository.

Please adhere to the collaboration policy on the course website – people you discussed the assignment solution with, or websites with source code you used should be listed in the submitted Jupyter notebook.

What/how to submit your work:

1. All your code should be included in a notebook named `sna-assignment.ipynb` that is provided in the cloned assignment repository.
2. Ensure that the images (particular visualizations of the mention graphs) are saved in the notebook as static images (png, jpg, etc.).
3. Commit and push your work to your github repository in order to submit it. Your last commit and push before the assignment deadline will be considered to be your submission. You can check your repository online to make sure that all required files have actually been committed and pushed to your repository.
4. A link to create a personal repository for this assignment is posted on QUERCUS.

Notes that you should pay attention to:

- This assignment is **graded entirely by the TA's** – your **submitted** Jupyter notebook must contain all experimental output – **notebooks with empty output cells will not be graded**.
- While there is **no AutoGrader** for this assignment, **you must still submit on github by the required deadline**. Timestamps are checked.

- As a backup, in case your cell data is lost, you should add and commit a `.pdf` version of your submitted `.ipynb` to your github repo. (This `.pdf` should show up when you browse your assignment repo via the web interface.)
- In case that your plotted graphs are cropped when saving the `.pdf` version of your submitted code, you can adjust the **Scale** options (e.g. Custom, 70) after selecting **File − Print**.
- Re-run the cell for creating the plots if it does not create any output for a while after being executed
- Please do not commit/push the data to your github Repository (github is not for data storage) – it should only contain your Jupyter notebook and supporting images/pdfs.

This assignment has *7 points* in total and point allocation is shown below:

- Grade allocation (7 points):
  - Q1: 1 point
  - Q2: 1 point
  - Q3: 1 point
  - Q4: 1 point
  - Q5: 1.5 point
  - Q6: 1.5 point

1 The Twitter Data

We will work with data taken from Twitter. We have provided a collection of tweets taken from one month of 2009. The data is stored in a csv file provided on QUERCUS (`tweets2009-06-0115.csv.zip`). Each tweet is a line with three fields: time, user and text of the tweet. Here is a snippet:

```
2009-06-11 16:59:45, ibbored, amberback #squarespace does?  Hot damn.  Now I want to
win more.
```

Twitter users abbreviate topics with hash-tags (#squarespace) and mention others with the @ sign (answerback). In this assignment, you will need to extract records like user names and hash-tags from the text, and analyze different patterns of communication in the given dataset.

Note: You are free to use your own Twitter dataset, either precompiled and found on the Internet or scraped yourself. Some additional existing datasets have been listed here:

- `https://github.com/shaypal5/awesome-twitter-data`
- `https://www.kaggle.com/data/35739`

You will need to ensure any datasets you collect or find will have the time, user, and the text of the tweet. Please indicate your data source in your lab submission along with the hash-tag you have selected and follow the same guidelines referenced in **Q1**.

2 In the Introductory lab

In this introductory lab, we will build a mention graph for a given hash-tag, and explore different ways to visualize graphs, and incorporate additional information. We will cover graph creation, manipulation, and analysis using `NetworkX`, complex sentiment analysis with `NRCLex`, and visualization using `Plotly`.

3 Main Assignment

Q1. Choose a hash-tag

You are required to choose a hash-tag to perform the social network analysis on. The chosen hash-tag should be unique and not shared with the any of your classmates. In order to keep track on the hash-tags being used by the other students, please refer to the discussion board on Piazza. Once you choose a hash-tag, please make sure it is still available and then post your chosen hash-tag on the discussion board.

- The first one to post a hash-tag will be the one to perform the analysis on this hash-tag.
- It is your responsibility to make sure the hash-tag is still available when you post it to Piazza.
- Make sure the hash-tag will be suitable (i.e., the hash-tag has been used in a large number of tweets and has been retweeted between users) for the analysis required in this assignment.
- If you want to change the hash-tag, please post another message declaring the old hash-tag (now available), and the new one (now allocated to you). As before, it is your responsibility to make sure the new hash-tag is available.

Q2. Build a Mention Graph

In this question you are required to build a mention graph for your chosen hash-tag. The mention graph is the mention relations between users. In this graph, each user is viewed as a node. If a user `Alice` mentions another user `Bob` in her tweet(s) with the @ sign, then an undirected edge connects `Alice` and `Bob`. The edge weight is the number of mentions in the tweets.

(a) How many nodes and how many edges are in your mention graph?

(b) Build a histogram of the graph nodes' degree (i.e., the degree distribution of the graph). What does the distribution in node degree tell you about how your network is structured?

(c) Build a log-log scatterplot with the node degree (x-axis) and fraction of nodes with that degree (y-axis). Does your mention graph exhibit a power law trend? If not, can you explain why that may be?

(d) For the highest-weighted edge, list the tweets (up to 25) for that edge and describe the interaction between these two users which explains why this edge has the highest weight. ***Note:*** edges are identified be the two nodes they connect, e.g., $\langle node1, node2\rangle$.

(e) Provide a visualization of the mention graph in which the edges visually (e.g., color, weight, etc.) reflects its weight (i.e., the number of mentions). ***Note:*** Be sure to use layouts to help make the visualizations clear.

Q3. Content Analysis

In the this question you are asked to perform a basic content analysis on your chosen hash-tag.

(a) Analyze the most frequent non-stopwords in all the tweets with the chosen hash-tag, and provide a basic description of the main themes.

(b) In a visualization of the mention graph add hover information for the nodes which lists the 3 most common words for each user. Indicate the nodes that have no words visually. For those nodes that have words, are the words similar for connected users or are the words different? If not the same, do the words have similar themes?

Q4. Centrality Analysis

In this question, you need to analyze centrality of users in the mention graph. Here is a list of `networkx` functions that calculate the different centrality measures:
`https://networkx.github.io/documentation/stable/reference/algorithms/centrality.html`
Note that PageRank lives in a different place in `networkX`:
`https://networkx.org/documentation/stable/reference/algorithms/link_analysis.html`

(a) Using PageRank and a 2^{nd} centrality measure of your choosing, calculate the centrality of the nodes on your graph based on each of the measures. Provide a mention graph visualization for each measure that demonstrates the centrality value of each node using a visual property (size, color, etc) for each of the centrality measures.

(b) List the 5 most central nodes for each of the centrality measures. Discuss the following:

1. Are the results between centrality measures similar or different? Explain a reason for the observed similarity or difference.
2. What centrality measure produced a more meaningful interpretation? Why?

Q5. Sentiment Analysis

The following questions will use the NRC (National Research Council) Lexicon to evaluate the content of the tweets (`https://github.com/metalcorebear/NRCLex`).

(a) Using `NRCLex`, evaluate each tweet and determine the number of words in each dimension of emotion. Construct a chart showing the number of words in each emotion. Which emotion is most frequent in the data? Does that make sense in the context of your chosen hash-tag? Explain why or why not by providing example tweets to support your claims

(b) For each user, determine which fraction of a user's total identified words are in each emotional affect (i.e., normalize the counts for each user). Identify the users with the highest fraction of words in each emotion. ***Note:*** You may wish to set a minimum number of tweets for user's to be considered.

(c) Construct a plot(s) to answer the following: Is there a correlation between any dimension of emotion and one of the centrality measures you evaluated? If you see a trend, please provide a hypothesis for why; if you do not see a trend, provide a hypothesis for why centrality of a user has no observed correlation with emotion in your particular dataset.

Q6. Link Prediction

You are going to predict the evolution of the network using link prediction.
`https://networkx.org/documentation/stable//reference/algorithms/link_prediction.html`

(a) Sort your collection of tweets, used to build the original mention graph in **Q2**, based on the time of their posting. Remove the last 20% (i.e., the most recent ones) and with the remaining 80% of the complete dataset construct a new mention graph. How many fewer nodes and edges are in your new mention graph?

(b) Select three different link prediction methods to be used. Explain why you selected these 3 methods. Which method do you think will output the most correctly predicted links?

(c) Using the *core nodes* in your new mention graph only (degree > 3), predict the 5 most probable new links using the selected 3 different link prediction methods. Visualize your mention graph for each prediction method. Indicate incorrect edge predictions (in red) and correct predictions (in green). Existing edges in the new mention graph should be included but make them subtle. Which method gave the most number of correct predictions? Hypothesize why it worked the best. Did the results match your expectation from **Q6b**? If not, why?

4 Helpful Links

NetworkX Documentation

`https://networkx.github.io/documentation/stable/index.html`

Plotly

`https://plot.ly/python/`